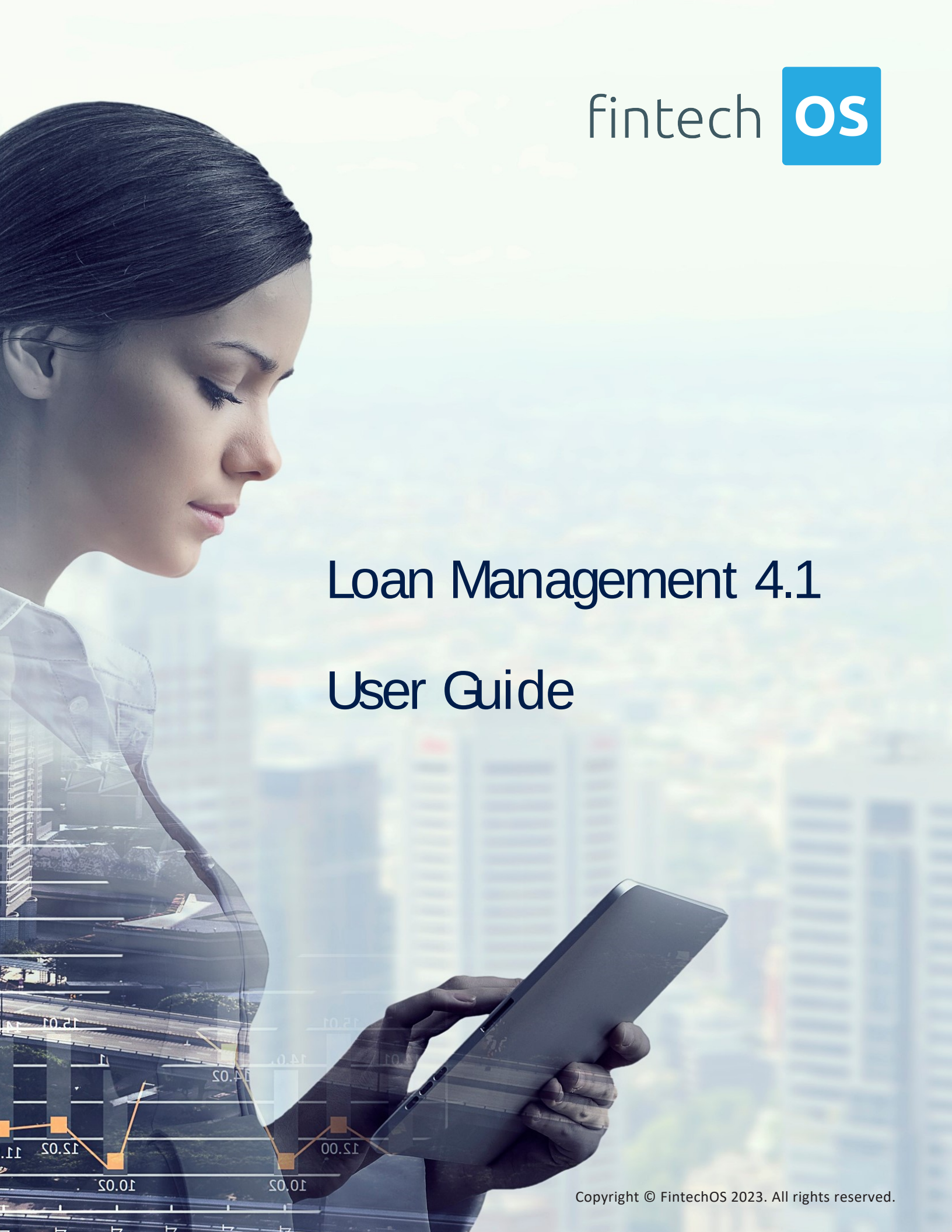


Loan Management 4.1

User Guide

Copyright © FintechOS 2023. All rights reserved.

TOC

Overview 8

Banking Product Factory 9

Customer 360 9

Subledger 9

Configurations for Loan Management 10

Getting Started with Loan Management 12

External Request Processing 17

Configuring the Async Engine 17

Managing Unprocessed Requests 37

Holiday 40

Creating Holiday Records 41

Allocation Method 42

Creating Allocation Methods 43

Exchange Rate 45

Creating Exchange Rates 46

Exchange Rate Type 47

Creating Exchange Rates Types 48

Loan Classification 49

Creating Loan Classification Records 51

Loan Periodicity 52

Creating Loan Periodicity Records 53

Operation Item 54

Creating Operation Item Records 56

Loan Management System Parameters 58

Managing Loan Management System Parameters 78
Transaction Types Used in Loan Management 80
Bank Account Transaction Queue 89
Bank Account Transaction Configurations 91
Transaction Types Covered Through Bank Account Transaction Operation Types in Loan Management 92
Loan Contract Approval Process Flow 95
Transaction Operation Type 98
Creating Bank Account Transaction Operation Types 99
Transaction Fee 102
Creating Transaction Fee Records 102
Transaction Fee List 104
Creating Transaction Fee List Records 105
Jobs 106
Security Roles for Loan Management 115
Predefined Loan Management Security Roles 116
Banks 152
Creating Bank Records 152
Creating External Bank Accounts 154
Sales Channels 155
Creating Sales Channel Records 156
Reconciliation Accounts 157
Creating Reconciliation Accounts 158
Reconciliation Account Settings 161
Creating Reconciliation Account Settings 162
Customers, Groups and Limits 164
Customers 165
Creating Customers 166

Groups 175
Limits 177
Group Exposure Types 177
Customer Exposure Types 178
Validations 179
Calculation of Available Limit Amount 182
Limit Statuses 183
Limit Versioning 184
Changing Limit Statuses 185
Managing Limits 187
Role-Based Limits 188
Role-Based Limits Validations 190
Managing Limit Types 191
Creating Limit Types 192
Creating Limits 193
Creating New Versions of Existing Limits 198
Collaterals 200
Managing Collaterals 201
Contracts 209
Contract Implementation Notes 209
Managing Contracts 209
Loans 211
Loan Contract Life Cycle and States 213
Contract Versioning 214
Changing Contract Statuses 214
Creating A New Unsecured Loan 216
Creating A New Secured Loan 236
Approving a Loan 240
Automated Actions After Contract Approval 242

Rejecting a Loan 246
Disbursing a Loan 248
Processing Loan Repayments 253
Working with Overdue Loans 283
Applying Payment Holiday to a Loan 286
Working with Grace 290
Working with Participants 292
Adding Participants 293
Blocking Participants 294
Working with Tranches 294
Adding Tranches to a Contract 295
Working with Covenants 296
Adding & Activating Covenants 297
Reviewing Covenants 298
Working with Contract Classification 300
Adding Classifications to a Contract 301
Working with Returns 301
Applying Fees and Commissions 310
Adding Fees 312
Closing a Loan With All Obligations Met 313
Rescheduling and Refinancing Loans 316
Writing Off a Loan 326
Changing the Interest Rate 331
Editing and Customizing Repayment Schedules 335
Manually Capture Notifications 357
Adding Repayment Notifications 357
Adding Repayment Notification Details 360
Approving Manual Repayment Notifications 361
Working with Documents 362
Adding Contract Documents 363

Treatment of Non Working Days for Schedule 365
Working with Limits 366
Creating New Versions of Existing Loan Contracts 369
Possible Changes on New Loan Contract Versions 371
Viewing a Contract's History 371
Viewing a Contract's Accounting Entries 372
Changing the Borrower of a Loan 375
Current Accounts 377
Current Account Contract Life Cycle and States 378
Contract Versioning 380
Changing Contract Statuses 380
Creating a Current Account 382
Approving a Current Account 400
Automated Actions After Contract Approval 402
Rejecting a Current Account 403
Working with Covenants 404
Adding & Activating Covenants 405
Reviewing Covenants 406
Working with Participants 408
Adding Participants 409
Blocking Participants 410
Working with Contract Classification 410
Adding Classifications to a Contract 411
Applying Fees and Commissions 412
Adding Fees 414
Working with Documents 415
Adding Contract Documents 416
Performing Transactions on Current Accounts 418
Processing Overdraft Repayments 431
Manually Capture Notifications 452

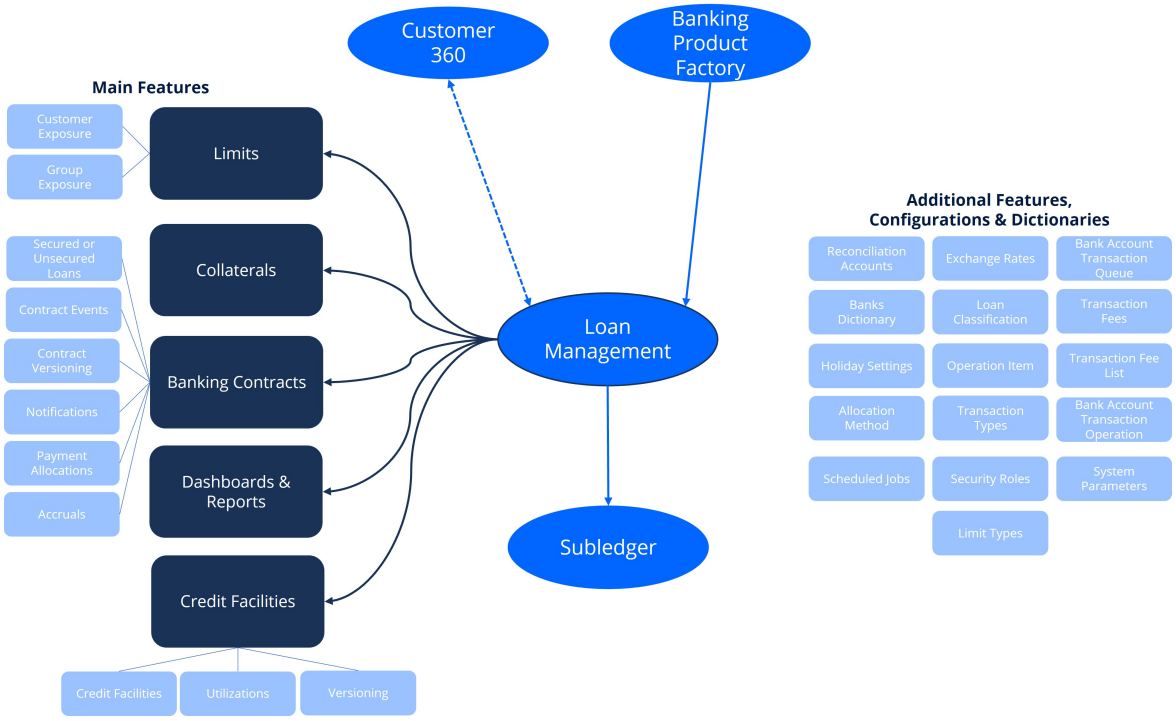
Adding Repayment Notifications 452
Adding Repayment Notification Details 455
Approving Manual Repayment Notifications 456
Closing a Current Account 457
Creating New Versions of Existing Current Account Contracts 462
Possible Changes on New Current Account with Overdraft Contract Versions 464
Viewing a Contract's History 464
Viewing a Contract's Accounting Entries 465
Credit Facilities 469
Credit Facility Implementation Notes 469
Business Logic 469
Managing Credit Facilities 470
Credit Facility Life Cycle and States 471
Credit Facility Versioning 472
Credit Facility Life Cycle 472
Changing Credit Facility Statuses 473
Creating Credit Facilities 475
Sending Credit Facilities for Approval 485
Approving Credit Facilities 485
Adding Utilizations to Credit Facilities 486
Approving Utilization Requests 490
Managing Credit Facility Utilization Details 491
Creating New Versions of Existing Credit Facilities 496
Viewing a Credit Facility's History 497
Dashboards and Reports 498

Overview

FintechOS **Loan Management** aims to help banks and/ or financial institutions with the management of records and processes during the life of the business relation with a customer, may it be on lending with underlying limits and collaterals, deposits, minimum current accounts capabilities, or credit facilities. Its automated processes scheduled to happen during close of day or start of day calculate cost elements and keep up the correct figures driven by the contracts inserted.

All the features in Loan Management are built using the capabilities of **FintechOS Studio**, and you can access its menus and dashboards when logged in **FintechOS Portal**.

The diagram below exhibits the main features of Loan Management, along with a series of configurations and dictionaries used to automate the complex banking processes performed by the system. Loan Management uses the banking products records defined within **Banking Product Factory**, and the customer records managed by **Customer 360**. Loan Management records are further used by **Subledger** to generate ledger entries.



Banking Product Factory

This is a powerful automation processor accessible in the FintechOS Studio that builds the products to be used in a customer journey, configures the interest, commissions and the life cycle of a product. Those products are later introduced into a customer journey or, when used in conjunction with Loan Management, they associate the products with transaction types. For more information, see Banking Product Factory.

Customer 360

Customer 360 is the central hub for collecting, aggregating, and processing banking customers' data for customers representing legal entities or individuals.

Subledger

Based on the transactions performed in Loan Management, Subledger logs, along with a company's financial transactions, details that enable the system to build ledger entries. For more information, see Subledger.

Configurations for Loan Management

Before starting to work with your Loan Management solution, there are a series of configurations that your implementation team should perform.

IMPORTANT!
FintechOS offers you a **Utilities** package for banking solutions, which you can use for testing purposes on your pre-production environments. This package helps you test the FintechOS banking solutions during the implementation phase of your project, using sample data. As such, you'd benefit from its content if you are a developer, a tester, a digital consultant, or a partner who works on implementing one of the FintechOS banking solutions.

This page contains topics that explain how Loan Management is configured to work and topics that assist you in configuring your Loan Management system.

Getting Started with Loan Management **12**
External Request Processing **17**
Configuring the Async Engine 17
Managing Unprocessed Requests 37
Holiday **40**
Creating Holiday Records 41
Allocation Method **42**
Creating Allocation Methods 43
Exchange Rate **45**
Creating Exchange Rates 46
Exchange Rate Type **47**
Creating Exchange Rates Types 48
Loan Classification **49**
Creating Loan Classification Records 51

Loan Periodicity **52**
Creating Loan Periodicity Records 53
Operation Item **54**
Creating Operation Item Records 56
Loan Management System Parameters **58**
Transaction Types Used in Loan Management **80**
Bank Account Transaction Queue **89**
Bank Account Transaction Configurations **91**
Transaction Types Covered Through Bank Account Transaction Operation Types in Loan Management 92
Loan Contract Approval Process Flow 95
Transaction Operation Type 98
Transaction Fee 102
Transaction Fee List 104
Jobs **106**
Security Roles for Loan Management **115**
Predefined Loan Management Security Roles 116
Banks **152**
Creating Bank Records 152
Creating External Bank Accounts 154
Sales Channels **155**
Creating Sales Channel Records 156
Reconciliation Accounts **157**
Creating Reconciliation Accounts 158
Reconciliation Account Settings 161

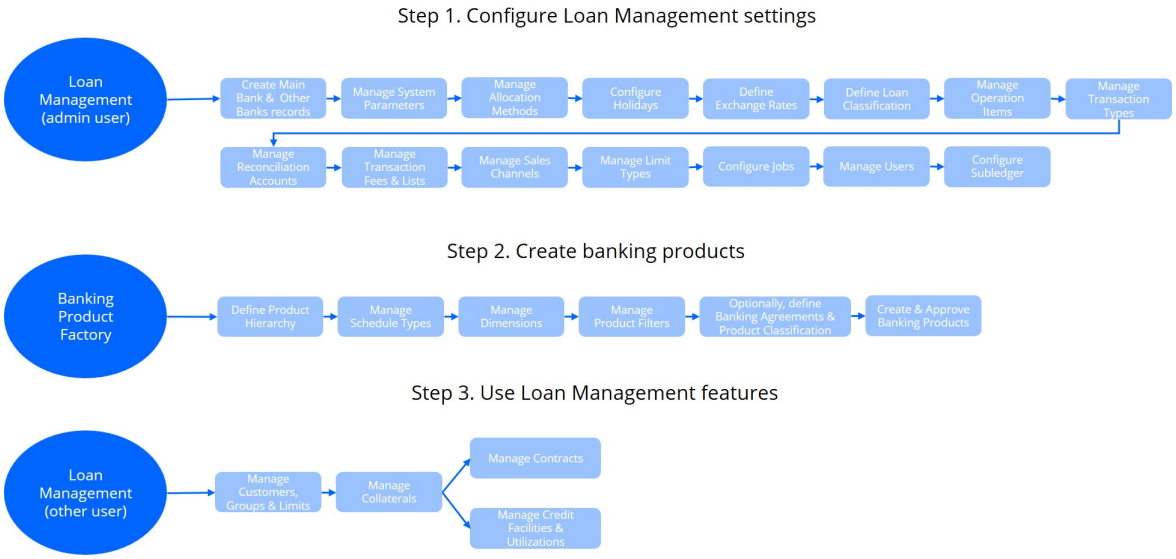
NOTE
If you need to re-index your solution to have a starting point for maintenance scripts within your implementation project, you can run the `92.fix.Indexes.Performance.sql` script that comes with the Loan Management packages. Due to high performance impact, the script is marked not to

run by default during installation, but you can check and start it manually at a later time.

Getting Started with Loan Management

After performing the installation process, Loan Management needs a series of configurations to be put in place before being ready for production. For example, it needs records for the main bank that uses the system, reconciliation accounts to be used for transactions, exchange rates information, holidays to be declared, specific settings for the Loan Management system parameters that indicate how the system should handle different situations or perform specific calculations, and so on.

This page is a step-by-step guide about what you have to set up, with links to detailed instructional pages related to each specific step. Follow through these steps after installing Loan Management and before declaring it ready for production.



Step 1. Configure Loan Management settings

Follow these steps to configure the settings needed by Loan Management:

1. Log into **FintechOS Portal** using a **user with administrator rights**.
   The user credentials for an administrator user are received from your FintechOS contact person. Insert the user name and the password associated with it to log into the FintechOS Portal.
2. Define your **main bank**.
   You must create a bank record to be used by Loan Management as the main financial institution. Use the **Core Banking Operational > Bank** menu and make sure the record has the `Main Bank` checkbox selected.
3. Manage your **Loan Management system parameters**.
   The system parameters used by Loan Management determine the behavior of all the contracts, transactions, limits, and other parts that make up your Loan Management system. See here the list of system parameters used by Loan Management, along with their descriptions.
4. Manage **allocation methods**.
   Loan Management uses allocation methods to determine the order in which credit items are prioritized when repaying loans, credit accounts, etc. Read here how to create and manage allocation methods, using the **Allocation Method** menu.
5. Enter **holidays** for the desired countries.
   Public holidays for each country are used in the product definition for the calendar years over which your financial institution's current business is spread. Use the **Holiday** menu to create holiday records specific to your financial institution's needs, as described here.
6. Define **exchange rate types** and enter **exchange rate** records.
   Exchange rates represent the value between the currencies of two countries on a given date. These rates are free-floating or fixed. Add exchange rate types using the **Exchange Rate Type** menu if you need to differentiate between exchange rates based on the currency market or business area. Read more information about creating exchange rate records and managing exchange rate types.
7. Define your **loan classification**.
   Financial institutions classify their existing loan contracts based upon the days past due (DPD), the number of days passed since repayment

due date without fully repaying the due amount. Since the provisions have an impact on the financial results of the bank, this is again driven by regulations and may vary in time or depending on country or region. Create loan classification records as described here.

8. Manage **operation items** specific for your business.
   Operation items are those items that relate to a bank's core business, such as all types of fees, commissions, principals, interests, advances, or penalty calculations. They can also be considered as balance types that add up to a certain deal or used in tracing what happened on a particular deal. Find here examples and information on creating operation item records.

9. Manage **transaction types**.
   Any transfer of funds between two bank accounts is recorded as a transaction. The transaction types are predefined for usage within Loan Management processes. Read here about different types of transactions used in Loan Management.

10. Define other **banks or financial institutions** with whom your main bank has business relations and **add external accounts**.
    Create bank records for the banking institutions with whom your main bank collaborates, using the **Loan Management Operational > Bank** menu. Add external bank accounts within these banks.

11. Define **reconciliation accounts** and default settings for the reconciliation accounts.
    Reconciliation is an accounting process that compares two sets of records to check that figures are correct and in agreement. Learn here how to manage the reconciliation accounts records. Read about setting up which reconciliation account for a specified currency should be used by Loan Management within a given period.

12. Manage **transaction fees and lists**.
    You can define different fees to be applied to bank account transactions. Using fee lists, you can attach fees with specified values to each bank account transaction operation type. When a transaction operation type is selected on a bank account transaction, Loan Management identifies the fee list and fee values and applies them

considering the current date of the transaction.

13. Define **sales channels** for your contracts.
You can create contracts through different channels:, such as the dedicated Loan Management menus in FintechOS Portal, API integration calls, or various customer journeys implemented within FintechOS accelerators. Manage the sales channels records, so that you can apply different pricing or to allow the selling a product on a specific channel.

14. Manage **limit types** for role-based limits.
You can create new limit types that are based on roles associated to contract participants specific to your business, and use them throughout Loan Management with all the functionality of any other default limit type.

15. Make sure a **JobServer** is up and running.
To perform the processes within Loan Management, a JobServer must be up and running. Manage server jobs as described here. Learn here about Loan Management's scheduled jobs.

16. Create **users** and allocate them appropriate **security roles**.
For appropriate access and rights within Loan Management, create users and allocate them appropriate Loan Management security roles. The following pages contain information related to this topic: Adding Users, Editing Users, and Security Roles.

17. Configure **Subledger**. The Subledger Add-On, installed over the Loan Management package, manages the accounting information needed for ledger reports and other financial statements. Access the Configurations page for information about the settings you need to perform.

Step 2. Create banking products

Use **Banking Product Factory** to create the banking products that your financial institution wants to offer to their customers via contracts. Loan Management integrates directly with Banking Product Factory, thus all the banking products with `Active` status are automatically available for you to use in Loan Management, when you create contracts. Before actually

creating the banking products, you should perform a series of configurations within Banking Product Factory.

> **NOTE**
> Once you install **Loan Management**, **Banking Product Factory** enhances with another section, the Lean Core Settings tab at the banking product level, that gives additional configuration ability to the banking product.

Step 3. Use Loan Management features

After configuring the Loan Management settings and creating banking products, you are ready to use the main features available in Loan Management, as described in the following steps:

1. Log into **FintechOS Portal** using a **user with an associated Loan Management security role**.
   Insert the user name and the password associated with it to log into the FintechOS Portal.
2. Add **customers** and **groups**, then set customer **limits**.
   Financial institutions deal with customers, either individuals or legal entities. Customers may be part of groups. In Loan Management, create customer records and attach them to groups. Monitor your financial institution's exposure for credit related activities by setting up limits for your customers. You can manage limits through a series of menus and reports available in Loan Management.
3. Register **collaterals**.
   Collateral management is the method of granting, verifying and managing collateral transactions in order to reduce credit risk in unsecured financial transactions. It is an essential and integral part of any financial institution's risk and regulatory compliance framework. Manage collateral records in Loan Management, as described here.
4. Manage **contracts**.
   Any agreement between a financial institution and a customer regarding the usage of a banking product is documented legally with a contract. In Loan Management, you can create contracts for your

financial institution's customers based on approvals. Read about contracts in the Contracts page.

5. Manage **credit facilities**.
   Credit facilities are groupings of multiple credit products that a customer has arranged with a bank under a single credit limit. Read here how to manage credit facility records.

External Request Processing

To improve the platform's overall performance through parallel processing, Loan Management sends by default the requests generated for processing outside of the platform, instead of being processed within the platform, through JobServer. Thus, Loan Management processes the requests in a parallel asynchronous way, through Async Engine.

Configuring the Async Engine

The configuration for Loan Management-Async Engine is as follows:

1. The **UseExternalProcessing** Loan Management system parameter must be set to `True` for sending the requests for processing outside of the platform. Otherwise, Loan Management processes the requests through JobServer.
2. For asynchronous transaction processing, the **Real Time Process** attribute on the desired transaction type must be set to **False**.
3. For asynchronous limit recalculation, the **AsyncLimitRecalculation** Loan Management system parameter must be set to **True**.
4. The **RepaymentNotificationFlowName**, **BankAccountTransactionFlowName**, **LoanApprovalFlowName** Loan Management system parameters must be used when a request for the specific flow is triggered.

5. The **Async Engine** web API client library has to be loaded on the platform using the Async Engine swagger file.
6. The Vault settings for the Studio, Portal, and JobServer components must contain a key named **AsyncEngineURL** with the value of the Async Engine's URL.

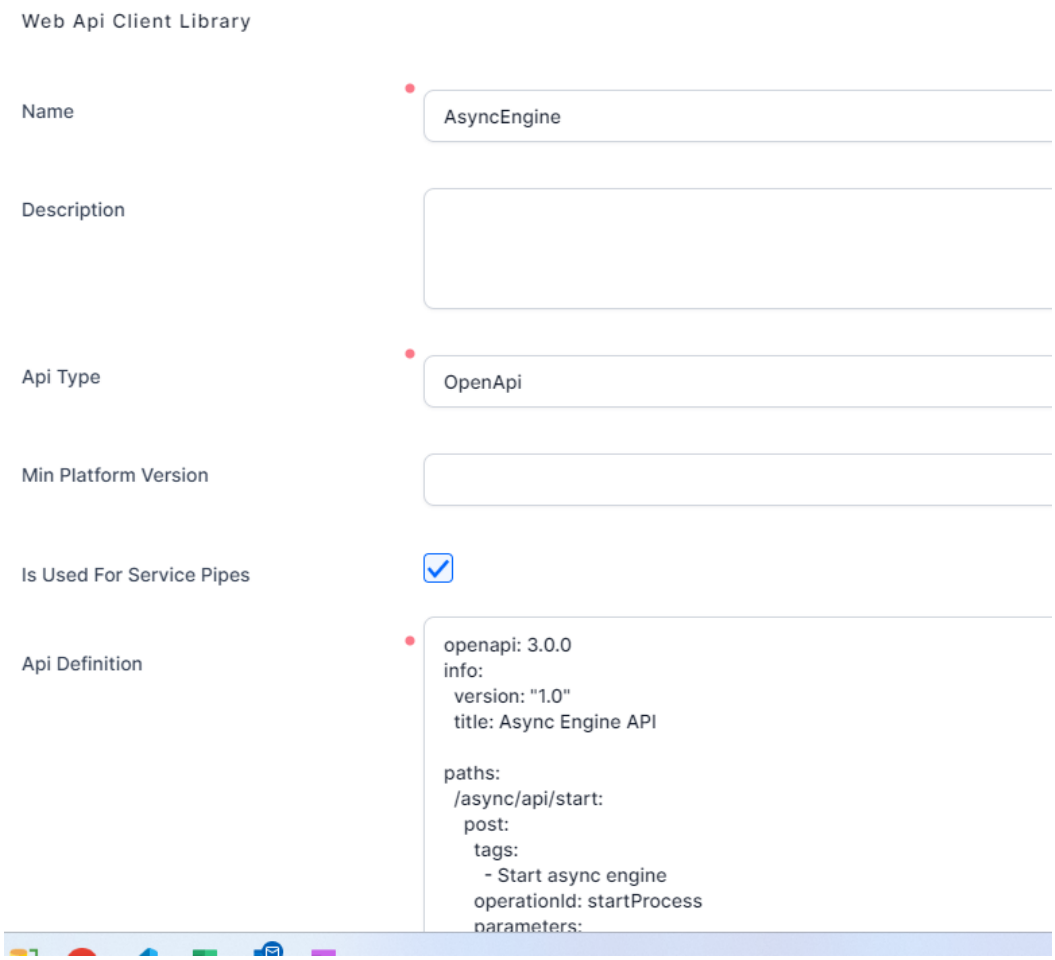
Read the following section for more detailed information:

Calling the External Request Processor

The asynchronous processing configuration is kept in the Vault.

> **IMPORTANT!**
> Configure the external request processor app service to use the `async-engine` folder in Vault.

During implementation, create a web API client library to enable Loan Management to call these configurations, making sure the library uses the latest swagger file found at `<server_address>/services/specifications/swagger-ui`.



IMPORTANT!
Set the `Is Used for service pipes` attribute to `True`.

Send the `AsyncEngineURL` parameter, stored in Vault, with the URL of the external request processor when importing the web api client library in the Loan Management context.

When the external request processor calls the Portal, the portal user client must be part of the Role defined in `rbac.config` vault parameter.

IMPORTANT!
Make sure you create a user for running the OpenAPI pipelines. The user must be part of the `Registered users` group.

While calling the external request processor using the `startProcess` operation, send the following parameters:

- `flowName` - mandatory;
- `correlationId` - optional, automatically generated if missing;
- `payload` - mandatory.

Configuring the Vault Parameters

You must configure a series of parameters so that your external request processor can receive and forward requests.

Add the `AsyncEngineURL` Vault parameter(s) to Loan Management Portal/ Studio/ JobServer. This is used when importing the web API client library. Please use the keys for single or for multiple instances, as described below:

AsyncEngineURL - for single instance configuration

```
"AsyncEngineURL_CoreBanking": "https://app-<async-engine-
instance>-<project_name>-<env>.azurewebsites.net",
"AsyncEngineURL_DistributionManagement": "https://app-
<async-engine-instance>-<project_name>-
<env>.azurewebsites.net",
"AsyncEngineURL_Origination": "https://app-<async-engine-
instance>-<project_name>-<env>.azurewebsites.net"
```

AsyncEngineURL - for multiple instances configuration

```
"AsyncEngineURL_CoreBanking": "https://app-<async-engine-
instance_LoanManagement>-<project_name>-
<env>.azurewebsites.net",
"AsyncEngineURL_DistributionManagement": "https://app-
<async-engine-instance_Distribution>-<project_name>-
<env>.azurewebsites.net",
"AsyncEngineURL_Origination": "https://app-<async-engine-
instance_Origination>-<project_name>-
<env>.azurewebsites.net"
```

NOTE
When using multiple Async Engine instances for the segregation of

processes, the parameters `flow.config` and `queue.config` must be split according to your configuration.

Single Instance Configuration

The following parameters are used in the external request processor's Vault parameters in a single instance configuration:

threadpool.config

```
{
  "poolSize": 10,
  "maxPoolSize": "1000",
  "maxQueueSize": "100"
}
```

azure.storage.config

```
{
        "accountName": "<storage account name>",
        "accountKey": "<storage acount key>",
        "suffix": "core.windows.net"
}
```

openid.config

```
{
    "use-resource-role-mappings":  "false",
    "autodetect-bearer-only":  "true",
    "ssl-required":  "external",
    "auth-server-url":  "<auth server url>",
    "principal-attribute":  "preferred_username",
    "resource":  "<client used for authentication>",
    "credentials":  {
                        "secret":  "<client secret>"
                    },
    "realm":  "<realm>"
}
```

openapi.url

```
https://<open api
url>.azurewebsites.net/ftosapi/automation-
processors/actions/
```

openapi.username
<username>

openapi.password
<password>

flow.config

```
[
  {
      "flowName": "Flow_CB_LoanApproval_AsyncEngine",
      "queueMappings": [
          {
              "sourceQueueName": "ftos-cb-loan-
approval",
              "endpointName": "FTOS_CB_
ProcessLoanApproval_AsyncEngine",
              "initial": true
          }
      ]
  },
  {
      "flowName": "Flow_CB_RepaymentNotification_
AsyncEngine",
      "queueMappings": [
          {
              "sourceQueueName": "ftos-cb-repayment-
notification",
              "endpointName": "FTOS_CB_
ProcessRepaymentNotification_AsyncEngine",
              "initial": true
          }
      ]
  },
  {
      "flowName": "Flow_CB_BankAcountTransaction_
AsyncEngine",
      "queueMappings": [
          {
              "sourceQueueName": "ftos-cb-
bankaccount-transaction",
              "endpointName": "FTOS_CB_
ProcessBankAccountTransaction_AsyncEngine",
              "initial": true
```

```
            }
        ]
    },
    {
        "flowName": "Flow_TPM_GenerateInvoiceDetails_
AsyncEngine",
        "queueMappings": [
            {
                "sourceQueueName": "ftos-tpm-find-
agrements",
                "endpointName": "FTOS_TPM_
FindMatchingAgreements_AsyncEngine",
                "targetQueueName": "ftos-tpm-build-
invoice-details",
                "initial": true
            },
            {
                "sourceQueueName": "ftos-tpm-build-
invoice-details",
                "endpointName": "FTOS_TPM_
BuildInvoiceDetails_AsyncEngine",
                 "targetQueueName": "ftos-tpm-
calculate-pricings"
            },
            {
                "sourceQueueName": "ftos-tpm-calculate-
pricings",
                "endpointName": "FTOS_TPM_
CalculatePricings_AsyncEngine",
                "targetQueueName": "ftos-tpm-generate-
invoice-details"
            },
            {
                "sourceQueueName": "ftos-tpm-generate-
invoice-details",
                "endpointName": "FTOS_TPM_
GenerateInvoiceDetails_AsyncEngine"
            }
        ]
    },
    {
        "flowName": "Flow_TPM_GenerateInvoices_
AsyncEngine",
        "queueMappings": [
```

```
            {
                "sourceQueueName": "ftos-tpm-generate-
invoice",
                "endpointName": "FTOS_TPM_
GenerateInvoice_AsyncEngine",
                "initial": true
            }
        ]
    },
    {
      "flowName": "Flow_TPM_PayInvoices_AsyncEngine",
      "queueMappings": [
        {
          "sourceQueueName": "ftos-tpm-pay-invoice",
          "endpointName": "FTOS_TPM_PayInvoice_
AsyncEngine",
          "initial": true,
          "maxThreads": 10
        }
      ]
    },
    {
      "flowName": "Flow_CB_LoanSubmission_AsyncEngine",
      "queueMappings": [
        {
          "sourceQueueName": "ftos-cb-customer",
          "endpointName": "FTOS_CB_LoanSubmission_
AsyncEngine",
          "targetQueueName": "ftos-cb-customerlimit-
create",
          "maxThreads": 10,
          "initial": true
        },
        {
          "sourceQueueName": "ftos-cb-customerlimit-
create",
          "endpointName": "FTOS_CB_LoanSubmission_
AsyncEngine",
          "targetQueueName": "ftos-cb-customerlimit-
approve",
          "maxThreads": 10
        },
        {
```

```
        "sourceQueueName": "ftos-cb-customerlimit-
approve",
        "endpointName": "FTOS_CB_LoanSubmission_
AsyncEngine",
        "targetQueueName": "ftos-cb-
settlementaccount-create",
        "maxThreads": 10
      },
      {
        "sourceQueueName": "ftos-cb-
settlementaccount-create",
        "endpointName": "FTOS_CB_LoanSubmission_
AsyncEngine",
        "targetQueueName": "ftos-cb-
settlementaccount-approve",
        "maxThreads": 10
      },
     {
        "sourceQueueName": "ftos-cb-
settlementaccount-approve",
        "endpointName": "FTOS_CB_LoanSubmission_
AsyncEngine",
        "targetQueueName": "ftos-cb-topup-create",
        "maxThreads": 10
      },
      {
        "sourceQueueName": "ftos-cb-topup-create",
        "endpointName": "FTOS_CB_LoanSubmission_
AsyncEngine",
        "targetQueueName": "ftos-cb-topup-approve",
        "maxThreads": 10
      },
      {
        "sourceQueueName": "ftos-cb-topup-approve",
        "endpointName": "FTOS_CB_LoanSubmission_
AsyncEngine",
        "targetQueueName": "ftos-cb-loancontract-
create",
        "maxThreads": 10
      },
      {
        "sourceQueueName": "ftos-cb-loancontract-
create",
```

```
        "endpointName": "FTOS_CB_LoanSubmission_
AsyncEngine",
        "targetQueueName": "ftos-cb-loancontract-
approve",
        "maxThreads": 10
      },
      {
        "sourceQueueName": "ftos-cb-loancontract-
approve",
        "endpointName": "FTOS_CB_LoanSubmission_
AsyncEngine",
        "maxThreads": 10
      }
    ]
  }

]
```

queue.config

```
[
        {
           "name": "ftos-cb-loan-approval",
           "maxMessages": 10,
           "timeToLive": 1
        },
        {
           "name": "ftos-cb-repayment-notification",
           "maxMessages": 10,
           "timeToLive": 1
        },
        {
           "name": "ftos-cb-bankaccount-transaction",
           "maxMessages": 10,
           "timeToLive": 1
        },
        {
           "name": "ftos-tpm-find-agrements",
           "maxMessages": 10,
           "timeToLive": 1
        },
        {
           "name": "ftos-tpm-build-invoice-details",
           "maxMessages": 10,
```

```
            "timeToLive": 1
         },
         {
            "name": "ftos-tpm-calculate-pricings",
            "maxMessages": 10,
            "timeToLive": 1
         },
         {
            "name": "ftos-tpm-generate-invoice-
details",
            "maxMessages": 10,
            "timeToLive": 1
         },
         {
            "name": "ftos-tpm-generate-invoice",
            "maxMessages": 10,
            "timeToLive": 1
         },
         {
      "name": "ftos-tpm-pay-invoice",
      "maxMessages": 10,
      "timeToLive": 1
   },
   {
      "name": "ftos-cb-customer",
      "maxMessages": 10,
      "timeToLive": 1
   },
   {
      "name": "ftos-cb-customerlimit-create",
      "maxMessages": 10,
      "timeToLive": 1
   },
  {
      "name": "ftos-cb-customerlimit-approve",
      "maxMessages": 10,
      "timeToLive": 1
   },
   {
      "name": "ftos-cb-settlementaccount-create",
      "maxMessages": 10,
      "timeToLive": 1
   },
```

```
    {
       "name": "ftos-cb-settlementaccount-approve",
       "maxMessages": 10,
       "timeToLive": 1
    },
    {
       "name": "ftos-cb-topup-create",
       "maxMessages": 10,
       "timeToLive": 1
    },
   {
       "name": "ftos-cb-topup-approve",
       "maxMessages": 10,
       "timeToLive": 1
    },
    {
       "name": "ftos-cb-loancontract-create",
       "maxMessages": 10,
       "timeToLive": 1
    },
    {
       "name": "ftos-cb-loancontract-approve",
       "maxMessages": 10,
       "timeToLive": 1
    }
    ]
```

rbac.config

```
 {
     "apiMappings": [
       {
         "url": "/api",
         "roles": [
           "<user profile created and used for service
 account roles> "
         ]
       }
     ],
     "applicationMappings": [
       {
         "url": "/actuator",
         "roles": [
```

```
            "<user profile created and used for service
  account roles>"
          ]
        }
      ],
      "swaggerMappings": [
        {
          "url": "/specifications",
          "roles": [
            "<user profile created and used for service
  account roles>"
          ]
        }
      ]
  }
```

redelivery.config

```
  {
      "default": {
          "count": 2,
          "delay": 2
      },
      "additional": [
          {
              "httpStatus": 502,
              "count": 2,
              "delay": 1
          }
      ]
  }
```

table.config

```
  {
      "statusTable": "statusTableAsyncEngine",
      "dlqTable": "dlqTableAsyncEngine"
  }
```

http.connection.config

```
  {
      "requestTimeout": "180000",
      "connectionTimeout": "120000",
      "socketTimeout": "230000",
```

```
    "totalConnections": "1000000",
    "routeConnections": "500000",
    "connectionTimeToLive": "1000"
}
```

Multiple Instances Configuration

The following parameters are used in the external request processor's Vault parameters in multiple instances configuration:

threadpool.config

```
{
  "poolSize": 10,
  "maxPoolSize": "1000",
  "maxQueueSize": "100"
}
```

azure.storage.config

```
{
       "accountName": "<storage account name>",
       "accountKey": "<storage acount key>",
       "suffix": "core.windows.net"
}
```

openid.config

```
{
    "use-resource-role-mappings":  "false",
    "autodetect-bearer-only":  "true",
    "ssl-required":  "external",
    "auth-server-url":  "<auth server url>",
    "principal-attribute":  "preferred_username",
    "resource":  "<client used for authentication>",
    "credentials":  {
                        "secret":  "<client secret>"
                    },
    "realm":  "<realm>"
}
```

openapi.url

```
https://<open api
url>.azurewebsites.net/ftosapi/automation-
processors/actions/
```

openapi.username

```
<username>
```

openapi.password

```
<password>
```

flow.config - for the Loan Management processes

```
[
  {
      "flowName": "Flow_CB_LoanApproval_AsyncEngine",
      "queueMappings": [
          {
              "sourceQueueName": "ftos-cb-loan-
approval",
              "endpointName": "FTOS_CB_
ProcessLoanApproval_AsyncEngine",
              "initial": true
          }
      ]
  },
  {
      "flowName": "Flow_CB_RepaymentNotification_
AsyncEngine",
      "queueMappings": [
          {
              "sourceQueueName": "ftos-cb-repayment-
notification",
              "endpointName": "FTOS_CB_
ProcessRepaymentNotification_AsyncEngine",
              "initial": true
          }
      ]
  },
  {
      "flowName": "Flow_CB_BankAcountTransaction_
AsyncEngine",
      "queueMappings": [
```

```
            {
                "sourceQueueName": "ftos-cb-
bankaccount-transaction",
                "endpointName": "FTOS_CB_
ProcessBankAccountTransaction_AsyncEngine",
                "initial": true
            }
        ]
    }
]
```

flow.config - for the Origination processes

```
[
  {
    "flowName": "Flow_CB_LoanSubmission_AsyncEngine",
    "queueMappings": [
      {
        "sourceQueueName": "ftos-cb-customer",
        "endpointName": "FTOS_CB_LoanSubmission_
AsyncEngine",
        "targetQueueName": "ftos-cb-customerlimit-
create",
        "maxThreads": 10,
        "initial": true
      },
      {
        "sourceQueueName": "ftos-cb-customerlimit-
create",
        "endpointName": "FTOS_CB_LoanSubmission_
AsyncEngine",
        "targetQueueName": "ftos-cb-customerlimit-
approve",
        "maxThreads": 10
      },
      {
        "sourceQueueName": "ftos-cb-customerlimit-
approve",
        "endpointName": "FTOS_CB_LoanSubmission_
AsyncEngine",
        "targetQueueName": "ftos-cb-
settlementaccount-create",
        "maxThreads": 10
      },
```

```
      {
        "sourceQueueName": "ftos-cb-
settlementaccount-create",
        "endpointName": "FTOS_CB_LoanSubmission_
AsyncEngine",
        "targetQueueName": "ftos-cb-
settlementaccount-approve",
        "maxThreads": 10
      },
     {
        "sourceQueueName": "ftos-cb-
settlementaccount-approve",
        "endpointName": "FTOS_CB_LoanSubmission_
AsyncEngine",
        "targetQueueName": "ftos-cb-topup-create",
        "maxThreads": 10
      },
      {
        "sourceQueueName": "ftos-cb-topup-create",
        "endpointName": "FTOS_CB_LoanSubmission_
AsyncEngine",
        "targetQueueName": "ftos-cb-topup-approve",
        "maxThreads": 10
      },
      {
        "sourceQueueName": "ftos-cb-topup-approve",
        "endpointName": "FTOS_CB_LoanSubmission_
AsyncEngine",
        "targetQueueName": "ftos-cb-loancontract-
create",
        "maxThreads": 10
      },
      {
        "sourceQueueName": "ftos-cb-loancontract-
create",
        "endpointName": "FTOS_CB_LoanSubmission_
AsyncEngine",
        "targetQueueName": "ftos-cb-loancontract-
approve",
        "maxThreads": 10
      },
      {
        "sourceQueueName": "ftos-cb-loancontract-
approve",
```

```
        "endpointName": "FTOS_CB_LoanSubmission_
 AsyncEngine",
        "maxThreads": 10
      }
    ]
  }
]
```

queue.config - for the Loan Management processes

```
[
        {
           "name": "ftos-cb-loan-approval",
           "maxMessages": 10,
           "timeToLive": 1
        },
        {
           "name": "ftos-cb-repayment-notification",
           "maxMessages": 10,
           "timeToLive": 1
        },
        {
           "name": "ftos-cb-bankaccount-transaction",
           "maxMessages": 10,
           "timeToLive": 1
        }
  ]
```

queue.config - for the Origination processes

```
[
  {
     "name": "ftos-cb-customer",
     "maxMessages": 10,
     "timeToLive": 1
  },
  {
     "name": "ftos-cb-customerlimit-create",
     "maxMessages": 10,
     "timeToLive": 1
  },
 {
     "name": "ftos-cb-customerlimit-approve",
     "maxMessages": 10,
```

```
      "timeToLive": 1
    },
    {
      "name": "ftos-cb-settlementaccount-create",
      "maxMessages": 10,
      "timeToLive": 1
    },
    {
      "name": "ftos-cb-settlementaccount-approve",
      "maxMessages": 10,
      "timeToLive": 1
    },
    {
      "name": "ftos-cb-topup-create",
      "maxMessages": 10,
      "timeToLive": 1
    },
   {
      "name": "ftos-cb-topup-approve",
      "maxMessages": 10,
      "timeToLive": 1
    },
    {
      "name": "ftos-cb-loancontract-create",
      "maxMessages": 10,
      "timeToLive": 1
    },
    {
      "name": "ftos-cb-loancontract-approve",
      "maxMessages": 10,
      "timeToLive": 1
    }
    ]
```

rbac.config

```
  {
      "apiMappings": [
        {
          "url": "/api",
          "roles": [
            "<user profile created and used for service
  account roles> "
          ]
```

```
      }
    ],
    "applicationMappings": [
      {
        "url": "/actuator",
        "roles": [
          "<user profile created and used for service
account roles>"
        ]
      }
    ],
    "swaggerMappings": [
      {
        "url": "/specifications",
        "roles": [
          "<user profile created and used for service
account roles>"
        ]
      }
    ]
}
```

redelivery.config

```
{
    "default": {
        "count": 2,
        "delay": 2
    },
    "additional": [
        {
            "httpStatus": 502,
            "count": 2,
            "delay": 1
        }
    ]
}
```

table.config

```
{
    "statusTable": "statusTableAsyncEngine",
    "dlqTable": "dlqTableAsyncEngine"
}
```

http.connection.config

```
{
    "requestTimeout": "180000",
    "connectionTimeout": "120000",
    "socketTimeout": "230000",
    "totalConnections": "1000000",
    "routeConnections": "500000",
    "connectionTimeToLive": "1000"
}
```

Managing Unprocessed Requests

If a request is not processed by the Async Engine regardless of the reason, you can view a message in the dedicated **DLQ dashboard** in your FintechOS Portal. For accessing the dashboard, perform the needed configurations during the implementation. Send the request for reprocessing by the Async Engine from the same dashboard, simply clicking a button next to that message.

Configuring the Messages and Access Rights for the DLQ Dashboard

The **DLQ Messages** (dead letter queue) dashboard stores messages about the requests that the Async Engine could not process for any reason. To view the messages and to send the requests back to processing, make sure you perform the following configurations:

1. In FintechOS Studio, select the **Admin > System Parameters** menu.
2. On the newly displayed **System Parameters list** page, find and double-click the **DLQInterfaceQueuesToBeShown** system parameter to open it for editing. This parameter represents the list of queues from which the messages are displayed in the **DLQ Messages** dashboard in FintechOS Portal.

3. On the **Edit System Parameter** page, edit the parameter value to display the messages from the desired flows. Enter the names of the flows one by one, separated by commas and with no spaces between them. For example:

   ```
   ftos-cb-loan-approval,ftos-cb-repayment-notification,ftos-cb-bankaccount-transaction,ftos-tpm-find-agrements,ftos-tpm-build-invoice-details,ftos-tpm-calculate-pricings,ftos-tpm-generate-invoice-details,ftos-tpm-generate-invoice,ftos-tpm-pay-invoice,ftos-cb-customer,ftos-cb-customerlimit-create,ftos-cb-customerlimit-approve,ftos-cb-settlementaccount-create,ftos-cb-settlementaccount-approve,ftos-cb-topup-create,ftos-cb-topup-approve,ftos-cb-loancontract-create,ftos-cb-loancontract-approve
   ```

   The flows must be specified in Vault.

   > **NOTE**
   >
   > The order of messages displayed in the dashboard depends on the order of the flows you mention here, so make sure you enter the most desired flow name first and then add them according to your priorities.

4. Click the **Save and Close** button.
5. Select the **Security > System Users** menu.
6. On the newly displayed **System Users list** page, find and double-click each user who should have access to the **DLQ Messages** dashboard.
7. Edit the system users, assigning them the **Async Engine Admin** security role.
8. Click the **Save and Close** button.

Now you are ready to view the **DLQ Messages** dashboard and send any requests for reprocessing.

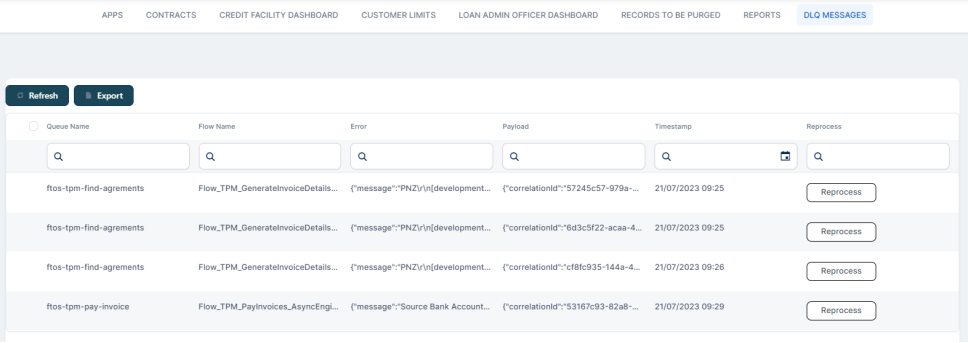
Viewing the DLQ Messages

Follow these steps to view the requests that the Async Engine could not process:

1. In FintechOS Portal's **Home** page, select the **DLQ Messages** dashboard.

> **NOTE**
>
> Your system user must have the **Async Engine Admin** security role to access the **DLQ Messages** dashboard.

2. View the messages in the list, if they exist. Each message contains the queue name, the flow name, the error message, the payload and the time stamp of the unprocessed request.



Reprocessing Failed Requests

You can send for processing each of the failed requests from the **DLQ Messages** dashboard one by one, as follows:

1. Click the **Reprocess** button in line with the desired request.
2. The request is sent back to the Async Engine for processing and an informative message is displayed on your screen.

Holiday

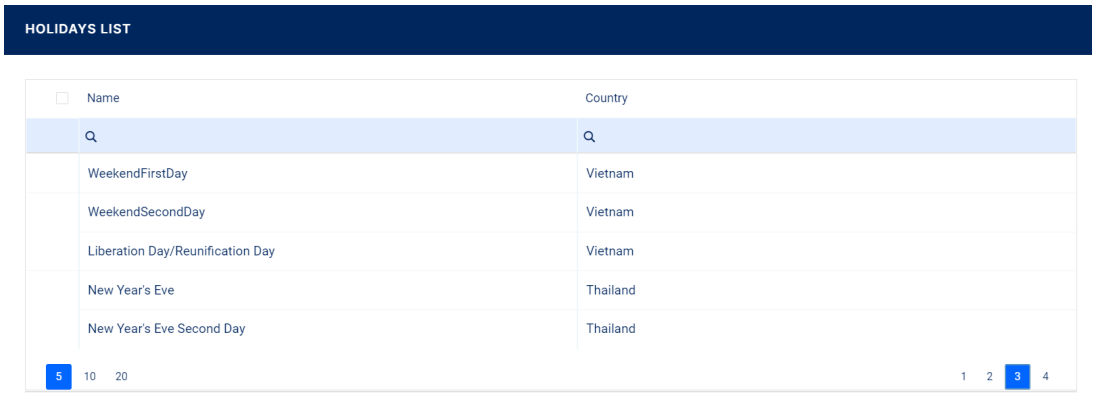
Financial institutions are usually closed and do not process payments or repayments for loans during holidays, hence the repayment schedules may be adapted to take the into consideration. In Loan Management, you can indicate the public holidays for the country used in the product definition, for the calendar years over which the financial institution's current business is spread.

Loan Management uses two methods for loan repayment processing when the holidays are taken into consideration in the schedule projections:

- `Shift forward`: the payment date is shifted to the first working day after the usual scheduled execution date.
- `Shift backward`: the payment date is shifted to the previous working day before the usual scheduled execution date.

In order to set up specific days when payments are not processed, follow these steps:

1. In FintechOS Portal, click the main menu icon and expand the **Admin Configuration** menu.
2. Click the **Holiday** menu item to open the **Holidays List** page.

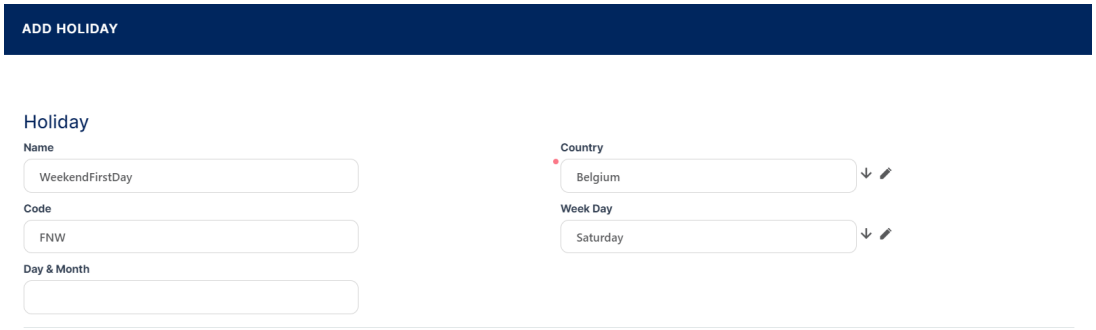


On the **Holidays List** page, you can add new holiday records or search, edit, and delete existing ones.

Creating Holiday Records

Follow these steps to create holiday records specific to your financial institution's country or organization:

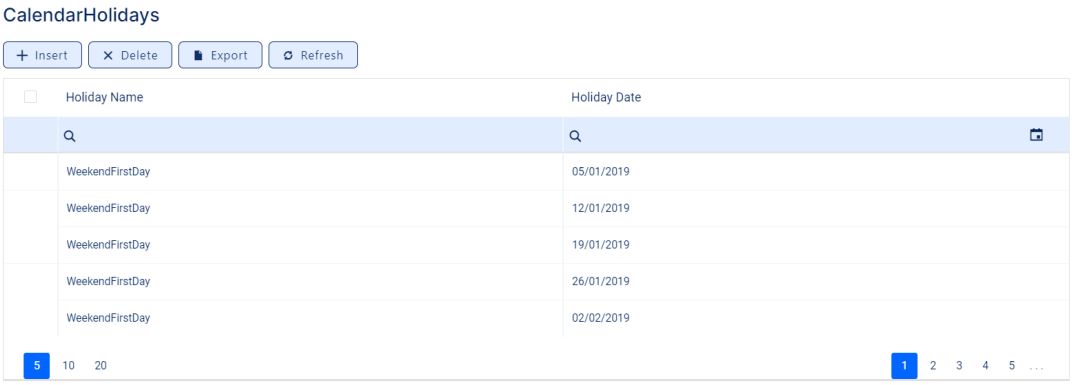
1. Click the **Insert** button on the top right side of the **Holiday List** page. The **Add Holiday** page is displayed.
2. Fill in the following fields:



- **Name** - Enter the name of the holiday.
- **Code** - Enter he code of the holiday.
- **Country** - Select the country in which the holiday rule applies to.
- **Week Day** - For repetitive holidays, select the weekday on which the holidays falls on.
- **Day and Month** - For holidays with a fixed date, enter the day and month, in the `dd.mm` format.

3. Click the **Save and Reload** button at the top right corner of the page. Loan Management populates the **Calendar Holidays** section with the exact dates of the

holidays.



You can insert, delete, or export the calendar holiday data.

Allocation Method

Allocation methods represent the order in which a financial institution proportionally allocates a portion of the payment to either fees, commissions, interest, and other credit items associated with the account, with the aim of closing the loan principal. FintechOS uses allocation methods to determine the order in which credit items are prioritized when repaying loans, credit accounts, and so on.

You can manage allocation methods though the **Allocation Method** menu, which stores information about the details and the banking products using those specific definitions in their setup. Follow these steps to manage such records:

1. In the FintechOS Portal, click the main menu icon and expand the **Admin Configuration** menu.
2. Click the **Allocation Method** menu item to open the **Allocation Method** page.



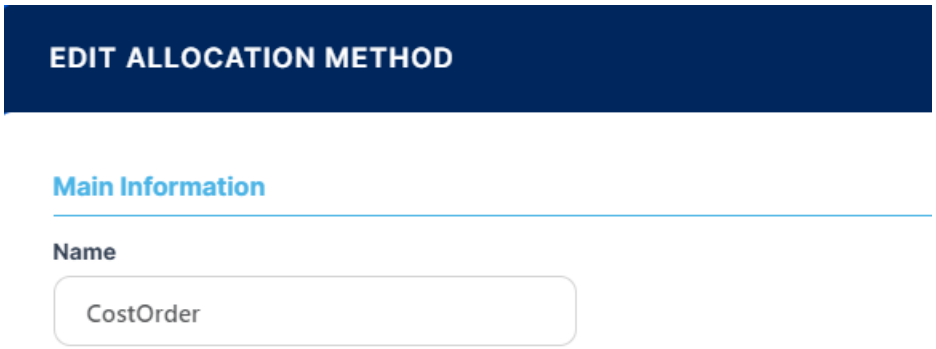
On the **Add Allocation Method** page, you can add new allocation methods or search, edit, and delete existing ones.

> **NOTE**
> The default allocation method used by Loan Management for manual repayment notifications that aren't linked to a contract is stored in the `ManualAllocationMethod` system parameter.

Creating Allocation Methods

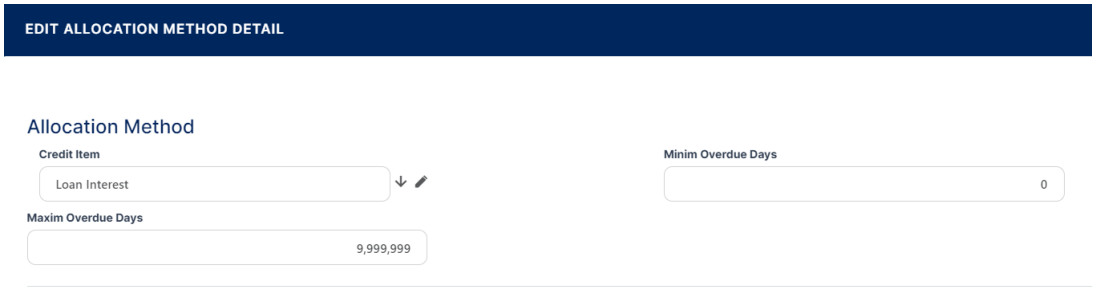
To create a new allocation method, follow these steps:

1. Click **Insert** on the **Allocation Method** page to ope the **Add Allocation Method** page.
2. Enter a **name** for the allocation method.



3. Click the **Save and Reload** button. The **Allocation Method Details** and the **Banking Products** sections are displayed.
4. In the newly displayed **Allocation Method Details** section, click **Insert** to open the **Add Allocation Method Details**.

5. Fill in the following fields:

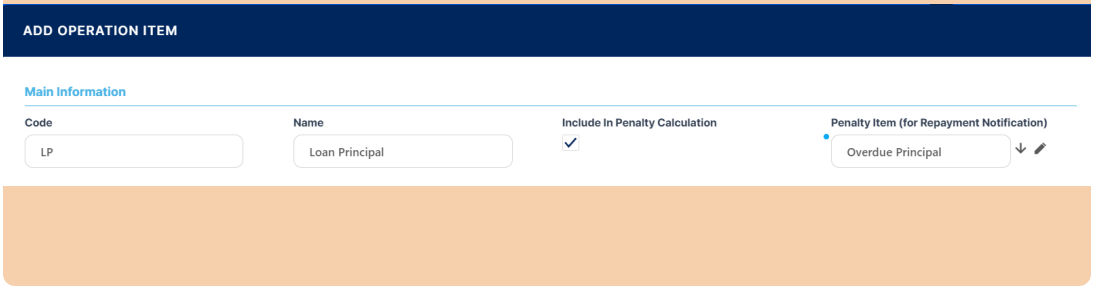


- **Credit Item** - Select the credit item of the allocation method.
- **Minimum Overdue Days** - Enter the minimum number of overdue days for the credit item.
- **Maximum Overdue Days** - Enter the maximum number of overdue days for the credit item.

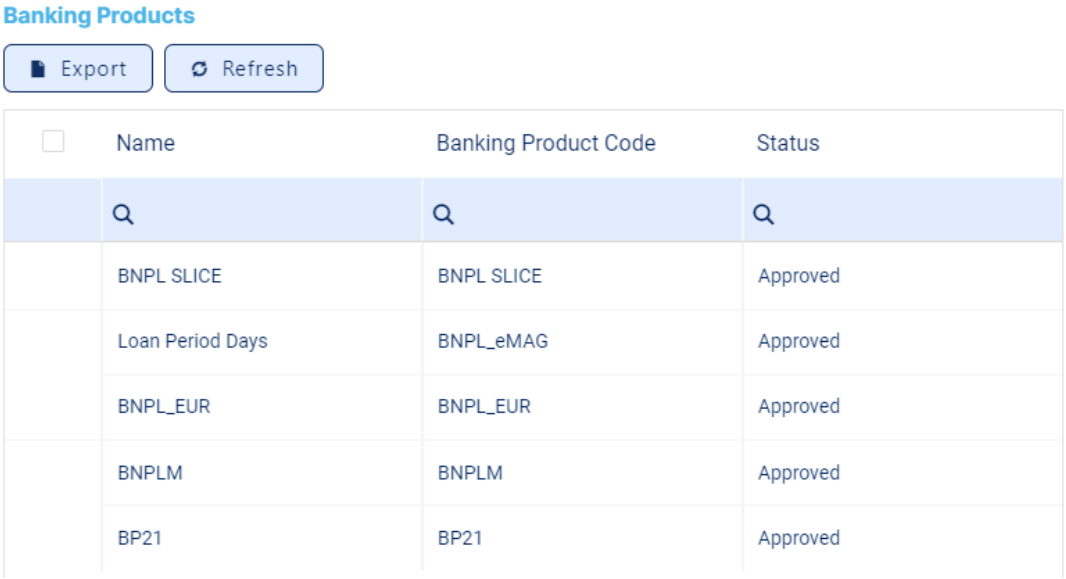
> **NOTE**
>
> The maximum and minimum number of overdue days is relevant when creating allocation methods as an account can have one or more loan principals. If, for example, there are two loan principals, the allocation method is applied based on the oldest one. Thus, depending on the date, the installment is allocated to the oldest loan principal and then moved to the other credit items.

> **IMPORTANT!**
>
> When editing credit items, the **IncludeInPenaltyCalculation** checkbox is available. If selected, the penalty amount is taken out of the account first.

For more details, see Operation Item.



6. Click the **Save and Close** button at the top right corner of the page.
7. Back on the **Edit Allocation Method** page, view the banking products associated with the allocation method previously created in the **Banking Products** section.



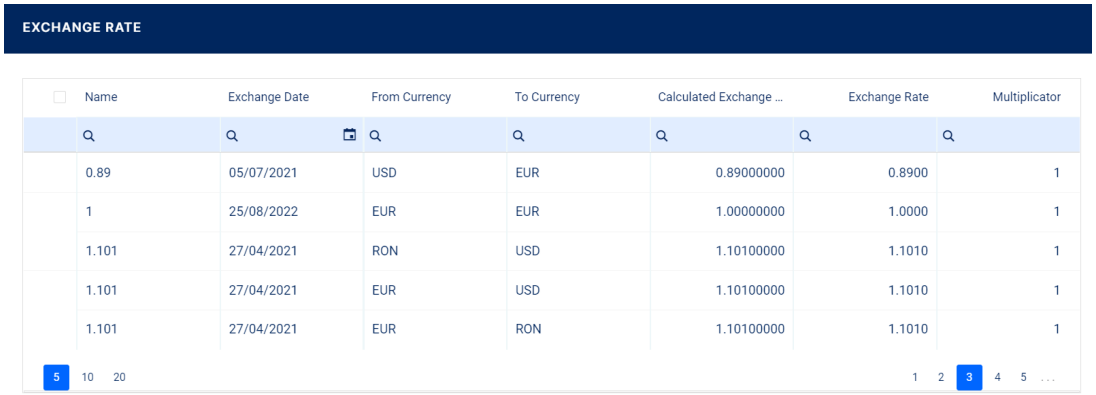
Exchange Rate

Exchange rates represent the value between the currencies of two countries on a given date. These rates are free-floating or fixed. In most cases, exchange rates are free-floating and the value can rise or fall based on market supply and demand. Fixed exchange rates have more restrictions and their value is set by the government.

Loan Management uses exchange rates in limit, collateral, credit facility, and third-party invoices calculations, when the contract and the attached limit/ collateral, the credit facility and its attached contracts, or the third-party invoice and its details are expressed in different currencies. Subledger also uses exchange rates to calculate all amounts using the accounting reference currency.

To manage exchange rates needed in financial operations, follow these steps:

1. In FintechOS Portal, click the main menu icon and expand the **Admin Configuration** menu.
2. Click the **Exchange Rate** menu item to open the **Exchange Rate** page.



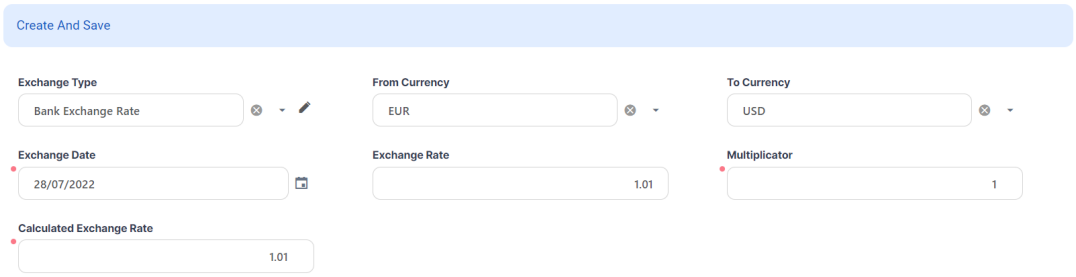
On the **Exchange Rate** page, you can add new exchange rates or search, edit, and delete existing ones.

Creating Exchange Rates

Follow these steps to create a new exchange rate:

1. Click the **Insert** button on the **Exchange Rate** page to open the **Create and Save** page.

2. Fill in the following fields:



- **Exchange Type** - Select the exchange type rate, if more than one was created in Loan Management.
- **Exchange Date** - Select the date of the exchange rate.
- **From Currency** - Select the currency from which the exchange rate is made.
- **To Currency** - Select the currency into which the exchange is performed.
- **Exchange Rate** - Enter the exact rate for the exchange to be applied on the exchange date.
- **Multiplicator** - You can multiply the exchange rate value with the whole number that you insert here. The default value is 1.
- **Calculated Exchange Rate** - Automatically completed with the exchange rate calculated based on the following formula: `ExchangeRate * Multiplicator`. You can edit the field.

3. Click the **Save and Close** button. Loan Management saves the exchange rate record.

Exchange Rate Type

Although Loan Management uses only the BER (Bank Exchange Rate) is used when defining exchange rates, if there is a need to differentiate between exchange rates based on currency market or business areayou can create exchange rate types.

To manage exchange rate types, follow these steps:

1. In FintechOS Portal, click the main menu icon and expand the **Admin Configuration** menu.
2. Click the **Exchange Rate Type** menu item to open the **Exchange Rate Types List** page.

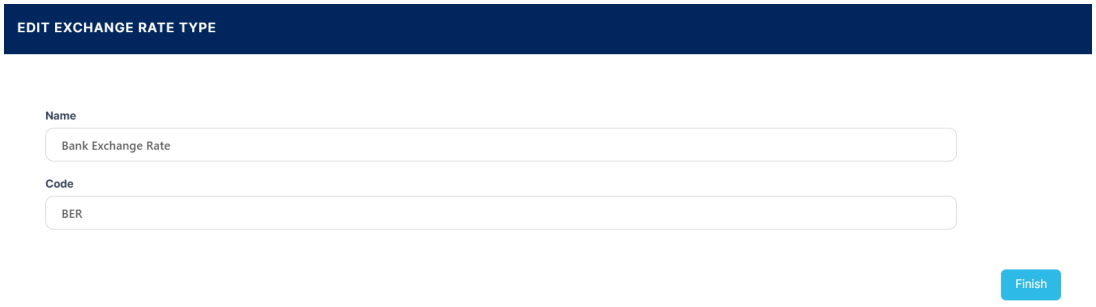


On the **Exchange Rate Type** page, you can add new exchange rates types or search, edit, and delete existing ones.

Creating Exchange Rates Types

Follow these steps to create a new exchange rate:

1. Click the **Insert** button on the top right side of the **Exchange Rate Types List** page. The **Add Exchange Rate Type** page is displayed.
2. Fill in the following fields:



   - **Name** - Enter the name of the exchange rate type.
   - **Code** - Enter the code of the exchange rate type.

3. Click **Finish** at the bottom right corner of the page.

Loan Classification

Financial institutions classify their existing loan contracts based upon the days past due (DPD), the number of days passed since repayment due date without fully repaying the due amount for the oldest unpaid repayment notification. In order to comply with the risk method calculation, the DPD (days past due) value is calculated as the number of days between the contract's due date and the current system date of Loan Management.

Banking being a highly regulated sector such requirements are usually enforced either with regional or local rules. The financial institutions can apply different provision percentages for principal or for interest for each contract, based on this classification: the higher the delay period, the higher the provision percentage applicable and the risk classification. Since the provisions have an impact on the financial results of the institution, this is again driven by regulations and may vary in time or depending on country or region.

IMPORTANT!

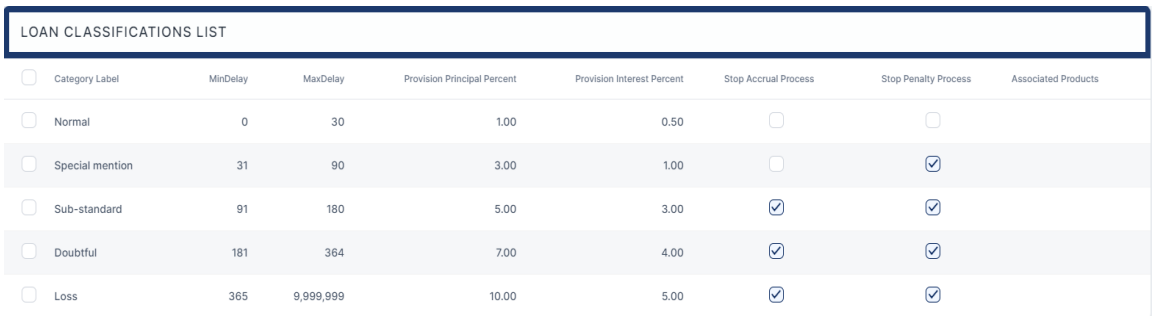
Loan classification works by risk contamination at the customer and the group levels. This means that if a loan contract belonging to a customer is classified as one of a higher risk due to delays in the repayment process, all the other loans of the customer and of the group where the customer is a member are further classified into that high-risk classification.
After the due amounts are paid and the contract is closed, either automatically or manually, Loan Management automatically resets the DPD days to zero.

The `UseContaminationForDPDCategory` Loan Management system parameter specifies whether Loan Management should use the risk contamination for loan classification or not.

The risk classification of loan contracts is automatically performed by the `Update Loan Classification (CB)` scheduled job based on the loan classification records' definition.

To manage loan classification records:

1. In FintechOS Portal, click the main menu icon and expand the **Admin Configurations** menu.
2. Click **Loan Classification** menu item to open the **Loan Classifications List** page.


LOAN CLASSIFICATIONS LIST

Category Label	MinDelay	MaxDelay	Provision Principal Percent	Provision Interest Percent	Stop Accrual Process	Stop Penalty Process	Associated Products
Normal	0	30	1.00	0.50	☐	☐	
Special mention	31	90	3.00	1.00	☐	☑	
Sub-standard	91	180	5.00	3.00	☑	☑	
Doubtful	181	364	7.00	4.00	☑	☑	
Loss	365	9,999,999	10.00	5.00	☑	☑	

On the **Loan Classifications List** page, you can add new classification records or search, edit, and delete existing ones. You can also:

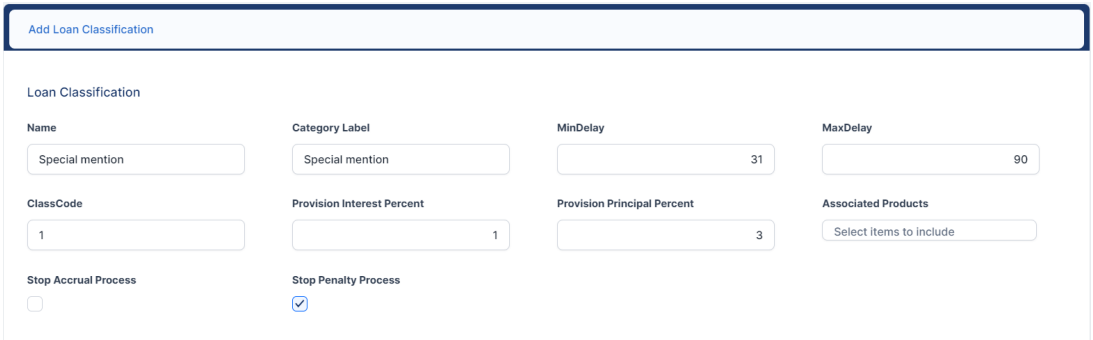
- Stop the accrual process for contracts that fall under a specific classification. Select the **Stop Accrual Process** checkbox next to a record to except all contracts within that category from the accrual calculation processes. All the following categories are automatically excepted from the accrual calculation processes.
- Include the contracts within a loan category into the accrual processes calculation by deselecting the **Stop Accrual Process** checkbox next to a record. You can do this only after deselecting the **Stop Accrual Process** checkboxes next to each of the lower categories.
- Stop the penalty process for contracts that fall under a specific classification. Select the **Stop Penalty Process** checkbox next to a record to except all contracts within that category from the penalty calculation processes. All the following categories are automatically excepted from the penalty calculation processes.
- Include the contracts within a loan category into the penalty processes calculation by deselecting the **Stop Penalty Process** checkbox next to a record. You can do this only after deselecting the **Stop Penalty Process** checkboxes next to each of the lower categories.

IMPORTANT!
You must have the **Loan Admin Officer** security role to select or deselect the **Stop Accrual Process** and **Stop Penalty Process** checkboxes.

Creating Loan Classification Records

Follow these steps to create new loan risk classification records:

1. Click **Insert** button on the **Loan Classifications List** page to display the **Add Loan Classification** page.
2. Fill in the following fields:



- **Name** - Enter the unique name of the risk classification record.
- **Category Label** - Enter the label of the risk classification record.
- **MinDelay/ MaxDelay**- Enter the minimum/ maximum number of days past since a repayment due date without performing the repayment for a loan contract in order for the contract to be classified in this risk classification.
- **ClassCode** - Enter a code for this risk classification. This code is used in automatic calculations for contracts classified in one of the risk categories.
- **Provision Principal Percent/ Provision Interest Percent** - Enter the provision percentage applicable to the principal/ interest amount of contracts falling into this loan risk classification.
- **Associated Products** - Optionally, select from the list the banking products which are the basis for contracts falling into this loan risk classification. If you don't select one or more banking products, then contracts based on any product can fall into this loan risk classification.

- **Stop Accrual Process** - If you select the checkbox, then the contracts that fall within this loan classification delay category are excluded from the accrual calculation processes. If a category is marked as true, all the following categories are automatically marked as true and excepted from the accrual calculation processes.
  You can include the contracts within a loan category into the accrual processes calculation by deselecting the **Stop Accrual Process** checkbox next to a record, within the **Loan Classifications List** page. You can do this only after deselecting the **Stop Accrual Process** checkboxes next to each of the lower categories.
- **Stop Penalty Process** - If you select the checkbox, then the contracts that fall within this loan classification delay category are excluded from the penalty calculation processes. If a category is marked as true, all the following categories are automatically marked as true and excepted from the penalty calculation processes.
  You can include the contracts within a loan category into the penalty processes calculation by deselecting the **Stop Penalty Process** checkbox next to a record, within the **Loan Classifications List** page. You can do this only after deselecting the **Stop Penalty Process** checkboxes next to each of the lower categories.

3. Click the **Save and Close** button.

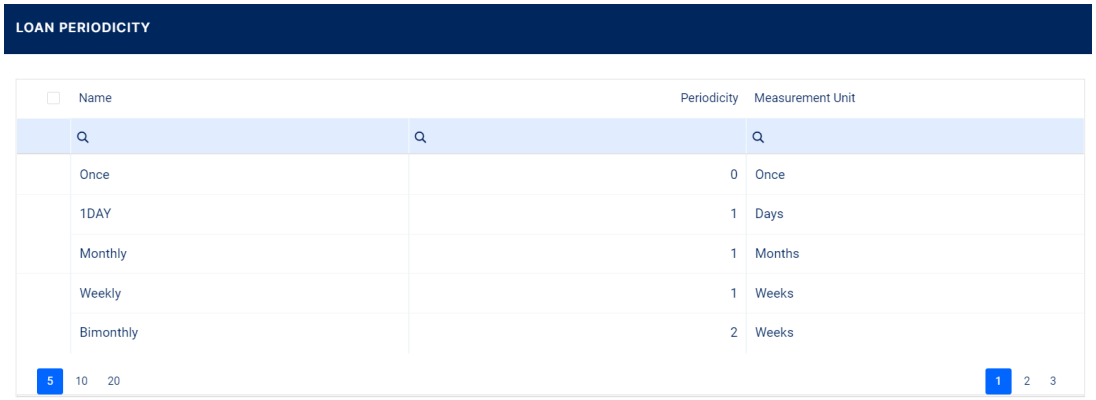
Loan Periodicity

Loan Management uses the loan periodicity dictionary used to define the regularity of payments. For example, payments related to loan contracts, commissions, or installments can be performed once, daily, monthly, yearly, and so on.

To manage loan periodicity records:

1. In FintechOS Portal, click the main menu icon and expand the **Admin Configurations** menu.

2. Click the **Loan Periodicity** menu item to open the **Loan Periodicity** page.

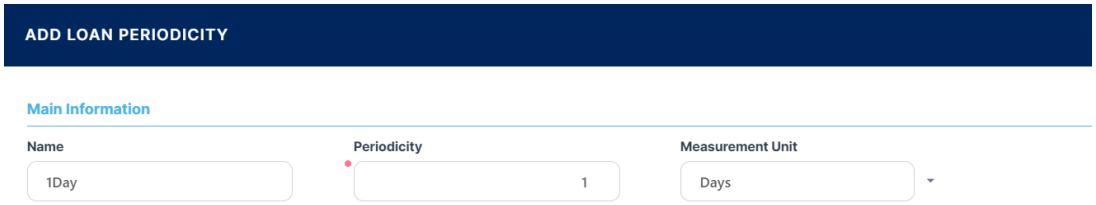


On the **Loan Periodicity** page, you can add new loan periodicity records or search, edit, and delete existing ones.

Creating Loan Periodicity Records

Follow these steps to create new loan periodicity records:

1. Click **Insert** on the **Loan Periodicity** page. The **Add Loan Periodicity** page is displayed.
2. Fill in the following fields:



- **Name** - Enter the name of the periodicity.
- **Periodicity** - Enter the number of measurement units for the periodicity.
- **Measurement Unit** - Select the measurement unit applicable for the periodicity from the drop-down. Possible values: `Days`, `Weeks`, `Months`, `Years`, and `Once`.

3. Click the **Save and Reload** button. The new loan periodicity is created and ready to be used.

Operation Item

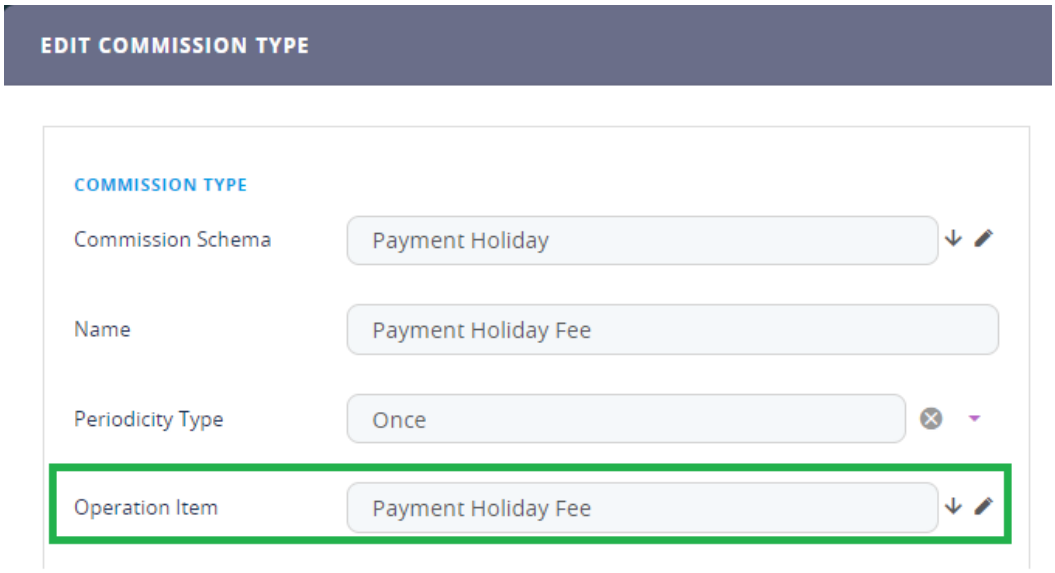
Operation items are those items that relate to a bank's core business, such as all types of fees, commissions, principals, interests, advances or penalty calculations. You can also consider them as balance types that add up to a certain deal or used in tracing what happened on a particular deal. For instance, on the first day of using a new loan, there is only the Loan Principal (LP) on that deal, or it can also have an Advance (ADV) in some particular cases or even a Frond-End Fee (FEF). As the deal progresses, depending on the definition and costs, as well as on repayments, the loan principal can become Paid Principal (PP) or Overdue Principal (OVP) if due amounts are not paid.

Loan Management uses operation items in the calculation processes of payment and repayment notifications, due amounts, accounting entries, and others.

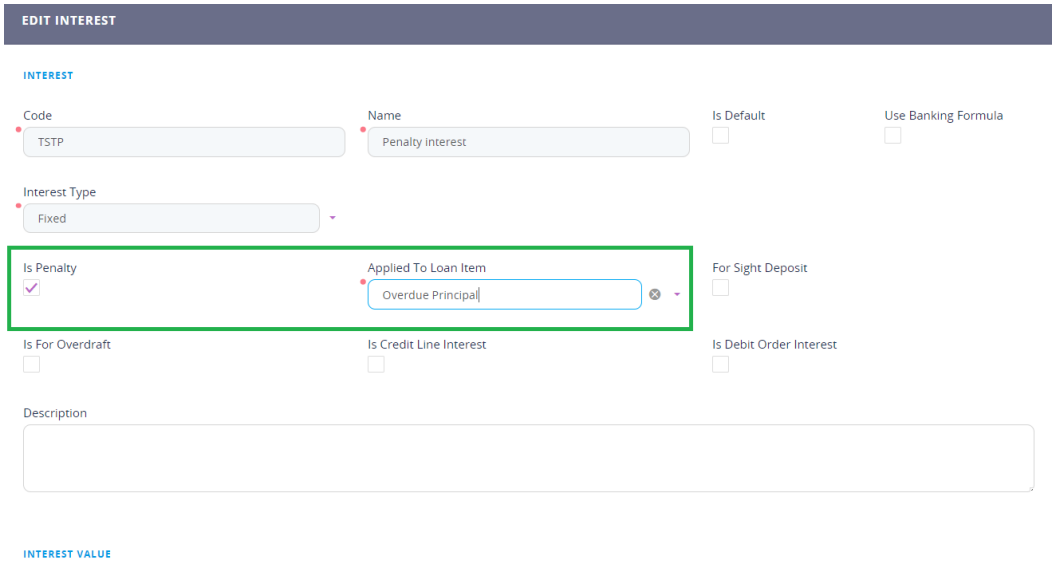
> **NOTE**
> If a repayment notification is not linked to a contract, then Loan Management takes the operation item value from the allocation method configured within the `ManualAllocationMethod` system parameter.
> If a repayment notification is created for a contract with `Closed` status, then Loan Management takes the operation item value from the allocation method selected at the banking product level.

Examples of Using Operation Items

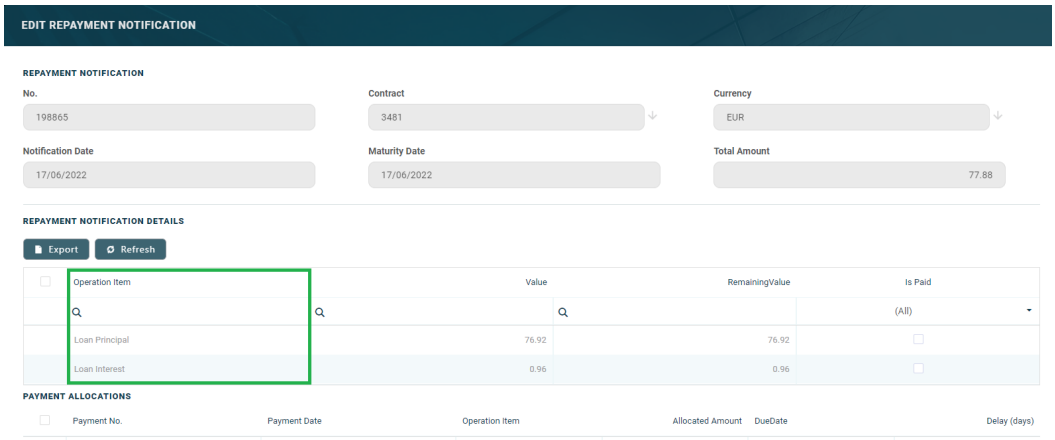
Example of using an operation item within a commission type definition:



Example of using an operation item within an interest definition:



Example of using an operation item within a repayment notification's details:



To manage operation item records:

1. In FintechOS Portal, click the main menu icon and expand the **Admin Configurations** menu.
2. Click the **Operation Item** menu to open the **Operation Item** page.

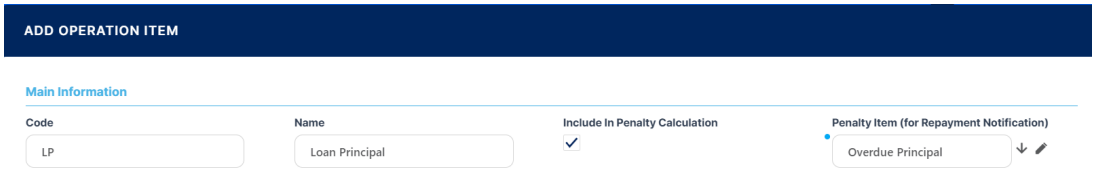


On the **Operation Item** page, you can add new operation item records or search, edit, and delete existing ones.

Creating Operation Item Records

Follow these steps to create new operation item records:

1. In the FintechOS Portal, click the **Insert** button on the top right side of the **Operation Item** page. The **Add Operation Item** page is displayed.
2. Fill in the following fields:



   - **Code** - Enter the unique code of the operation item.
   - **Name** - Enter the name of the operation item.
   - **Include In Penalty Calculation** - Select the checkbox to mark this new item as an operation item used for penalty calculation.

     > **NOTE**
     >
     > If a banking product has in its attached interests list an interest with **Is General = True**, then at the contract level the penalty percent is applied to all operation items that are overdue and are marked with **Include In Penalty Calculation = True**. Read more information about interests and how to define them on the **Interests** page within the Banking Product User Guide.

   - **Penalty Item (for Repayment Notification)** - If `Include In Penalty Calculation = True`, select from the list an existing operation item in whose penalty calculation this new item must participate.
3. Click the **Save and Close** button.

Loan Management System Parameters

The system parameters used by Loan Management determine the behavior of all the contracts, transactions, limits, and other parts that make up your Loan Management system.

> **NOTE**
> Do not confuse the Loan Management system parameters with the FintechOS system parameters, stored in the `systemparameter` and `systemParameterOnPortalProfile` entities!

Predefined Loan Management System Parameters

Here's the list of system parameters used by Loan Management, along with their description:

AccountingAnalyticChar

It represents the analytic character used when displaying decimal numbers.

Module that uses the system parameter: Loan Admin

Parameter type: Text

Default value: **.**

AccountingRealTime

It specifies if all accounting entries are generated real-time (for `True` value) or on demand (for `False` value).

Module that uses the system parameter: Loan Admin

Parameter type: Boolean

Default value: **`False`**

AdvanceNotification

It specifies if the advance to be paid in a contract is displayed in a new repayment notification record (for `False` value) or included in the front-end fee repayment notification (for `True` value).

Module that uses the system parameter: Loan Admin

Parameter type: Boolean

Default value: **`False`**

AsyncContractApproval

When set to `True`, the approval of a contract only switches the change of business status of the contract to `Approved`, and the following approval processes, such as generating a main bank account, updating the contract with the main bank account number, if the `Auto Disbursement = True` then perform the disbursement event, generate a repayment schedule, generate and then process repayment notifications, are written to a queue and then are treated asynchronously by the `Async Contract Approval (CB) scheduled job`. When set to `False`, the approval processes are all synchronous, at the moment when you trigger the approval.

Module that uses the system parameter: Loan Admin

Parameter type: Boolean

Default value: **`False`**

AsyncLimitRecalculation

The parameter indicates how the limit recalculation on repayment in handled by Loan Management, real-time (for `False` value) or asynchronously (for `True` value) through the `Async Limit Recalculation` scheduled job.

Module that uses the system parameter: Loan Admin

Parameter type: Boolean

Default value: **`False`**

BankAccountTransactionFeeMarkDown

This parameter instructs the system how to process bank account transaction operations. Possible values:

- `Total` - the system creates 4 operations – 2 for debit and 2 for credit for the transaction value and sum(fees)
- `Individual` - the system creates 2 * (1+NoFees) operations – debit and credit for the transaction value and each fee.

Module that uses the system parameter: Loan Admin

Parameter type: Text

Default value: `Total`

BankAccountTransactionFlowName

The parameter is used for the external request processing through Async Engine, when triggering a request for the bank account transaction flow.

Module that uses the system parameter: Loan Admin

Parameter type: Text.

Default value: `Flow_CB_BankAcountTransaction_AsyncEngine`

CalculateAccrualEarlyRepayment

It specifies whether the accrual and provision should be calculated for early repayments with the event value equal to a part of contract's unpaid amount (partial early repayments) or only for full

early repayments.

- For `True` value, the accrual and provision is calculated for any early repayment event value.
- For `False` value, the accrual and provision is calculated only for full early repayment event value.

For each early repayment event, the accrual and provision is calculated only if it was not calculated before for the current system date.

Module that uses the system parameter: Loan Admin

Parameter type: Boolean

Default value: Set it according to the bank's policy.

CalculatedProvisions

It specifies whether Loan Management should calculate (for `True` value) or not (for `False` value) the provisions in the accruals and provisions processes.

Module that uses the system parameter: Loan Admin

Parameter type: Boolean

Default value: `False`.

CalendarYearEnd

The maximum year in the calendar to be used when generating holidays for calendars. Format: YYYY.

Module that uses the system parameter: Loan Admin

Parameter type: Whole Number

Default value: `2100`.

CalendarYearStart

The minimum year in the calendar to be used when generating holidays for calendars. Format: YYYY

Module that uses the system parameter: Loan Admin

Parameter type: Whole Number

Default value: 2019.

CacheControlPanel

It is used to cache the system date within the FTOS_CB_SystemDate library, for True value. The same date is also used in the Control Panel within the **Utilities** package.

Modules that use the system parameter: Banking Product, Loan Management

Parameter type: Bool

Default value: False.

CreditFacilityLimitPercent

It represents the default limit of credit facility records.

Module that uses the system parameter: Loan Admin

Parameter type: Whole Number

Default value: 30.

CurrentAccount_WithOverdraft_DaysBeforeExpire

It represents the number of days before the overdraft feature's expiration date of a current account when the contract based on that banking product gets displayed in the **Soon to Expire Ovedrafts** dashboard.

Module that uses the system parameter: Loan Admin

Parameter type: Whole Number

Default value: 30.

CustomerToContractDirectDebitSettlementAcc

It handles the change on a customer's contracts once the Direct Debit Settlement Account attribute at the customer level is switched to true or false.

- If CustomerToContractDirectDebitSettlementAcc = False, the changes from the customer level for direct debit settlement do not impact existing contracts, and only the manual repayment notifications of the affected customer change their status accordigly.
- If CustomerToContractDirectDebitSettlementAcc = True, the changes at the customer level for direct debit settlement impact current contracts. All the customer's existing contracts' Direct Debit Settlement Account settings are changed according to the setting at the customer level, and all the repayment notifications associated to the customer change their status accordingly.

Module that uses the system parameter: Loan Admin

Parameter type: Boolean

Default value: Set it according to the bank's policy.

DaysBeforePurge

It represents the default number of calendar days that a record will be kept in Draft status before it is purged. The records that are due to be purged on the current day and have their transaction type's **To Be Purged** field marked as True are displayed in the Records To Be Purged Dashboard, within the section specific to the record's transaction type. The job performing the deletion is Delete Purged Entries and it should be scheduled at the bank's level.

The custom job error records are also purged at the interval given by this parameter.

Module that uses the system parameter: Loan Admin

Parameter type: Whole Number

Default value: Set it according to the bank's policy.

DaysFutureInstallmentsReport

It represents the default number of days before an installment's due date in order for that installment to be included in the **Future Installments** report within the Reports Dashboard.

Module that uses the system parameter: Loan Admin

Parameter type: Whole Number

Default value: 15.

DaysPastDueInstallmentsReport

It represents the default number of days after an unpaid installment's due date in order for that installment to be included in the **Past Due Installments** report within the Reports Dashboard.

Module that uses the system parameter: Loan Admin

Parameter type: Whole Number

Default value: 25.

DashboardCurrency

It specifies the currency in which all the amounts are displayed within the **Contracts Overview** section of the Loan Admin Officer Dashboard.

Module that uses the system parameter: Loan Admin

Parameter type: Lookup. To entity: Currency

Default value: EUR

DashboardDefaultLastXDays

It represents the default number of days considered when running the reports within the Loan Admin Officer Dashboard.

Module that uses the system parameter: Loan Admin

Parameter type: Numeric

Default value: 365

DefaultIntervalLimitsReport

It represents the default number of months considered when running the reports within the Limit Report dashboard.

Module that uses the system parameter: Loan Admin

Parameter type: Whole Number

Default value: 12

DefaultSalesChannelAPI

It represents the default sales channel for contracts defined via API integration.

Module that uses the system parameter: Loan Admin

Parameter type: Lookup. To entity `SalesChannel`

Default value: Set it according to the bank's policy.

DefaultSalesChannelBackOffice

It represents the default sales channel for contracts defined through the Loan Management user interface.

Module that uses the system parameter: Loan Admin

Parameter type: Lookup. To entity `SalesChannel`

Default value: Set it according to the bank's policy.

DefaultWriteOffReconciliationAccountBankingProduct

It represents the financial institution's internal write-off account defined for the system's default currency. The write-off transaction takes the remaining amount from the internal write-off account and credits the loan account. The value is used at the product level, within the Write Off section of the Lean Core Settings tab, but you can change the account according to your needs.

Module that uses the system parameter: Loan Admin

Parameter type: Lookup. To entity: `Bank Account`

Default value: Select an internal reconciliation account according to the financial institution's policy.

DelayDaysForBlockNewContractApproval

It represents the default number of delay days for blocking the approval of new loan contracts for customers who have overdue payments. New contract approval is blocked by Loan Management if the customer has overdue days >= the value of the `DelayDaysForBlockNewContractApproval` parameter.

Module that uses the system parameter: Collection

Parameter type: Whole Number

Default value: `0`

DepositAggregateItemValues

It specifies if the deposit interest is split in two lines or displayed in one line.

- For `False` value, the system splits the **Deposit interest to recover** in two lines (- paid interest -> recover all; sight interest to pay, pay all).
- For `True` value, the system displays the **Deposit interest to recover** in one line with the aggregate value.

Module that uses the system parameter: Loan Admin

Parameter type: Boolean

Default value: `False`.

EarlyRepaymentFee_IndividualNotification

It specifies whether to generate a separate repayment notification for the early repayment fee of a contract (for `True` value) or include the fee into the repayment notification containing the actual early repayment amount (for `False` value).

Module that uses the system parameter: Loan Admin

Parameter type: Boolean

Default value: `False`.

ExchangeRate_UseLatest

It specifies whether Loan Management should use the latest available exchange rates for calculations or not.

Module that uses the system parameter: Loan Admin

Parameter type: Boolean

Default value: `True`.

FrontEndFee

It specifies the commission type used for automatic notification on contract approval (Inclusion)/ or notification daily process (Exclusion).

Module that uses the system parameter: Loan Admin

Parameter type: Lookup. To entity `CommissionType`

Default value: `Front-end Fee`.

IncludeDAEIntoScheduleCalculation

It specifies whether Loan Management should include APRC (annual percentage rate of charge) for repayment schedule calculations or not.

Module that uses the system parameter: Loan Admin

Parameter type: Boolean

Default value: `True`.

LimitMandatoryForIndividuals

It specifies whether Loan Management should validate the limits for individual customers or only validate them for legal entity customers.

- For `False` value, Loan Management does not validate any limits for the individual customers.
- For `True` value, Loan Management validates all the limits for the individual customers the same way it does for legal entity customers.

Module that uses the system parameter: Loan Admin

Parameter type: Boolean

Default value: `True`.

LoanApprovalFlowName

The parameter is used for the external request processing through Async Engine, when triggering a request for the loan contract approval flow.

Module that uses the system parameter: Loan Admin

Parameter type: Text.

Default value: `Flow_CB_LoanApproval_AsyncEngine`

LogScheduleJobError

It specifies whether job errors should be logged (for `True` value) as custom job error records or not (for `False` value). The custom job error records are purged at the interval given by the `DaysBeforePurge` parameter.

Module that uses the system parameter: Loan Admin

Parameter type: Boolean

Default value: `False`.

ManualAllocationMethod

The default allocation method used by Loan Management for manual repayment notification unlinked to a contract or when contracts have products without allocation method defined.

Module that uses the system parameter: Collection

Parameter type: Lookup. To entity: `AllocationMethod`

Default value: `CostOrder`

ManualGraceRepayment

The parameter is used to set up the maturity date on manual repayment notification, if the notification is not linked to a contract.

Module that uses the system parameter: Collection

Parameter type: Whole Number

Default value: Set it according to the bank's policy.

ManualPenaltyInterestList

The parameter holds the penalty interest list used for penalty calculation for manual repayment notifications that are not linked to a contract.

Module that uses the system parameter: Banking Product

Parameter type: Lookup. To entity: `InterestList`

Default value: `ManualPenaltyInterestList`

ManualRepaymentFee

It specifies whether a banking product can have only one **Repayment Fee** type commission on its **Commission List** or more. This parameter affects the **Contract Event** page.

- For `False` value, the banking product has only one Repayment Fee commission type on its commission list.
- For `True` value, the banking product's commission list displays all the commissions stored in the `Commission` entity with type Repayment Fee.

Read more information about the effects of this parameter's value in the Transaction Fees section.

Module that uses the system parameter: Loan Admin

Parameter type: Boolean

Default value: `True`.

PurgeScheduleJobLogDays

It specifies the number of days used to select the old data from the schedule job log to be purged.

Module that uses the system parameter: Loan Admin

Parameter type: Whole Number

Default value: 1.

ReconciliationAccountTreatment

Specifies how Loan Management treats situations when the funds of the reconciliation account associated with the banking product used in the contract would go below zero if a disbursement event would be approved.

- For `NoMessage` value, there is no error or warning message displayed if the disbursement event that is being approved would result in a negative balance of the associated reconciliation account. The event can be approved and the balance can go below zero.
- For `Warning` value, there is a warning message displayed if the disbursement event that is being approved would result in a negative balance of the associated reconciliation account. The event can be approved and the balance can go below zero.
- For `Error` value, there is an error message displayed if the disbursement event that is being approved would result in a negative balance of the associated reconciliation account. The event can't be approved and the balance can't go below zero.

This is a system-wide setting, applicable to events for contracts based on all banking products without a specified `Negative balance treatment` value. Loan Management also takes into consideration the settings used at the banking product level (the `Negative balance treatment` field's value next to `Reconciliation Account`). Thus, if the value is specified at the banking product level, then that value takes precedence over the system parameter's setting.

Module that uses the system parameter: Banking Product

Parameter type: Option Set. Values from option set: `WarningErrorTreatment`

Default value: `Warning`.

RepaymentFee

The commission type used for notification daily process (Exclusion).

Module that uses the system parameter: Loan Admin

Parameter type: Lookup. To entity: `CommissionType`

Default value: `Repayment Fee`

RepaymentNotificationFlowName

The parameter is used for the external request processing through Async Engine, when triggering a request for the repayment notification flow.

Module that uses the system parameter: Loan Admin

Parameter type: Text.

Default value: `Flow_CB_RepaymentNotification_AsyncEngine`

ThirdPartyPaymentIsNet

It specifies whether Loan Management should generate one or two bank account transactions and payments for a third-party agreement invoice when the invoice's status is changed from **Approved** to **Unpaid**.

For `False` value, two bank account transactions are generated with two payments:

- One transaction with **source** account = `Settlement Account` and **destination** account = `Reconciliation Account` with the value of `Total Amount To Recover`;
- Another transaction with **source** account = `Reconciliation Account` and **destination** account = `Settlement Account` with the value of `Total Amount To Pay.`

For `True` value, Loan Management calculates the difference between `Total Amount To Recover` and `Total Amount To Pay`. Only one bank account transaction is generated and only one payment, representing the non-zero value between the `Total Amount To Recover` and the `Total Amount To Pay`, as follows:

- If `Total Amount To Recover - Total Amount To Pay > 0`, a new bank account transaction is generated with **source** account = `Settlement Account` and **destination** account = `Reconciliation Account`, and a payment is generated for the invoice.
- If `Total Amount To Recover - Total Amount To Pay = 0`, a bank account transaction is generated, and the transaction's status changes to **Paid**.

- If `Total Amount To Recover - Total Amount To Pay < 0`, a new bank account transaction is generated with **source** account = `Reconciliation Account` and **destination** account = `Settlement Account`, and a payment is generated for the invoice.

When the payments are approved, the invoice's status becomes **Paid**.

Module that uses the system parameter: Loan Admin

Parameter type: Boolean

Default value: `False`

ThirdPartyRole

It contains the list of allowed roles to choose from in the third-party agreement form.

Module that uses the system parameter: Banking Product

Parameter type: Text

Default value: `Merchant,Insurer,Broker,Agent`

Unusage

It specifies the commission for not using the funds. The commission type used for Credit Facility accrual daily process.

Module that uses the system parameter: Loan Admin

Parameter type: Lookup. To entity: `CommissionType`

Default value: `False`.

Usage

It specifies the commission for usage of funds. The commission type used for Credit Facility accrual daily process.

Module that uses the system parameter: Loan Admin

Parameter type: Lookup. To entity: `CommissionType`

Default value: `Commission Usage Monthly`

UseCF

It specifies whether your installation uses the Credit Facility module or not.

- For `True` value, the Credit Facility module is used, the **Credit Facility** menu item is displayed within the Loan Management Operational menu, and the credit facility features are available in the Loan Admin Officer Dashboard.
- For `False` value, the Credit Facility module isn't used, the **Credit Facility** menu item is not displayed within the Loan Management Operational menu, and the credit facility features are not available in the Loan Admin Officer Dashboard.

Module that uses the system parameter: Loan Admin

Parameter type: Boolean

Default value: `True`

UseContaminationForDPDCategory

Loan classification works by risk contamination at the customer and the group levels. This means that if a loan contract belonging to a customer is classified as one of a higher risk due to delays in the repayment process, all the other loans of the customer and of the group where the customer is a member are further classified into

that high-risk classification. Read more about loan classification in this dedicated page.

This parameter specifies whether Loan Management should use the risk contamination for loan classification or not.

- For `True` value, risk contamination is used for loan classification, thus one unpaid contract affecting all the loan contracts of that customer.
- For `False` value, risk contamination is not used for loan classification, thus unpaid contracts don't affect other loan contracts of the same customer.

The parameter affects only the EOD and SOD jobs.

Module that uses the system parameter: Loan Admin

Parameter type: Boolean

Default value: `True`

UseExternalProcessing

It specifies whether the requests generated in Loan Management are sent for processing outside of the platform, for example through Camel async engine, or within the platform, through jobserver.

- For `True` value, the requests are sent for processing outside of the platform, thus improving the platform's overall performance through parallel processing.
- For `False` value, the requests are processed within the platform, through jobserver.

Module that uses the system parameter: Loan Admin

Parameter type: Boolean

Default value: True

UseGLModule

It specifies whether your installation uses the Subledger module or not. The Subledger module comes within a different digital asset than Loan Management, thus its use is optional.

- For True value, the GL module is used and the **GL Settings** tab is displayed at banking product level.
- For False value, the GL module is not used and the **GL Settings** tab does not display at banking product level.

Module that uses the system parameter: Loan Admin

Parameter type: Boolean

Default value: True

UseTPM

It specifies if the Distribution Management package is installed on your system. If UseTPM = True, the system triggers the queueing of the transactions based on their contract and event ids. Also:

- For True value, the Distribution Management related features are available in the Loan Admin Officer Dashboard.
- For False value, the Distribution Management related features are not available in the Loan Admin Officer Dashboard.

Module that uses the system parameter: Loan Admin

Parameter type: Boolean

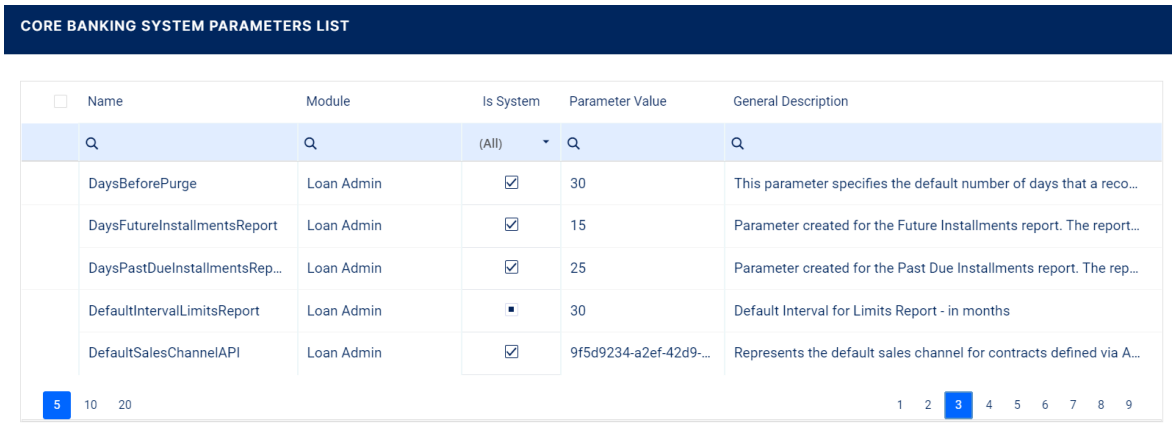
Default value: False. It is automatically updated to True when the Distribution Management package is installed.

Managing Loan Management System Parameters

> **IMPORTANT!**
> You must have the `system administrator` user right to view and manage the Loan Management system parameters.

In order to manage the system parameters used by your FintechOS Loan Management installation, follow these steps:

1. In FintechOS Portal, click the main menu icon and expand the **Admin Configurations** menu.
2. Click **Loan Management System Parameter** menu item to open the **Loan Management System Parameters List** page.



On the **Loan Management System Parameters List** page, you can add new system parameters or search, edit, and delete existing ones.

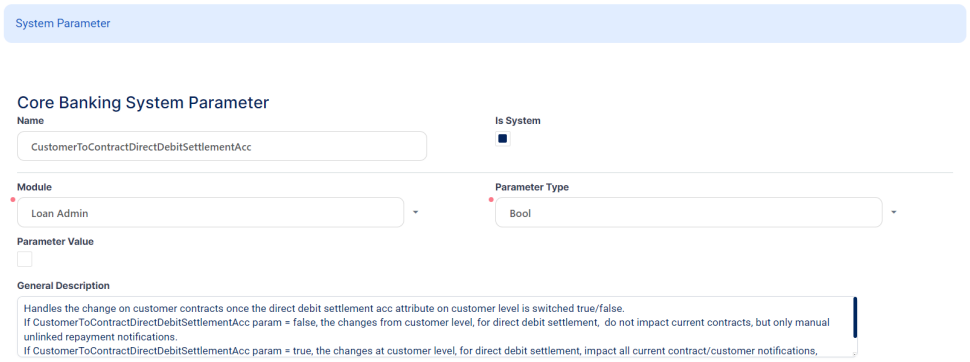
> **IMPORTANT!**
> You can't delete parameters marked as **Is System** or edit anything else except their value.

Creating Loan Management System Parameters

Follow these steps to create new system parameters to be used with Loan Management:

1. Click **Insert** button on the **Loan Management System Parameters List** page to display the **System Parameter** page.

2. Fill in the following fields:



   - **Name** - Enter a suggestive name for the parameter.
   - **Is System** - Select the checkbox to specify that the parameter cannot deleted or edited, except its value. Leave the checkbox empty if this parameter can be changed or deleted.
   - **Module** - Select the Loan Management module that uses the system parameter.
   - **General Description** - Enter a detailed description for the system parameter.
   - **Parameter Type** - Select the data type of the parameter. Possible values: `Text`, `Date`, `Date Time`, `Invariant Date`, `Whole Number`, `Numeric`, `Option Set` and `Entity`.

3. Fill in the rest of the fields, depending on the selected data type:
   - **Entity** - For Lookup parameter type, select the entity from where you need to pick a record as parameter value.
   - **Option Set** - For Option Set parameter type, select the option set from where you need to pick a value as parameter value.
   - **Parameter Value** - Enter the value of the default parameter. Depending on the selected parameter type, you can either enter a value, select the checkbox or select record:

- For `Text, Date, Date Time, Invariant Date, Whole Number, Numeric` parameter types, enter the desired value taking in consideration the data type's format.
- For `Boolean` parameter types, select the checkbox to specify a `True` value, or deselect for a `False` value.
- For `Lookup` parameter type, select the desired record from the previously selected entity that acts as parameter value.
- For `Option Set` parameter types, select the desired value from the previously selected option set that acts as parameter value.

4. Click the **Save and Reload** button.
5. Make sure you create scripts that actually use the new parameter, to instruct Loan Management to perform the tasks you wish.

Transaction Types Used in Loan Management

Any transfer of funds between two bank accounts is recorded as a transaction. There are different types of transactions used in the financial world.

You can manage the transaction types in the FintechOS Portal's **Admin Configurations -> Transaction Type** or **General Ledger Configurations -> Transaction Type** menu. See more details about managing transaction types in the **SubledgerUser Guide**. You can also manage transaction type records in the FintechOS Studio's **Product Factory > Banking Product Dictionaries > Transaction Type** menu.

Before being approved and used within contracts, each banking product must have its allowed transaction types specified in the **Associated Transaction** tab. For more information, see Banking Product Factory.

Predefined Transaction Types

You can use the following transaction types are predefined in Loan Management processes:

- **Accruals and Provisions** - It represents the funds set aside to cover future expenses. A provision is aimed at covering a probable future expense, or reduction in the value of an asset. An accrual is a type of provision where revenue or expenses are recorded when a transaction occurs rather than when payment is received or made. Can't be purged.
- **Agreement** - It represents a binding contract between the bank and a third-party entity (agent, broker, insurer, etc.) to formalize an agreement to financially compensate the third-party for the intermediation of selling banking products or services to customers, or compensate the bank for the inter-mediation of selling the third-party's products or services to customers, and to compensate the bank for managing the contract with the third-party. Can be purged.
- **Deposit Liquidation** - It represents the way of closing the deposit account, so the entire amount is transferred in the current account and the deposit account is closed. If the liquidation occurs at the maturity date, the interest will also be paid. If the liquidation occurs on any other day except the maturity date, the customer will receive the sight interest (if a sight interest is configured). Can't be purged.
- **Disbursement** - It represents the actual delivery of funds from a bank account to the customer. The repayment schedule gets calculated or recalculated. Can be purged.
- **DownPaymentAndFee** - It represents a "payment in" operation type transaction, that can be used when instrumenting a payment at contract approval, for down-payment and fees. Can't be purged. Not used at the moment. It can be used for Buy Now Pay Later products-based contracts, to record the first scheduled payment and the front-end commissions and fees, as a copy of the Top-Up transaction on the settlement account.
- **Early Repayment** - It represents the early return of funds previously borrowed from a lender. The repayment schedule is updated. Can be purged.

- **Early Termination Deposit** - It represents the way of closing the deposit account applicable when the deposit is terminated before schedule. Can be purged.
- **Interest Capitalization** - It represents the addition of the unpaid interest value to the principal balance. Can't be purged.
- **Loan Contract** - It represents a binding contract between two or more parties to formalize a loan process. Can be purged.
- **Overdraft Payment** - It represents an amount of money that a customer with a bank account is temporarily allowed to owe to the bank. Can't be purged.
- **Payment Deposit** - It represents an amount of money paid into an account as part of a payment schedule. Can't be purged.
- **Payment Holiday** - It represents taking a break of any number of installments for the generated schedule. Can be purged.
- **Repayment** - It represents the act of paying back money previously borrowed from a lender by manually repaying an installment from the schedule. Can't be purged.
- **Repayment Notification** - It represents a notification sent for a repayment. At the due date of every installment, an automatic notification is generated by Loan Management. Can't be purged.
- **Reschedule Overdues** - It represents an operation where overdue installments are merged to the following installments and they are no longer collecting penalties. The repayments schedule gets updated. Can be purged.
- **Reschedule Debt** - It represents an operation that updates the balance with the amount rescheduled. Can't be purged.
- **Returned Amount or Goods** - It represents the transaction through which a customer returns all or part of a loan or mortgage in a short while after contract creation, if the banking product was defined to allow such transactions, and the commissions already paid by the customer as front-end fees marked as returnable are paid back. This transaction type only accepts Return Fee commission types. Upon transaction approval, a new contract version is automatically created.

Can be purged.

- **Third-Party Invoice** - It represents the invoice through which the amounts automatically calculated based on an agreement are recorded in Loan Management.
- **Top-Up Account** - It represents adding amounts to the account before the value drains down to zero. Can be purged.
- **Transfer between my bank accounts** - It represents the process of moving funds between the same customer's bank accounts. Can be purged.
- **Withdraw** - It represents removing funds from a bank account. Can be purged.
- **WriteOff** - It represents a way of writing off the remaining amounts for non-performing loans. Can't be purged.

IMPORTANT!
If a transaction type is marked as an automatic transaction (`Is Automatic Transaction = True`), then that transaction type cannot be selected in the **Events** page when creating contract events. Check the **SubledgerUser Guide** for more information about defining transaction types.
The transaction types that cannot be purged cannot be deleted from the system. Their **To Be Purged** field within the **Transaction Type** page is marked as `False`, cannot be edited and is hidden.
For each transaction type that can be purged, Loan Management displays a tab in the **Records To Be Purged dashboard** only if their **To Be Purged** field is marked as `True`.

Read about which transaction types are typically used for each type of banking products in the Banking Product Factory user guides, within each banking product's **Associated Transactions** section.

Term Loan Contracts-Specific Transactions

When you add events for contracts created based on term loan banking products, the following transaction types can be selected, assuming that they were added previously at banking product level: **Disbursement**, **Early Repayment**, **Payment Holiday**, **Repayment**

Notification, **Reschedule Overdues**, **Reschedule Debt**.

Current Account Contracts-Specific Transactions

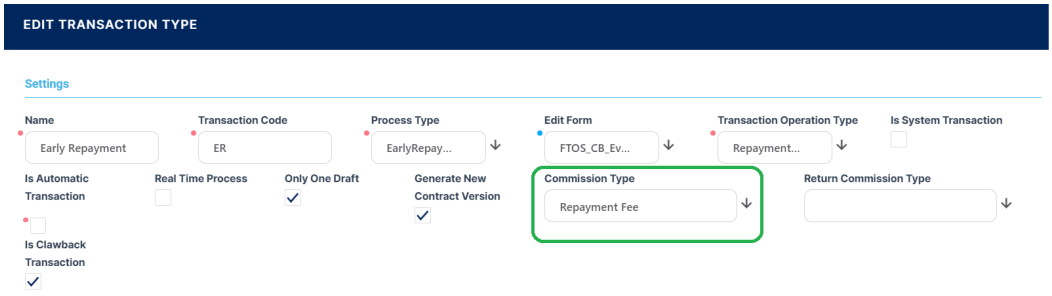
When you add events for contracts created based on current account banking products, the following transaction types can be selected, assuming that they were added previously at banking product level: **Top-Up Account**, **Transfer between my bank accounts**, **Withdraw**.

> **NOTE**
> The transactions can be performed only in the same currency.

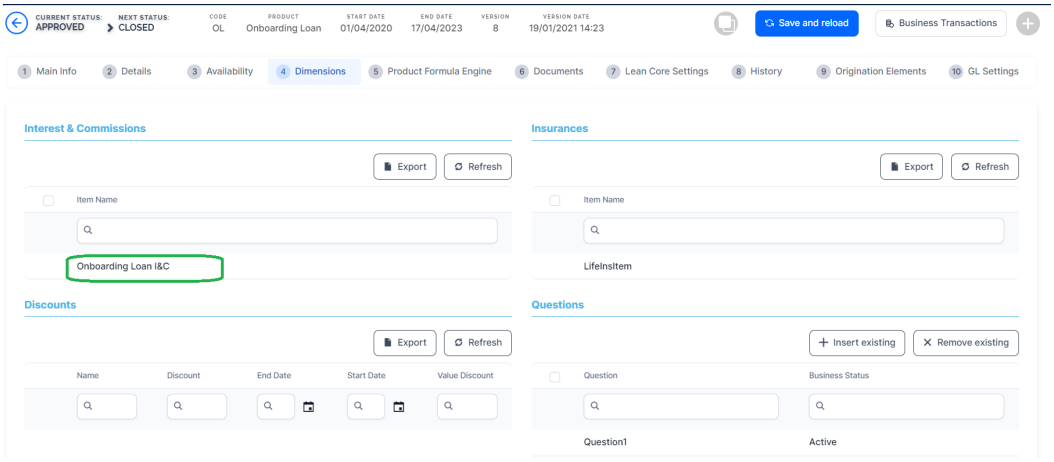
Transaction Fees

Some transactions have a fee collected at the event validation for each contract. For these transactions, repayment notifications for those fees are automatically generated when an event gets to the **Approved** status. For Early Repayment transaction there is a repayment fee, and for Payment Holiday transactions there is a payment holiday fee. These fees are automatically selected from the banking product.

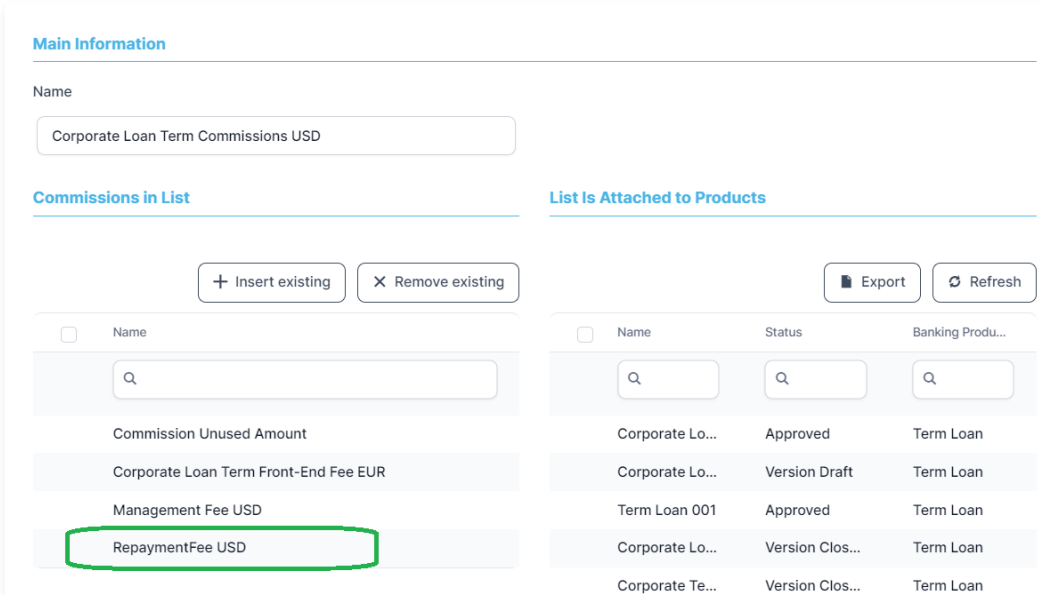
When defining the transaction type, you can select the commission type for the fee:



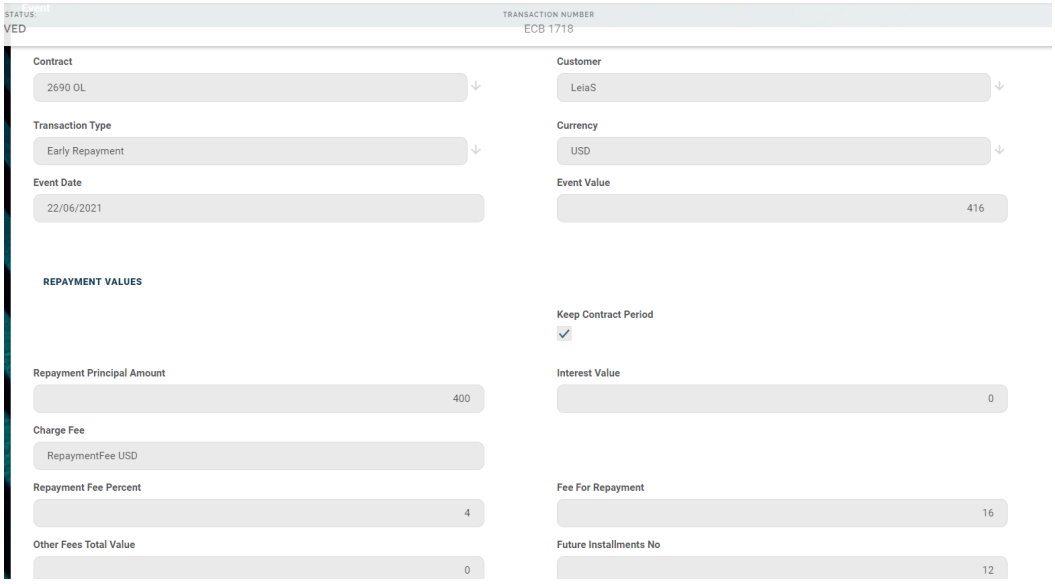
Let's say your contract uses a banking product that has this type of fee attached to it:



Edit Commission List



Loan Management uses that fee for collection at the event level, for example 4% out of 400 USD, thus the customer must pay 416 USD in order to make an early repayment. This amount is notified at the approval of the event.



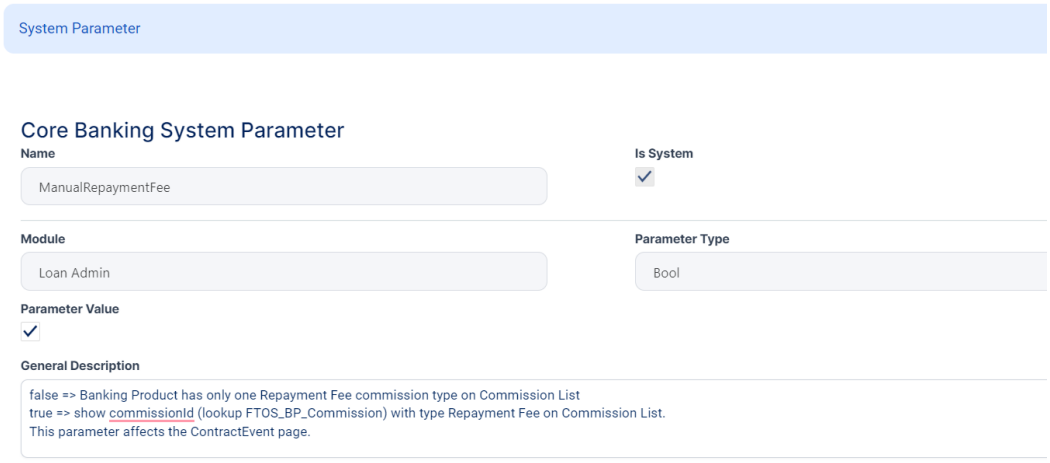
For a payment holiday that affects future installments, only the payment holiday fee gets notified.

The transaction types used for loan contracts that collect a fee at event approval are the following:

- **Payment Holiday** transactions, with an associated commission type of **Payment Holiday Fee**.
- **Reschedule Overdues** transactions, with an associated commission type of **Repayment Fee**.
- **Early Repayment** transactions, with an associated commission type of **Repayment Fee**.

> **NOTE**
> Disbursements don't have this setup for collecting a fee at event approval.

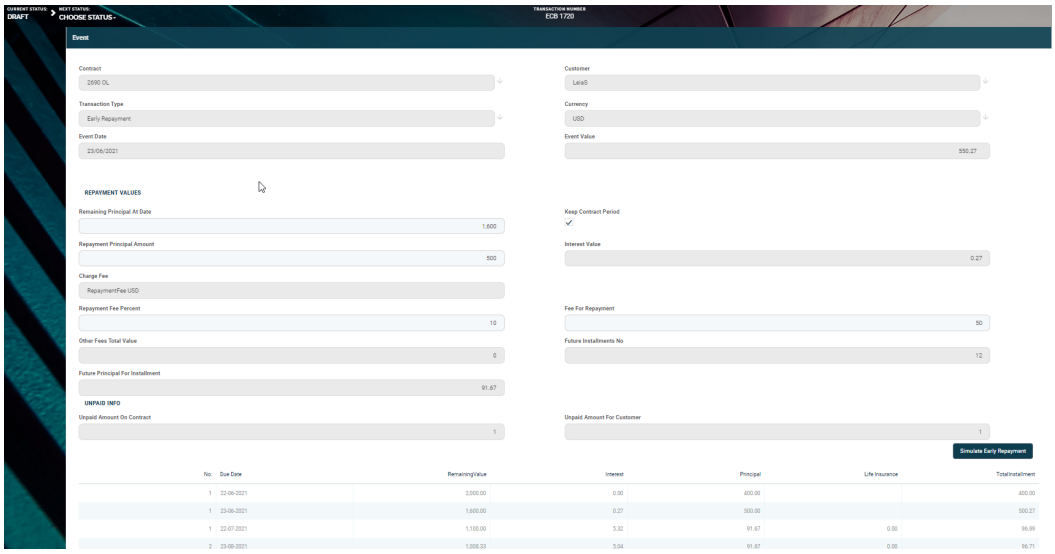
The Loan Management system parameter `ManualRepaymentFee`, having a default value `False`, is used mostly for early repayment.



IMPORTANT!
When the value of the `ManualRepaymentFee` parameter is `False`, the early repayment fee is not negotiable, and the fee values are selected exactly as they are defined in the banking product.

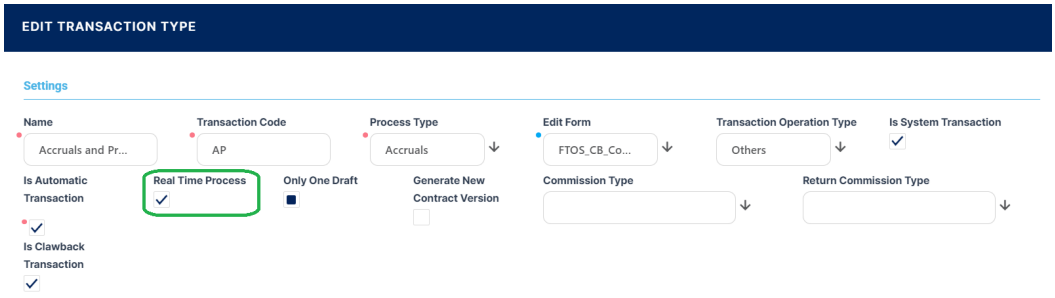
When the value of the `ManualRepaymentFee` parameter is **True**, the early repayment fee is negotiable, and the credit officer that is operating the contract event can change the default value that is coming out of the banking product. If the fee is a percentage, then they can change the fee percentage or the fee value. If the fee is not a percentage, then they can change only the fee value. Other related values are automatically updated.

In the example below, having a `ManualRepaymentFee` parameter set on `True`, Loan Management allows changing the default repayment fee percentage of 3.5% out of 500 USD to 10%, resulting in a fee for repayment amount of 50 USD.



Real-Time or Queued Transaction Processing

The transactions made on bank accounts can be processed in real-time, when the transaction is approved, or at a later time, after being placed in a queue and taken for processing by a specialized scheduled job. The real-time processing depends on the **Real Time Process** checkbox being selected or not at every transaction type's level:



Each time a transaction is performed on a bank account, the system verifies its transaction type's **Real Time Process** field. If the value is `True`, then the transaction is processed right away. If the value is `False`, then the transaction is inserted as a record in the `BankAccountTransactionQueue` entity, with the `isProcessed` attribute set to `False` and `isWithError` set to `False`. The `Bank Account Transaction Queue Processing` scheduled job runs every 1 minute, taking the top 10 records from the entity

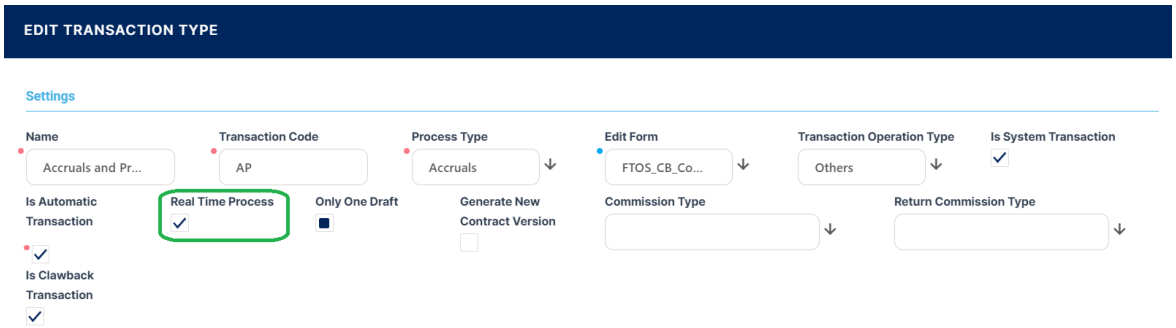
with the attribute `isProcessed = False`, and processing the transactions. Any errors encountered on processing are logged in the `errorMessage` attribute. The `Bank Account Transaction Queue Cleanup` scheduled job runs once each night and cleans up the already processed transaction records with `isWithError = False`.

As a user with admin rights, you can view the transactions within the queue in the **Bank Account Transaction Queue** menu:



Bank Account Transaction Queue

Loan Management processes transactions made on bank accounts in real-time, when the transaction is approved, or at a later time, after placing the transactions in a queue and being taken for processing by a specialized scheduled job. The real-time processing depends on the **Real Time Process** checkbox being selected or not at every transaction type's level:



Each time a transaction is performed on a bank account, the system verifies its transaction type's **Real Time Process** field. If the value is `True`, then the transaction is processed right away. If the value is `False`, then the transaction is inserted as a record in the `BankAccountTransactionQueue` entity, with the `isProcessed` attribute set to `False` and `isWithError` set to `False`. The `Bank Account Transaction Queue Processing` scheduled job runs every 1 minute, taking the top 10 records from the entity with the attribute `isProcessed = False`, and processing the transactions. Any errors encountered on processing are logged in the `errorMessage` attribute. The `Bank Account Transaction Queue Cleanup` scheduled job runs once each night and cleans up the already processed transaction records with `isWithError = False`.

> **NOTE**
> You can view the transactions within the queue if you have the **Loan Admin Officer** security role.

To view the bank account transactions within the queue, follow these steps:

1. In FintechOS Portal, click the main menu icon and expand the **Admin Configurations** menu.
2. Click the **Bank Account Transaction Queue** menu to open the **Bank Account Transaction Queue** page.



Here you can view all the bank account level transactions that were placed in the queue, no matter their processing status. The columns of the table contain the following information:

- **Contract No** - The contract on which the transaction was performed.
- **Customer Name** - The name of the customer on whose contract the transaction was performed.
- **Event Date** - The date when the transaction event occured.
- **Event Type** - The type of the transaction event.
- **Event No** - The number of the transaction event.
- **Event Amount** - The amount of the transaction event.
- **Is Processed** - The checkbox specifies whether the transaction was already processed by the Bank Account Transaction Queue Processing scheduled job or not.
- **Is With Error** -The checkbox specifies whether there was an error when processing the transaction or not.
- **Error Message** - The text of the error encountered when processing the job if such an error was encountered during processing.

On the **Bank Account Transaction Queue** page, you can search for a specific transaction by filling in any or all the column headers of the displayed records list.

Bank Account Transaction Configurations

Transactions between bank accounts generate debit operations on a source bank account and credit operations on a destination bank account. The bank accounts can be reconciliation accounts (accounts defined at the banking product level serving as source for disbursements or accounts defined at the transaction fee level for collecting fees), user bank accounts (current, deposit or credit accounts), or external bank accounts (accounts from other banks).

In Loan Management, the transactions between bank accounts are created in `Draft` status. When you change a transaction's status changes to `Approved`, Loan Management automatically generates bank account operations: Debit Operations for

the Source Account, and Credit Operations for the Destination Account. It also updates the balance for the source or the destination accounts if they are bank accounts defined within Loan Management.

You can define fees to be added to bank account transactions. Using fee lists, you can attach fees with specified values to each bank account transaction type. The lists can further be filtered. When you select a transaction operation type on a bank account transaction, Loan Management identifies the fee list and fee values and applies them considering the data of the transaction. Loan Management creates operations of debit and credit for both transaction value and fees. The number of credit and debit operations created by Loan Management is managed through the `BankAccountTransactionFeeMarkDown` system parameter.

You can configure the bank account transaction operations and fees by managing the records within FintechOS Portal's dedicated menu, **Admin Configurations > Bank Account Transactions**.

This page contains a series of topics that assist you in configuring how Loan Management manages transactions between bank accounts:

Transaction Types Covered Through Bank Account Transaction Operation Types in Loan Management **92**

Loan Contract Approval Process Flow **95**

Transaction Operation Type **98**

Transaction Fee **102**

Transaction Fee List **104**

Transaction Types Covered Through Bank Account Transaction Operation Types in Loan Management

Loan Management currently covers within its automated processes the following transaction types through the bank account transaction operation types:

Transaction	Code	Usage	Generates Accounting Entry	Is Return Transaction	Is System Transaction	Is Automatic	New Contract Version	Source Entity	Bank Account Transaction Operation Type
Disbursement	DSB	Loan Account	yes	no	no	no	no	Contract Event	Disbursement
Revert Disbursement	RDSB	Loan Account	yes	no	no	yes	no	Contract Event	Disbursement
Repayment	RP	Loan Account	yes	no	no	yes	no	Payment	PaymentIn
Top-Up Account	TPCA	Current Account	yes	no	no	no	no	Contract Event	PaymentIn
Payment Deposit	PD	Deposit Account	yes	no	no	yes	no	Payment Notification	PaymentIn
Withdraw	W	Current/ Deposit Account	yes	no	no	no	no	Contract Event	Payment Out

Transaction	Code	Usage	Generates Accounting Entry	Is Return Transaction	Is System Transaction	Is Automatic	New Contract Version	Source Entity	Bank Account Transaction Operation Type
Transfer between my bank accounts	TR	Current Account	no	no	no	no	no		Payment Out
Early Termination Deposit	RPD	Deposit Account	yes	no	no	yes	no	PaymentNotification	Payment Out
Revert Transfer between my bank accounts	RTR	Current Account	yes	no	no	yes	no	ContractEvent	Payment Out
Deposit Liquidation	DLQ	Deposit Account	yes	no	no	no	no	ContractEvent	Payment Out
Reschedule Overdues	RO	Loan Account	no	no	no	no	yes		RecoverDebt
Repayment Notification	RN	Loan Account	yes	no	no	yes	no	RepaymentNotification	RepaymentContract

Transaction	Code	Usage	Generates Accounting Entry	Is Return Transaction	Is System Transaction	Is Automatic	New Contract Version	Source Entity	Bank Account Transaction Operation Type
Early Repayment	ER	Loan Account	yes	no	no	no	yes	`ContractEvent`	RepaymentContract
Overdraft Payment	ODP	Loan Account	yes	no	no	no	no	`ContractEvent`	RepaymentContract
Returned Amount or Goods	RG	Loan Account	yes	yes	no	no	yes	`ContractEvent`	RepaymentContract
Write Off	WO	Loan Account	no	no	no	no	no	`ContractEvent`	WriteOff

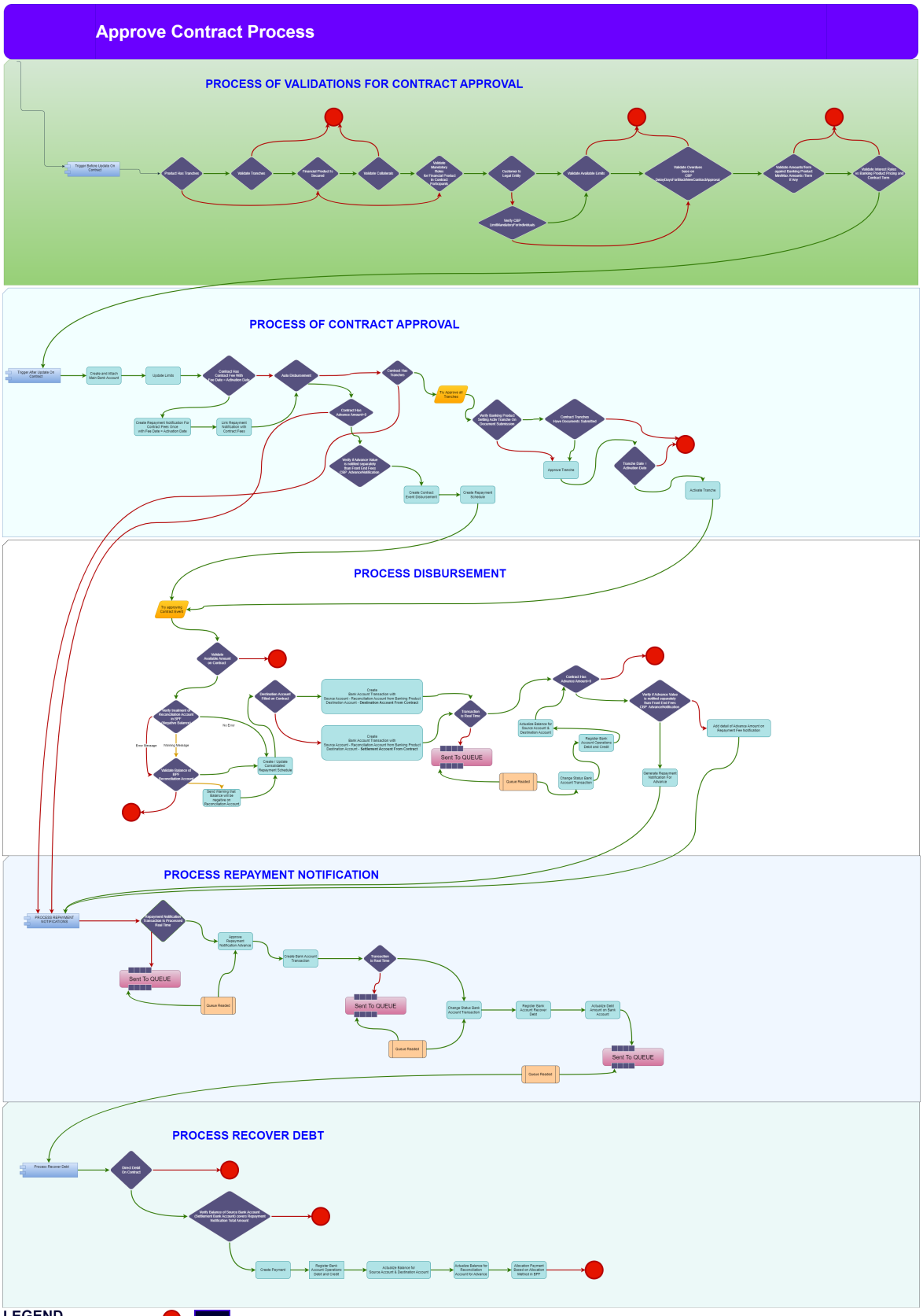
Usually, these transactions are operated behind the scenes, on the server-side, on a higher-order entity like `Contract` (once a credit contract changes its state to `Approved` for auto disbursement) or `ContractEvent` (when the initiated event of transferring money from a current bank account to a deposit bank account of the same user is approved and the selected `Transaction Type = Transfer between my bank accounts`).

Loan Contract Approval Process Flow

Loan contracts in Loan Management go through a series of processes during their life time, from contract approval to full recovery of debts. The following transaction types are illustrated in the below diagram:

- **Loan Contract** - This is the transaction performed when the contract is being validated for approval and actually approved;
- **Disbursement** - This is the transaction performed when the funds are actually delivered from a bank account to the customer;
- **Repayment Notification** - This is the transaction performed when a repayment is generated.
- **Repayment** - This is the transaction performed when the funds are actually collected from the customer and the notification is recovered.

If no external triggers are put in place (such as an early repayment or a return of goods event), then this is the flow applicable for each loan contract in Loan Management. The orchestration of the processes is performed either internally, through Job Server (see Jobs page) or through external request processing.


Approve Contract Process
PROCESS OF VALIDATIONS FOR CONTRACT APPROVAL
PROCESS OF CONTRACT APPROVAL
PROCESS DISBURSEMENT
PROCESS REPAYMENT NOTIFICATION
PROCESS RECOVER DEBT
Sent To QUEUE
Queue Resided
LEGEND
Green Arrow = TRUE
Red Arrow = FALSE
Yellow Arrow = WARNING
BPF = Banking Product

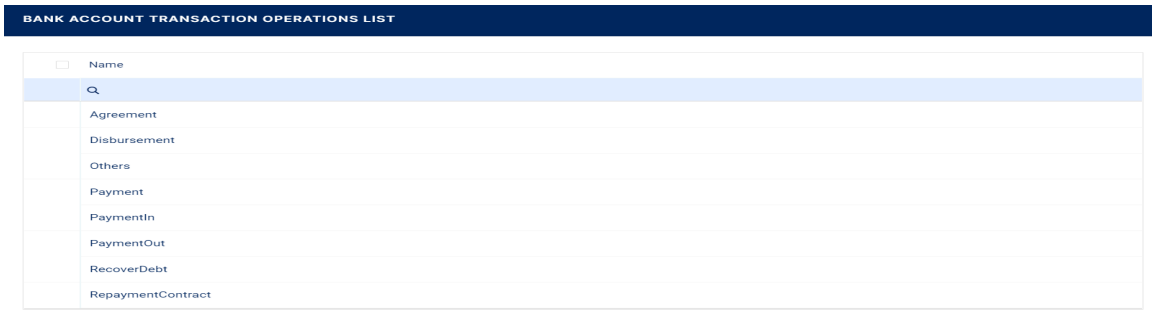
Transaction Operation Type

A bank account transaction is any amount that moves in or out of a bank account. There are different types of operations that affect bank account transactions. Loan Management uses the following types of bank account transaction operations:

- `Payment` – usual transaction operation for a bank account transaction;
- `PaymentIn` – transaction operated into a bank account;
- `PaymentOut` – transaction operated from a bank account;
- `RepaymentContract` – transaction operation type used when a contract repayment is registered. In this case, Loan Management debits the current account of the customer and credits the reconciliation account allocated to the banking product. This is an internal type of bank account transactions operation.
- `Disbursement` – transaction operation type used when a credit contract disbursement is registered. In this case, Loan Management debits the reconciliation account allocated to the banking product and credits the current account of the customer. This is an internal type of bank account transactions operation.
- `RecoverDebt` – transaction operation type used for direct debit in case of loan credit. Loan Management automatically registers a debt for an installment for which the amount in the current account associated to the loan credit doesn't cover the debt amount. This is an internal type of bank account transactions operation.

To manage bank account transaction operation types in Loan Management, follow these steps:

1. In FintechOS Portal, click the main menu icon and expand the **Admin Configurations** menu.
2. Expand **Bank Account Transaction** menu and click **Bank Account Transaction Operation** menu item to open the **Bank Account Transaction Operations List** page.

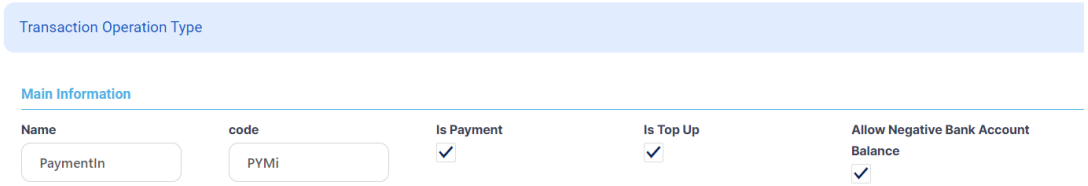


On the **Bank Account Transaction Operations List** page, you can add new bank account transaction operation types or search, edit, and delete existing ones.

Creating Bank Account Transaction Operation Types

Follow these steps to create new bank account transaction operation type records:

1. Click **Insert** on the **Bank Account Transaction Operations List** page to display the **Transaction Operation Type** page.
2. Fill in the following fields:



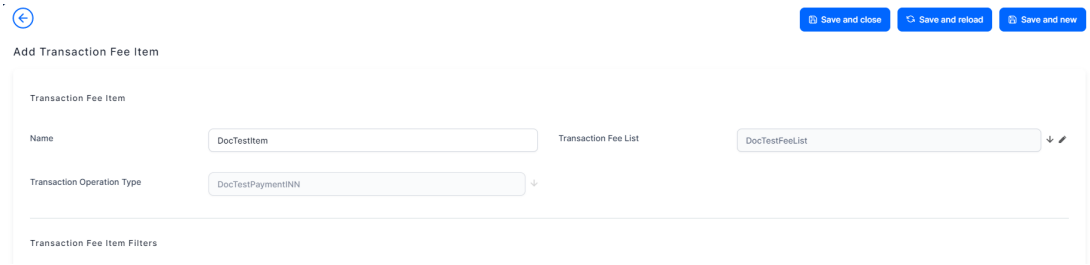
   - **Name** - Enter the name of the bank account transaction operation record.
   - **Code** - Enter a code for this transaction operation record.
   - **Is Payment** - Select this checkbox if the bank account transaction operation record represents a payment within Loan Management.
3. If you marked `Is Payment` as `True`, then fill in these newly displayed fields:
   - **Is Top-Up** - Select this checkbox if the bank account transaction operation record represents a top-up type payment. This field is displayed only if the `Is Payment` field is selected.
   - **Allow Negative Bank Account Balance** - Select this checkbox if the bank account allows the existence of a negative balance. This field is displayed only if the `Is Payment` field is selected.

4. Click the **Save and Reload** button.

   In the **Transaction Fee Items** section displayed after saving the record, you can attach multiple transaction fee items, each serving a different purpose through filters (fees for payment to a specific country, fees made in a specific currency, fees for transfers having a certain minimum or maximum amount). Note that fees can also be combined in fee lists. You can add as many items as you need.

5. To add a new item, click the **Insert** button within the **Transaction Fee Items** section.

6. On the displayed **Add Transaction Fee Item** page, fill in the following fields:

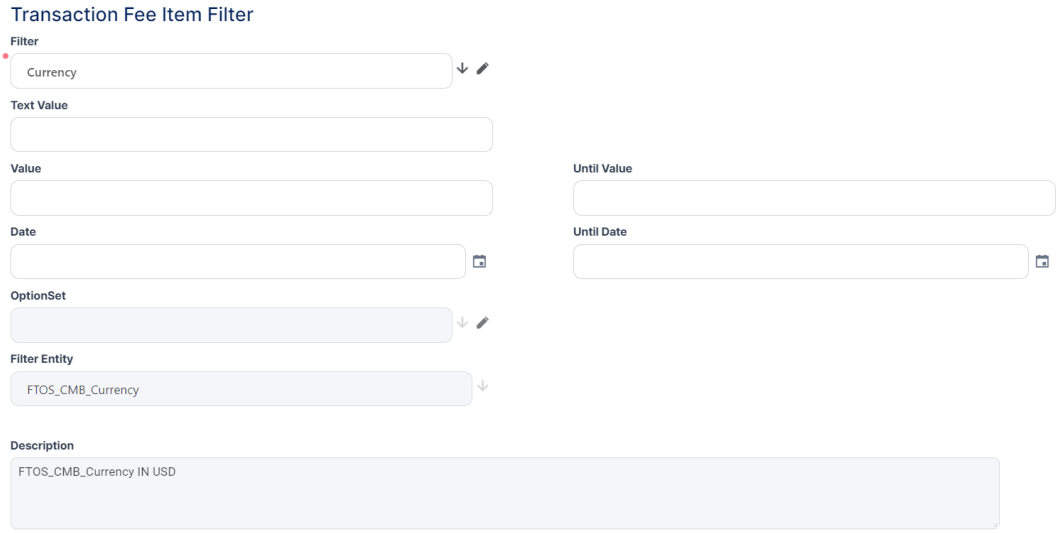


   - **Name** - Enter the name of the transaction fee item record.
   - **Transaction Fee List** - Select the transaction fee list associated to the transaction fee item.
   - **Transaction Operation Type** - Automatically completed with the bank account transaction operation record being edited and it cannot be changed.

7. Click the **Save and Reload** button.
   In the **Transaction Fee Item Filters** section displayed after saving the record, you can add filters for the transaction fee item. You can add as many item filters as you need.

8. To add a new item filter, click the **Insert** button within the **Transaction Fee Item Filters** section.

9. On the newly displayed **Add Transaction Fee Item Filter**, fill in the following fields to specify the filtering criteria for the transaction fee item filter:



- **Filter** - Select the filter for the transaction fee item filter.
- **Description** - Enter a description for the transaction fee item filter.

10. Optionally, fill in the following fields:
    - **Text Value** - Enter the text value of the filter, if applicable.
    - **Value/ Until Value** - Enter the starting/ ending value of the interval for the filter if applicable.
    - **Date/ Until Date** - Enter the starting/ ending date for the filter if applicable.
    - **OptionSet** - This field is automatically completed with the option set of the filter previously selected in the `Filter` field if that filter is of option set type.
    - **Filter Entity** - This field is automatically completed with the entity of the filter previously selected in the `Filter` field if that filter is of entity type.
11. Click the **Save and Close** button.

Transaction Fee

You can define different fees to be applied to bank account transactions. Using fee lists, you can attach fees with specified values to each bank account transaction operation type. When you select a transaction operation type on a bank account transaction, Loan Management identifies the fee list and fee values and applies them considering the current date of the transaction.

To manage transaction fee records:

1. In FintechOS Portal, click the main menu icon and expand the **Admin Configurations** menu.
2. Expand **Bank Account Transaction** menu and click **Transaction Fee** menu item to open the **Transaction Fees List** page.



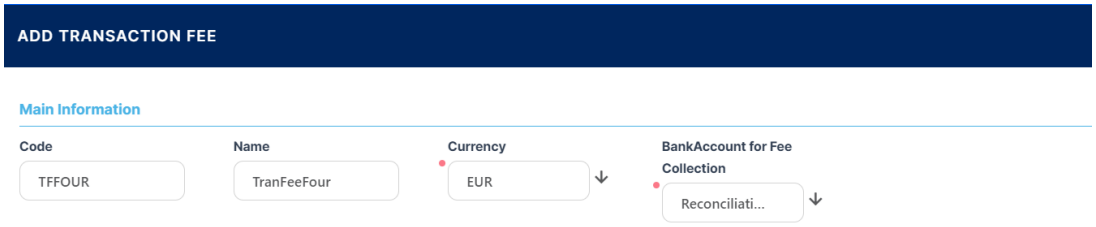
On the **Transaction Fees List** page, you can add new transaction fee records or search, edit, and delete existing ones.

Creating Transaction Fee Records

Follow these steps to create new transaction fee records:

1. Click **Insert** on the **Transaction Fees List** page to display the **Add Transaction Fee** page.

2. Fill in the following fields:



Transaction Fee Values

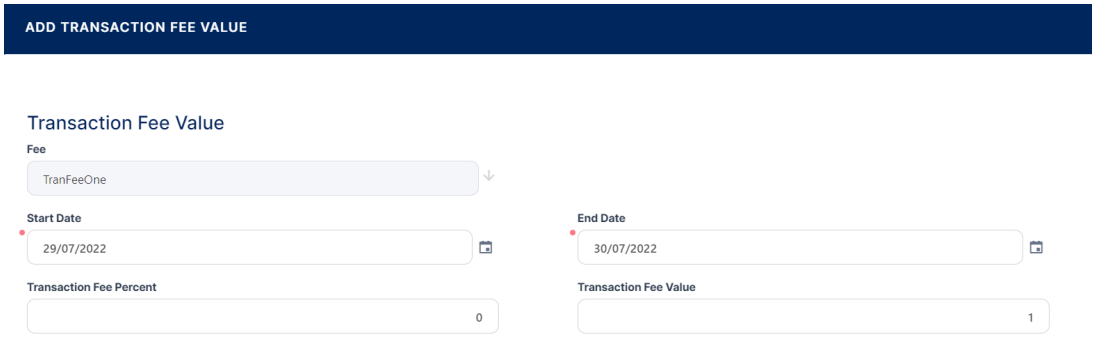
- Code - Enter a code for this transaction fee record.
- Name - Enter the name of the transaction fee record.
- Currency - Select the currency of the fee.
- Bank Account for Fee Collection - Select the bank account for the fee collection.

3. Click the **Save and Reload** button.

   You can manage the values of the transaction fee within the newly displayed **Transaction Fee Values** section.

4. Click **Insert** within the **Transaction Fee Values** section to add a new value for the fee. The **Add Transaction Fee Value** page is displayed.

5. Fill in the following fields:



- **Fee** - Automatically completed with the transaction fee for which you are inserting values.
- **Start Date/ End Date** - Select the interval when the value is active.

- **Transaction Fee Percent** - Enter the percent from the bank account transaction applied as fee value if the value is defined as a percentage.

  OR

- **Transaction Fee Value** - The value of the transaction fee, expressed in the transaction fee currency if the value is not defined as percentage.

6. Click the **Save and Close** button. The new value for the transaction fee is saved.

> **NOTE**
>
> You can add as many values as needed, as long as the validity periods of the value don't overlap. The fee values are identified and applied considering the current date of the transaction.

Transaction Fee List

Loan Management uses lists to group previously defined transaction fees. The transaction fee lists are attached to each bank account transaction type. The lists can further be filtered. When you select a transaction operation type on a bank account transaction, Loan Management identifies the fee list and fee values and applies them considering the current date of the transaction.

To manage transaction fee list records:

1. In FintechOS Portal, click the main menu icon and expand the **Admin Configurations** menu.
2. Expand **Bank Account Transaction** menu and click **Transaction Fee List** menu item to open the **Transaction Fees Lists List** page.

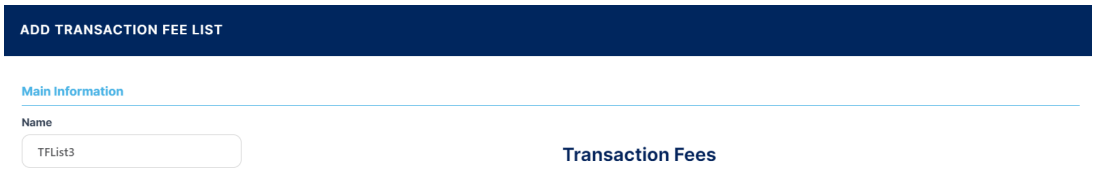


On the **Transaction Fees Lists List** page, you can add new transaction fee list records or search, edit, and delete existing ones.

Creating Transaction Fee List Records

Follow these steps to create new transaction fee list records:

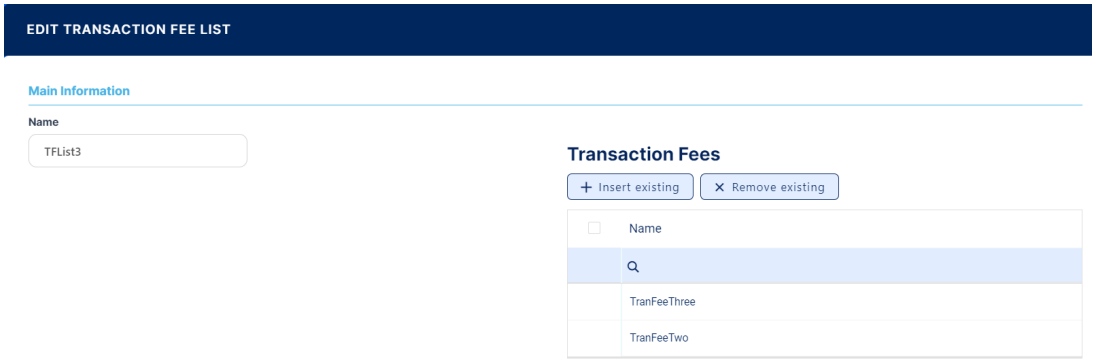
1. Click the **Insert** button on the **Transaction Fees Lists List** page to open the **Add Transaction Fee List** page.
2. Fill in the name of the transaction fee list record.



3. Click the **Save and Reload** button.

   A new section, **Transaction Fees**, is displayed after saving the record. You can manage the actual transaction fees of the list within this section.
4. Click **Insert existing** in the **Transaction Fees** section to add a new transaction fee to the list.
5. In the newly displayed pop-up window, select one or more transaction fees and

click **OK** to attach them to your list.



6. Click the **Save and Close** button. You can add as many transaction fees as needed to your list.

Jobs

Jobs are automated procedures that perform certain tasks, running at a specific time or on a recurring schedule. Read detailed information about scheduling jobs in the FintechOS Studio User Guide's dedicated page.

Loan Management comes with the following scheduled jobs that perform repetitive banking procedures for the purpose of closing one day and opening the next one, or for processing or cleaning up queued transactions:

Start Of Day (SOD) Job

The **Loan Management Start Of Day** job is made up of a series of services, which are run one by one each day before the banking day is initiated.

The following services run as part of this job, in this order:

1. **Start Job Log**

   If another SOD or EOD job is not currently running within Loan Management, the service inserts a record into the `ScheduleJobLog` entity, containing information about the job starting time.

2. **Disburse Contracts With Approved Tranches In Current Date**
   The service performs the disbursements of contract tranches approved on the current date (the day that is about to be opened/ initiated).
3. **Set Limit Available Amount Due To FX Change**
   The service recalculates the available amount on limits depending on the currency's exchange rate on a specific day. The service does not take into consideration past versions of the limit up to a given day. It only takes into consideration the current limit version and uses those results for calculating the limit available value.
4. **Set Collateral Available Amount And Contract Collateral Value Due To FX Change**
   The service recalculates the available amount on collaterals and the collateral value on contracts depending on the currency's exchange rate on a specific day.
5. **Set Contract Amount (Overdraft) Due To Plan Due Date Reached (Increase/ Decrease)**
   The service increases/ decreases the limit amount on overdraft contracts that reached their reevaluation plan due date.
6. **Set Credit Facility Amount Due To Plan DueDate Reached (Increase/ Decrease)**
   The service increases/ decreases the limit amount on credit facilities that reached their reevaluation plan due date.
7. **Set Credit Facility Available Amount Due To FX Change**
   The service recalculates the available amount on credit facilities depending on the currency's exchange rate on a specific day.
8. **Set Contract Covenant Status Due To Review Date Reached**

   The service changes the status of covenants on contracts that reached their review date.
9. **Set Contract Category Based on Overdue Days**
   The service changes the category of contracts based on overdue days.
10. **Schedule versioning after stop accrual**
    The service creates new schedule versions for the contracts that were marked for stopping the accrual process calculation. The new version

of the schedule has the `Process of Loss` reason and all the un-notified installments are updated.

11. **Calculate Accrual For Overdraft Contracts and Current Accounts With Overdraft Contracts**
    The service recalculates the accrual for utilization and for unusage for overdraft contracts and for current account with overdraft contracts.
12. **Calculate Accrual For Credit Facility**
    The service recalculates the accrual for utilization and for unusage for credit facilities.
13. **Calculate Accrual And Provisions For Loan Contracts**
    The service recalculates the accrual and provisions for loan contracts.
14. **Calculate Overdraft Debt to recover**
    The service recalculates the overdraft debt still to recover from contracts.
15. **Calculate Penalties for Contracts and Credit Facility**
    The service recalculates the penalties for contracts and for credit facilities.
16. **Calculate Interests For Deposit Contracts**
    The service recalculates the interests for deposit contracts.
17. **End Job Log**
    The service updates the same record that the **Start Job Log** service inserted within the `ScheduleJobLog` entity, writing information about the job ending time.

End Of Day (EOD) Job

The **Loan Management End Of Day** job is made up of a series of services, which are run one by one each day after the banking day is closed.

The following services run as part of this job, in this order:

1. **Start Job Log**

   If another SOD or EOD job is not currently running within Loan Management, the service inserts a record into the `ScheduleJobLog` entity, containing information about the job starting time.

2. **Set Limit Expired**
   The service sets the limits which are about to expire in the current day as Expired.

3. **Set Accrual Amounts on Repayment Schedule for Overdraft Contract and Current Account with Overdraft**
   The service recalculates the accrual for utilization on repayment schedules for overdraft contracts and for current account with overdraft contracts.

4. **Run Recover Debts**
   The service performs the direct debit transactions to recover all debts at that point in process.

5. **Set Accrual Amounts Unusage Commission Values on Repayment Schedule for Contracts**
   The service recalculates the accrual for unusage commissions on repayment schedules for contracts.

6. **Update Expiry Date for Overdraft installment**
   The service updates all the expiry dates for overdraft installments for the current account with overdraft contracts.

7. **Generate Repayment Notifications**
   The service generates the repayment notifications for all the contracts with due amounts on schedule on that day.

8. **Recover Debts After Generate Notification**
   The service performs the direct debit transactions to recover all debts after generating the repayment notifications.

9. **Run Payment Allocation**
   The service performs the payment allocation operations for unallocated or partially allocated payments.

10. **Generate Accounting Entries**
    The service generates the accounting entries for all transactions performed on that day.
11. **Set Limit Available Amount Due To FX Changes**
    The service sets the limit amounts available to all contracts due to exchange rates changes.
12. **End Job Log**
    The service updates the same record that the **Start Job Log** service inserted within the `ScheduleJobLog` entity, writing information about the job ending time.

Bank Account Transaction Queue Processing (CB) Job

The **Bank Account Transaction Queue Processing** scheduled job processes all the bank account transactions that were placed in a queue instead of being processed in real-time at transaction approval. Each time a transaction is performed on a bank account, the system verifies its transaction type's **Real Time Process** field. If the value is `True`, then the transaction is processed right away. If the value is `False`, then the transaction is inserted as a record in the `BankAccountTransactionQueue` entity, with the `isProcessed` attribute set to `False` and `isWithError` set to `False`. The **Bank Account Transaction Queue Processing** scheduled job runs every 1 minute, taking the top 10 records from the entity with the attribute `isProcessed = False`, and processing the transactions. Any errors encountered on processing are logged in the `errorMessage` attribute.

The following service runs as part of this job, in this order:

1. **Start Job Log**

   If another SOD or EOD job is not currently running within Loan Management, the service inserts a record into the `ScheduleJobLog` entity, containing information about the job starting time.
2. **BankAccountTransactionQueue_Process**
   The service takes the top 10 records from the bank account transactions queue (the `BankAccountTransactionQueue` entity)

with the attribute `isProcessed = False`, and processes the transactions. After each processing, the `isProcessed`, `isWithError`, and `errorMessage` attributes are updated.

3. **End Job Log**
   The service updates the same record that the **Start Job Log** service inserted within the `ScheduleJobLog` entity, writing information about the job ending time.

Bank Account Transaction Queue Cleanup (COB) Job

The **Bank Account Transaction Queue Cleanup** scheduled job runs once each night and deletes the already processed transaction records with the `isWithError` attribute value `False.`

The following service runs as part of this job:

1. **BankAccountTransactionQueue_Cleanup**
   The service deletes the records from the bank account transactions queue (the `BankAccountTransactionQueue` entity) with the attribute `isProcessed = True` and `isWithError = False`.

Close Contracts (COB) Job

The **Close Contracts** scheduled job closes automatically all contracts with `Automatic Closure = True` and `Real Time Closure = False`, with zero available amount and with no further amounts to be recovered, that have `Balance Off Date` filled in and `Closure Date = Current Date`. The job runs by default at 11:00 PM system time.

The following services run as part of this job, in this order:

1. **Start Job Log**

   If another SOD or EOD job is not currently running within Loan Management, the service inserts a record into the `ScheduleJobLog` entity, containing information about the job starting time.

2. **CloseContract**
   The service closes all contracts with `Automatic Closure = True` and `Real Time Closure = False`, with zero available amount and with no further amounts to be recovered, that have `Balance Off Date` filled in and `Closure Date = Current Date`.
3. **End Job Log**
   The service updates the same record that the **Start Job Log** service inserted within the `ScheduleJobLog` entity, writing information about the job ending time.

Close Contracts RealTime(CB) Job

The **Close Contracts RealTime** scheduled job closes automatically all contracts with `Automatic Closure = True` and `Real Time Closure = True`, with zero available amount and with no further amounts to be recovered. The job runs by default every 5 seconds, between 06:00 AM and 07:59 PM system time.

The following services run as part of this job, in this order:

1. **Start Job Log**

   If another SOD or EOD job is not currently running within Loan Management, the service inserts a record into the `ScheduleJobLog` entity, containing information about the job starting time.
2. **CloseContractRealTime**
   The service closes all contracts with `Automatic Closure = True` and `Real Time Closure = True`, with zero available amount and with no further amounts to be recovered.
3. **End Job Log**
   The service updates the same record that the **Start Job Log** service inserted within the `ScheduleJobLog` entity, writing information about the job ending time.

Run Loan Payment Allocation (CB) Job

The **Run Loan Payment Allocation** scheduled job processes almost in real-time the payment allocation operations for payments with **Unallocated** or **Partially Allocated** statuses. The job runs by default every 5 seconds, between 06:00 AM and 07:59 PM system time.

The following services run as part of this job, in this order:

1. **Start Job Log**

   If another SOD or EOD job is not currently running within Loan Management, the service inserts a record into the `ScheduleJobLog` entity, containing information about the job starting time.

2. **Debt_to_Payment**
   The service processes the payment allocation operations for unallocated or partially allocated payments.

Auto Process Manual Repayment Notifications (CB) Job

The **Auto Process Manual Repayment Notifications (CB)** scheduled job facilitates the automatic transition of manual repayment notifications from the **Approved** status into **Pending Recover** or **In Recovery** statuses, thus allowing manual notifications to be automatically processed for payment allocation. The job runs by default at 7:00 PM system time.

The following service runs as part of this job:

1. **Auto Process Manual Repayment Notifications**
   The service processes the manual repayment notification record in **Approved** status created on the current date of the system, changing their status depending on the direct debit setting at the customer level. If the `Direct Debit Settlement Account` field at the contract level = `True`, then the manual notification's status changes to **In Recovery**, otherwise it changes to **Pending Recover**.

Async Contract Approval (CB) Job

The **Async Contract Approval (CB)** scheduled job performs all the contract approval-related activities, such as generating a main bank account, updating the contract with the main bank account number, if the `Auto Disbursement = True` then perform the disbursement event, generate a repayment schedule, generate and then process repayment notifications, when the `AsyncContractApproval` Loan Management system parameter is set to `True`. The job runs by default every 5 seconds.

The following service runs as part of this job:

1. **Async Contract Approval Queue Processing**
   The service processes the records from the `Async Contracts Approval Queue` entity, performing every task related to a contract approval, except the change of business status of the contract to `Approved`, which happened already.

Async Limit Recalculation Job

The **Async Limit Recalculation** scheduled job performs the limit recalculation for the limit/ customer/ group queued in the `Daily Limit Recalculation Queue` entity, when the `AsyncLimitRecalculation` Loan Management system parameter is set to `True`. The job runs by default every 10 seconds.

The following service runs as part of this job:

1. **FTOS_CB_DailyLimitRecalculationFromQueue**
   The service processes the records from the `Daily Limit Recalculation Queue` entity, performing the limit recalculation for the limit/ customer/ group queued in the entity.

Process Contracts For Auto WriteOff Job

The **FTOS_CB_ProcessContractsForAutoWriteOff** scheduled job automatically performs write-off transactions for non-performing loan contracts, after their maturity day passes and the remaining value on the contract is under the threshold specified at the banking product level. You can enable the job to run according to your financial institution's needs,

usually at the end of day.

The following service runs as part of this job:

1. **FTOS_CB_ProcessContractsForAutoWriteOff**
   The service first identifies the non-performing loan contracts with notified amounts still not recovered. Then, it calculates each contract's maximum value for the write-off transaction as `the Remaining Principal (absolute main balance) + Accrued Interest + unpaid values from Repayment Notifications (interest, commissions, penalties, or insurances)`, and identifies which values are under the write-off threshold defined at the banking product level, placing the affected contracts in a processing queue. Lastly, the service processes the queue items and performs write-off transactions for the identified contracts, taking the remaining amount from the specified internal write-off account and crediting the loan's main bank account for the loan principal, respectively the product's self bank account for all the other operation items.

Security Roles for Loan Management

A security role is a set of privileges and levels of access to various actions/ functions within the FintechOS Platform. Read the Security Roles topic for detailed information.

Loan Management has a set of predefined security roles specific for banking-related business needs. Financial institutions can use these security roles to grant their employees access rights within its systems. Read the Users topic for information about associating security roles to users in FintechOS Studio.

You can also create new security roles to fit your financial institution's business needs.

Predefined Loan Management Security Roles

Loan Management comes with the following predefined security roles that grant specific access rights to the users associated with one or more of these roles to the FintechOS Platform's actions and functions:

- **Loan Admin Officer** - Users with this security role have `read, write` and `authorize` access rights to the **Loans** and **Accounting** records in **all modules** of Loan Management within **their organization**.
- **Accounting Read** - Users with this security role have `read` access rights to the **Accounting** module of Loan Management within **their organization**.
- **Corporate Credit Officer** - Users with this security role have access rights to `read` and `write` access rights to the **Contract**, **Contract Version**, **Contract Event**, **Limits**, **Collateral Register**, **Credit Facility** and **Credit Facility Version** records of Loan Management within **their organization**.
- **Retail Credit Officer** - Users with this security role have access rights to `read` and `write` access rights to the **Contract**, **Contract Version**, **Contract Event**, **Collateral Type** and **Collateral Register** records of Loan Management within **their organization**.
- **Accounting Officer** - Users with this security role have `read` and `write` access rights to the **Accounting** module of Loan Management within **their organization**.
- **Supervisor Risk Officer** - Users with this security role have `read` access rights to the **Contracts**, **Collateral Register** and **Credit Facility** records of Loan Management within **their organization**, and `authorization` rights to the **Limits** records, all within **their organization**.
- **Risk Officer** - Users with this security role have `read` access rights to the **Contracts**, **Collateral Register** and **Credit Facility** records, and `write` access rights to the **Limits** records of Loan Management within **their organization**.
- **Async Engine Admin** - Users with this security role can access the **DLQ Messages** dashboard and resend unprocessed requests to the Async Engine.

IMPORTANT!
If your user has more than one role, then your access rights are cumulative and you are granted the highest access right.

Read the following sections for detailed information about each security role's access rights to entities and endpoints within the FintechOS Platform:

Loan Admin Officer

A user with this security role has the following access rights to records in the FintechOS Platform's entities within **their organization**:

Entity	Insert	Read	Update	Delete
Account	No	Yes	Yes	Yes
AccountRelOwnership	No	Yes	No	Yes
Address	No	Yes	No	Yes
approvalTask	No	Yes	No	Yes
businessunit	Yes	Yes	Yes	Yes
BWstatus	Yes	Yes	Yes	Yes
entity	Yes	Yes	Yes	No
entitystatus	Yes	Yes	Yes	Yes
Division	No	Yes	No	Yes
Document	No	Yes	No	Yes
GroupAccount	Yes	Yes	Yes	Yes
GroupMember	No	Yes	No	Yes
UnitType	No	Yes	No	Yes
BandedInterest	No	Yes	No	Yes
BankingProduct	No	Yes	No	Yes
BankingProductAgreement	No	Yes	No	Yes
BankingProductDiscount	No	Yes	No	Yes
BankingProductDocument	No	Yes	No	Yes
BankingProductMandatoryRoles	Yes	Yes	Yes	Yes
BankingProductType	No	Yes	No	No
ClassificationType	No	Yes	No	Yes
CollateralType	Yes	Yes	Yes	Yes
Commission	Yes	Yes	Yes	Yes
CommissionSchema	Yes	Yes	Yes	Yes
CommissionType	Yes	Yes	Yes	Yes
CommissionValue	Yes	Yes	Yes	Yes

Entity	Insert	Read	Update	Delete
Covenant	Yes	Yes	Yes	Yes
Feature	No	Yes	No	Yes
Formula	No	Yes	No	Yes
FormulaType	Yes	Yes	Yes	Yes
GLAccounts	Yes	Yes	Yes	Yes
InsuranceItem	No	Yes	No	Yes
Interest	No	Yes	No	Yes
InterestCommissionItem	Yes	Yes	Yes	Yes
InterestCommissionItemFilter	Yes	Yes	Yes	Yes
InterestRateMatrix	No	Yes	No	Yes
InterestValue	No	Yes	No	Yes
PaymentScheduleType	No	Yes	No	Yes
ProductAvailabilityItemFilter	No	Yes	No	Yes
ProductClassification	No	Yes	No	Yes
ProductCovenant	No	Yes	No	Yes
ProductDestinationType	No	Yes	No	Yes
ProductDisbursement	No	Yes	No	Yes
ProductGuaranteeStructure	No	Yes	No	Yes
ProductQuestion	No	Yes	No	Yes
TestScenario	No	Yes	No	Yes
BankAccount	Yes	Yes	Yes	Yes
BankAccountOperation	Yes	Yes	Yes	Yes
CollateralRegister	Yes	Yes	Yes	Yes
CollateralRegister_BW	Yes	No	Yes	Yes
CollateralRegisterDocuments	Yes	Yes	Yes	Yes
CollateralRegisterOwner	Yes	Yes	Yes	Yes
CollateralRegisterParticipants	Yes	Yes	Yes	Yes
CollateralRegisterRank	Yes	Yes	Yes	Yes
Contract	Yes	Yes	Yes	Yes
Contract_BW	Yes	Yes	Yes	Yes
ContractAccrualAndProvision	Yes	Yes	Yes	Yes
ContractBorrowers	Yes	Yes	Yes	Yes
ContractClassification	Yes	Yes	Yes	Yes
ContractCollateral	Yes	Yes	Yes	Yes
ContractCorrectionEntry	Yes	Yes	Yes	Yes
ContractCorrectionEntry_BW	Yes	No	Yes	Yes
ContractCorrectionEntryDetail	Yes	Yes	Yes	Yes
ContractCovenant	Yes	Yes	Yes	Yes

Entity	Insert	Read	Update	Delete
ContractCovenant_BW	Yes	Yes	Yes	Yes
ContractDisbursement	Yes	Yes	Yes	Yes
ContractDisbursementTranche	Yes	Yes	Yes	Yes
ContractDisbursementTranche_BW	Yes	Yes	Yes	Yes
ContractDiscount	Yes	Yes	Yes	Yes
ContractDocument	Yes	Yes	Yes	Yes
ContractDocument_BW	Yes	Yes	No	Yes
ContractEvent	Yes	Yes	Yes	Yes
ContractEvent_BW	Yes	No	Yes	Yes
ContractExtended	Yes	Yes	Yes	Yes
ContractFee	Yes	Yes	Yes	Yes
ContractGuarantor	Yes	Yes	Yes	Yes
ContractInterestRate	Yes	Yes	Yes	Yes
ContractParticipant	Yes	Yes	Yes	Yes
ContractParticipant_BW	Yes	Yes	Yes	Yes
ContractPenalty	Yes	Yes	Yes	Yes
ContractReevaluation	Yes	Yes	Yes	Yes
ContractRepaymentSchedule	Yes	Yes	Yes	Yes
ContractRepaymentScheduleDetail	Yes	Yes	Yes	Yes
ContractRepaymentScheduleDisb	Yes	Yes	Yes	Yes
ContractRepaymentScheduleDisbDet	Yes	Yes	Yes	Yes
ContractRepaymentScheduleVersion	Yes	Yes	Yes	Yes
ControlPanel	No	Yes	No	No
CovenantResolution	Yes	Yes	Yes	Yes
CreditFacility	No	Yes	No	Yes
CreditFacilityAccrual	No	Yes	No	Yes
CreditFacilityDetail	Yes	Yes	Yes	Yes
CreditFacilityFee	No	Yes	No	Yes
CreditFacilityFeeValue	No	Yes	No	Yes
CreditFacilityParticipant	No	Yes	No	Yes
CreditFacilityPlan	No	Yes	No	Yes
CreditFacilityProduct	No	Yes	No	Yes
CustomerLimit	Yes	Yes	Yes	Yes
CustomerLimit_BW	Yes	Yes	Yes	Yes
CustomerLimitType	Yes	Yes	Yes	Yes
DailyContractInterest	Yes	Yes	Yes	Yes

Entity	Insert	Read	Update	Delete
DelayCategory	Yes	Yes	Yes	Yes
DepositValueCalculation	Yes	Yes	Yes	Yes
Holiday	Yes	Yes	Yes	Yes
OperationItem	No	Yes	No	Yes
Payment	Yes	Yes	Yes	Yes
PaymentAllocation	Yes	Yes	Yes	Yes
PaymentNotification	Yes	Yes	Yes	Yes
PeriodicityType	Yes	Yes	Yes	Yes
ReconciliationAccountSettings	Yes	Yes	Yes	Yes
RepaymentNotification	No	Yes	No	Yes
RepaymentNotificationDetail	No	Yes	No	Yes
SalesChannel	No	Yes	No	No
SystemParameter	Yes	Yes	Yes	Yes
TransactionOperationType	Yes	Yes	Yes	Yes
VersioningReason	No	Yes	No	No
WeekDay	Yes	Yes	Yes	Yes
AccountType	No	Yes	No	Yes
Action	No	Yes	No	Yes
Activity	No	Yes	No	Yes
Country	No	Yes	No	Yes
Currency	Yes	Yes	Yes	Yes
EntityStatusSettings	Yes	Yes	Yes	Yes
AccountingChart	Yes	Yes	Yes	Yes
AccountingEntry	Yes	Yes	Yes	Yes
AccountingJournal	Yes	Yes	Yes	Yes
AccountingScope	Yes	Yes	Yes	Yes
AccountingSystem	Yes	Yes	Yes	Yes
Journal	Yes	Yes	Yes	Yes
LegalEntity	Yes	Yes	Yes	Yes
LegalEntitySystem	Yes	Yes	Yes	Yes
OperationTransaction	Yes	Yes	Yes	Yes
OperationTransactionValue	Yes	Yes	Yes	Yes
TransactionAccountingModel	Yes	Yes	Yes	Yes
TransactionItemAccountingConfig	Yes	Yes	Yes	Yes
TransactionType	Yes	Yes	Yes	Yes
TransactionValueType	Yes	Yes	Yes	Yes
Activity	No	Yes	No	Yes
Agreement	Yes	Yes	Yes	Yes

Entity	Insert	Read	Update	Delete
Agreement_BW	Yes	Yes	Yes	Yes
AgreementPricing	Yes	Yes	Yes	Yes
Invoice	Yes	Yes	Yes	Yes
Invoice_BW	Yes	Yes	Yes	Yes
InvoiceDetail	Yes	Yes	Yes	Yes
VersionSettings	Yes	Yes	Yes	Yes
optionset	No	Yes	No	Yes
optionsetitem	No	Yes	No	Yes
systemuser	Yes	Yes	Yes	Yes

A user with this security role can access the following endpoints:

Endpoint
FTOS_BP_CheckFromToDates
FTOS_BP_CheckMandatoryRoleXLimitType
FTOS_BP_CommissionSchemaDetail
FTOS_BP_GetBankingProductDefaultValues
FTOS_BP_GetBankingProductInfo
FTOS_BP_GetBankingProductPeriod
FTOS_BP_GetBankingProductTypeValues
FTOS_BP_GetCommissionInfo
FTOS_BP_GetCommissionTypeInfo
FTOS_BP_GetInterestCommissionItem
FTOS_BP_GetPeriodicityTypeInfo
FTOS_BP_GetProductInterestCommissionList
FTOS_CB_AddUpdateContractPaymentHoliday
FTOS_CB_CalculateContractCustomValues
FTOS_CB_CalculateContractCustomValues_ForDisbursement
FTOS_CB_CalculateInvoiceAmounts
FTOS_CB_CalculateMaturityDate_BA
FTOS_CB_CheckCustomerRole
FTOS_CB_CheckInvoiceHasDetails
FTOS_CB_CheckLimitTypeRole
FTOS_CB_CheckScheduleImported
FTOS_CB_ContractCommissionAmountCalc
FTOS_CB_CountContractInterests
FTOS_CB_DeleteDocument
FTOS_CB_DeleteRepaymentSchedule
FTOS_CB_Disbursement_WarningInsuficientFunds

Endpoint
FTOS_CB_DisplayFinancedAmountEventForm
FTOS_CB_EditContractPenaltyInterestRate
FTOS_CB_EntityVersion_Agreement
FTOS_CB_EntityVersion_Contract
FTOS_CB_EntityVersion_CustomerLimit
FTOS_CB_getBandedInterestObject
FTOS_CB_GetBlockAmountOnContract
FTOS_CB_GetClosureOfContracts
FTOS_CB_GetCommissionDetail
FTOS_CB_GetContractCollateralInfo
FTOS_CB_GetContractEventEditUrl
FTOS_CB_GetContractEventRepaymentScheduleIds
FTOS_CB_GetContractExtendedData
FTOS_CB_GetContractInfo
FTOS_CB_GetContractInterestRates
FTOS_CB_GetContractPeriodicity
FTOS_CB_GetContractRepaymentSchedule
FTOS_CB_GetContractsForLimit
FTOS_CB_GetContractsWithPaymentHolidayPossibility
FTOS_CB_GetCreditFacilityLimitPercent
FTOS_CB_GetDataSourceChartAgreement
FTOS_CB_GetDataSourceChartContractOverview
FTOS_CB_GetDataSourceChartCreditFacility_LoanOfficerAdminRetail
FTOS_CB_GetDataSourceClosingContractsChart
FTOS_CB_GetDataSourceFutureInstallmentsReport
FTOS_CB_GetDataSourceNewContractsChart
FTOS_CB_GetDataSourcePastDueInstallmentsReport
FTOS_CB_GetDisbursementRepaymentSchedule
FTOS_CB_GetEarlyRepaymentValues
FTOS_CB_GetExchangeRate
FTOS_CB_GetExchangeRateOfCollateralRegisterToContract
FTOS_CB_GetFixedVariableInterest
FTOS_CB_GetGLOnContract
FTOS_CB_GetGroupInfo
FTOS_CB_GetInstallment_Principal_InstallmenNo_Values
FTOS_CB_GetInterestReferencePeriod
FTOS_CB_GetInvoiceDetails
FTOS_CB_GetLimitTypeByCustomer

Endpoint
FTOS_CB_GetNoOfContractTranches
FTOS_CB_GetNoOfCurrentAccountsForCurrencyId
FTOS_CB_GetProductDetails
FTOS_CB_GetProductInterestValue
FTOS_CB_GetProductMinInterestRate
FTOS_CB_GetPurgeableRecordsByCode
FTOS_CB_GetReasonDetails
FTOS_CB_GetRemainingNotifications
FTOS_CB_GetSalesChannelByName
FTOS_CB_GetScheduleColumns
FTOS_CB_GetSoonToExpireCurrentAccountWithOverdrafts
FTOS_CB_GetSystemInvariantDate
FTOS_CB_GetSystemParameter
FTOS_CB_GetTransactionTypeByCode
FTOS_CB_GetWorkingDate
FTOS_CB_PaymentHolidaySchedule
FTOS_CB_PaymentScheduleFields
FTOS_CB_PaymentScheduleFieldsDisb
FTOS_CB_PeriodicityType
FTOS_CB_ProcessAccrualsAndProvisions
FTOS_CB_RecalculateEarlyRepaymentSchedule
FTOS_CB_RecalculateSchedule
FTOS_CB_RecalculateScheduleOnDemand
FTOS_CB_releaseCollaterals
FTOS_CB_SetInterestRate
FTOS_CB_UpdateActivationDate
FTOS_CB_UpdateTranche
FTOS_CB_ValidateOverdraftExpireFields
FTOS_CB_VerifyInsertedContractInterestRates
FTOS_CheckLicense
FTOS_GetDataSourceChartContract
FTOS_GetDataSourceChartContractEvents
FTOS_GetDataSourceChartCreditFacility
FTOS_GL_CheckToBePurgedTransaction
FTOS_GL_GenerateAccountingEntry
FTOS_GL_GetTransactionTypeDetails
FTOS_GL_GetTransactionTypeDetailsBulk
FTOS_TPM_CalculateCommissionAppliedTo

Endpoint
FTOS_TPM_CallFormula
FTOS_TPM_GetAgreementBusinessStatusDisplayName
FTOS_TPM_GetAgreementCommissionsDetails
FTOS_TPM_GetAgreementDetails
FTOS_TPM_GetInvoiceDetailContracts
FTOS_TPM_GetTransactionTypesForClawback
FTOS_VerifyUsersCompetence

Accounting Read

A user with this security role has the following access rights to records in the FintechOS Platform's entities within **their organization**:

Entity	Insert	Read	Update	Delete
Account	No	Yes	No	No
approvalTask	No	Yes	No	No
businessunit	No	Yes	No	No
entity	No	Yes	No	No
entitystatus	No	Yes	No	No
GroupMember	No	Yes	No	No
BankingProduct	No	Yes	No	No
BankingProductType	No	Yes	No	No
Commission	No	Yes	No	No
CommissionSchema	No	Yes	No	No
CommissionType	No	Yes	No	No
CommissionValue	No	Yes	No	No
FormulaType	No	Yes	No	No
GLAccounts	No	Yes	No	No
Interest	No	Yes	No	No
PaymentScheduleType	No	Yes	No	No
ProductClassification	No	Yes	No	No
BankAccount	No	Yes	No	No
BankAccountOperation	No	Yes	No	No
Contract	No	Yes	No	No
ContractAccrualAndProvision	No	Yes	No	No
ContractClassification	No	Yes	No	No
ContractCollateral	No	Yes	No	No
ContractCovenant	No	Yes	No	No

Entity	Insert	Read	Update	Delete
ContractDisbursementTranche	No	Yes	No	No
ContractDocument	No	Yes	No	No
ContractEvent	No	Yes	No	No
ContractFee	No	Yes	No	No
ContractGuarantor	No	Yes	No	No
ContractParticipant	No	Yes	No	No
ContractPenalty	No	Yes	No	No
ContractReevaluation	No	Yes	No	No
ContractRepaymentSchedule	No	Yes	No	No
ContractRepaymentScheduleDetail	No	Yes	No	No
ContractRepaymentScheduleVersion	No	Yes	No	No
CreditFacility	No	Yes	No	No
CreditFacilityAccrual	No	Yes	No	No
CreditFacilityDetail	No	Yes	No	No
CreditFacilityFee	No	Yes	No	No
CreditFacilityFeeValue	No	Yes	No	No
CreditFacilityParticipant	No	Yes	No	No
CreditFacilityPlan	No	Yes	No	No
CreditFacilityProduct	No	Yes	No	No
CustomerLimit	No	Yes	No	No
CustomerLimit_BW	No	Yes	No	No
CustomerLimitType	No	Yes	No	No
OperationItem	No	Yes	No	No
Payment	No	Yes	No	No
PaymentNotification	No	Yes	No	No
PeriodicityType	No	Yes	No	No
RepaymentNotification	No	Yes	No	No
TransactionOperationType	No	Yes	No	No
VersioningReason	No	Yes	No	No
WeekDay	No	Yes	No	No
AccountType	No	Yes	No	No
Currency	No	Yes	No	No
EntityStatusSettings	No	Yes	No	No
AccountingChart	No	Yes	No	No
AccountingEntry	No	Yes	No	No
AccountingJournal	No	Yes	No	No
AccountingScope	No	Yes	No	No

Entity	Insert	Read	Update	Delete
AccountingSystem	No	Yes	No	No
Journal	No	Yes	No	No
LegalEntity	No	Yes	No	No
LegalEntitySystem	No	Yes	No	No
OperationTransaction	No	Yes	No	No
OperationTransactionValue	No	Yes	No	No
TransactionAccountingModel	No	Yes	No	No
TransactionItemAccountingConfig	No	Yes	No	No
TransactionType	No	Yes	No	No
TransactionValueType	No	Yes	No	No
Agreement	No	Yes	No	No
Agreement_BW	No	Yes	No	No
AgreementPricing	No	Yes	No	No
Invoice	No	Yes	No	No
Invoice_BW	No	Yes	No	No
InvoiceDetail	No	Yes	No	No
optionset	No	Yes	No	No
optionsetitem	No	Yes	No	No

A user with this security role can access the following endpoints:

Endpoint
FTOS_BP_CheckFromToDates
FTOS_BP_CheckMandatoryRoleXLimitType
FTOS_BP_CommissionSchemaDetail
FTOS_BP_GetBankingProductInfo
FTOS_BP_GetCommissionInfo
FTOS_BP_GetCommissionTypeInfo
FTOS_BP_GetPeriodicityTypeInfo
FTOS_BP_GetProductInterestCommissionList
FTOS_CB_CalculateContractCustomValues
FTOS_CB_CalculateInvoiceAmounts
FTOS_CB_CalculateMaturityDate_BA
FTOS_CB_CheckCustomerRole
FTOS_CB_CheckInvoiceHasDetails
FTOS_CB_CheckLimitTypeRole
FTOS_CB_EditContractPenaltyInterestRate
FTOS_CB_EntityVersion_Agreement
FTOS_CB_EntityVersion_Contract

Endpoint
FTOS_CB_EntityVersion_CustomerLimit
FTOS_CB_getBandedInterestObject
FTOS_CB_GetBlockAmountOnContract
FTOS_CB_GetClosureOfContracts
FTOS_CB_GetCommissionDetail
FTOS_CB_GetContractEventEditUrl
FTOS_CB_GetContractInfo
FTOS_CB_GetContractRepaymentSchedule
FTOS_CB_GetContractsForLimit
FTOS_CB_GetCreditFacilityLimitPercent
FTOS_CB_GetDataSourceChartAgreement
FTOS_CB_GetDataSourceChartContractOverview
FTOS_CB_GetDataSourceChartCreditFacility_LoanOfficerAdminRetail
FTOS_CB_GetDataSourceClosingContractsChart
FTOS_CB_GetDataSourceNewContractsChart
FTOS_CB_GetExchangeRate
FTOS_CB_GetGLOnContract
FTOS_CB_GetGroupInfo
FTOS_CB_GetInstallment_Principal_InstallmenNo_Values
FTOS_CB_GetInterestReferencePeriod
FTOS_CB_GetInvoiceDetails
FTOS_CB_GetLimitTypeByCustomer
FTOS_CB_GetNoOfCurrentAccountsForCurrencyId
FTOS_CB_GetProductInterestValue
FTOS_CB_GetProductMinInterestRate
FTOS_CB_GetReasonDetails
FTOS_CB_GetSalesChannelByName
FTOS_CB_GetSystemInvariantDate
FTOS_CB_GetSystemParameter
FTOS_CB_GetTransactionTypeByCode
FTOS_CB_PeriodicityType
FTOS_CB_UpdateActivationDate
FTOS_GetDataSourceChartContract
FTOS_GetDataSourceChartContractEvents
FTOS_GetDataSourceChartCreditFacility
FTOS_GL_GenerateAccountingEntry
FTOS_TPM_CalculateCommissionAppliedTo
FTOS_TPM_CallFormula

Endpoint
FTOS_TPM_GetAgreementBusinessStatusDisplayName
FTOS_TPM_GetAgreementCommissionsDetails
FTOS_TPM_GetAgreementDetails
FTOS_TPM_GetInvoiceDetailContracts
FTOS_TPM_GetTransactionTypesForClawback
FTOS_VerifyUsersCompetence

The following dashboards can be viewed by a user with this security role:

Dashboards
RequestContractEvent
CustomerLimit

Corporate Credit Officer

A user with this security role has the following access rights to records in the FintechOS Platform's entities within **their organization**:

Entity	Insert	Read	Update	Delete
Account	No	Yes	Yes	No
AccountRelOwnership	No	Yes	No	No
approvalTask	No	Yes	No	No
attribute	No	Yes	No	No
businessunit	No	Yes	No	No
BWstatus	No	Yes	No	No
entity	No	Yes	No	No
entitystatus	No	Yes	No	No
Document	No	Yes	Yes	No
GroupAccount	No	Yes	No	No
GroupMember	No	Yes	No	No
BankingProduct	No	Yes	No	No
BankingProductType	No	Yes	No	No
CollateralType	No	Yes	Yes	No
Commission	No	Yes	No	No
Covenant	No	Yes	No	No
Interest	No	Yes	No	No
InterestCommissionItem	No	Yes	No	No
InterestCommissionItemFilter	No	Yes	No	No
PaymentScheduleType	No	Yes	No	No

Entity	Insert	Read	Update	Delete
ProductClassification	No	Yes	No	No
BankAccount	No	Yes	No	No
BankAccountOperation	No	Yes	No	No
CollateralRegister	Yes	Yes	Yes	No
CollateralRegister_BW	Yes	Yes	No	No
CollateralRegisterDocuments	Yes	Yes	Yes	No
CollateralRegisterOwner	Yes	Yes	Yes	No
CollateralRegisterParticipants	Yes	Yes	Yes	No
CollateralRegisterRank	Yes	Yes	Yes	No
Contract	Yes	Yes	Yes	No
Contract_BW	Yes	Yes	No	No
ContractAccrualAndProvision	Yes	Yes	Yes	No
ContractBorrowers	Yes	Yes	Yes	No
ContractClassification	No	Yes	No	No
ContractCollateral	Yes	Yes	Yes	No
ContractCorrectionEntry	Yes	Yes	Yes	No
ContractCorrectionEntryDetail	Yes	Yes	Yes	No
ContractCovenant	Yes	Yes	Yes	Yes
ContractDisbursement	Yes	Yes	Yes	No
ContractDisbursementTranche	Yes	Yes	Yes	Yes
ContractDiscount	Yes	Yes	Yes	No
ContractDocument	Yes	Yes	Yes	Yes
ContractDocument_BW	Yes	Yes	No	Yes
ContractEvent	Yes	Yes	Yes	No
ContractEvent_BW	Yes	Yes	No	No
ContractFee	Yes	Yes	Yes	Yes
ContractGuarantor	Yes	Yes	Yes	No
ContractParticipant	Yes	Yes	Yes	Yes
ContractParticipant_BW	Yes	Yes	Yes	Yes
ContractPenalty	Yes	Yes	Yes	No
ContractPenaltyDetail	Yes	Yes	Yes	No
ContractReevaluation	Yes	Yes	Yes	No
ContractRepaymentSchedule	Yes	Yes	Yes	No
ContractRepaymentScheduleDetail	Yes	Yes	Yes	No
ContractRepaymentScheduleDisb	Yes	Yes	Yes	No
ContractRepaymentScheduleDisbDet	Yes	Yes	Yes	No

Entity	Insert	Read	Update	Delete
ContractRepaymentScheduleVersion	Yes	Yes	Yes	No
ControlPanel	No	Yes	No	No
CreditFacility	Yes	Yes	Yes	No
CreditFacility_BW	Yes	Yes	No	No
CreditFacility_BWA	Yes	Yes	No	No
CreditFacilityAccrual	Yes	Yes	Yes	No
CreditFacilityDetail	Yes	Yes	Yes	No
CreditFacilityDetail_BW	Yes	Yes	No	No
CreditFacilityDetail_BWA	Yes	Yes	No	No
CreditFacilityFee	Yes	Yes	Yes	No
CreditFacilityFeeValue	Yes	Yes	Yes	No
CreditFacilityParticipant	Yes	Yes	Yes	No
CreditFacilityPlan	Yes	Yes	Yes	No
CreditFacilityProduct	Yes	Yes	Yes	No
CustomerLimit	Yes	Yes	Yes	Yes
CustomerLimit_BW	Yes	Yes	Yes	Yes
CustomerLimitType	Yes	Yes	Yes	Yes
DailyContractInterest	Yes	Yes	Yes	No
DelayCategory	No	Yes	No	No
DepositValueCalculation	Yes	Yes	Yes	No
OperationItem	No	Yes	No	No
PaymentAllocation	Yes	Yes	Yes	No
PaymentNotification	Yes	Yes	Yes	No
PeriodicityType	No	Yes	No	No
RepaymentNotification	Yes	Yes	Yes	No
VersioningReason	No	Yes	No	No
AccountType	No	Yes	No	No
Currency	No	Yes	No	No
EntityStatusSettings	No	Yes	No	No
AccountingEntry	No	Yes	No	No
TransactionType	No	Yes	No	No
VersionSettings	No	Yes	No	No
systemuser	No	Yes	No	No

A user with this security role can access the following endpoints:

Endpoint
FTOS_BP_CheckFromToDates

Endpoint
FTOS_BP_CheckMandatoryRoleXLimitType
FTOS_BP_GetBankingProductInfo
FTOS_BP_GetBankingProductsForCreditFacility
FTOS_BP_GetCommissionInfo
FTOS_BP_GetProductInterestCommissionList
FTOS_CB_CalculateContractCustomValues
FTOS_CB_CalculateContractCustomValues_ForDisbursement
FTOS_CB_CalculateMaturityDate_BA
FTOS_CB_CheckCustomerRole
FTOS_CB_CheckLimitTypeRole
FTOS_CB_ContractCommissionAmountCalc
FTOS_CB_DeleteDocument
FTOS_CB_DeleteRepaymentSchedule
FTOS_CB_DisplayFinancedAmountEventForm
FTOS_CB_EditContractPenaltyInterestRate
FTOS_CB_EntityVersion_Agreement
FTOS_CB_EntityVersion_Contract
FTOS_CB_EntityVersion_CreditFacility
FTOS_CB_EntityVersion_CustomerLimit
FTOS_CB_getBandedInterestObject
FTOS_CB_GetBlockAmountOnContract
FTOS_CB_GetClosureOfContracts
FTOS_CB_GetCommissionDetail
FTOS_CB_GetContractCollateralInfo
FTOS_CB_GetContractEventEditUrl
FTOS_CB_GetContractExtendedData
FTOS_CB_GetContractRepaymentSchedule
FTOS_CB_GetContractsForLimit
FTOS_CB_GetCreditFacilityInfo
FTOS_CB_GetCreditFacilityLimitPercent
FTOS_CB_GetCreditFacilityParticipantInfo
FTOS_CB_GetDataSourceChartAgreement
FTOS_CB_GetDataSourceChartContractOverview
FTOS_CB_GetDataSourceChartCreditFacility_LoanOfficerAdminRetail
FTOS_CB_GetDataSourceClosingContractsChart
FTOS_CB_GetDataSourceFutureInstallmentsReport
FTOS_CB_GetDataSourceNewContractsChart
FTOS_CB_GetDataSourcePastDueInstallmentsReport

Endpoint
FTOS_CB_GetEarlyRepaymentValues
FTOS_CB_GetExchangeRate
FTOS_CB_GetExchangeRateOfCollateralRegisterToContract
FTOS_CB_GetFixedVariableInterest
FTOS_CB_GetGLOnContract
FTOS_CB_GetGroupInfo
FTOS_CB_GetInstallment_Principal_InstallmenNo_Values
FTOS_CB_GetInterestReferencePeriod
FTOS_CB_GetLimitTypeByCustomer
FTOS_CB_GetNoOfContractTranches
FTOS_CB_GetNoOfCurrentAccountsForCurrencyId
FTOS_CB_GetPeriodicity
FTOS_CB_GetProductInterestValue
FTOS_CB_GetProductMinInterestRate
FTOS_CB_GetReasonDetails
FTOS_CB_GetRecoverInterest
FTOS_CB_GetRemainingNotifications
FTOS_CB_GetSalesChannelByName
FTOS_CB_GetSoonToExpireCurrentAccountWithOverdrafts
FTOS_CB_GetSystemInvariantDate
FTOS_CB_GetSystemParameter
FTOS_CB_GetTransactionTypeByCode
FTOS_CB_PaymentHolidaySchedule
FTOS_CB_PaymentScheduleFieldsDisb
FTOS_CB_PeriodicityType
FTOS_CB_ProcessAccrualsAndProvisions
FTOS_CB_RecalculateEarlyRepaymentSchedule
FTOS_CB_RecalculateRescheduleOverduesSchedule
FTOS_CB_RecalculateSchedule
FTOS_CB_releaseCollaterals
FTOS_CB_SelectNotificationForReschedule
FTOS_CB_UpdateActivationDate
FTOS_CB_UpdateContractDelayCategory_Filtered
FTOS_CB_UpdateTranche
FTOS_CB_ValidateOverdraftExpireFields
FTOS_GetCustomersAvailableForCreditFacility
FTOS_GetDataSourceChartContract
FTOS_GetDataSourceChartCreditFacility

Endpoint
FTOS_TPM_GetTransactionTypesForClawback
FTOS_VerifyUsersCompetence

Retail Credit Officer

A user with this security role has the following access rights to records in the FintechOS Platform's entities within **their organization**:

Entity	Insert	Read	Update	Delete
Account	No	Yes	No	No
AccountRelOwnership	No	Yes	No	No
Address	No	Yes	No	No
approvalTask	No	Yes	No	No
businessunit	Yes	Yes	Yes	No
BWstatus	Yes	Yes	Yes	No
entity	Yes	Yes	Yes	No
entitystatus	Yes	Yes	Yes	No
Division	No	Yes	No	No
Document	No	Yes	No	No
GroupAccount	Yes	Yes	Yes	No
GroupMember	No	Yes	No	No
UnitType	No	Yes	No	No
BandedInterest	No	Yes	No	No
BankingProduct	No	Yes	No	No
BankingProductAgreement	No	Yes	No	No
BankingProductDiscount	No	Yes	No	No
BankingProductDocument	No	Yes	No	No
BankingProductType	No	Yes	No	No
ClassificationType	No	Yes	No	No
CollateralType	Yes	Yes	Yes	No
Commission	Yes	Yes	Yes	No
CommissionSchema	Yes	Yes	Yes	No
CommissionType	Yes	Yes	Yes	No
CommissionValue	Yes	Yes	Yes	No
Covenant	Yes	Yes	Yes	No
Feature	No	Yes	No	No
Formula	No	Yes	No	No
FormulaType	Yes	Yes	Yes	No

Entity	Insert	Read	Update	Delete
GLAccounts	No	Yes	No	No
InsuranceItem	No	Yes	No	No
Interest	No	Yes	No	No
InterestCommissionItem	Yes	Yes	Yes	No
InterestCommissionItemFilter	Yes	Yes	Yes	No
InterestRateMatrix	No	Yes	No	No
InterestValue	No	Yes	No	No
PaymentScheduleType	No	Yes	No	No
ProductAvailabilityItemFilter	No	Yes	No	No
ProductClassification	No	Yes	No	No
ProductCovenant	No	Yes	No	No
ProductDestinationType	No	Yes	No	No
ProductDisbursement	No	Yes	No	No
ProductGuaranteeStructure	No	Yes	No	No
ProductQuestion	No	Yes	No	No
TestScenario	No	Yes	No	No
BankAccount	Yes	Yes	Yes	No
BankAccountOperation	Yes	Yes	Yes	No
CollateralRegister	Yes	Yes	Yes	No
CollateralRegister_BW	Yes	Yes	Yes	No
CollateralRegisterDocuments	Yes	Yes	Yes	No
CollateralRegisterOwner	Yes	Yes	Yes	No
CollateralRegisterParticipants	Yes	Yes	Yes	No
CollateralRegisterRank	Yes	Yes	Yes	No
Contract	Yes	Yes	Yes	No
Contract_BW	Yes	Yes	Yes	No
ContractAccrualAndProvision	Yes	Yes	Yes	No
ContractBorrowers	Yes	Yes	Yes	No
ContractClassification	Yes	Yes	Yes	No
ContractCollateral	Yes	Yes	Yes	No
ContractCorrectionEntry	Yes	Yes	Yes	No
ContractCorrectionEntry_BW	Yes	Yes	Yes	No
ContractCorrectionEntryDetail	Yes	Yes	Yes	No
ContractCovenant	Yes	Yes	Yes	No
ContractCovenant_BW	Yes	Yes	Yes	No
ContractDisbursement	Yes	Yes	Yes	No
ContractDisbursementTranche	Yes	Yes	Yes	No

Entity	Insert	Read	Update	Delete
ContractDisbursementTranche_BW	Yes	Yes	Yes	No
ContractDiscount	Yes	Yes	Yes	No
ContractDocument	Yes	Yes	Yes	Yes
ContractDocument_BW	Yes	Yes	No	Yes
ContractEvent	Yes	Yes	Yes	No
ContractEvent_BW	Yes	Yes	Yes	No
ContractFee	Yes	Yes	Yes	No
ContractGuarantor	Yes	Yes	Yes	No
ContractParticipant	Yes	Yes	Yes	No
ContractParticipant_BW	Yes	Yes	Yes	No
ContractPenalty	Yes	Yes	Yes	No
ContractReevaluation	Yes	Yes	Yes	No
ContractRepaymentSchedule	Yes	Yes	Yes	No
ContractRepaymentScheduleDetail	Yes	Yes	Yes	No
ContractRepaymentScheduleDisb	Yes	Yes	Yes	No
ContractRepaymentScheduleDisbDet	Yes	Yes	Yes	No
ContractRepaymentScheduleVersion	Yes	Yes	Yes	No
ControlPanel	No	Yes	No	No
CovenantResolution	Yes	Yes	Yes	No
CreditFacility	No	Yes	No	No
CreditFacilityAccrual	No	Yes	No	No
CreditFacilityDetail	Yes	Yes	Yes	No
CreditFacilityFee	No	Yes	No	No
CreditFacilityFeeValue	No	Yes	No	No
CreditFacilityParticipant	No	Yes	No	No
CreditFacilityPlan	No	Yes	No	No
CreditFacilityProduct	No	Yes	No	No
CustomerLimit	Yes	Yes	Yes	Yes
CustomerLimit_BW	Yes	Yes	Yes	Yes
CustomerLimitType	Yes	Yes	Yes	Yes
DailyContractInterest	Yes	Yes	Yes	No
DepositValueCalculation	Yes	Yes	Yes	No
OperationItem	No	Yes	No	No
Payment	Yes	Yes	Yes	No
PaymentAllocation	Yes	Yes	Yes	No
PaymentNotification	Yes	Yes	Yes	No

Entity	Insert	Read	Update	Delete
PeriodicityType	Yes	Yes	Yes	No
RepaymentNotification	No	Yes	No	No
RepaymentNotificationDetail	No	Yes	No	No
VersioningReason	No	Yes	No	No
WeekDay	Yes	Yes	No	No
AccountType	No	Yes	No	No
Action	No	Yes	No	No
Activity	No	Yes	No	No
Country	No	Yes	No	No
Currency	Yes	Yes	Yes	No
EntityStatusSettings	Yes	Yes	Yes	No
AccountingEntry	No	Yes	No	No
TransactionType	No	Yes	No	No
Activity	No	Yes	No	No
Agreement	Yes	Yes	Yes	No
Agreement_BW	Yes	Yes	Yes	No
AgreementPricing	Yes	Yes	Yes	No
Invoice	Yes	Yes	Yes	No
Invoice_BW	Yes	Yes	Yes	Yes
InvoiceDetail	Yes	Yes	Yes	No
VersionSettings	Yes	Yes	Yes	No
systemuser	Yes	Yes	Yes	No

A user with this security role can access the following endpoints:

Endpoint
FTOS_BP_CheckFromToDates
FTOS_BP_CheckMandatoryRoleXLimitType
FTOS_BP_CommissionSchemaDetail
FTOS_BP_GetBankingProductDefaultValues
FTOS_BP_GetBankingProductInfo
FTOS_BP_GetCommissionInfo
FTOS_BP_GetCommissionTypeInfo
FTOS_BP_GetPeriodicityTypeInfo
FTOS_BP_GetProductInterestCommissionList
FTOS_CB_AddUpdateContractPaymentHoliday
FTOS_CB_CalculateContractCustomValues
FTOS_CB_CalculateContractCustomValues_ForDisbursement
FTOS_CB_CalculateInvoiceAmounts

Endpoint
FTOS_CB_CalculateMaturityDate_BA
FTOS_CB_CheckCustomerRole
FTOS_CB_CheckInvoiceHasDetails
FTOS_CB_CheckLimitTypeRole
FTOS_CB_ContractCommissionAmountCalc
FTOS_CB_DeleteDocument
FTOS_CB_DeleteRepaymentSchedule
FTOS_CB_DisplayFinancedAmountEventForm
FTOS_CB_EditContractPenaltyInterestRate
FTOS_CB_EntityVersion_Agreement
FTOS_CB_EntityVersion_Contract
FTOS_CB_EntityVersion_CustomerLimit
FTOS_CB_getBandedInterestObject
FTOS_CB_GetBlockAmountOnContract
FTOS_CB_GetClosureOfContracts
FTOS_CB_GetCommissionDetail
FTOS_CB_GetContractCollateralInfo
FTOS_CB_GetContractEventEditUrl
FTOS_CB_GetContractExtendedData
FTOS_CB_GetContractInfo
FTOS_CB_GetContractRepaymentSchedule
FTOS_CB_GetContractsForLimit
FTOS_CB_GetContractsWithPaymentHolidayPossibility
FTOS_CB_GetCreditFacilityLimitPercent
FTOS_CB_GetDataSourceChartAgreement
FTOS_CB_GetDataSourceChartContractOverview
FTOS_CB_GetDataSourceChartCreditFacility_LoanOfficerAdminRetail
FTOS_CB_GetDataSourceClosingContractsChart
FTOS_CB_GetDataSourceFutureInstallmentsReport
FTOS_CB_GetDataSourceNewContractsChart
FTOS_CB_GetDataSourcePastDueInstallmentsReport
FTOS_CB_GetEarlyRepaymentValues
FTOS_CB_GetExchangeRate
FTOS_CB_GetExchangeRateOfCollateralRegisterToContract
FTOS_CB_GetFixedVariableInterest
FTOS_CB_GetGLOnContract
FTOS_CB_GetGroupInfo
FTOS_CB_GetInstallment_Principal_InstallmenNo_Values

Endpoint
FTOS_CB_GetInterestReferencePeriod
FTOS_CB_GetInvoiceDetails
FTOS_CB_GetLimitTypeByCustomer
FTOS_CB_GetNoOfContractTranches
FTOS_CB_GetNoOfCurrentAccountsForCurrencyId
FTOS_CB_GetProductInterestValue
FTOS_CB_GetProductMinInterestRate
FTOS_CB_GetReasonDetails
FTOS_CB_GetRemainingNotifications
FTOS_CB_GetSalesChannelByName
FTOS_CB_GetSoonToExpireCurrentAccountWithOverdrafts
FTOS_CB_GetSystemInvariantDate
FTOS_CB_GetSystemParameter
FTOS_CB_GetTransactionTypeByCode
FTOS_CB_GetWorkingDate
FTOS_CB_PaymentHolidaySchedule
FTOS_CB_PaymentScheduleFields
FTOS_CB_PaymentScheduleFieldsDisb
FTOS_CB_PeriodicityType
FTOS_CB_ProcessAccrualsAndProvisions
FTOS_CB_RecalculateEarlyRepaymentSchedule
FTOS_CB_RecalculateSchedule
FTOS_CB_releaseCollaterals
FTOS_CB_SetInterestRate
FTOS_CB_UpdateActivationDate
FTOS_CB_UpdateContractDelayCategory_Filtered
FTOS_CB_UpdateTranche
FTOS_CB_ValidateOverdraftExpireFields
FTOS_CheckLicense
FTOS_GetDataSourceChartContract
FTOS_GetDataSourceChartContractEvents
FTOS_GetDataSourceChartCreditFacility
FTOS_TPM_CalculateCommissionAppliedTo
FTOS_TPM_CallFormula
FTOS_TPM_GetAgreementBusinessStatusDisplayName
FTOS_TPM_GetAgreementCommissionsDetails
FTOS_TPM_GetAgreementDetails
FTOS_TPM_GetInvoiceDetailContracts

Endpoint
FTOS_TPM_GetTransactionTypesForClawback
FTOS_VerifyUsersCompetence

Accounting Officer

A user with this security role has the following access rights to records in the FintechOS Platform's entities within **their organization**:

Entity	Insert	Read	Update	Delete
Account	No	Yes	No	No
approvalTask	No	Yes	No	No
businessunit	No	Yes	No	No
entity	No	Yes	No	No
entitystatus	Yes	Yes	Yes	No
GroupMember	No	Yes	No	No
BankingProduct	No	Yes	No	No
BankingProductType	No	Yes	No	No
Commission	No	Yes	No	No
CommissionSchema	No	Yes	No	No
CommissionType	Yes	Yes	Yes	No
CommissionValue	No	Yes	No	No
FormulaType	No	Yes	No	No
GLAccounts	Yes	Yes	Yes	No
Interest	No	Yes	No	No
PaymentScheduleType	No	Yes	No	No
ProductClassification	No	Yes	No	No
BankAccount	No	Yes	No	No
BankAccountOperation	No	Yes	No	No
Contract	No	Yes	No	No
ContractAccrualAndProvision	No	Yes	No	No
ContractClassification	No	Yes	No	No
ContractCollateral	No	Yes	No	No
ContractCovenant	No	Yes	No	No
ContractDisbursementTranche	No	Yes	No	No
ContractDocument	No	Yes	No	No
ContractEvent	No	Yes	No	No
ContractFee	No	Yes	No	No
ContractParticipant	No	Yes	No	No

Entity	Insert	Read	Update	Delete
ContractPenalty	No	Yes	No	No
ContractReevaluation	No	Yes	No	No
ContractRepaymentSchedule	No	Yes	No	No
ContractRepaymentScheduleDetail	No	Yes	No	No
ContractRepaymentScheduleVersion	No	Yes	No	No
CreditFacility	No	Yes	No	No
CreditFacilityAccrual	No	Yes	No	No
CreditFacilityDetail	No	Yes	No	No
CreditFacilityFee	No	Yes	No	No
CreditFacilityFeeValue	No	Yes	No	No
CreditFacilityParticipant	No	Yes	No	No
CreditFacilityPlan	No	Yes	No	No
CreditFacilityProduct	No	Yes	No	No
CustomerLimit	No	Yes	No	No
CustomerLimit_BW	No	Yes	No	No
CustomerLimitType	No	Yes	No	No
OperationItem	No	Yes	No	No
Payment	No	Yes	No	No
PaymentNotification	No	Yes	No	No
PeriodicityType	No	Yes	No	No
RepaymentNotification	No	Yes	No	No
TransactionOperationType	Yes	Yes	Yes	No
VersioningReason	No	Yes	No	No
WeekDay	No	Yes	No	No
AccountType	No	Yes	No	No
Currency	No	Yes	No	No
EntityStatusSettings	No	Yes	No	No
AccountingChart	Yes	Yes	Yes	No
AccountingEntry	Yes	Yes	Yes	No
AccountingJournal	Yes	Yes	Yes	No
AccountingScope	Yes	Yes	Yes	No
AccountingSystem	Yes	Yes	Yes	No
Journal	Yes	Yes	Yes	No
LegalEntity	Yes	Yes	Yes	No
LegalEntitySystem	Yes	Yes	Yes	No
OperationTransaction	Yes	Yes	Yes	No
OperationTransactionValue	Yes	Yes	Yes	No

Entity	Insert	Read	Update	Delete
TransactionAccountingModel	Yes	Yes	Yes	No
TransactionItemAccountingConfig	Yes	Yes	Yes	No
TransactionType	Yes	Yes	Yes	No
TransactionValueType	Yes	Yes	Yes	No
Agreement	No	Yes	No	No
Agreement_BW	No	Yes	No	No
AgreementPricing	No	Yes	No	No
Invoice	No	Yes	No	No
Invoice_BW	No	Yes	No	No
InvoiceDetail	No	Yes	No	No
optionset	No	Yes	No	No
optionsetitem	No	Yes	No	No

A user with this security role can access the following endpoints:

Endpoint
FTOS_BP_CheckFromToDates
FTOS_BP_CheckMandatoryRoleXLimitType
FTOS_BP_CommissionSchemaDetail
FTOS_BP_GetBankingProductInfo
FTOS_BP_GetCommissionInfo
FTOS_BP_GetCommissionTypeInfo
FTOS_BP_GetPeriodicityTypeInfo
FTOS_BP_GetProductInterestCommissionList
FTOS_CB_CalculateContractCustomValues
FTOS_CB_CalculateInvoiceAmounts
FTOS_CB_CalculateMaturityDate_BA
FTOS_CB_CheckCustomerRole
FTOS_CB_CheckInvoiceHasDetails
FTOS_CB_CheckLimitTypeRole
FTOS_CB_DisplayFinancedAmountEventForm
FTOS_CB_EditContractPenaltyInterestRate
FTOS_CB_EntityVersion_Agreement
FTOS_CB_EntityVersion_Contract
FTOS_CB_EntityVersion_CustomerLimit
FTOS_CB_getBandedInterestObject
FTOS_CB_GetBlockAmountOnContract
FTOS_CB_GetClosureOfContracts
FTOS_CB_GetCommissionDetail

Endpoint
FTOS_CB_GetContractEventEditUrl
FTOS_CB_GetContractInfo
FTOS_CB_GetContractRepaymentSchedule
FTOS_CB_GetContractsForLimit
FTOS_CB_GetCreditFacilityLimitPercent
FTOS_CB_GetDataSourceChartAgreement
FTOS_CB_GetDataSourceChartContractOverview
FTOS_CB_GetDataSourceChartCreditFacility_LoanOfficerAdminRetail
FTOS_CB_GetDataSourceClosingContractsChart
FTOS_CB_GetDataSourceNewContractsChart
FTOS_CB_GetExchangeRate
FTOS_CB_GetGLOnContract
FTOS_CB_GetGroupInfo
FTOS_CB_GetInstallment_Principal_InstallmenNo_Values
FTOS_CB_GetInterestReferencePeriod
FTOS_CB_GetInvoiceDetails
FTOS_CB_GetLimitTypeByCustomer
FTOS_CB_GetNoOfCurrentAccountsForCurrencyId
FTOS_CB_GetProductInterestValue
FTOS_CB_GetProductMinInterestRate
FTOS_CB_GetReasonDetails
FTOS_CB_GetSalesChannelByName
FTOS_CB_GetSystemInvariantDate
FTOS_CB_GetSystemParameter
FTOS_CB_GetTransactionTypeByCode
FTOS_CB_PeriodicityType
FTOS_CB_UpdateActivationDate
FTOS_GetDataSourceChartContract
FTOS_GetDataSourceChartContractEvents
FTOS_GetDataSourceChartCreditFacility
FTOS_GL_GenerateAccountingEntry
FTOS_TPM_CalculateCommissionAppliedTo
FTOS_TPM_CallFormula
FTOS_TPM_GetAgreementBusinessStatusDisplayName
FTOS_TPM_GetAgreementCommissionsDetails
FTOS_TPM_GetAgreementDetails
FTOS_TPM_GetInvoiceDetailContracts
FTOS_TPM_GetTransactionTypesForClawback
FTOS_VerifyUsersCompetence

Supervisor Risk Officer

A user with this security role has the following access rights to records in the FintechOS Platform's entities within **their organization**:

Entity	Insert	Read	Update	Delete
Account	No	Yes	Yes	No
AccountRelOwnership	No	Yes	No	No
approvalTask	No	Yes	Yes	No
businessunit	No	Yes	No	No
entity	No	Yes	No	No
entitystatus	No	Yes	No	No
GroupAccount	No	Yes	No	No
GroupMember	No	Yes	No	No
BankingProductType	No	Yes	No	No
CollateralType	No	Yes	No	No
Commission	No	Yes	No	No
CommissionSchema	No	Yes	No	No
CommissionType	No	Yes	No	No
CommissionValue	No	Yes	No	No
Covenant	No	Yes	No	No
FormulaType	No	Yes	No	No
Interest	No	Yes	No	No
PaymentScheduleType	No	Yes	No	No
ProductClassification	No	Yes	No	No
BankAccount	No	Yes	No	No
BankAccountOperation	No	Yes	No	No
CollateralRegister	No	Yes	No	No
CollateralRegisterDocuments	No	Yes	No	No
CollateralRegisterOwner	No	Yes	No	No
CollateralRegisterParticipants	No	Yes	No	No
CollateralRegisterRank	No	Yes	No	No
Contract	No	Yes	No	No
ContractAccrualAndProvision	No	Yes	No	No
ContractClassification	No	Yes	No	No
ContractCollateral	No	Yes	No	No
ContractCorrectionEntry	No	Yes	No	No
ContractCorrectionEntryDetail	No	Yes	No	No
ContractCovenant	No	Yes	No	No
ContractDisbursementTranche	No	Yes	No	No

Entity	Insert	Read	Update	Delete
ContractDocument	No	Yes	No	No
ContractEvent	No	Yes	No	No
ContractFee	No	Yes	No	No
ContractGuarantor	No	Yes	No	No
ContractParticipant	No	Yes	No	No
ContractPenalty	No	Yes	No	No
ContractPenaltyDetail	No	Yes	No	No
ContractReevaluation	No	Yes	No	No
ContractRepaymentSchedule	No	Yes	No	No
ContractRepaymentScheduleDetail	No	Yes	No	No
ContractRepaymentScheduleDisb	No	Yes	No	No
ContractRepaymentScheduleVersion	No	Yes	No	No
CreditFacility	No	Yes	No	No
CreditFacility_BW	No	Yes	No	No
CreditFacilityAccrual	No	Yes	No	No
CreditFacilityDetail	No	Yes	No	No
CreditFacilityFee	No	Yes	No	No
CreditFacilityFeeValue	No	Yes	No	No
CreditFacilityParticipant	No	Yes	No	No
CreditFacilityPlan	No	Yes	No	No
CreditFacilityProduct	No	Yes	No	No
CustomerLimit	Yes	Yes	Yes	Yes
CustomerLimit_BW	Yes	Yes	Yes	Yes
CustomerLimitType	Yes	Yes	Yes	Yes
OperationItem	No	Yes	No	No
Payment	No	Yes	No	No
PaymentAllocation	No	Yes	No	No
PaymentNotification	No	Yes	No	No
PeriodicityType	No	Yes	No	No
RepaymentNotification	No	Yes	No	No
RepaymentNotificationDetail	No	Yes	No	No
VersioningReason	No	Yes	No	No
WeekDay	No	Yes	No	No
AccountType	No	Yes	No	No
Currency	No	Yes	No	No
EntityStatusSettings	No	Yes	No	No
AccountingEntry	No	Yes	No	No

Entity	Insert	Read	Update	Delete
TransactionType	No	Yes	No	No
Agreement	No	Yes	No	No
Agreement_BW	No	Yes	No	No
AgreementPricing	No	Yes	No	No
Invoice	No	Yes	No	No
Invoice_BW	No	Yes	No	No
InvoiceDetail	No	Yes	No	No
VersionSettings	No	Yes	No	No
userCompetence	No	Yes	No	No

A user with this security role can access the following endpoints:

Endpoint
FTOS_BP_CheckFromToDates
FTOS_BP_CheckMandatoryRoleXLimitType
FTOS_BP_CommissionSchemaDetail
FTOS_BP_GetBankingProductInfo
FTOS_BP_GetCommissionInfo
FTOS_BP_GetCommissionTypeInfo
FTOS_BP_GetPeriodicityTypeInfo
FTOS_BP_GetProductInterestCommissionList
FTOS_CB_CalculateContractCustomValues
FTOS_CB_CalculateInvoiceAmounts
FTOS_CB_CalculateMaturityDate_BA
FTOS_CB_CheckCustomerRole
FTOS_CB_CheckInvoiceHasDetails
FTOS_CB_CheckLimitTypeRole
FTOS_CB_DisplayFinancedAmountEventForm
FTOS_CB_EditContractPenaltyInterestRate
FTOS_CB_EntityVersion_Agreement
FTOS_CB_EntityVersion_Contract
FTOS_CB_EntityVersion_CustomerLimit
FTOS_CB_getBandedInterestObject
FTOS_CB_GetBlockAmountOnContract
FTOS_CB_GetClosureOfContracts
FTOS_CB_GetCommissionDetail
FTOS_CB_GetContractEventEditUrl
FTOS_CB_GetContractEventFee
FTOS_CB_GetContractInfo

Endpoint
FTOS_CB_GetContractRepaymentSchedule
FTOS_CB_GetContractsForLimit
FTOS_CB_GetCreditFacilityInfo
FTOS_CB_GetCreditFacilityLimitPercent
FTOS_CB_GetDataSourceChartAgreement
FTOS_CB_GetDataSourceChartContractOverview
FTOS_CB_GetDataSourceChartCreditFacility_LoanOfficerAdminRetail
FTOS_CB_GetDataSourceClosingContractsChart
FTOS_CB_GetDataSourceFutureInstallmentsReport
FTOS_CB_GetDataSourceNewContractsChart
FTOS_CB_GetDataSourcePastDueInstallmentsReport
FTOS_CB_GetEarlyRepaymentValues
FTOS_CB_GetExchangeRate
FTOS_CB_GetGLOnContract
FTOS_CB_GetGroupInfo
FTOS_CB_GetInstallment_Principal_InstallmenNo_Values
FTOS_CB_GetInterestReferencePeriod
FTOS_CB_GetInvoiceDetails
FTOS_CB_GetLimitTypeByCustomer
FTOS_CB_GetNoOfContractTranches
FTOS_CB_GetNoOfCurrentAccountsForCurrencyId
FTOS_CB_GetProductInterestValue
FTOS_CB_GetProductMinInterestRate
FTOS_CB_GetReasonDetails
FTOS_CB_GetSalesChannelByName
FTOS_CB_GetSystemInvariantDate
FTOS_CB_GetSystemParameter
FTOS_CB_GetTransactionTypeByCode
FTOS_CB_PaymentHolidaySchedule
FTOS_CB_PaymentScheduleFields
FTOS_CB_PaymentScheduleFieldsDisb
FTOS_CB_PeriodicityType
FTOS_CB_ScheduleVersionFields
FTOS_CB_UpdateActivationDate
FTOS_Entity Version
FTOS_GetDataSourceChartContract
FTOS_GetDataSourceChartContractEvents
FTOS_GetDataSourceChartCreditFacility

Endpoint
FTOS_TPM_CalculateCommissionAppliedTo
FTOS_TPM_CallFormula
FTOS_TPM_GetAgreementBusinessStatusDisplayName
FTOS_TPM_GetAgreementCommissionsDetails
FTOS_TPM_GetAgreementDetails
FTOS_TPM_GetInvoiceDetailContracts
FTOS_TPM_GetTransactionTypesForClawback
FTOS_VerifyUsersCompetence

Risk Officer

A user with this security role has the following access rights to records in FintechOS Platform's entities:

Entity	Insert	Read	Update	Delete
Account	No	Yes	Yes	No
AccountRelOwnership	No	Yes	No	No
Address	No	Yes	No	No
approvalTask	No	Yes	No	No
businessunit	No	Yes	No	No
entity	No	Yes	No	No
entitystatus	No	Yes	No	No
Division	No	Yes	No	No
Document	No	Yes	No	No
GroupAccount	No	Yes	No	No
GroupMember	No	Yes	No	No
UnitType	No	Yes	No	No
BankingProductType	No	Yes	No	No
CollateralType	No	Yes	No	No
Commission	No	Yes	No	No
CommissionSchema	No	Yes	No	No
CommissionType	No	Yes	No	No
CommissionValue	No	Yes	No	No
Covenant	No	Yes	No	No
FormulaType	No	Yes	No	No
Interest	No	Yes	No	No
PaymentScheduleType	No	Yes	No	No
ProductClassification	No	Yes	No	No

Entity	Insert	Read	Update	Delete
BankAccount	No	Yes	No	No
BankAccountOperation	No	Yes	No	No
CollateralRegister	No	Yes	No	No
CollateralRegisterDocuments	No	Yes	No	No
CollateralRegisterOwner	No	Yes	No	No
CollateralRegisterParticipants	No	Yes	No	No
CollateralRegisterRank	No	Yes	No	No
Contract	Yes	Yes	No	No
ContractAccrualAndProvision	No	Yes	No	No
ContractClassification	No	Yes	No	No
ContractCollateral	No	Yes	No	No
ContractCorrectionEntry	No	Yes	No	No
ContractCorrectionEntryDetail	No	Yes	No	No
ContractCovenant	No	Yes	No	No
ContractDisbursementTranche	No	Yes	No	No
ContractDocument	No	Yes	No	No
ContractEvent	No	Yes	No	No
ContractFee	No	Yes	No	No
ContractGuarantor	No	Yes	No	No
ContractParticipant	No	Yes	No	No
ContractPenalty	No	Yes	No	No
ContractReevaluation	No	Yes	No	No
ContractRepaymentSchedule	No	Yes	No	No
ContractRepaymentScheduleDetail	No	Yes	No	No
ContractRepaymentScheduleDisb	No	Yes	No	No
ContractRepaymentScheduleVersion	No	Yes	No	No
CreditFacility	No	Yes	No	No
CreditFacility_BW	No	Yes	No	No
CreditFacilityAccrual	No	Yes	No	No
CreditFacilityDetail	No	Yes	No	No
CreditFacilityFee	No	Yes	No	No
CreditFacilityFeeValue	No	Yes	No	No
CreditFacilityParticipant	No	Yes	No	No
CreditFacilityPlan	No	Yes	No	No
CreditFacilityProduct	No	Yes	No	No
CustomerLimit	Yes	Yes	Yes	Yes
CustomerLimit_BW	Yes	Yes	Yes	Yes

Entity	Insert	Read	Update	Delete
CustomerLimitType	Yes	Yes	Yes	Yes
OperationItem	No	Yes	No	No
Payment	No	Yes	No	No
PaymentAllocation	No	Yes	No	No
PaymentNotification	No	Yes	No	No
PeriodicityType	No	Yes	No	No
RepaymentNotification	No	Yes	No	No
RepaymentNotificationDetail	No	Yes	No	No
VersioningReason	No	Yes	No	No
WeekDay	No	Yes	No	No
AccountType	No	Yes	No	No
Action	No	Yes	No	No
Activity	No	Yes	No	No
Currency	No	Yes	No	No
EntityStatusSettings	No	Yes	No	No
AccountingEntry	No	Yes	No	No
TransactionType	No	Yes	No	No
Activity	No	Yes	No	No
Agreement	No	Yes	No	No
Agreement_BW	No	Yes	No	No
AgreementPricing	No	Yes	No	No
Invoice	No	Yes	No	No
Invoice_BW	No	Yes	No	No
InvoiceDetail	No	Yes	No	No
VersionSettings	Yes	Yes	Yes	No

A user with this security role can access the following endpoints:

Endpoints
FTOS_BP_CheckFromToDates
FTOS_BP_CheckMandatoryRoleXLimitType
FTOS_BP_CommissionSchemaDetail
FTOS_BP_GetBankingProductInfo
FTOS_BP_GetCommissionInfo
FTOS_BP_GetCommissionTypeInfo
FTOS_BP_GetPeriodicityTypeInfo
FTOS_BP_GetProductInterestCommissionList
FTOS_CB_CalculateContractCustomValues
FTOS_CB_CalculateInvoiceAmounts

Endpoints
FTOS_CB_CalculateMaturityDate_BA
FTOS_CB_CheckCustomerRole
FTOS_CB_CheckInvoiceHasDetails
FTOS_CB_CheckLimitTypeRole
FTOS_CB_DisplayFinancedAmountEventForm
FTOS_CB_EditContractPenaltyInterestRate
FTOS_CB_EntityVersion_Agreement
FTOS_CB_EntityVersion_Contract
FTOS_CB_EntityVersion_CustomerLimit
FTOS_CB_getBandedInterestObject
FTOS_CB_GetBlockAmountOnContract
FTOS_CB_GetClosureOfContracts
FTOS_CB_GetCommissionDetail
FTOS_CB_GetContractEventEditUrl
FTOS_CB_GetContractEventFee
FTOS_CB_GetContractInfo
FTOS_CB_GetContractRepaymentSchedule
FTOS_CB_GetContractsForLimit
FTOS_CB_GetCreditFacilityInfo
FTOS_CB_GetCreditFacilityLimitPercent
FTOS_CB_GetDataSourceChartAgreement
FTOS_CB_GetDataSourceChartContractOverview
FTOS_CB_GetDataSourceChartCreditFacility_LoanOfficerAdminRetail
FTOS_CB_GetDataSourceClosingContractsChart
FTOS_CB_GetDataSourceFutureInstallmentsReport
FTOS_CB_GetDataSourceNewContractsChart
FTOS_CB_GetDataSourcePastDueInstallmentsReport
FTOS_CB_GetEarlyRepaymentValues
FTOS_CB_GetExchangeRate
FTOS_CB_GetGLOnContract
FTOS_CB_GetGroupInfo
FTOS_CB_GetInstallment_Principal_InstallmenNo_Values
FTOS_CB_GetInterestReferencePeriod
FTOS_CB_GetInvoiceDetails
FTOS_CB_GetLimitTypeByCustomer
FTOS_CB_GetNoOfContractTranches
FTOS_CB_GetNoOfCurrentAccountsForCurrencyId
FTOS_CB_GetProductInterestValue

Endpoints
FTOS_CB_GetProductMinInterestRate
FTOS_CB_GetReasonDetails
FTOS_CB_GetSalesChannelByName
FTOS_CB_GetSystemInvariantDate
FTOS_CB_GetSystemParameter
FTOS_CB_GetTransactionTypeByCode
FTOS_CB_PaymentHolidaySchedule
FTOS_CB_PaymentScheduleFields
FTOS_CB_PaymentScheduleFieldsDisb
FTOS_CB_PeriodicityType
FTOS_CB_ScheduleVersionFields
FTOS_CB_UpdateActivationDate
FTOS_GetDataSourceChartContract
FTOS_GetDataSourceChartContractEvents
FTOS_GetDataSourceChartCreditFacility
FTOS_TPM_CalculateCommissionAppliedTo
FTOS_TPM_CallFormula
FTOS_TPM_GetAgreementBusinessStatusDisplayName
FTOS_TPM_GetAgreementCommissionsDetails
FTOS_TPM_GetAgreementDetails
FTOS_TPM_GetInvoiceDetailContracts
FTOS_TPM_GetTransactionTypesForClawback
FTOS_VerifyUsersCompetence

Async Engine Admin

A user with this security role can access the **DLQ Messages** dashboard and the following endpoints:

Endpoint
FTOS_AE_GetMessagesFromQueues
FTOS_AE_ReprocessMessage

Banks

A bank is a financial institution licensed to receive deposits and make loans. Loan Management needs to have some basic information about your main bank or financial institution and your branches network, as well as about other banks or financial institutions with whom you are in a business relationship. Such information includes name, bank identification, branches, and bank accounts.

To manage bank records in Loan Management:

1. In FintechOS Portal, click the main menu icon and expand the **Loan Management Operational** menu.
2. Click **Bank** menu item to open the **Banks List** page.

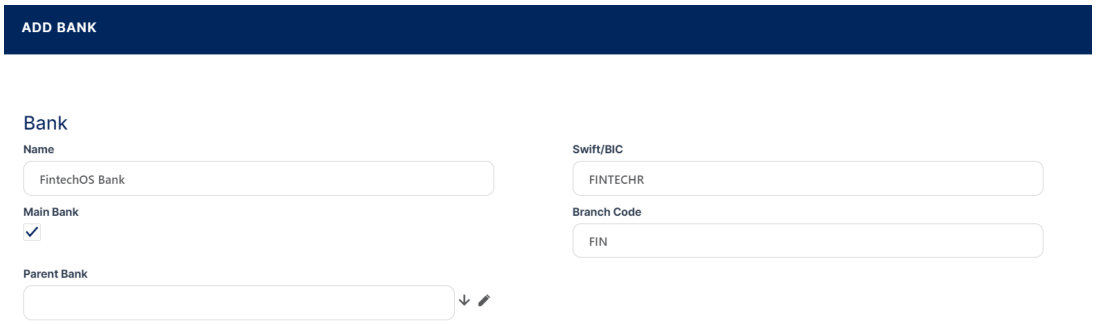


On the **Banks List** page, you can add new bank records or search, edit, and delete existing ones. You can also create external bank accounts for customers.

Creating Bank Records

Follow these steps to create new bank records:

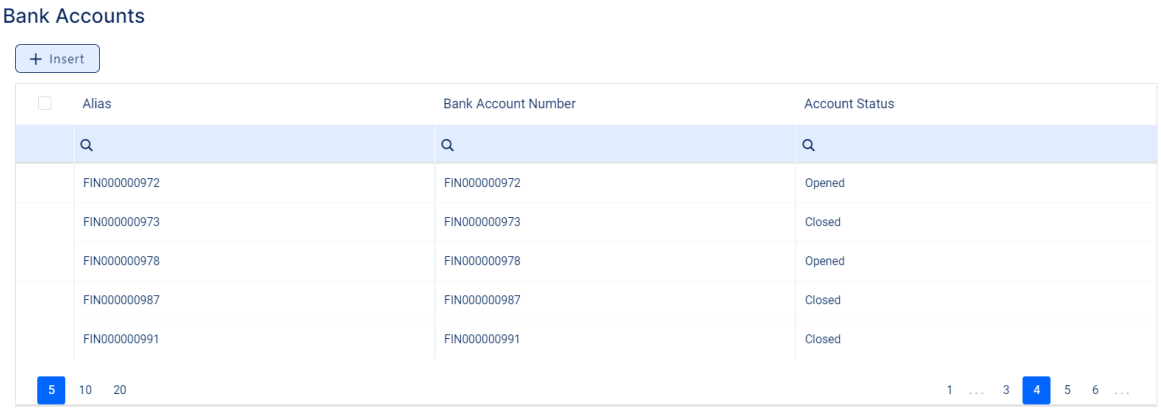
1. Click **Insert** on the **Banks List** page to display the **Add Bank** page.
2. Fill in the following fields:



- **Name** - Enter the name of the bank/ financial institution.
- **Main Bank** - If the checkbox to mark this record as the main bank, the one where all bank accounts are created when a new contract is approved.

3. Optionally, insert the following information:
   - **Swift/BIC** - Enter the SWIFT / BIC codes of the bank (maximum length 11 characters).
     - First 4 characters - Represent the bank code (alphabetic)
     - Next 2 characters - ISO 3166-1 alpha-2 country code (alphabetic)
     - Next 2 characters - Location code (alphanumeric) (passive participants have 1 in the second character)
     - Last 3 characters - Determine the branch code, optional (XXX for main branch/ office) (alphanumeric)
   - **Parent Bank** - Select the parent bank of the new record if the newly entered bank is a branch.
   - **Branch Code** - Enter the code of the branch (maximum length 4 characters).
4. Click the **Save and Reload** button. The bank record is saved and the **Bank Accounts** section is displayed.

For an existent bank record, view all the accounts opened for that bank record in your system within the **Bank Accounts** section. If the bank is marked as `Main Bank`, then all the accounts created for customers when approving a new contract are listed here. Each account shows the alias, the number, and the status of the bank account.

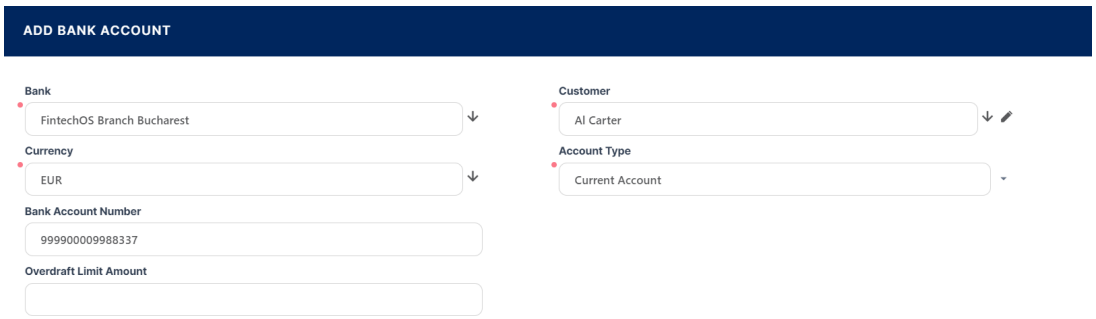


Creating External Bank Accounts

You can create bank accounts opened at banks other than your main bank. These accounts are known as `external accounts`, being marked with attribute `isExtern = True`, and they are created within your system, without interfering with the other bank's accounts.

Follow these steps to create new external bank accounts:

1. On the **Banks List** page, double-click the desired bank record to edit it.
2. On the newly displayed **Edit Bank** page, make sure the bank is not marked as `Main Bank`.
3. Under the **Bank Account** section, click the **Insert** button to open the **Add Bank Account** page.
4. Fill in the following fields:



- **Bank** - This field is automatically completed with the bank where you are opening the external bank account.
- **Customer** - Select the customer for whom you are opening the account.
- **Currency** - Select the currency of the account.
- **Account Type** - Select the type of the account. Possible values are current, savings, fixed deposit, term deposit, and loan term account.

5. Optionally, insert the following information:
   - **Bank Account Number** - Enter the bank account number recorded within the external bank's systems. This is not the bank account number in Loan Management, which is automatically generated when you save the record.
   - **Overdraft Limit Amount** - Enter an overdraft limit amount, if applicable.
6. Click the **Save and Close** button. The external bank account record is saved in the **Opened** status and is ready to be used for referencing in bank documents.

Sales Channels

In Loan Management, you can create contracts through different channels: the dedicated Loan Management menus in FintechOS Portal, API integration calls, or various customer journeys implemented within FintechOS accelerators. The information about the sales channel for each contract is stored at the contract level, so it can be used by financial institutions, for example for different pricing or for selling a product on a specific channel.

To manage the sales channel that your financial institution uses:

1. In FintechOS Portal, click the main menu icon and expand the **Admin Configurations** menu.
2. Click **Sales Channels** menu item to open the **Sales Channels List** page.



On the **Sales Channels List** page, you can create a new sales channel record, edit an record from the list by double-clicking it, delete, export or find a record.

> **IMPORTANT!**
> You need to have the **Loan Admin Officer** security role attached to your user to access the **Sales Channel** menu.

Creating Sales Channel Records

Follow these steps to create new sales channel records:

1. Click the **Insert** button on the **Sales Channels List** page to open the **Add Sales Channel** page is displayed.
2. Fill in the **Name** field with the new sales channel's desired name.



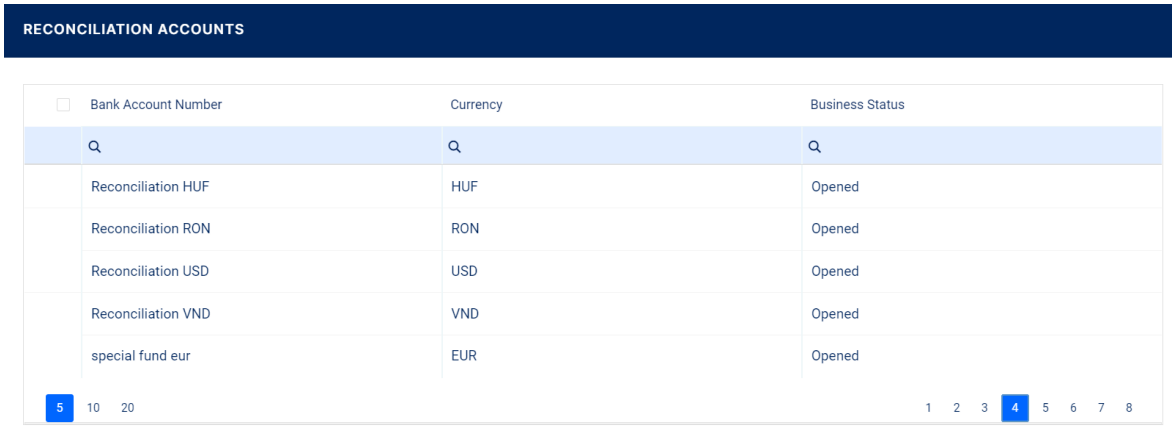
3. Click the **Save and Close** button. The record is now saved and you can use it to specify a contract's origin.

Reconciliation Accounts

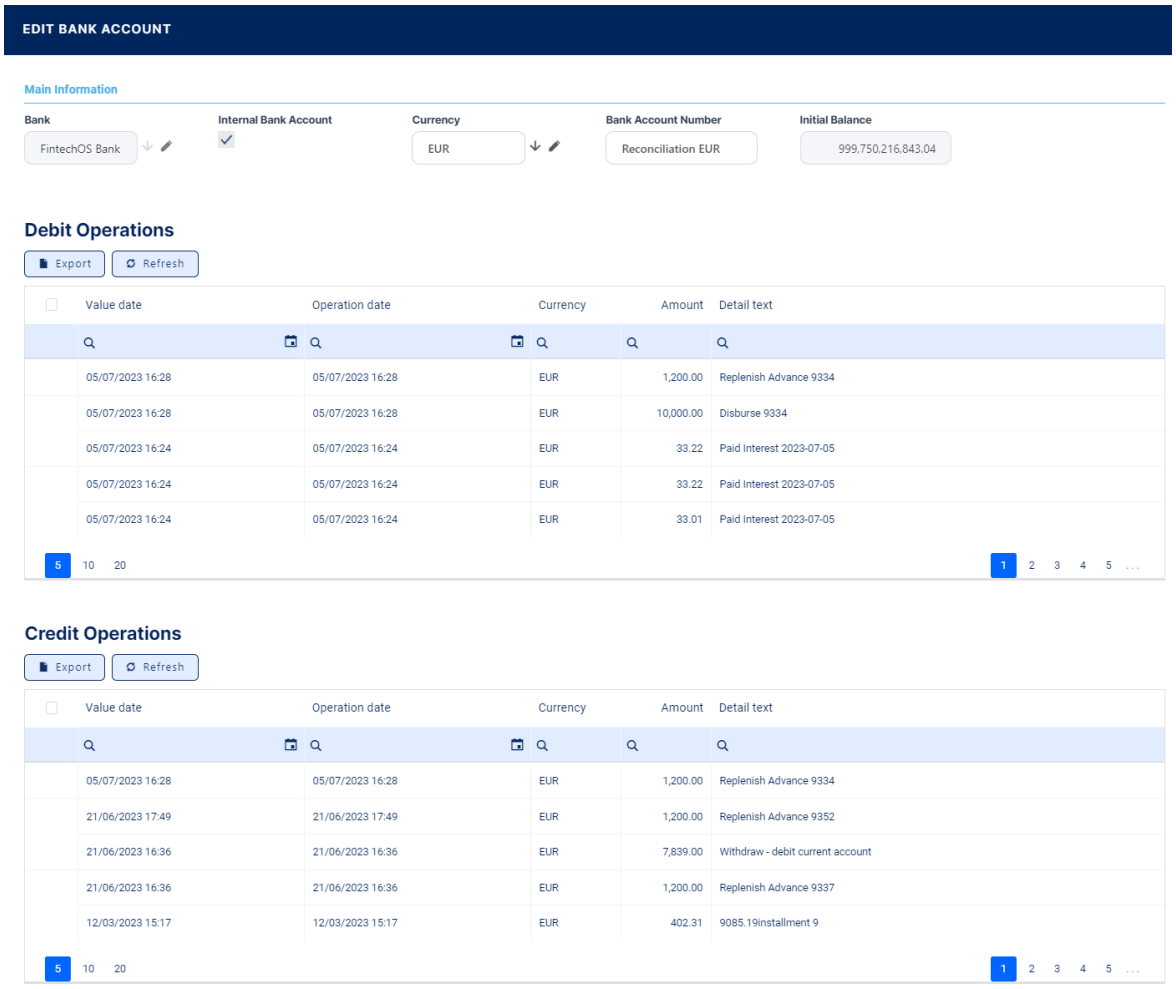
Reconciliation is an accounting process that compares two sets of records to check that figures are correct and in agreement. Reconciliation also confirms that accounts in the general ledger are consistent, accurate, and complete. Loan Management uses reconciliation accounts in its accounting processes and in the product definition itself as tools for monitoring the activity for a specific product or groups of products. When creating a banking product, you must choose a reconciliation account within the **Associated Transactions** tab of the banking product. These accounts are later used by the contracts based on those banking products when performing debit or credit transactions. Reconciliation accounts are also known as "self-bank accounts" or "internal bank accounts".

To manage reconciliation accounts:

1. In FintechOS Portal, click the main menu icon and expand the **Loan Management Operational** menu.
2. Click **Reconciliation Accounts** to open the **Reconciliation Accounts** page.



On the **Reconciliation Accounts** page, you can create a new reconciliation account record, edit an record from the list by double-clicking it, delete, export or find a record. You can also view the debit and credit operations performed through each reconciliation account by double-clicking the desired account and observing the **Debit Operations** and **Credit Operations** sections.



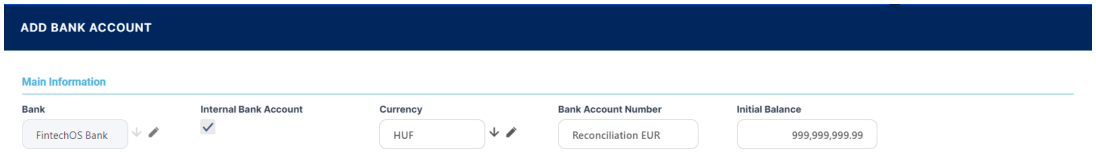
NOTE

You can open reconciliation accounts in every currency, but for the sake of automating some processes, Loan Management allows you to define which reconciliation account opened in a specific currency should be used within a period of time. Thanks to these settings, Loan Management determines automatically the reconciliation account to be used for a currency at a specific date. Read about these settings on the "Reconciliation Account Settings" on page 161 page.

Creating Reconciliation Accounts

Follow these steps to create reconciliation accounts:

1. In the FintechOS Portal, click the **Insert** button on the top right side of the **Reconciliation Accounts** page. The **Add Bank Account** page is displayed.
2. Fill in the following fields:



   - **Bank** - Automatically completed with the bank or financial institution marked as `Main Bank` in the system. You can't change this value.
   - **Internal Bank Account** - This checkbox specifies that the account is an internal bank account, used for reconciliation. Automatically checked as `True`. You can't change this value.
   - **Currency** - Select from the list the currency of the reconciliation account.
   - **Bank Account Number** - Enter the bank account number for the reconciliation account.
   - **Initial Balance** - Edit the reconciliation account's initial amount, which is automatically completed with the value of 999,999,999.00. You need the initial balance especially for those accounts that are used for debit purposes, representing the source for some transactions.
3. Click the **Save and Reload** button. The reconciliation account is saved and its status becomes **Opened**, ready to be used.

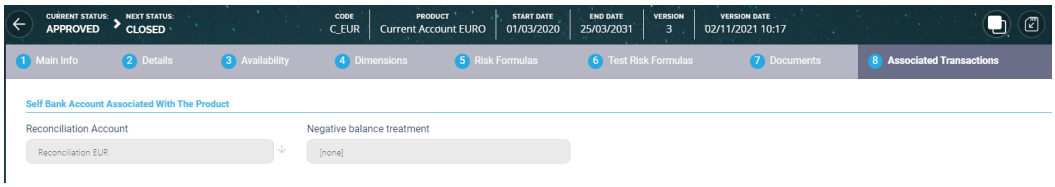
The **Debit Operations** and **Credit Operations** sections are now displayed, still empty. You can see new records in these two sections when transactions are performed for contracts based on banking products that use this reconciliation account. The following information is displayed about each transaction:

- **Value Date** - The date when the transaction was requested in the system.
- **Operation Date** - The date when the transaction was operated by the system.
- **Currency** - The currency of the transaction.
- **Amount** - The amount of the transaction.

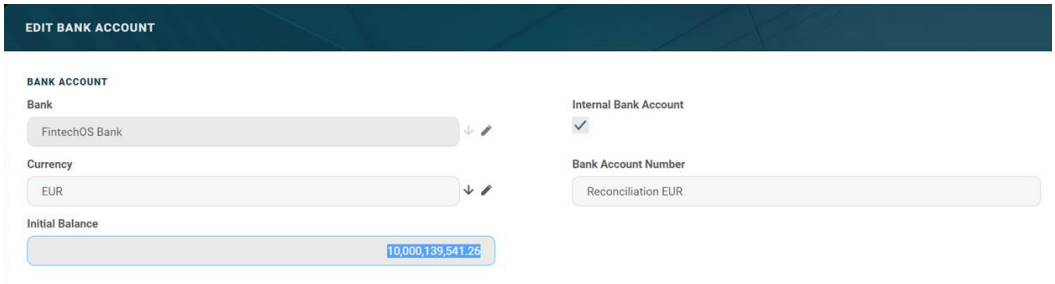
- **Detail Text** - The text representing information about the transaction, such as event type, repayment notification number, due date, and so on.

Example of Reconciliation Accounts Usage in Loan Management

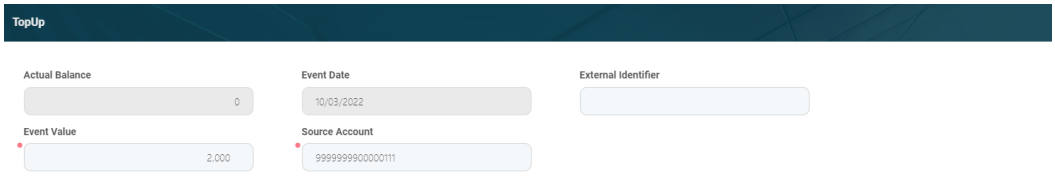
Let's consider the product definition of the `Current Account EURO` banking product, where the value selected for the **Reconciliation Account** field = `Reconciliation EUR`:



Checking the balance of the `Reconciliation EUR` account in the Reconciliation Accounts menu, we see the **Initial Balance** = 10,000,139,541.26:

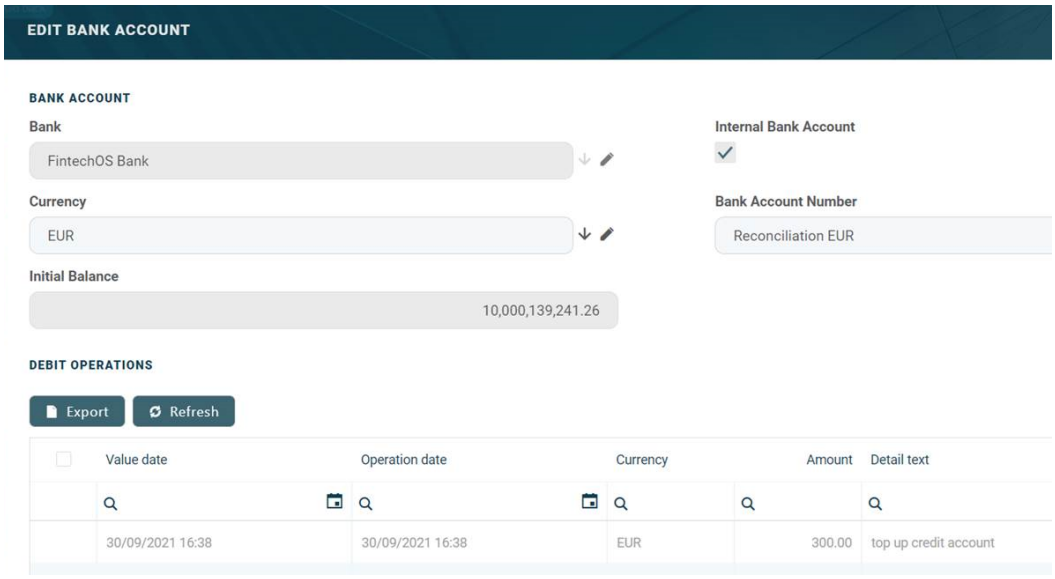


Using an approved contract based on the `Current Account EURO` banking product, we inserted and approved a **Top-Up Account** transaction type, with an **Event Value** = 300:



Checking back to the `Reconciliation EUR` account in the Reconciliation Accounts menu, the balance of the reconciliation account is updated to reflect the transaction just inserted above. The new balance value is

`10,000,139,241.26`, with a difference of -300 from the previous value. Observe that the same debit transaction is listed in the **Debit Operations** section:



Reconciliation Account Settings

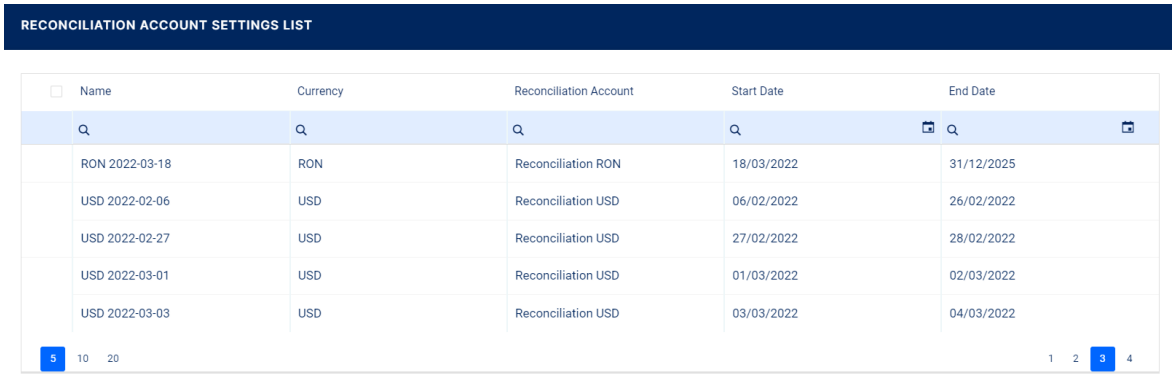
Reconciliation accounts can be opened in every currency, but for the sake of automating some processes, Loan Management allows you to define which reconciliation account opened in a specific currency should be used within a period of time. Thanks to these settings, Loan Management determines automatically the reconciliation account to be used for a currency at a specific date. For example, it validates the existence of a setting for a reconciliation account for a specific currency upon third-party agreement or agreement version approval. Loan Management also checks whether the reconciliation account setting has continuity for the entire validity period of the agreement. The reconciliation account is then automatically determined for usage within the third-party invoicing process.

The **Reconciliation Account Settings** menu item, accessible within the Portal's **Admin Configurations** menu, allows you to configure the default settings of reconciliation accounts.

IMPORTANT!
You must have the associated role of Loan Admin Officer to view, insert, update, or delete reconciliation account settings records.

To configure the default settings of reconciliation accounts:

1. In FintechOS Portal, click the main menu icon and expand the **Admin Configurations** menu.
2. Click the **Reconciliation Account Settings** menu item to open the **Reconciliation Account Settings List** page.



On the **Reconciliation Account Settings List** page, you can create a new reconciliation account setting, edit an record from the list by double-clicking it, delete, export or find a record.

NOTE
You can only delete a setting if the reconciliation account associated to it is not part of a third-party invoice.
You can only edit the `End Date` of a setting if the reconciliation account associated to it is part of a third-party invoice. The `End Date` must be >= than the current system date.

Creating Reconciliation Account Settings

Follow these steps to create reconciliation account settings:

1. Click **Insert** on the **Reconciliation Account Settings List** page to open the **Reconciliation Account Settings** page.

2. Fill in the following fields:



   - **Currency** - Select from the list the currency of the reconciliation account.
   - **Reconciliation Account** - Select the reconciliation account that Loan Management should automatically use for operations in the specified currency.
   - **Start Date** - Select the starting date for Loan Management to use this reconciliation account for operations in the specified currency.
   - **End Date** - Select the ending date for Loan Management to use this reconciliation account for operations in the specified currency. Make sure that `End Date>= Start Date`.

3. Click the **Save and Reload** button.

   Loan Management checks whether the start dates and end dates don't overlap for reconciliation accounts defined for the same currency. If the validation passes, the reconciliation account setting is saved with a unique name in the form of `Currency Code + Start Date`.

Customers, Groups and Limits

Financial institutions deal with customers, either individuals or legal entities. Customers may be part of groups. Loan Management enables you to manage customers and groups with the aid of dedicated menus.

To perform credit related activities, financial institutions can monitor their exposure by setting up limits for their customers. You can manage limits through a series of menus and reports available in Loan Management.

> **NOTE**
> You must record customers, groups (if applicable), and limits (depending on your system's configuration) to Loan Management before creating loan contracts for those customers.

This page contains a series of topics that explain how Loan Management is configured to work with customers, groups, and limits:

Customers **165**

Creating Customers 166

Groups **175**

Limits **177**

Managing Limits 187

Role-Based Limits 188

Managing Limit Types 191

Creating Limits 193

Creating New Versions of Existing Limits 198

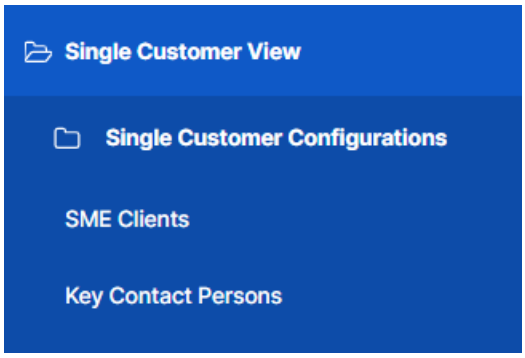
Customers

A customer is an individual or a legal entity who has an account with a bank. Opening an account is the crucial element in establishing the bank-customer relationship. You must create a record for each of your financial institution's customers in Loan Management. The customer records are stored in the Account entity.

IMPORTANT!
Perform complex customer management operations using the **Customer 360** solutions, as described here.
You can also manage customers and their relevant information from Loan Management.

Creating Customers Using the Customer 360 Menus

You can manage customers via the **Single Customer View** dedicated menus: **SME Clients**, **Banking Retail Clients** and **Key Contact Person**.



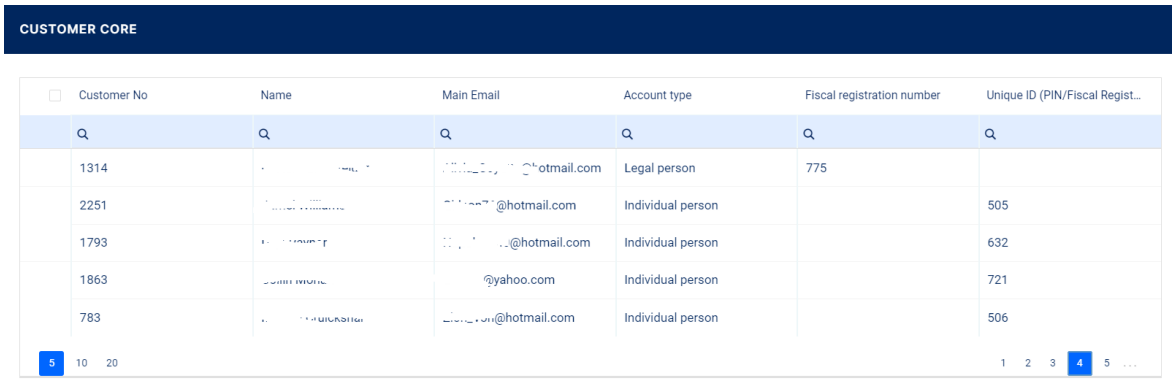
Create new customer records for customers following the steps described in the Adding Companies page.

After creating a record for a customer, proceed to filling in their details as described in the Managing Detailed Company Information page.

> **NOTE**
> A customer has to be in the `Prospect` or `Customer` status to be selected when creating contracts.

Alternatively, follow these steps to manage customers in Loan Management:

1. In FintechOS Portal, click the main menu icon and expand the **Loan Management Operational** menu.
2. Click **Customer Core** menu item to open the **Customer Core** page.



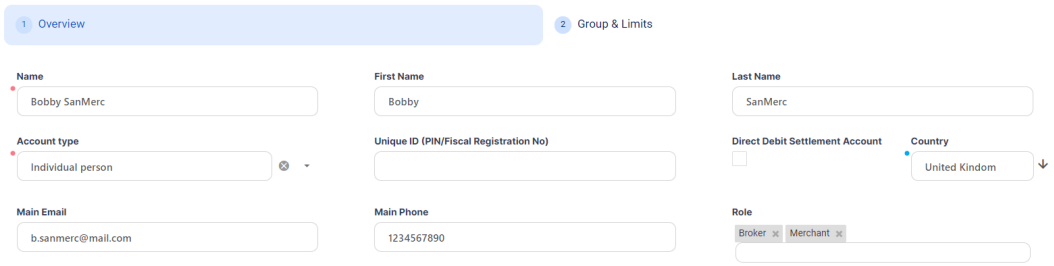
On the **Customer Core** page, you can add new customer records or search, edit, and delete existing ones.

Creating Customers

Follow these steps to create new customer records:

1. Click **Insert** on the **Customer Core** page to open the **Overview** page, the first in the customer creation process.

2. Fill in the following fields:



   - **Name** - Enter the name of the customer.
   - **Account Type** - Select the customer type from the possible options: `Legal person`, `Individual person` or `Self-employed individual`.

3. Optionally, fill in the following fields:
   - **First Name** - Enter the first name of the customer, if this is not a legal person.
   - **Last Name** - Enter the last name of the customer, if this is not a legal person.
   - **Unique ID (PIN/Fiscal Registration No)** - Enter the customer's unique ID:
     - For an `Individual customer` enter their personal identification number (PIN).
     - For a `Legal person` or a `Self employed individual` customer enter their fiscal registration number.
   - **Direct Debit Settlement Account** - Select this checkbox if the direct debit settlement for repayment notifications should be turned on for this customer, regardless of the settings at the contracts level.
     You can choose whether this setting should impact all the customer's existing contracts or not with the aid of a Loan Managementsystem parameter.
   - **Country** - Select the country of the customer. This can affect `Country Exposure` limits for customers part of groups.
   - **Main Email** - Enter the email address of the customer.
   - **Main Phone** - Enter the phone number of the customer.

- **Role** - Select as many roles as you wish for this customer to be able to have within contracts. The values are displayed from the `Contract Role` option set.

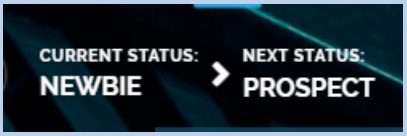
> **NOTE** Roles can affect the limits that can be set up for a customer. If a customer has a role, then they can have role-based limits that are configured with a limit type associated to the same role. For example, if the customer is declared as Merchant, they can have a limit based on a limit type with the associated Merchant role.

4. Click the **Save and Reload** button.

   The customer is saved in **Newbie** status, with minimum information, such as an auto-generated customer number, but you can now add more information about the customer in the following sections: **Company Representatives**, **Products**, **Bank Accounts** and **Collateral Register**. The **Groups & Limits** tab helps you with the configuration of group membership and limits for this customer.

5. Scroll through each of the newly displayed sections and fill in the necessary information, as described within the sections.

6. Select the **Groups & Limits** tab and fill in the necessary information, as described here.

> **NOTE**
> Change the customer's status `Prospect` or `Customer` to select that customer when you create contracts, as pictured here:
>
> 
>

Manage Company Representatives

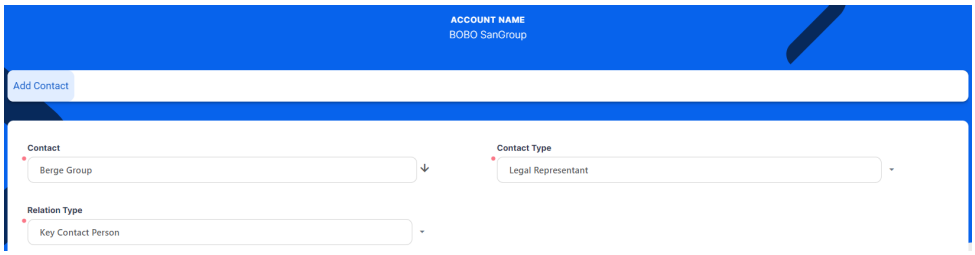
This section lists the customer’s legal representatives, such as administrators, affiliates, owners, or other key contact persons.



In this section, you can add new representatives, view the existing ones by clicking the desired record from the list, delete records from the list and export the list.

To add a new representative to the customer, follow these steps:

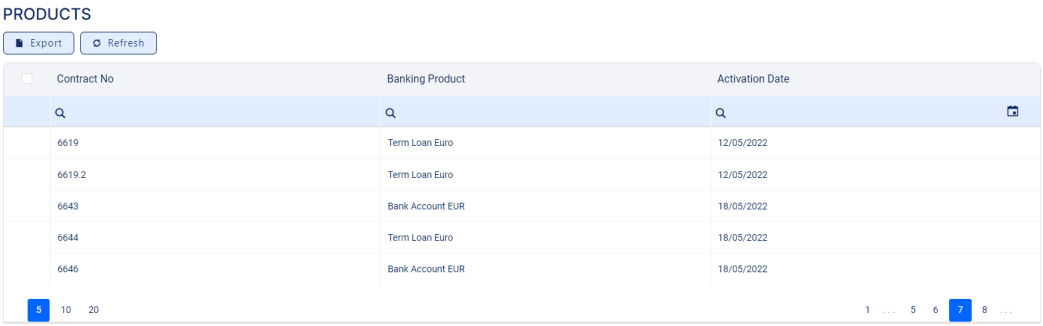
1. Click the **Insert** button within the **Company Representatives** section to open the **Add Contact** page.
2. Fill in the following fields:



   - **Contact** - Select an existing customer from the database as your new customer's representative.
   - **Contact Type** - Select the contact type from the drop-down list.
   - **Account Relation Type** - Select the representative's type of relation with the customer from the drop-down list of possible values.
3. Click the **Save and Close** button.

View Products

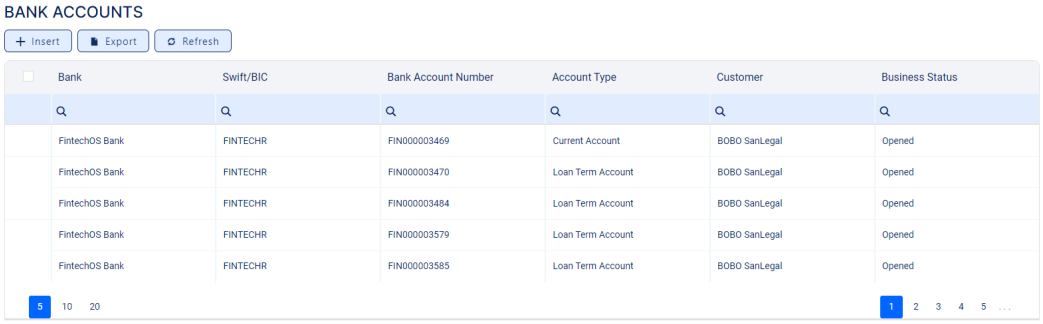
View the contracts opened for the customer in the **Products** section. The list displays information about the number of the contract, the banking product, and the activation date.



Click the desired record from the list to open the **Contract** page with the selected contract's information.

View Bank Accounts

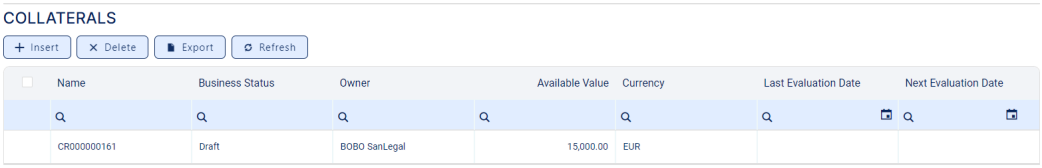
View the customer’s bank accounts in the **Bank Accounts** section, with details such as name of the bank, Swift/BIC, account number and type, customer, and business status.



Click the desired record from the list to open the **Bank Account** page with the selected bank account's information.

Manage Collateral Register

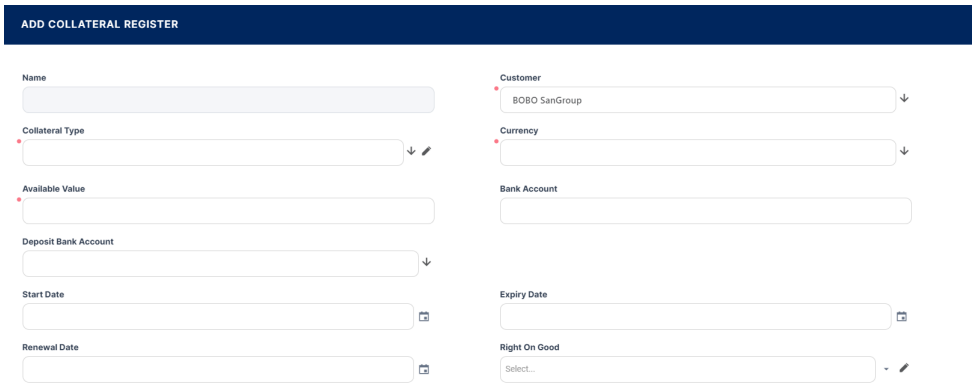
Manage the customer’s registered collaterals in the **Collaterals** section, displaying details such as name, business status, owner, available value, currency, last and next evaluation dates.



In this section, you can add new collateral register records, view the existing ones by clicking the desired record from the list, delete records from the list and export the list.

To add a new collateral register record to the customer, follow these steps:

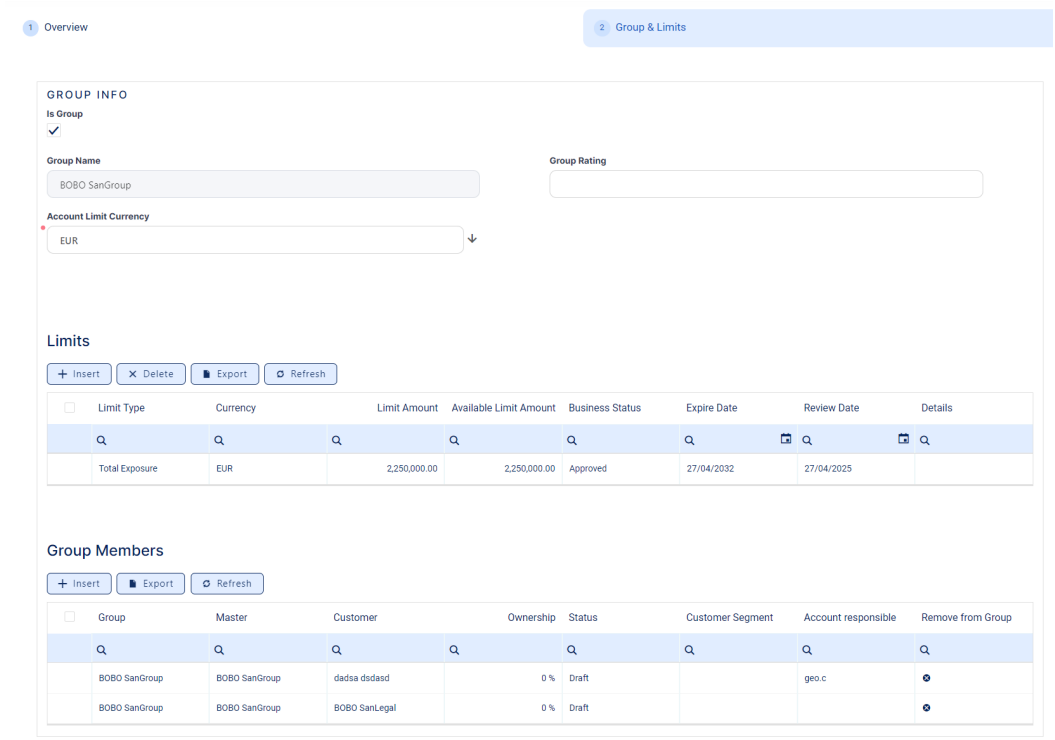
1. Click the **Insert** button within the **Collateral Register** section to display the **Add Collateral Register** page, with the `Customer` field automatically completed with the current customer's name.



2. Follow the steps described in the Registering Collaterals page of this guide.

Manage Groups & Limits

Access the **Group & Limits** tab to manage the limits set for the customer. If the customer is a group, then all the limits applicable for the group members are listed here, as well as details about the group members. To learn more about how limits and groups work, see the Limits page.



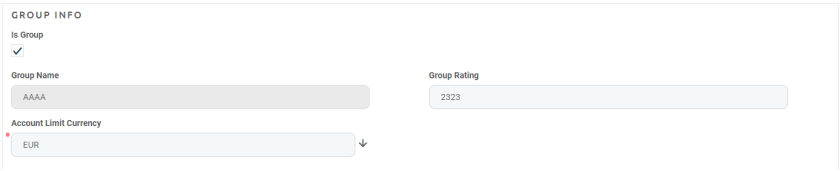
> **NOTE**
> You can add both legal entity and individual customers to groups. This can be helpful if you need to monitor group exposure for a household or a company and its shareholders together.
> When the `LimitMandatoryForIndividuals` system parameter is set to `True`, Loan Management performs limit validations for a group containing individual customers the same way as for groups composed solely of legal persons.

Go through the sections available within this tab to perform the following tasks:

Configure Group Info

You can specify whether a customer represents a group of companies in the **Group Info** section.

1. Fill in the following information within this section:

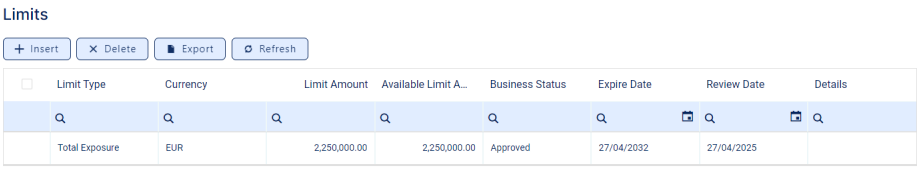


   - **Is Group** - Select the checkbox if the customer represents a group.
   - **Group Name** - This field is displayed only if the customer represents a group and it is automatically completed with the current customer's name.
   - **Account Limit Currency** - Select the currency of the customer limit.

2. If the customer is a group, fill in the **Group Rating** by entering the rating of the group.
3. Click the **Save and Reload** button.

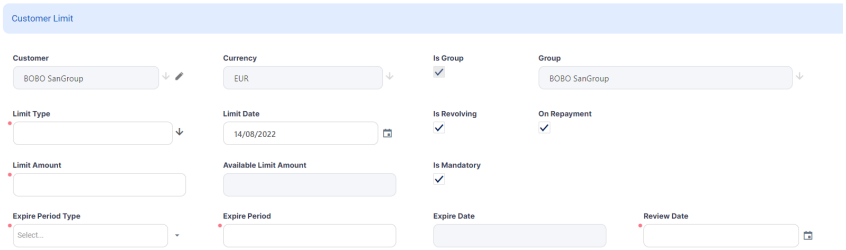
Set Limits

Set the customer's limits in the **Limits** section. If the customer is a group, then all the limits applicable for the group, coming from group members, are listed here.



To add a new limit record to the customer, follow these steps:

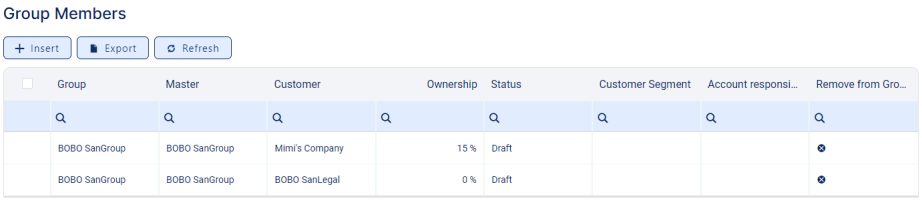
1. Click the **Insert** button within the **Limits** section to display the **Customer Limit** page, with the customer, the group and the currency automatically completed with the current customer's values.



2. Follow the steps described in the Creating Limits page of this guide.

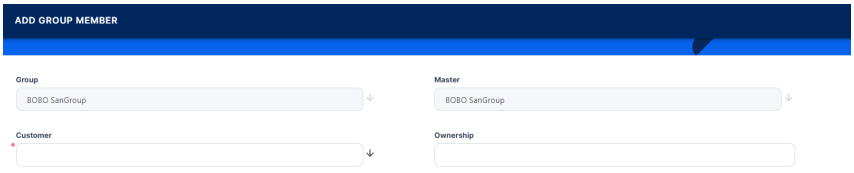
Manage Members/ Group Members Section

Manage the members/ group members' details in the **Members** or **Group Members** section. Here you can add new member records, view the existing ones by clicking the desired record from the list, and export the list.



To add a new member or group member to the customer, follow these steps:

1. Click the **Insert** button within the **Members/Group Members** section to display the **Add Member/ Add Group Member** page, with the master and the group automatically completed with the current customer's values.



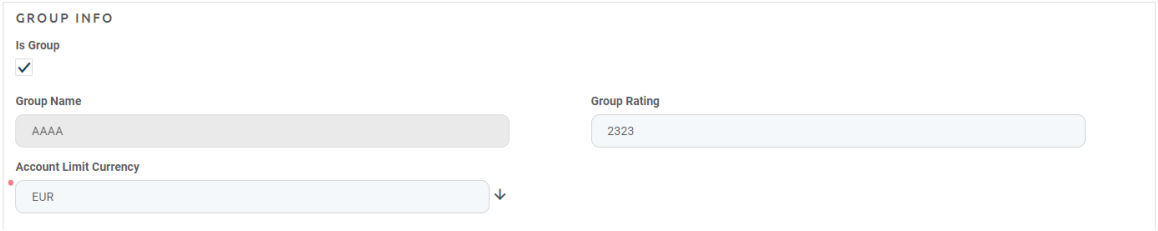
2. Follow the instructions from the Groups page of this guide, selecting a customer to add as a member, and entering an ownership percentage.

Groups

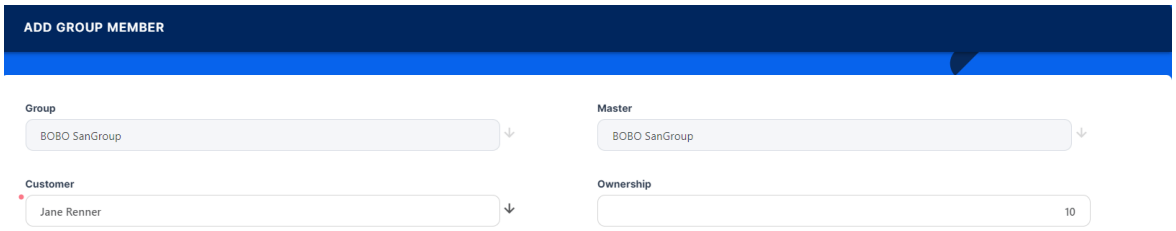
Loan Management allows you to define not only customers, but groups of customers as well. A corporate group or group of companies is a collection of parent and subsidiary corporations that function as a single economic entity through a common source of control. You can add both legal entity and individual customers to groups, or you can create groups of individual customers. This can be helpful if you need to monitor group exposure for a household or a company and its shareholders together.

IMPORTANT!
Complex group management operations are performed by the **Customer 360** apps. For detailed information, see the **Groups Info** section within the Customer 360 Legal Entities user guide.
You can also manage customers groups' relevant information from Loan Management.

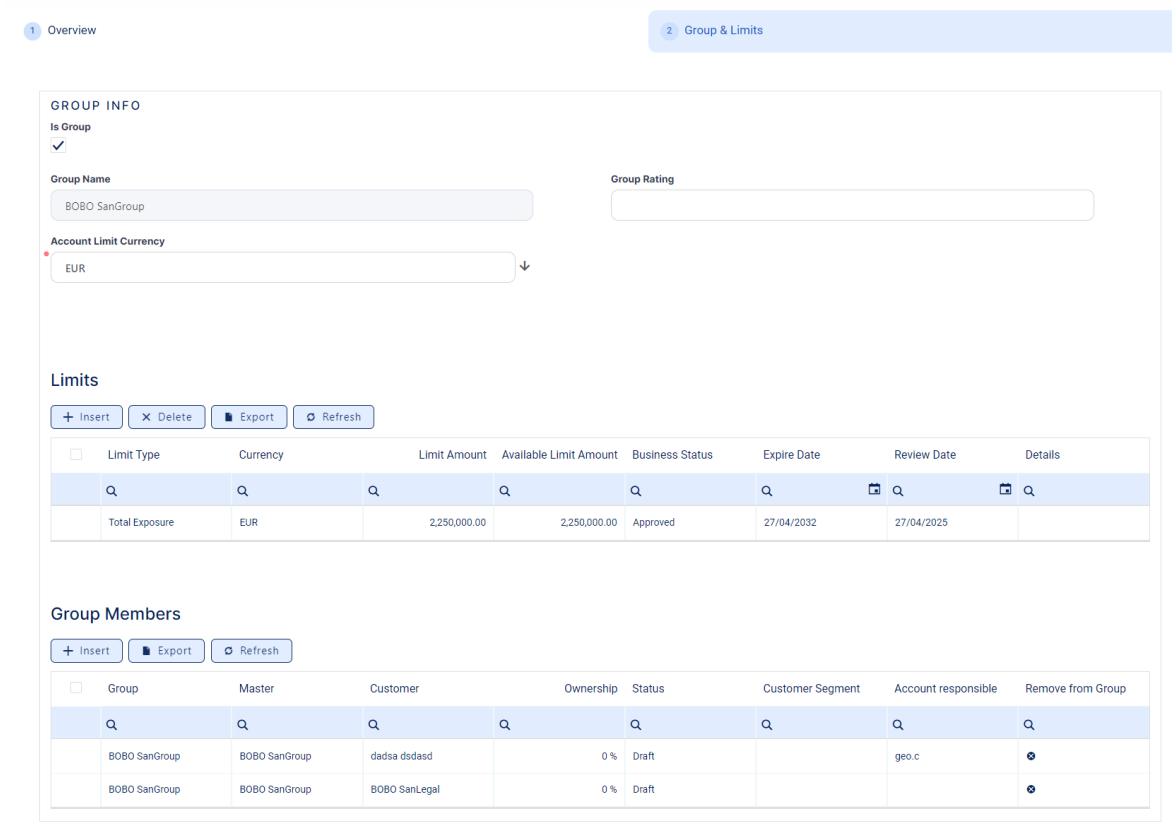
To define a group, select the checkbox **Is Group** from the **Group&Limits** tab of the **Customer** page, accessible through the **Loan Management Operational > Customer Core** menu. After you select the checkbox, the group name is automatically populated with the customer's name and you can insert a rating and a limit currency for that group.



Add new members to the group by clicking the **Insert** button from the **Group Members** section. If a member has other members associated with it, they are all displayed in the same section. For a more clear picture of the group, you can insert an ownership percent, determined by dividing the number of shares they own by the number of outstanding shares.



You can add subsidiary companies (sometimes referred to as child companies) to a company by clicking the **Insert** button from the **Group Members** tab. Therefore, a customer's child company is displayed in the **Group Members** tab.



If a customer is a child company for more than one company part of different groups, it impacts the available limit amount of the group to which it was first added.

NOTE
When a member is added to or deleted from a group, Loan Management automatically recalculates the limits of the group. The limits of the deleted member become as they were before entering the group.
If a member is moved from one group to another via API integration, the limits of both affected groups are automatically recalculated in real time.

Limits

The exposure is the risk a financial institution is taking on for writing the loan. Every time a financial institution grants any type of credit facility to a customer (a loan), the financial institution monitors its exposure to various financial indicators, which can negatively affect the customer and the institution itself. The financial institution uses various algorithms to calculate their exposure to the risks, but this calculation simply adds up to their exposure.

When referring to a loan, this page refers to all types of loans: unsecured loan, secured loan, overdraft, promissory note, working capital loan, and so on.

In FintechOSLoan Management an exposure can be related to a group or to a customer.

The approval of limits is subject to validation, depending on the type of customer. These validations are detailed below.

Group Exposure Types

- **Total Exposure** - the total amount of principal the lender is willing to allow the group to use in lending products at a certain moment in time.
- **Country Exposure** - the limit placed by a financial institution on the number of loans that can be given to borrowers in a particular country. They are used to control the financial institution' risk exposure to particular regions.

- **Company Exposure** - the exposure of a particular company in the group the lender is willing to allow for granting loans.
- **Product Type Exposure** - the maximum amount of credit an institution extends to the group for a specific type of product.
- **Product Exposure** - the maximum amount of credit an institution extends to the group for a specific product.
- **Exchange Exposure** - the risk a company undertakes when making financial transactions in foreign currencies. All currencies can experience periods of high volatility which can adversely affect profit margins, if suitable strategies are not in place to protect cash flow from sudden currency fluctuations.

Customer Exposure Types

- **Total Exposure** - the total amount of principal the lender is willing to allow the borrower to use in lending products at a certain moment in time.
- **Product Type Exposure** - the maximum amount of credit a financial institution extends to the customer for a specific type of product.
- **Product Exposure** - the maximum amount of credit a financial institution extends to the customer for a specific product.
- **Exchange Exposure** - the risk a company undertakes when making financial transactions in foreign currencies. All currencies can experience periods of high volatility which can adversely affect profit margins, if suitable strategies are not in place to protect cash flow from sudden currency fluctuations.

NOTE
You can define new limit types that are based on roles associated to contract participants specific to your business, and use them throughout Loan Management with all the functionality of any other default limit type. Read the dedicated page to learn how to manage limit type records.

Validations

IMPORTANT!
The `LimitMandatoryForIndividual` Loan Management system parameter allows banks to specify whether their system should validate limits for individual customer, the same way it validates limits for legal entity customers. The limits for legal entities and groups are validated by Loan Management by default.

For Customers Not Belonging to Groups

- **Total Exposure** is validated to be unique.
- **Product Type Exposure** is validated against the approved and active total exposure set on the customer.
- **Product Exposure** is validated against the Product Type Exposure if exists. If a Product Type Exposure does not exist, it is validated against the Total Exposure.
- **Exchange Exposure** is validated against Total Exposure.

For Customers Belonging to Groups

- **Total Exposure** is validated to be unique and it is validated against the Total Exposure set on the group.
- When a group defines a Company Exposure, a Total Exposure is automatically created for that company.
- All the other limits are validated against their correspondent set on the customer's group, if exists. If the correspondent does not exist, there are validated against Total Exposure from the group.
- The account limit currency is automatically filled in with the group limit currency.

NOTE
Both legal entity and individual customers can be added to groups. This can be helpful if you need to monitor group exposure for a household or a

company and its shareholders together.
When `LimitMandatoryForIndividual` = `True`, limit validations for a group containing individual customers happen the same way as for groups composed solely of legal persons.

For Groups

- **Total Exposure** is validated to be unique.
- **Product Type Exposure** is validated against the approved and active Total Exposure set on the group.
- **Product Exposure** is validated against the Product Type Exposure if it exists. If a Product Type exposure does not exist, it is validated against the Total Exposure.
- **Company Exposure** is validated against the approved and active total exposure set on the group.
- **Country Exposure** is validated against the approved and active total exposure set on that group.
- **Exchange Exposure** is validated against Total Exposure.
- You can define as many limits with the same Type (on Group or on Customer) as long as only one Limit (Type) is in **Approved** status. After setting the limits, the loan approval is validated against those limits, as detailed below:
    - If there is not at least one limit set at the customer or group level, the approval of the loan is not possible and an explicit error is displayed.
    - Contract maturity date cannot exceed the limit's expiry date and an explicit error is displayed.
    - The loan amount cannot exceed the corresponding limit amount. If not, an explicit error is displayed.

NOTE
When a member is added to or deleted from a group, Loan Management

automatically recalculates the limits of the group. The limits of the deleted member become as they were before entering the group.
If a member is moved from one group to another via API integration, the limits of both affected groups are automatically recalculated in real time.

For Role-Based Limits

Role-based limits have all the functionality of any other system limit type. The limits defined for participants at the contract level can be updated according to the contract's value. If a limit is set as revolving, it is replenished with capital repayments.

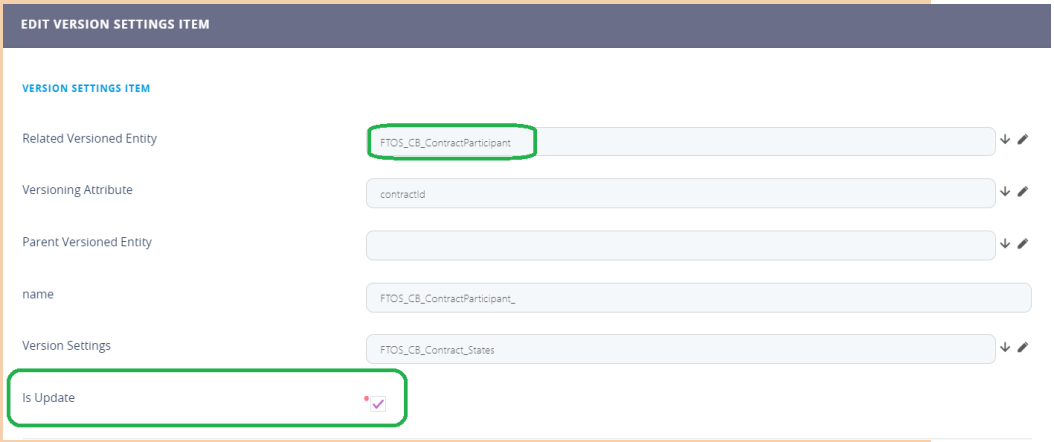
For contracts based on a banking product with a mandatory role configured at the product level, Loan Management checks whether the contract contains a participant with the same role. For example, for a banking product with `Merchant` mandatory role, if Loan Management doesn't find a participant with this `Merchant` role on the contract, then an error message informs you that "*Contract participants are blocking disbursement (Merchant)!*". In this case, add a participant with the `Merchant` role to the contract.

If the `Search Limit` checkbox was selected for the mandatory role at the banking product level, then Loan Management checks whether there is a contract participant whose limit is of the limit type associated with the same role. In the example above, Loan Management checks the existence of a participant who has a `Merchant Exposure` type limit.

If the existing limit's available amount is smaller than the value of the contract, then Loan Management checks the limit's `Is Mandatory` field. If `Is Mandatory = True`, then an error is raised that the limit is reached and the contract cannot be approved, otherwise, a warning is presented but the contract can be approved.

NOTE
The `Is Mandatory` field's value cannot be changed from `False` to `True` when versioning a limit until Available Limit Amount >= 0.

IMPORTANT!
The way the system is configured by default, there are no validations at **Contract Version** approval for contract participants' limits. If this is desired, the version settings for `Contract Participants` need to be changed from `IsUpdate= true` to `IsUpdate = false`.



Calculation of Available Limit Amount

After loan approval, the available amount for each corresponding limit is recalculated by subtracting the loan amount from the limit amount. When calculating the group limit available amount, the application takes into account all group members. If the limit currency and loan currency are different, the application automatically converts the loan amount using the current exchange rate.

All group and customer limits are updated daily in accordance with the exchange rate. This is done via a job called `Daily Limit Recalculation`.

If a limit is revolving (`Is Revolving = True` at the limit level), then the limit is a revolving limit, meaning that the Available Amount of the limit is replenished either on each repayment of the principal or on loan contract closure, depending on the `On Repayment` field's value. If `Is Revolving = False`, then the limit is not revolving in any circumstances.

At a revolving limit's level, if `On Repayment = True`, then the Available Amount of the limit is replenished on each repayment of the principal with the repayment value. If `On Repayment = False`, then the limit amount is replenished on loan contract closure with the amount of the contract.

If a customer that already has approved contracts becomes a member of a group, all its active limits are suspended. The same applies when excluding a customer from a group.

If a customer is a child company for more than one company which are part of different groups, it should have impact on the available limit amount on the group to which it was first added, unless if it was already part of a group.

Limits Lifecycle

Limit Statuses

The four-eyes principle is applicable for limits in FintechOSLoan Management, meaning that a record should be approved by a second bank employee, with higher authorization rights. This is enabled via approval task FintechOS Platform capabilities and thus it is also a bank's responsibility to set proper security roles and access rights to its users, in order to make sure that the same user can't insert and also authorize the same record.

A limit record has the following business workflow statuses:

- **Draft** - the status of a newly created limit record that was not yet sent for approval. While in this status, you can edit some fields, but you can't use it in contracts. Send the record to approval after editing all the necessary details.
- **Pending** - this is a system status applied to limits or limit versions sent for approval, but not yet approved. No updates of the records are available in this system status.
- **LimitApproved** - the status of a limit record after being authorized for use throughout Loan Management by a user with customer limits approval competencies. While in this status, you cannot edit the record's details. If you need to alter the limit's details, create a new version based on the current limit.
- **Closed** - the last status of a limit, after manually closing it or after creating a new version based on the current version. No updates are allowed on the record. The limit record cannot be used anymore.

- **Suspended** - the status of limit records which are suspended at the moment and cannot be used. If a customer is introduced into a group, the customer's limits are all suspended automatically until the limit records are reviewed and new versions are created for them with updated information.
- **Expired** - the status of limits whose availability has expired, thus the record cannot be used anymore. You can edit a limit's expiration date to a future date by creating a new version.

> **IMPORTANT!**
> For the limit to be applied, it must be in **Approved** status.

Limit Versioning

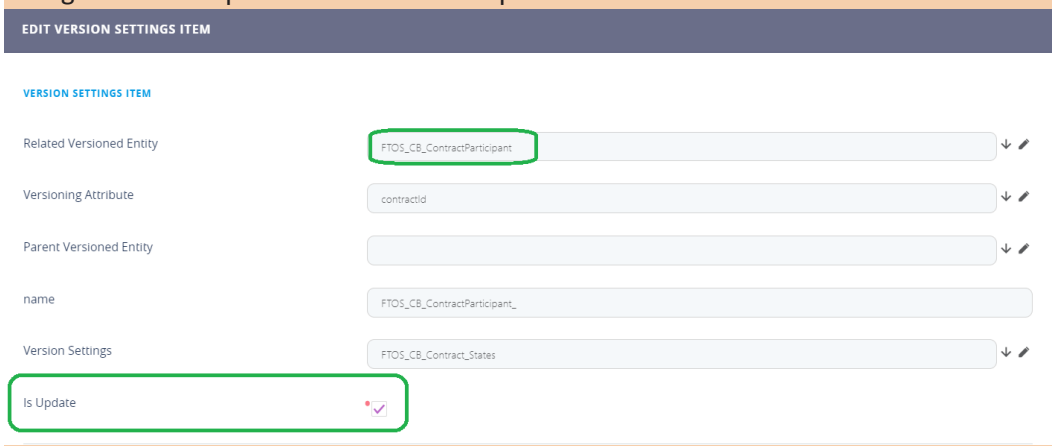
Loan Management allows you to create new versions for an existing limit if you need to modify an existing approved limit.

A limit version can have the following statuses:

- **Version Draft** - the status of a newly created limit version record that was not yet sent for approval. While in this status, you can edit some fields. Send the record to approval after editing all the necessary details.
- **Approved** - the status of a limit version record after being authorized by a user with customer limits approval competencies. While in this status, you cannot edit the record's details.
- **Version Closed** - the last status of a limit version, after manually closing it or after creating another new version based on the current version. No updates are allowed on the record.

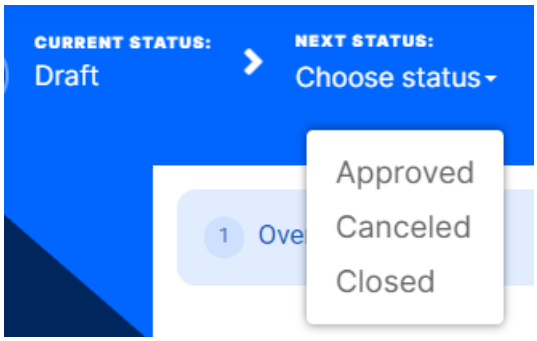
> **IMPORTANT!**
> The way the system is configured by default, there are no validations at **Contract Version** approval for contract participants' limits. If this is desired, the version settings for `Contract Participants` needs to be

changed from `IsUpdate= true` to `IsUpdate = false`.

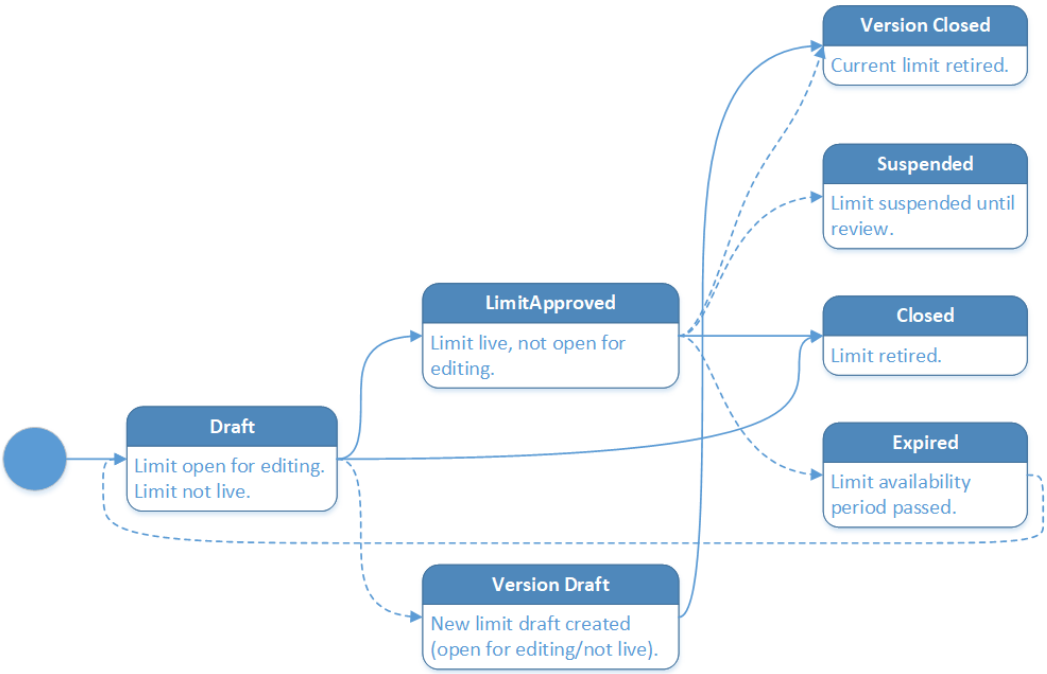


Changing Limit Statuses

You can manage a limit's life cycle by changing its status from the top right corner of the screen.



The limit status transitions are illustrated below:



Note that:

- Once a record is live, its settings can no longer be modified.
- If you want to update the details of a live limit, you must create a new limit version.
- When you create a new limit version, the current version is retired and moved to history; no updates are allowed on the retired version.
- Every limit version starts in a draft state and must go through an approval process before going live.
- Only one version of a limit can be live at one time.

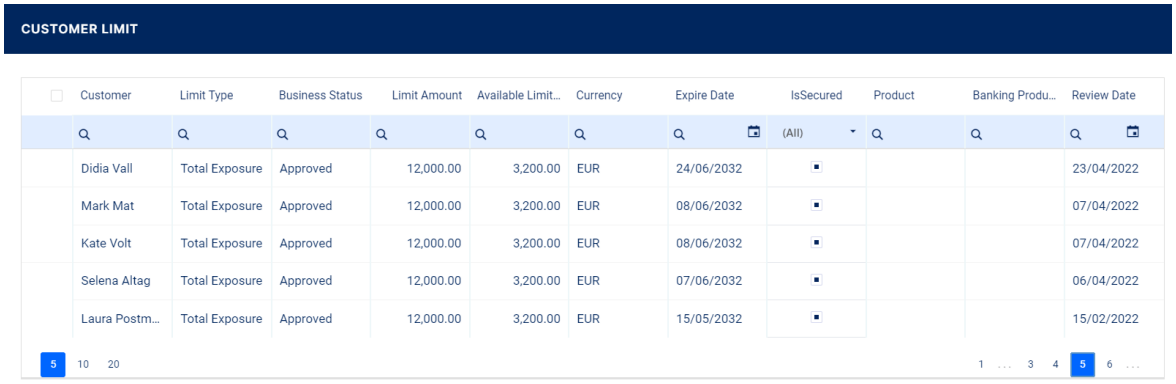
NOTE
As a best practice, new records or new versions of existing records created on a specific day should be approved on the same day.

Managing Limits

> **NOTE**
> You must have the **Corporate Credit Officer**, **Retail Credit Officer**, or **Risk Officer** security roles to add and update limits. Other associated roles allow you only to read limits information.

To manage limit records:

1. In FintechOS Portal, click the main menu icon and expand the **Loan Management Operational** menu.
2. Click **Customer Limit** menu item to open the **Customer Limits List** page.



On the **Customer Limits List** page, you can create a new limit record for a customer, delete records in Draft status, and search for a specific record. You can also edit the information for limits in **Draft** or **Version Draft** status, or create new versions for approved limits to change their information.

Alternatively, you can also manage limits at a customer level from the **Customer Core** menu, by selecting a customer from the list and managing their limit records within the **Groups & Limits** tab.

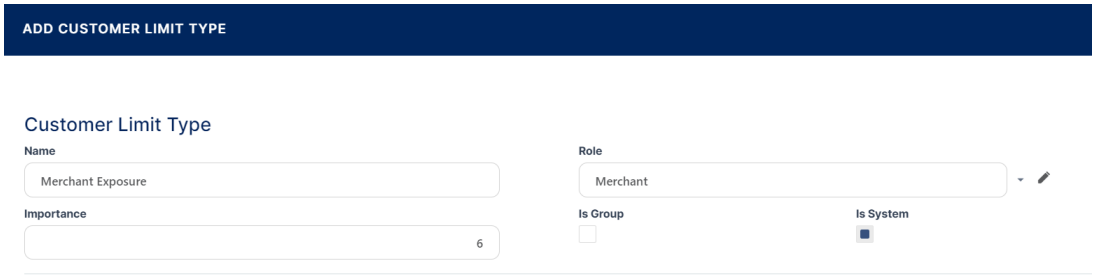
Role-Based Limits

Role-based limit capabilities allow you to manage limits for different customer types, such as merchants. Using role-based limits, the limit for a customer who is a merchant within several contracts can be configured properly, allowing the customer to take loans until they reach their set limit.

To use role-based limits within your contracts, follow these steps:

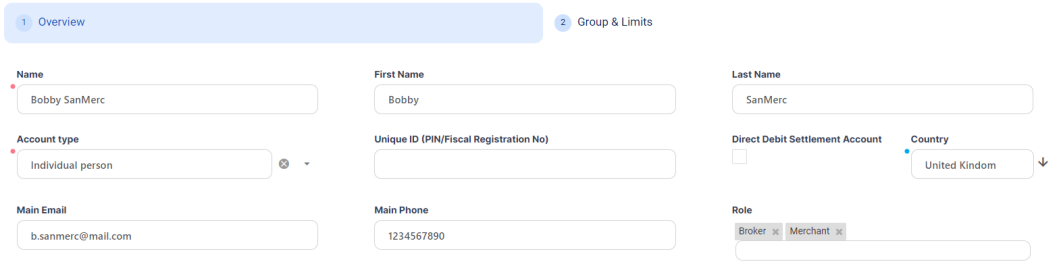
1. Define new limit types that are based on roles associated with contract participants specific to your business.

   You can use these new limit types throughout Loan Management with all the functionality of any other default limit type. For example, you can configure a `Merchant Exposure` limit type, to enable the creation of limits for customers who have the `Merchant` role associated at a customer level.



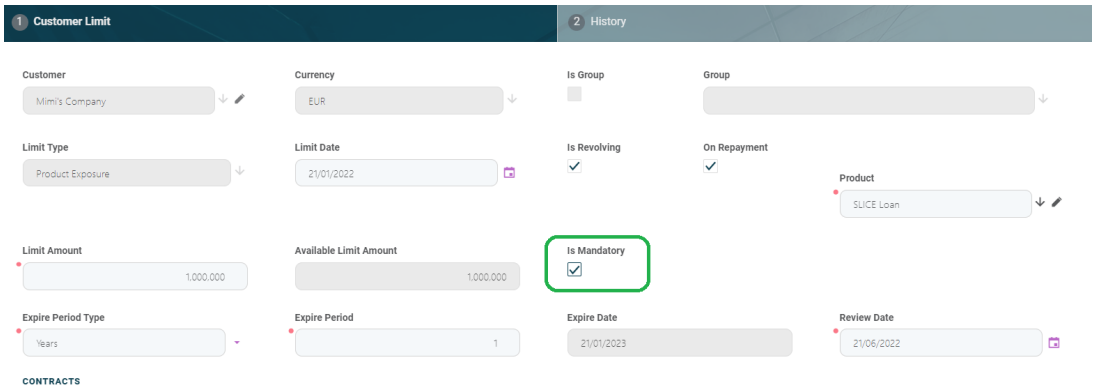
2. Associate the same role to the customer.

   You can associate as many roles as you need for a customer to be able to have within contracts, using the `Role` field added to the **Customer** page accessible through the **Customer Core** menu.
   Limits defined for roles at a customer level are treated as system limits and they are affected by contracts where that specific customer plays that role. For example, if the customer has a `Merchant` role, you can define them a `Merchant Exposure` type limit.



3. Decide whether a customer limit is a mandatory limit or not.

   The `Is Mandatory` field's value within the **Customer Limit** page configures the limit validation at the contract level. When the existing limit's available amount is smaller than the value of the contract, Loan Management checks the limit's `Is Mandatory` field. If `Is Mandatory = True`, then an error is raised that the limit is reached and the contract cannot be approved, otherwise, if `Is Mandatory = False`, a warning is presented but the contract can be approved. The default value is `True`.

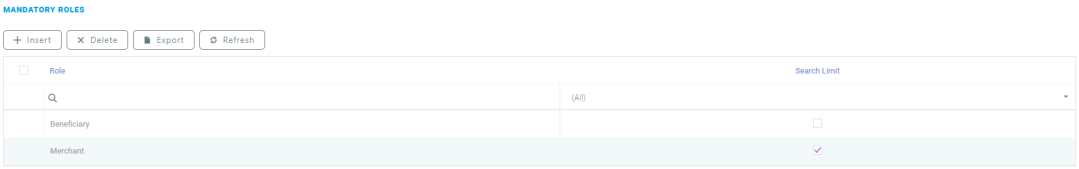


4. Configure mandatory roles at the banking products level.

   The **Mandatory Roles** section within the **Availability** tab at the banking product level allows you to add the roles of the participants that are mandatory to exist at the contract level for contracts based on this banking product. In other words, when creating contracts based on banking products with the `Merchant` role in this section, you must add a customer with the same `Merchant` role as a contract participant, otherwise, the contract cannot be approved.

   When `Search Limit` is selected for a role on a banking product, Loan Management

checks if the contract participant with this role has an attached limit configured with a limit type associated with the same role.



> **NOTE**
> Verify the access rights for users with Corporate Credit Officer, Retail Credit Officer, and Risk Officer security roles. The out-of-the-box settings for these security roles allow users to add and update limits, while users with other associated roles can only read limit information. Update the access rights according to your financial institution's needs.

Role-Based Limits Validations

Role-based limits have all the functionality of any other system limit type. The limits defined for participants at the contract level can be updated according to the contract's value. If a limit is set as revolving, it is replenished with capital repayments.

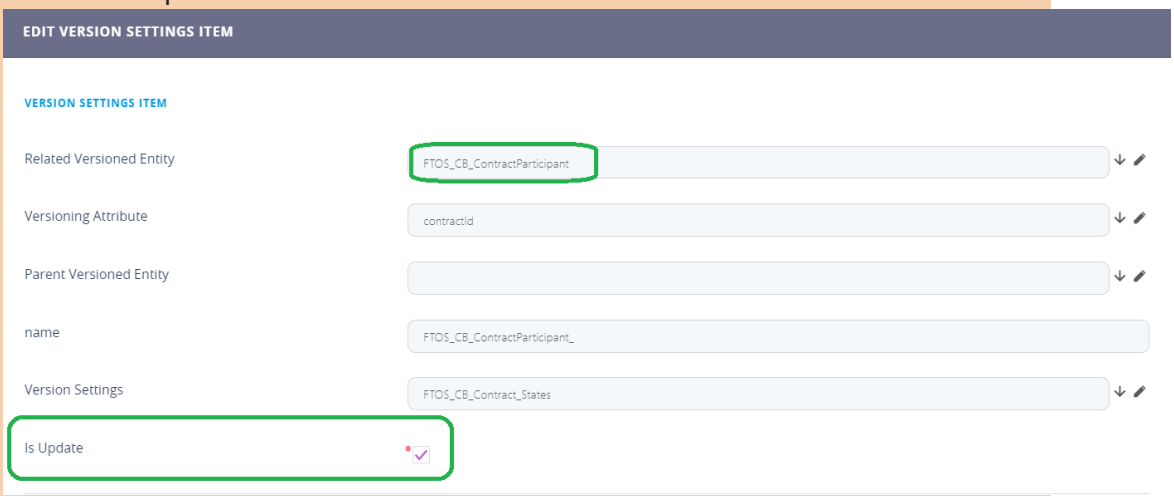
For contracts based on a banking product with a mandatory role configured at the product level, Loan Management checks whether the contract contains a participant with the same role. For example, for a banking product with `Merchant` mandatory role, if Loan Management doesn't find a participant with this `Merchant` role on the contract, then an error message informs you that "*Contract participants are blocking disbursement (Merchant)!*". In this case, add a participant with the `Merchant` role to the contract.

If the `Search Limit` checkbox was selected for the mandatory role at the banking product level, then Loan Management checks whether there is a contract participant whose limit is of the limit type associated with the same role. In the example above, Loan Management checks the existence of a participant who has a `Merchant Exposure` type limit.

If the existing limit's available amount is smaller than the value of the contract, then Loan Management checks the limit's `Is Mandatory` field. If `Is Mandatory = True`, then an error is raised that the limit is reached and the contract cannot be approved, otherwise, a warning is presented but the contract can be approved.

NOTE
The `Is Mandatory` field's value cannot be changed from `False` to `True` when versioning a limit until Available Limit Amount >= 0.

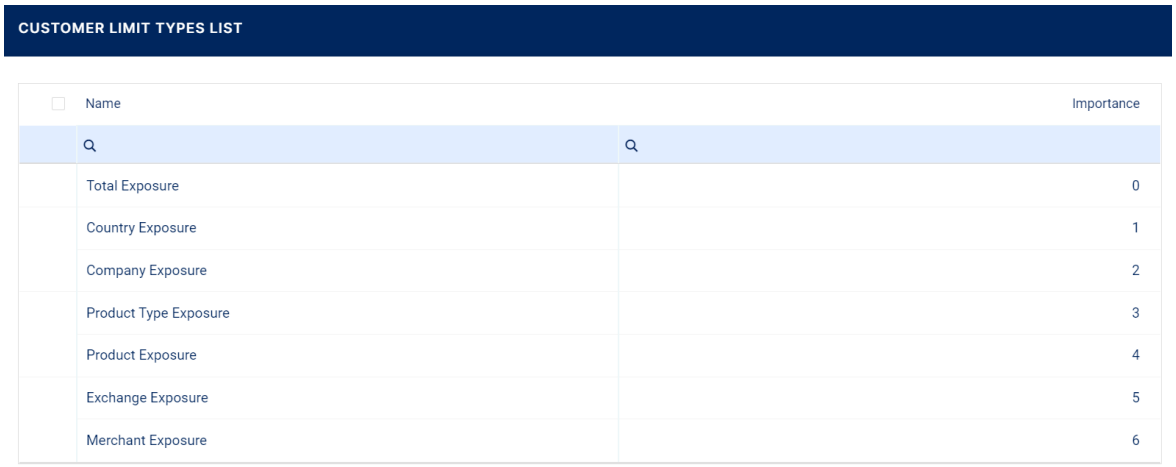
IMPORTANT!
The way the system is configured by default, there are no validations at **Contract Version** approval for contract participants' limits. If this is desired, the version settings for `Contract Participants` need to be changed from `IsUpdate= true` to `IsUpdate = false`.



Managing Limit Types

You can define new limit types that are based on roles associated to contract participants specific to your business, and use them throughout Loan Management with all the functionality of any other default limit type. To manage limit type records:

1. In FintechOS Portal, click the main menu icon and expand the **Admin Configurations** menu.
2. Click **Customer Limit Type** menu item to open the **Customer Limit Types List** page.



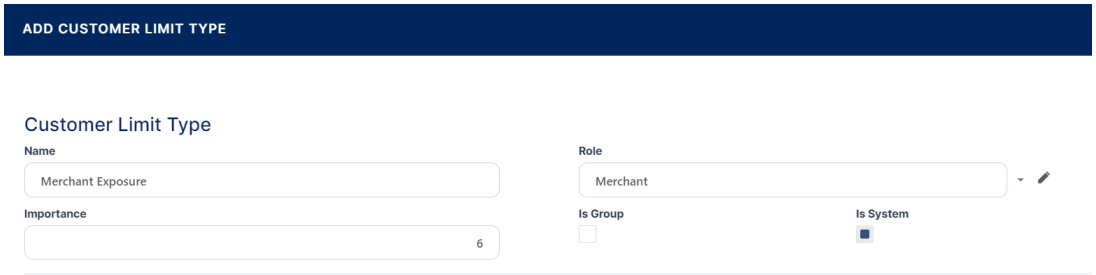
On the **Customer Limit Types List** page, you can add new limit type records or search, edit, and delete existing ones.

> **IMPORTANT!**
> You can only edit or delete limit types that are not marked as `Is System`. Limit types that come with the Loan Management packages are considered system limit types and they are applicable to the customer of a contract.

Creating Limit Types

To create a new limit record for a customer, follow these steps:

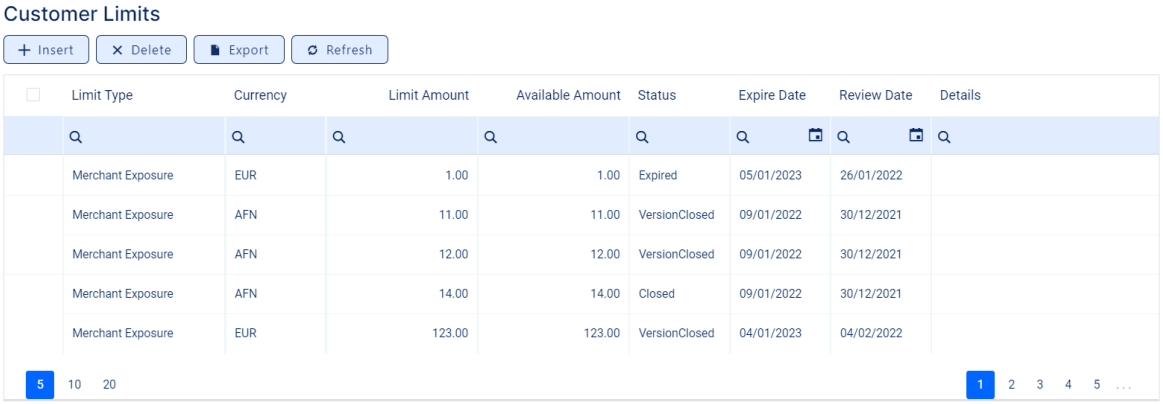
1. Click **Insert** on the **Customer Limit Types List** page to open the **Add Customer Limit Type** page.
2. Fill in the following fields:



- **Name** - Enter the name of the limit type.
- **Role** - Select the role of the contract participant for which this limit type can be used.
- **Importance** - Enter the order in which this limit is considered by the system. The lower the number, the higher the limit type's importance during the limit calculations.
- **Is Group** - Select this checkbox if the limit type is applicable to groups.
- **Is System** - This read-only field marks a record as system limit type, and only the limit types that come within the Loan Management packages are marked as system limit types and they are applicable to the customer of a contract. You can only edit or delete limit types that are not marked as `Is System`.

3. Click the **Save and Reload** button.

You can view the existing customer limits affected by this limit type in the **Customer Limits** section. Only customers with the same role selected on their customer record have their customer limit displayed here.



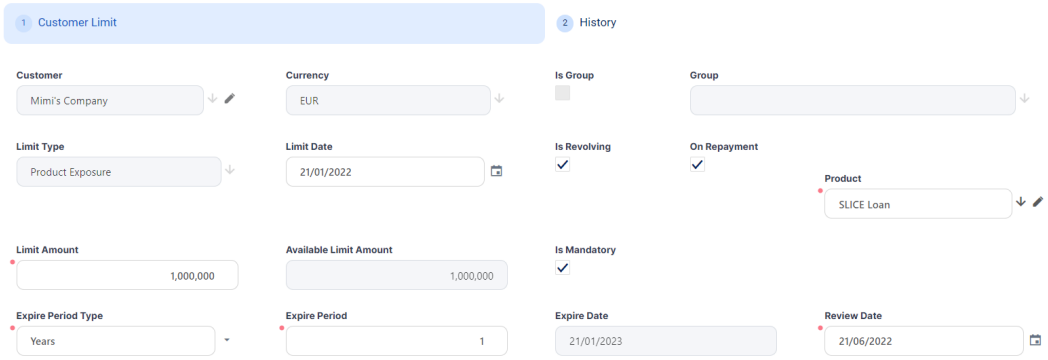
Creating Limits

IMPORTANT!
The `LimitMandatoryForIndividual` Loan Management system parameter

allows financial institutions to specify whether their system should validate limits for individual customer, the same way it validates limits for legal entity customers. The limits for legal entities and groups are validated by Loan Management by default.

To create a new limit record for a customer, follow these steps:

1. In the FintechOS Portal, click the main menu icon and expand the **Loan Management Operational** menu.
2. Click **Customer Limit** menu item to open the **Customer Limits List** page.
3. Click the **Insert** button to display the **Customer Limit** page.



Alternatively, click the **Add New Customer Limit** button in the **Customer Limits** dashboard to display the same **Customer Limit** page.

Or you can manage limits at a customer level from the **Customer Core** menu, by selecting a customer from the list and managing their limit records within the **Groups & Limits** tab.

4. Fill in the following fields:

- **Customer** - Select the customer to whom the limit is associated.
- **Currency** - Select the currency for this limit.
- **Is Group** - Select the checkbox if the customer represents a group.
- **Group** - If the customer is not a group, this field is read-only. If the customer is a legal entity that is also a group, the name of the group is automatically filled in.

- **Limit Type** - Select the type from the list `Total Exposure`, `Product Type Exposure`, `Product Exposure`, `Exchange Exposure`, `Country Exposure`, `Company Exposure`, each predefined type explained in the Limits page, or `Role-based limits` (such as `Merchant Exposure`) associated with the same role that the customer has in its record.
  If a customer has a role, then they can have role-based limits configured with a limit type associated to the same role. For example, if the customer is declared as Merchant, they can have a limit based on a limit type with the associated Merchant role.

  > **IMPORTANT!**
  > The correlation between the limits and group is important as the limits on the parent entity affect the child entities. If the customer is a group, two additional types of exposures are available: country and company exposure.

- **Limit Date** - Enter the date when the limit becomes active. It is automatically completed with the current date, but it can be changed.
- **Is Revolving** - If you select this checkbox, then the limit is a revolving limit, meaning that the Available Amount of the limit is replenished either on each repayment of the principal or on loan contract closure, depending on the `On Repayment` field's value. If the checkbox is not selected, then the limit is not revolving in any circumstances. The default value is `True`.
- **On Repayment** - You can only select this if `Is Revolving = True`. If selected, then the Available Amount of the limit is replenished on each repayment of the principal with the repayment value. If the checkbox is not selected for a revolving limit, then the limit amount is replenished on contract closure with the amount of the contract. The default value is `True`.

> **NOTE**
>
> You can't modify the `On Repayment` and `Is Revolving` fields after limit approval. If you must change these properties, close the limit and open a new limit with the required setup.

- **Limit Amount** - Select the amount representing the limit for the credit.
- **Available Limit Amount** - This field is automatically completed by Loan Management with the remaining amount, e.g. if the total exposure was $5 million, a credit was given for $3 million, $2 million is still available.
- **Is Mandatory** - Selected by default. At limit validation at the contract level, when the existing limit's available amount is smaller than the value of the contract, then Loan Management checks the limit's `Is Mandatory` field. If `Is Mandatory = True`, then an error is raised that the limit is reached and the contract cannot be approved, otherwise, if `Is Mandatory = False`, a warning is presented but the contract can be approved.

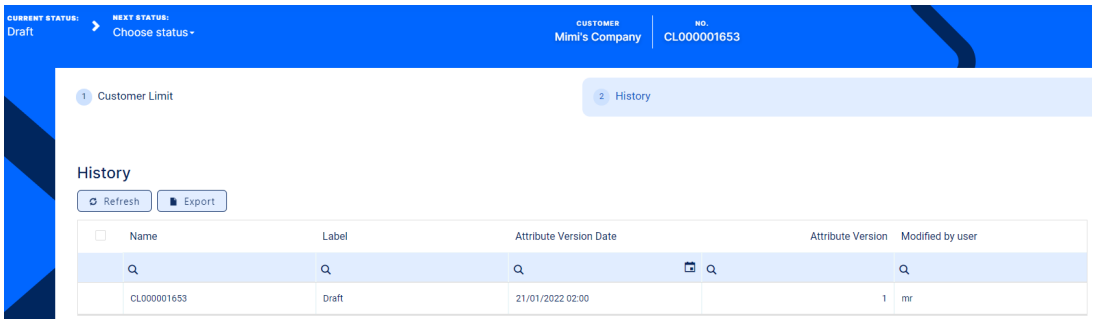
> **NOTE**
>
> The `Is Mandatory` field's value cannot be changed from `False` to `True` when versioning a limit until Available Limit Amount >= 0.

- **Expire Period Type** - Select from the list the period type applicable for this limit: `Days, Weeks, Months, Years`, or `Once`.
- **Expire Period** - Insert the number for the period, e.g. 4., i.e. 4 months.
- **Expire Date** - This field is automatically completed with the date when the limit expires, as calculated based on the values entered in the previous fields.
- **Review Date** - Select a date when the limit is reviewed.

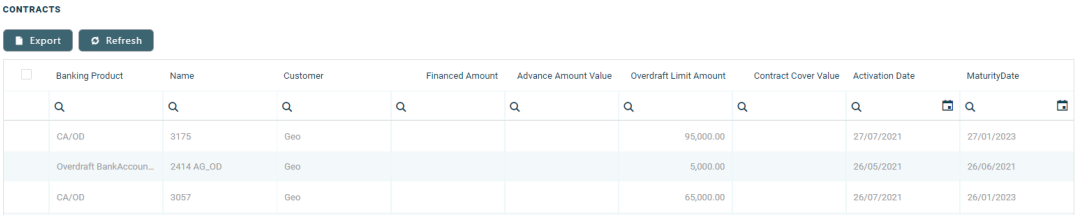
5. Depending on the type of exposure selected, fill in these additional fields:

- For `Exchange exposure`, select the **Exchange currency limit** from the list.
- For `Product Exposure`, select the **Product** from the list. From the total exposure amount, you can set a limited amount to be given on a certain product. For example, for a corporate term loan to give only $2 million dollars while the total exposure is $3 million.
- For `Product Type Exposure`, select the **Product Type** from the list. If the type is secured by an asset, then select the **Product type is secured** checkbox.
- For `Country Exposure`, select the **Country** where the limit is available.
- For `Company Limit`, select the **Company** for which the limit applies from the list of group members. See Groups for details about groups.

6. Click the **Save and Reload** button. Loan Management saves the limit and generates a number for a record, then displays it at the top of the page along with the name of the customer. The **History** tab is also displayed, containing information about each version of the record.



The existing contracts affected by this limit are displayed in the **Contracts** section.



7. Repeat to add as many limits as needed.

IMPORTANT!

For the same customer, only one limit exposure can be added.

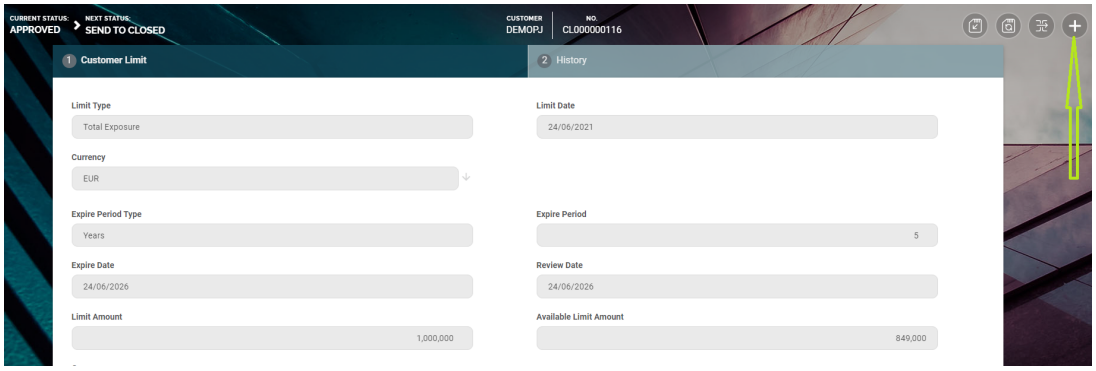
8. Send the limit record to approval by changing its status into **Send to Approved**.
9. As a user with customer limits approval rights, approve the record, so that it becomes active in the system. Approve the limit from the **Customer Limits** dashboard **>Customer Limits Approval Requests** tab or from the **Approval Tasks** menu.

Creating New Versions of Existing Limits

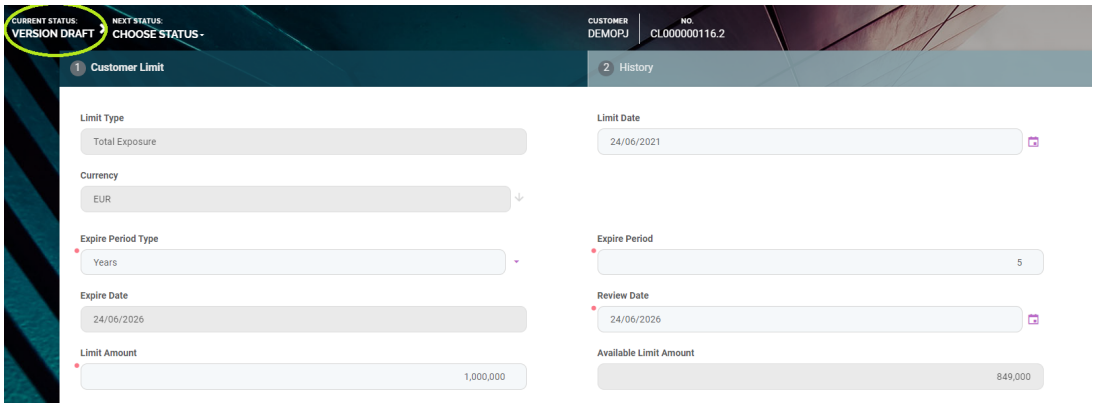
The limits are set up for versioning. Thus, if the details of an approved limit have to be updated, a new version of the record must be created.

To create a new version for a record with the **Approved** status, follow these steps:

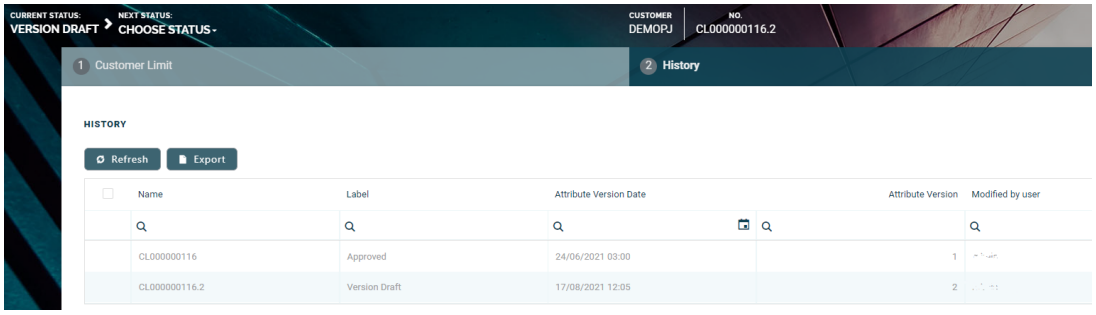
1. Double-click the limit record selected for updates.
2. Click the **New Version** button in the top right corner of the page.



A new version of the limit is created, with **Version Draft** status.



3. Edit the desired fields in the **Customer Limit** tab. You cannot edit the limit type, currency, group and available limit amount.
4. Click the **Save and Reload** button. The number of the record is automatically updated and displayed at the top of the page. The **History** tab is also displayed, containing information about each version of the record.



If the version draft record is approved, then the original record transitions into the **Version Closed** status and the secondary version becomes the **Approved** currently active limit record.

Read more details about versioning a record on the How to Version an Entity Record page.

Collaterals

A collateral is a property, such as securities, items of value, pledged by a borrower to protect the interests of the lender. A lender can seize the collateral from a borrower if the latter fails to repay a loan according to the agreed terms. A collateral acts as a guarantee that the lender receives the amount lent even if the borrower does not repay the loan as agreed. For example, when contracting a mortgage, the bank asks the customer to provide their house as collateral. If the customer fails to meet the repayment terms of their mortgage, the bank has the right to take ownership of the house. The bank can then sell the house in order to recoup the money that it lent to the customer.

Collateral management is the method of granting, verifying and managing collateral transactions in order to reduce credit risk in unsecured financial transactions. It is an essential and integral part of any financial institution's risk and regulatory compliance framework.

There is a wide range of possible collaterals used to hedge credit exposure with various degrees of risk:

- Cash Collateral: Fixed Deposit, Stocks, etc.
- Real estate: Property, Land
- Other: metals, commodities, etc.

Collateral System Statuses

In Loan Management, a registered collateral has the following statuses:

- **Draft** - the status of a newly created collateral registration record that was not yet cleared to be used. While in this status, you can edit its fields. Change the status of the record to **Active** after editing all the necessary details in order to use it later in contracts.
- **Active** - the status of a collateral registration record after being authorized for usage in contracts.

- **Owned** - the collateral is being used by a contract. It is linked to a loan or any other secured product (overdraft, bank guarantee, etc).
- **Released** - the status of a collateral after closing the contract to which it was attached.

> **IMPORTANT!**
> In order to use the collateral as a guarantee for covering a secured loan contract, it must have the **Active** status.

Managing Collaterals

Perform the following steps to manage collaterals in FintechOS Loan Management:

1. Add new guarantee types, if your financial institutions wishes to work with guarantee types other than the default ones. If not, skip to Registering Collaterals.
2. Create collateral types based on the newly added guarantee types, if applicable. If not, skip to Registering Collaterals.
3. Register a collateral before using it within a contract. Registration is performed based on a collateral type.

The first two steps are usually performed during Loan Management configuration, while the 3rd step is performed each time you must register an asset as a collateral, to be later on used to cover a contract.

> **IMPORTANT!**
> You must first register a collateral so that you can use it as a guarantee for covering a secured loan contract.

Setting Guarantee Types

There are two guarantee types defined by default in the `GuaranteeTypes` option set that cover most of the business requirements: `Real Estate` and `Cash Collateral`.

if your financial institutions wishes to work with guarantee types other than the default ones, add new guarantee types following these steps:

1. In FintechOS Studio, click the main menu icon, expand the **Admin** menu, and click **Option Sets** to open the **Option Sets List** page.
2. Find the `GuaranteeTypes` option set, storing the guarantee types, and double-click it to open the **Edit Option Set** page.
3. In the **Option Set Items** section, click the **Insert** button to open the **Add Option Set Item** page.
4. Add the details of the new guarantee type by filling in the following fields:

- **Name** - Enter the name of the guarantee type.
- **Display Name** - Enter the display name of the guarantee type.
- **Value** - Enter the value of the guarantee type.
- **Status Id** - Select the status of the item within the option set: active or inactive. Default value: `Active`.

5. Click the **Save and Close** button. The id of the option set item is automatically generated when saving the record.

Guarantee types are further used to define collateral types.

Creating Collateral Types

In FintechOS Loan Management there is a large range of predefined collateral types:

- Predefined collaterals for Cash Collateral guarantee: Fixed Deposit, Stock, Bonds.
- Predefined collaterals for Real Estate guarantee: Land and Property.

You can create new ones for your business specific needs. To add new collateral types, follow the steps described in the Collateral Type page within the Banking Product Factory user guide.

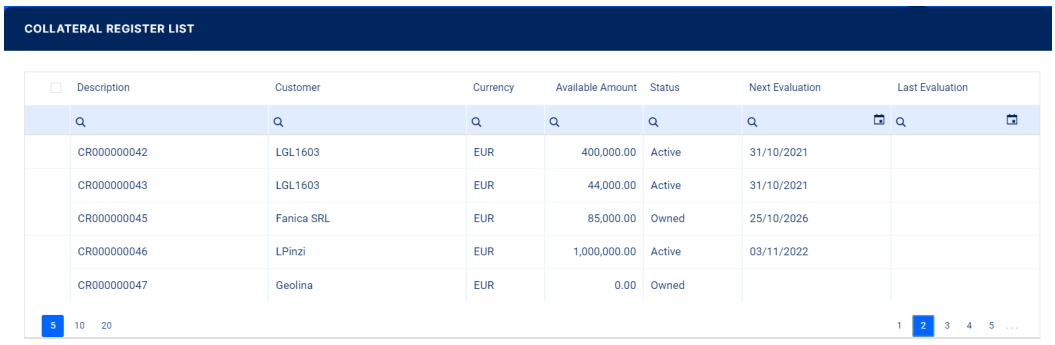
Registering Collaterals

You should register a collateral to support managing secured loans. You can attach a registered collateral to multiple contracts if the contracts total amount does not exceed the collateral available amount.

In order to use the collateral as a guarantee for covering a secured loan contract, you must first insert it in the collateral register. Follow these steps to register a collateral:

1. In the FintechOS Portal, click the main menu icon and expand the **Loan Management Operational** menu.
2. Click **Collateral Register** menu item to open the **Collateral Register List** page.

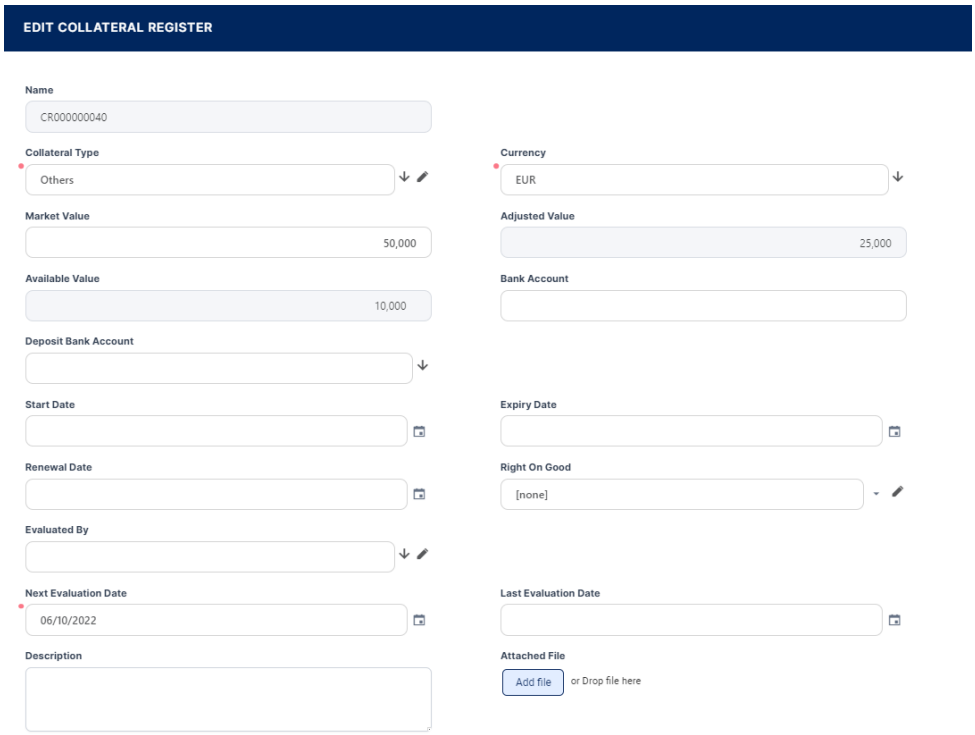
   On the **Collateral Register List** page, you can add new collateral records or search, edit, and delete existing ones.



> **NOTE**
> Collateral register records with an Owned status cannot be deleted. See Collateral System Statuses for more details.

3. Click the **Insert** button on the **Collateral Register List** page to add a new collateral register.

4. In the newly displayed **Add Collateral Register** page, fill in the following fields:



- **Name** - Automatically completed by Loan Management after saving the record, it displays the id of the collateral.
- **Customer** - Select the customer who owns the collateral. A collateral may have many owners. If this is your case, enter the rest of the owners in the **Collateral Owners** section as guarantors.
  Only displayed when registering a new collateral.

- **Collateral Type** - Select the type of the collateral from the drop-down list.
- **Currency** - Select the currency of the collateral. It can be different from the currency of the contract which uses this collateral as a guarantee.
- **Next Evaluation Date** - For Real Estate collaterals, you can insert the next evaluation date. Only displayed when editing an existing collateral register.

5. Optionally, fill in the following fields:
   - **Market Value** - Enter the market value of the collateral which is taken into consideration. Only applicable for Real Estate and Others collateral types.
   - **Adjusted Value** - The percent that should be covered by the collateral is set in at the banking product level, in the `Collateral Cover Percent` field. Only applicable for Real Estate and Others collateral types, it is automatically calculated by Loan Management.
   - **Available Value** - If the collateral is used to cover other loans, Loan Management automatically calculates the remaining value and displays it in this read-only field.
   - **Bank Account** - Enter the customer's current account so that the funds within the account can serve as a guarantee.
   - **Deposit Bank Account** - Select the deposit bank account of the customer from the list of accounts with Open status and type different than Loan Term Account. Only applicable for Cash collateral types.
   - **Start Date** - Select the start date for the collateral registration.
   - **Expiry Date** - Select the end date for the collateral registration.
   - **Renewal Date** - Select the date when the collateral registration is renewed.
   - **Evaluated By** - Select the customer who evaluated the collateral.

- **Right On Good** - Select the type of rights held on the collateral goods. Possible values: `Full Property`, `Naked Property` and `Usufruct`.
- **Last Evaluation Date** - For Real Estate collaterals, you can insert the previous evaluation date. Only displayed when editing an existing collateral register.
- **Description** - Enter a suggestive description for the collateral.
- **Attached File** - Attach files relevant for the collateral.

6. Click the **Save and Reload** button.

After saving the record, fill in the new sections displayed in the page, with specific information:

Add Collateral Register Rank

In the **Collateral Register Rank** section, you can insert, delete or export collateral register ranks.

To add a rank:

1. Click **Insert** and fill in the following fields:
   - **Parity On Rank** - Select the checkbox to mark the collateral with parity on rank.
   - **Rank** - Select the rank of the collateral.
   - **Owner** - Select the customer who owns the collateral.
2. Click the **Save and Close** button.

View Contract Collaterals

The list within the **Contract Collateral** section is automatically generated, displaying the contracts where the current collateral is used as guarantee, if such contracts exist. The following information is displayed:

- **Contract** - The id of the contract where the collateral is attached.
- **Status** - The status of the contract.
- **Collateral Register Value** - The value of the registered collateral.
- **Collateral Register Value Usage (%)** - The percent from the collateral used for coverage within the contract.
- **Value in Contract Currency** - The value of the collateral expressed in the currency of the contract.

To edit a collateral attached to a contract from this list:

1. Double-click the record and perform the desired updates in the **Edit Contract Collateral** page.
2. Click the **Save and Close** button.

Manage Collateral Owners

A collateral may have multiple owners. The customer whom you previously entered before saving the collateral register record becomes the main owner. To add other registered customers who partially own the collateral, use the **Collateral Owners** section of the **Edit Collateral Register** page. The other owners of the collateral are considered guarantors of the contract, and they should consent on this. They are stored in the `Collateral Register Owner` entity.

To add an owner:

1. Click **Insert** and fill in the following fields:
    - **Collateral** - Automatically filled in by Loan Management with the id of the collateral register record.

- **Customer** - Select the customer who partially owns the collateral and becomes a guarantor for contracts where the collateral is used.

2. Click the **Save and Close** button.

Manage Collateral Register Participants

In the **Collateral Register Participants** section, you can insert, delete or export customers who participate to the collateral in a specific role such as notary, valuer, etc.

To add a participant:

1. Click **Insert** and fill in the following fields:
   - **Participant** - Select the customer who is a participant to the collateral.
   - **Participant Role** - Select the role of the customer in this collateral.
2. Click the **Save and Close** button.

Manage Collateral Register Documents

In the **Collateral Register Documents** section, you can insert, delete or export collateral documents.

To add a document:

1. Click **Insert** and fill in the following fields:
   - **Document Type** - Select the type of the document that is uploaded for the collateral.
   - **Collateral File** - Attach the file to be uploaded.
2. Click the **Save and Close** button.

Contracts

Any agreement between a bank or a financial institution and a customer regarding the usage of a banking product is documented legally with a contract.

FintechOS Loan Management allows financial institutions to create banking product agreements (contracts) for their customers based on approvals. This is how Loan Management indicates the selling of a banking product, recording a contract to reflect the product and negotiated details within the origination process.

Contract Implementation Notes

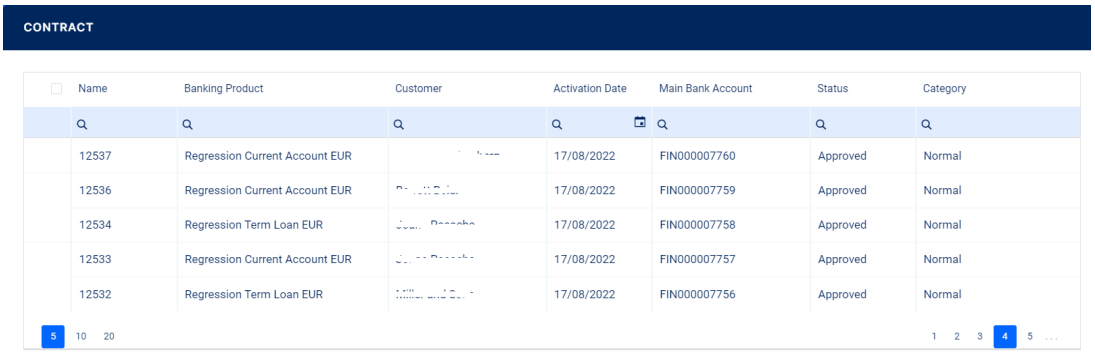
- The contract approval is made according to the specifications of the financial institution set during the implementation process.
- The integration is done according to the financial institution's requirements.
- The contract should be approved only after the advance is paid. This should be applicable for loan contracts with a stipulated advance amount or percent to be paid on the first disbursement day (for merchant loans), in order to avoid having to close newly created contracts if the process of collecting the advance and potential fees fails.

Managing Contracts

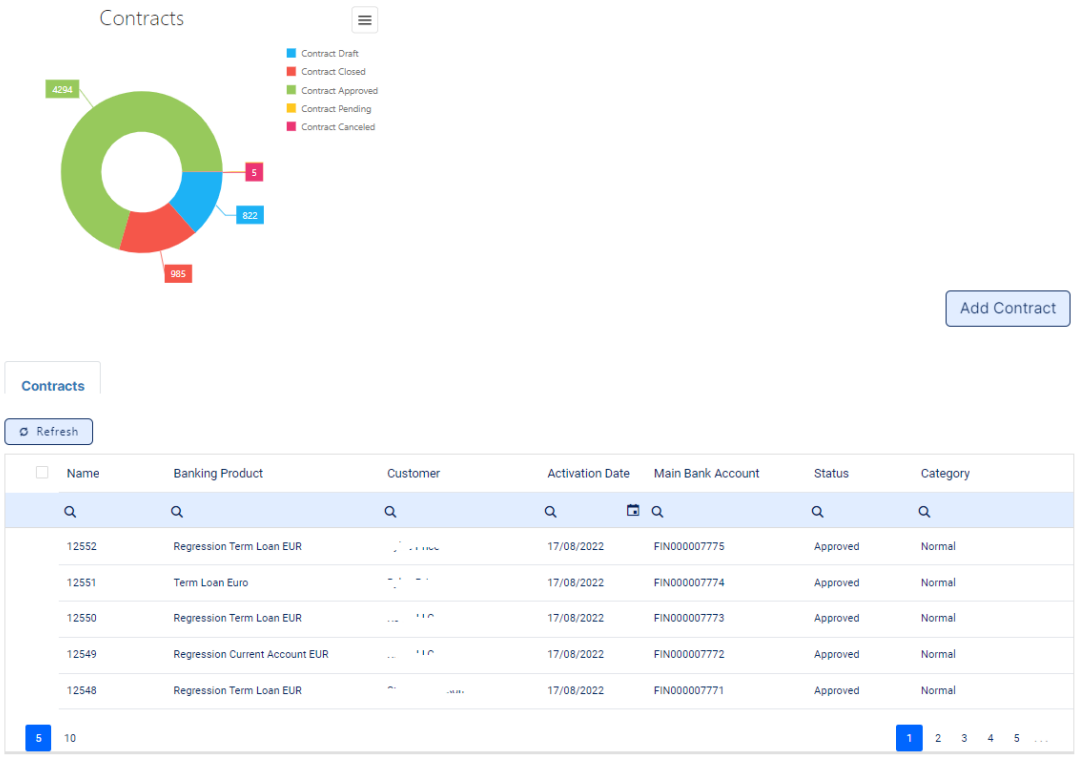
> **NOTE**
> Loan Management enables you to manage contracts via the user interface or via integration through APIs. For information about the available endpoints, please visit the Loan Management Developer Guide.
> For information about managing contracts via the user interface, continue reading this page.

To manage contracts:

1. In FintechOS Portal, click the main menu icon and expand the **Loan Management Operational** menu.
2. Click **Contract** menu item to open the **Contracts** page.

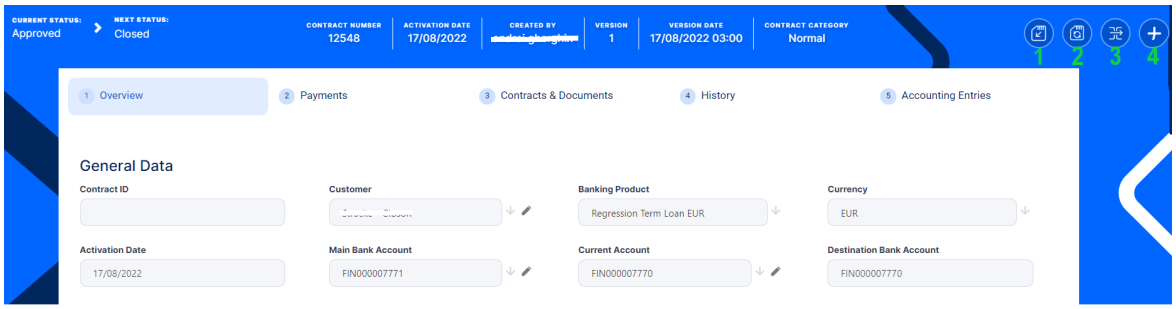


Alternatively, you can select the **Contracts dashboard**.



On the **Contracts** page, you can add new contract records or search, edit, and delete existing ones in Draft status.

Loan Management has a series of buttons that help shorten the processes of managing contracts. Such actions aid you quickly streamline several processes. In the selected contract's page, in the top right corner, a series of buttons (depicted below in green) trigger different actions.



- Button 1 is the **Save and close** button.
- Button 2 is the **Save and reload** button.
- Button 3 opens the **Contract Business Workflow Transitions List**.
- Button 4 creates a **new version of the existing contract**. The new version has to be approved before the customer starts using it. If it is NOT approved, then the initial version can still be used.

Loans

A loan is a banking product which defines a loan for a specific amount, that has a specified repayment schedule and either a fixed or floating interest rate. Loan Management enables you to create contracts based on term loan products and to manage such contracts. Read about the operations that you can perform for the various types of loans: revolving loans, unsecured personal loans, secured loans, credit facility, SME loans.

Loan Contract Life Cycle and States 213
Creating A New Unsecured Loan 216
Creating A New Secured Loan 236

Approving a Loan 240
Rejecting a Loan 246
Disbursing a Loan 248
Processing Loan Repayments 253
Working with Overdue Loans 283
Applying Payment Holiday to a Loan 286
Working with Grace 290
Working with Participants 292
Working with Tranches 294
Working with Covenants 296
Working with Contract Classification 300
Working with Returns 301
Applying Fees and Commissions 310
Closing a Loan With All Obligations Met 313
Rescheduling and Refinancing Loans 316
Writing Off a Loan 326
Changing the Interest Rate 331
Editing and Customizing Repayment Schedules 335
Manually Capture Notifications 357
Working with Documents 362
Treatment of Non Working Days for Schedule 365
Working with Limits 366
Creating New Versions of Existing Loan Contracts 369
Viewing a Contract's History 371
Viewing a Contract's Accounting Entries 372
Changing the Borrower of a Loan 375

Loan Contract Life Cycle and States

The four-eyes principle is applicable for all contracts in FintechOSLoan Management, meaning that a record should be approved by a second financial institution employee, with higher authorization rights. This is enabled via approval task FintechOS Platform capabilities and thus it is also a financial institution's responsibility to set proper security roles and access rights to its users, in order to make sure that the same user can't insert and also authorize the same record.

A contract record has the following business workflow statuses:

- **Draft** - the status of a newly created contract record that was not yet sent for approval. While in this status, you can edit some fields, but you can't add events (payments) to it. Send the record to approval after editing all the necessary details.
- **Pending** - this is a system status applied to contracts sent for approval, but not yet approved. No updates are available in this system status.
- **Approved** - the status of a contract record after being authorized by a user with contract approval competencies. While in this status, you cannot edit the record's details, but you can add events to it within the **Payments** tab. If you need to alter the contract's details, create a new version based on the current contract.

  > **NOTE**
  >
  > Each event must also be approved by a user with contract approval competencies, otherwise, the transaction is not performed by the system.
  >
  > New contract approval is blocked by Loan Management if the customer has overdue days >= the value of the `DelayDaysForBlockNewContractApproval` parameter.

- **Closed** - the last status of a contract, after manually closing it or after creating a new version based on the current version. No updates are allowed on the record.

- **Canceled** - the status of a contract after manually canceling it straight from the **Draft** status. No updates are allowed on the record.

> **NOTE**
> Change the contract's status to **Approved** so that the customer can use the contract and in order to apply transactions to it.

Contract Versioning

Loan Management allows you to create new versions for an existing contract if you need to modify an existing approved contract. New versions are automatically created when the payment schedule is modified - that implies any increase/ decrease, change of costs, reschedule or payment holiday transactions.

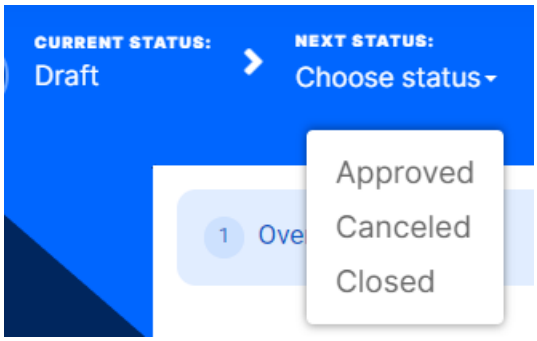
A contract version can have the following statuses:

- **Contract Version Draft** - the status of a newly created contract version record that was not yet sent for approval. While in this status, you can edit some fields, but you can't add events (payments) to it. Send the record to approval after editing all the necessary details.
- **Approved** - the status of a contract version record after being authorized by a user with contract approval competencies. While in this status, you cannot edit the record's details, but you can add events to it within the **Payments** tab.
- **Contract Version Closed** - the last status of a contract version, after manually closing it or after creating another new version based on the current version. No updates are allowed on the record.

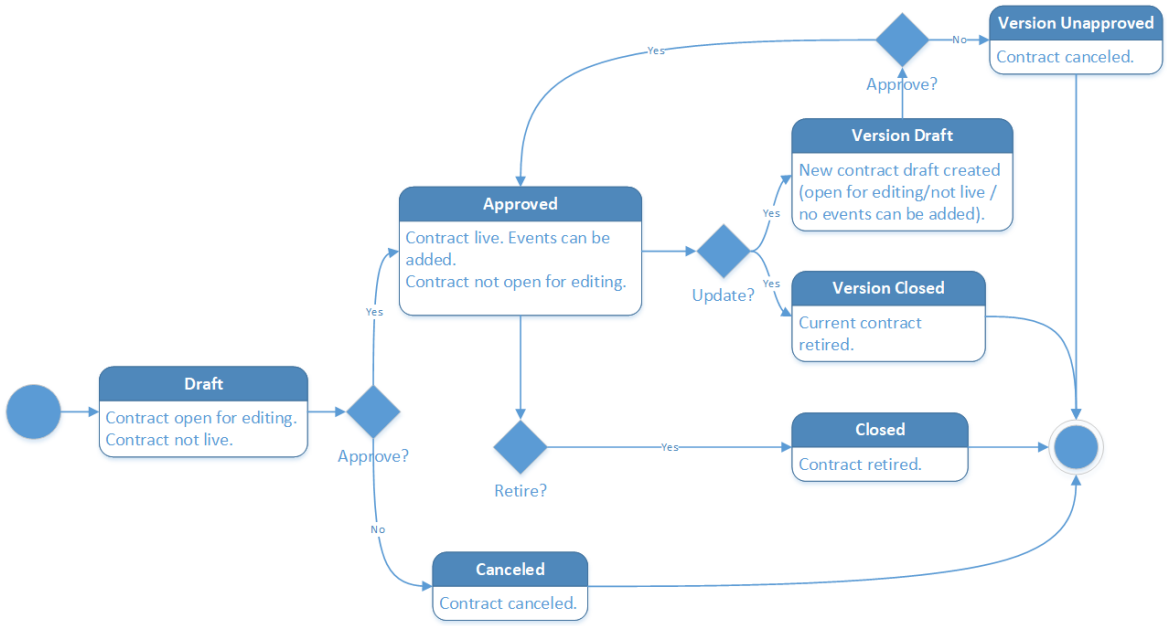
> **NOTE**
> All existing versions of the contract in **Contract Version Draft** status are automatically changed to **Contract Version Closed** when a payment event outside regular schedule is approved for that contract.

Changing Contract Statuses

You can manage a contract's life cycle by changing its status from the top right corner of the screen.



The contract status transitions are illustrated below:



Note that:

- Once a record is live, its settings can no longer be modified.
- If you want to update the details of a live contract, you must create a new contract version.
- When you create a new contract version, the current version is retired and moved to history; no updates are allowed on the retired version.
- Every contract version starts in a draft state and must go through an approval process

before going live.

- Only one version of a contract can be live at one time.

> **NOTE**
> As a best practice, new records or new versions of existing records created on a specific day should be approved on the same day.

Creating A New Unsecured Loan

An unsecured loan is a loan that doesn't require any type of collateral. Instead of relying on a borrower's assets as security, financial institutions approve unsecured loans based on a borrower’s creditworthiness.

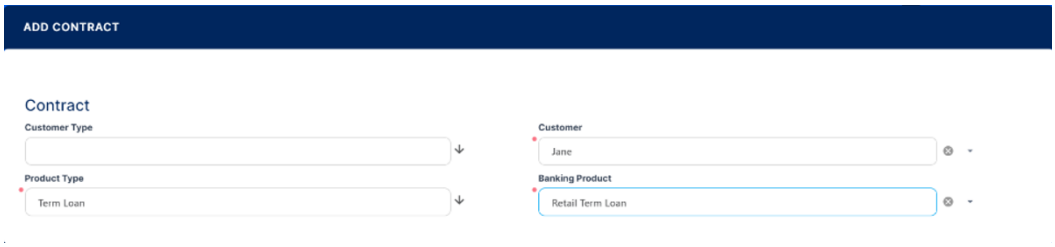
Before creating a unsecured term loan contract, make sure that:

- the customer is recorded in Loan Management,
- a settlement account (a current account contract for the same customer) is set up for the desired currency,
- and the limits are configured according to Loan Management's setup.

To create a new contract:

1. Add Minimum Contract Data

1. Open the **Contracts** page as described in the Managing Contracts section.
2. Click the **Insert** button to display the **Add Contract** page, the initial page when you insert any type of contract.



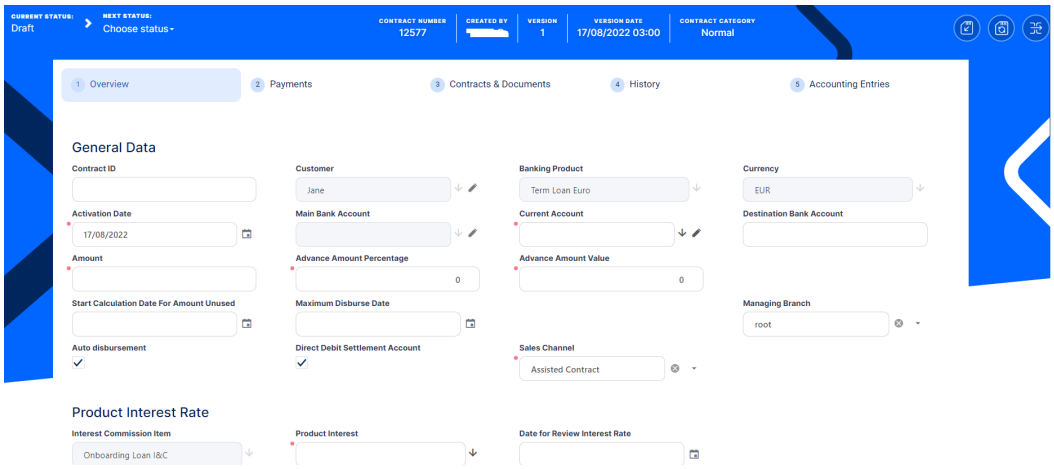
3. Fill in the following fields:

- **Customer Type** - Optionally, select the type of the customer for the contract, to filter the displayed customers in the next field.
- **Customer** - Select from the list the customer for whom you are creating a contract.
- **Product Type** - Select from the list the product type to filter the list of banking products accordingly.
- **Banking Product** - Select from the list the desired banking product.

> **NOTE**
> Be careful when choosing the values for the previously mentioned fields because you can't modify them after saving the contract!
> Make sure that you select `Term` Loan in the Product Type field and an `Unsecured` Loan banking product in the Banking Product field.

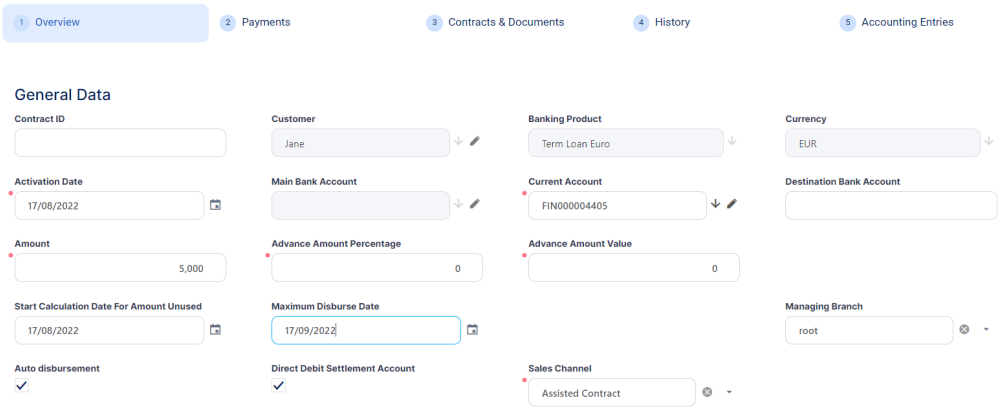
4. Click the **Save and Reload** button.

Loan Management saves the contract in **Draft** status, with minimum default information, such as an auto-generated contract number, created by, version and version number. The previously provided details are kept on screen in the **General Data** section, but they are no longer available for update. The **Currency** has been updated from the banking product level.



Proceed to the next steps where the details about the contract are captured and validated against the underlying product, setting the basic elements for the creation of a contract such as customer, banking product, account, interest rate, participants, tranches, fees, and contract covenants, within the newly displayed **Overview** tab.

2. Add General Data to the Contract



1. Fill in or modify the following information:
    - **Current Account** - Select the current account to be used for settlement.

- **Destination Bank Account** - Enter the destination bank account number, an account where the disbursements should be performed, if it's different from the current account selected previously. If the destination bank account belongs to a different customer than the one owning the contract, then any disbursement event happens in that customer's account, and the operation transaction record at the Subledger level mentions the name of that customer in the **Destination Partner** field. For example, if you input the bank account of a merchant, then the disbursement happens in the merchant's account and the operation transaction record mentions the merchant's name in the **Destination Partner** field.
- **Amount** - Enter the actual amount of credit for the contract.
- **Advance Amount Percentage** & **Advance Amount Value** - Use these in case the product requires a first installment to be claimed on granting the loan itself, or as approval condition - used for orchestrating BNPL where the customer needs to provide a certain amount before benefiting from the loan; requires orchestration of the payment and the approval of such contract should happen only on instruction that payment for advance has been supplied. The 2 fields change one based on the value inserted in the other, so if you insert a percentage the amount is updated based on loan amount.

> **NOTE** Limit validations for contracts with advance amount >0 are performed for Amount - Advance Amount.
>
> When the contract is activated, the available limit amount is decreased with the (Amount - Advance Amount) value.
>
> `Auto Disbursement = True` and cannot be changed for contracts with advance amount >0.
>
> If the advance amount is changed back to zero, then the

value of the `Auto Disbursement` checkbox becomes the default value set at the banking product level and can be edited.

- **Sales Channel** - Select the channel through which the contract is created.

2. Optionally, fill in or modify the following information:
   - **Contract ID** - Enter a contract ID other than the contract number generated automatically by Loan Management when you saved the contract.
   - **Activation Date** - Modify the date when the contract is activated. It is automatically completed with the system date.
   - **Start Calculation Date For Amount Unused** - Enter the date when commitment fee starts being applied. There are instances when, because the loan is granted, the financial institution needs to reserve those funds and make sure they are available when the customer asks for a disbursement, provided all other conditions are met. For such cases, when the financial institution does not generate income from interest, they might want to have a minimum income and thus the commitment fee (commissions with `Commission Unsuage` type). They can also allow for an interval for the amounts to be used and start applying such commissions post that interval.
   - **Maximum Disburse Date** - Select the maximum date by which the loan should be used for the approved contract. It can be that it is required because of internal policies or legislation – such in case of an investment or a mortgage, if you do not use the funds for 6 months, there might be a need for a reassessment. If not selected, this date is calculated based on the `Maximum Period Disburse After Activation (Months)` from the banking product level, and `Maximum Disburse Date = Maturity Date - 1`.

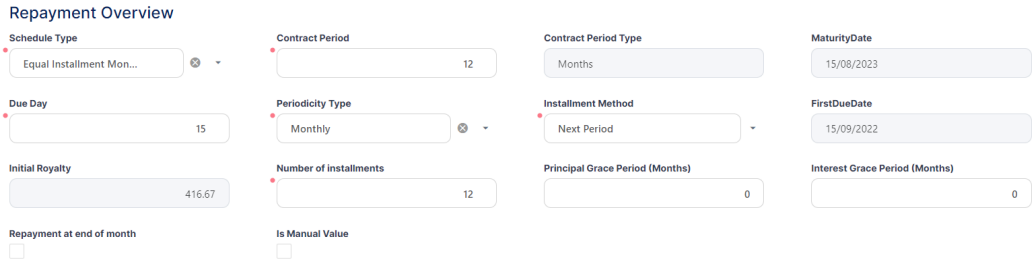
- **Managing Branch** - This represents the branch of the organization where the contract was created. Suppose you work in a branch or credit center, and you need cases to be linked to a specific location so that you can properly allocate them for further actions. It is automatically completed at contract saving time, but you can select another branch from the list.
- **Auto disbursement** - Select this checkbox if the financed amount must be automatically disbursed on the approval of the contract. If selected, Loan Management performs the disbursement transaction immediately after contract approval, and the funds are moved to the settlement account or destination account as per instructions. The auto disbursement property is set at banking product level, but it can be modified at the contract level. The following validations are performed for this checkbox:
    - If the contract has multiple tranches, then `Auto disbursement = False` and it cannot be edited.
    - If `Auto disbursement = True` and the contract approval date = activation date, then Loan Management does not generate a new version for the contract.
    - If `Auto disbursement = True` and the contract approval date > activation date, then Loan Management generates a new version for the contract.
- **Direct Debit Settlement Account** - Select this checkbox if the automated settlement of repayment notifications (the direct debit settlement account) functionality is turned on at the contract level. The value of the checkbox was set at the banking product level, but it can be modified at the contract level. The checkbox can be edited in all the statuses of a contract except `Version Closed`, `Closed`, and `Canceled`.

> **NOTE** The `Direct Debit Settlement Account` setting at the customer level takes precedence over the setting at the contract level when creating new contracts. For existing contracts, Loan Management applies the setting configured within the `CustomerToContractDirectDebitSettlementAcc` system parameter.

3. Click the **Save and Reload** button.

3. Enter Repayment Information for the Contract

In the **Repayment Overview** section you should enter term, schedule type and first due date so that Loan Management can properly build the repayment schedule. Optionally you can set a grace period, mention if the due date should always be on the last day of the month and if there is any manual value for the installment.



1. Fill in or modify the following information specific to the contract's repayment schedule:
    - **Schedule Type** - Select the payment schedule type to be used to calculate the installments of this contract. You can select one of the payment schedule types associated to the underlying banking product in the **Details** tab > **Associated Payment Schedule Types**

list. Loan Management uses the schedule type to build the repayment plan with equal instalments or liniar payments, include fees on the schedule and arrive to the day basis to be used for interest calculation (30/360).

- **Contract Period** - Edit the term of the loan that was automatically completed with the number of contract period of contract period type as it was defined at banking product level, according to your needs. The contract period is used together with Contract Period Type and Periodicity Type. They all need to be in sync and also in sync with the schedule definition itself, and if there are multiple definitions allowed on the product, make sure to pick those working together.
- **Contract Period Type** - This field is automatically completed with the contract period type as it was defined at banking product level. You can't edit this value.
- **Maturity Date** - This field is automatically completed with the contract maturity date, calculated based on the values of the Contract Period, Contract Period Type, Due Date and Activation Date.
- **Due Day** - Enter the exact day of month for installment repayment. If it is set to 31, then the system takes the last day of month. If you manually select the `First Due Date`, then this field is automatically completed and not editable. If the periodicity and the repayments are set to every 30 days, Loan Management defaults the due date based on the activation date.
- **Periodicity Type** - Select the time interval for the repayment schedule. Possible values are set at the banking product level. If the periodicity is set to `Once`, then the payment happens one time, at loan maturity. You can only select from periodicity types with the same measurement unit as the selected schedule type's contract period type. For example, if the value in the `Contract Period Type` is `Days`, you can only select a periodicity type whose measurement unit is in days.

- **Installment Method** - Select the installment method to calculate if the first due date is set into the current month or in the next month. Possible values:
    - `Actual Period`, with the first installment's due date calculated within the same calendar month;
    - `Next Period`, with the first installment's due date calculated within the next calendar month after contract approval.
- **First Due Date** - Select the date of the first due installment. If you manually select the `Due Day`, then this field is automatically completed as calculated based on the information within the Due Date, Periodicity Type and Installment Method, and it is not editable.
- **Initial Royalty** - This field is automatically completed with the value of an installment. The field is displayed and can be filled in if the selected schedule type is of type `Equal Installment`. You can edit this value. If at the selected payment schedule type's level the `Installment Value Custom` field is `False`, then the `Initial Royalty` field at the contract level is read only.
- **Initial Principal Value** - This field is automatically completed with the value of the principal within an installment. The field is displayed and can be filled in if the selected schedule type is of type `Equal Principal`. You can edit this value. If at the selected payment schedule type's level the `Installment Value Custom` field is `False`, then the `Initial Principal Value` field at the contract level is read only.
- **Number of installments** - This field is automatically completed with the number of installments to be paid for this contract, calculated based on previously defined values.
- **Principal Grace Period (Months)** - This field is displayed only if the banking product allows a principal grace period. Enter a value in months for the grace period allowed for principal repayment for this contract. The value inserted in this field should be

between the minimum and maximum grace period set at the banking product level.

- **Interest Grace Period (Months**) - This field is displayed only if the banking product allows an interest grace period. Enter a value in months for the grace period allowed for interest repayment for this contract. The value inserted in this field should be between minimum and maximum grace period set at the banking product level.

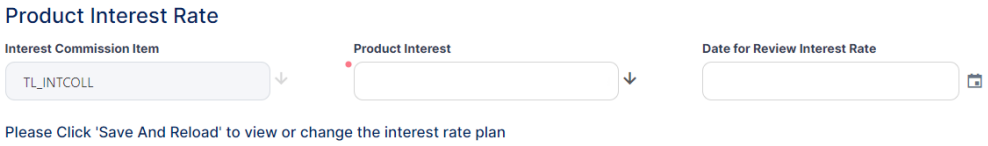
  > **NOTE** You can change the grace period only until the first installment is notified.

- **Repayment at end of month** - If you select this checkbox, then the due day of the contract is automatically set to the last day of the month, and the repayment schedule is calculated with an installment in the last day of month.
- **Is Manual Value** - If you select this checkbox, then you can manually enter the value for royalty or principal, thus overriding the values automatically calculated by Loan Management.
  - If `Is Manual Value = False`, then the `Initial Royalty` and the `Initial Principal Value` fields are read-onlyand cannot be modified.
  - If at the selected payment schedule type's level the `Installment Value Custom` field is `False`, then the `Is Manual Value` field at the contract level is read only.
  - If `Installment Value Custom = True`, then the `Is Manual Value` field at the contract level is editable, with `False` default value.

2. Click the **Save and Reload** button.

4. Manage Product Interest Rate for the Contract

Enter the details about the Product Interest Rate applied to the loan. Depending on the product definition again, you have a list of interest definitions that you can bring along to the contract.



To manage the product interest rate as it must be applied to this contract:

1. Fill in or modify the following fields:

- **Interest Commission Item** - This field is automatically completed with the interest & commission item defined at the product level, if only one item is found at the product level. If the selected product has more items, you must select one from the list.
- **Product Interest** - Select from the interest to be applied for this contract. Only the interests associated to the selected banking product are displayed within the list. Penalty interests cannot be selected here.
- **Date for Review Interest Rate** - Enter the date for reviewing the interest rate applicable for the remaining amount. This date must be between `Activation Date` and `Maturity Date`, otherwise, an error is displayed.
  For variable interest, this field is automatically completed with the `Reference Rate Date` + `Reference Interest Period` of the underlying interest definition, from the base type interest attached to variable interest. You can edit this field. For months where the date is over lapsed, the last day of the month is used for the calculation (for example, if you specify 30, then in February the system takes the last day, which can be the 28th or the 29th).

2. If the underlying interest definition has referenced a variable interest rate, the details included other fields for you to complete:

- **Margin** – The margin applicable on top of the variable interest rate.
- **Reference Rate Date** – The date to be considered in order to arrive to the applicable rate for the underlying variable interest (EURIBOR as of 30th June 2022).
- **Reference Rate** – The underlying rate for the variable interest as captured in Loan Management for the date above.

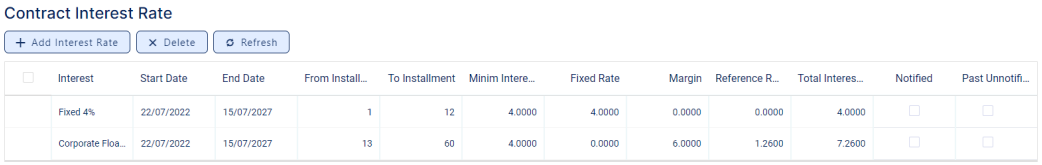
3. Click the **Save and Reload** button.

> **NOTE**
> Fill in any other mandatory fields from the **General Data** and **Repayment Overview** sections, otherwise you can't successfully save the contract.

5. Manage Contract Level Interest & Penalty Interest Rate

Define the information about the contract interest rate (or rates, if you selected a `Collection` type interest rate in the previous **Product Interest Rate** section) in a table format, in the section **Contract Interest Rate** section, which appears only after saving the selected product interest rates. The **Contract Penalty Interest Rate** section, similar to the one above, helps you manage penalty interest rates applicable to the contract when there are overdue amounts according to the definition of penalty interest and included operation item elements. If the penalty interest has multiple interest values, at contract generation Loan Management brings only the penalty interest value for which the current date falls in the interest value interval.

You can edit the tables cells, so you can customize the interest rates selected at the product level, if the interest and commission list was defined as `negotiable`, to obtain the desired interest rates configuration at the contract level. You can also add or delete interest rates, using the **Add Interest Rate**, **Add Penalty Interest Rate**, respectively the **Delete** buttons above the tables. Thus, the tables enables you to work with multiple interest rates at the contract level.



> **NOTE**
> The information disappears if you change the product interest, tenor, first due date, maturity date, contract period, or activation date. In this case, save the contract again to display the updated information.

To customize the information specific to each of the contract's **interest rates**:

1. In the **Contract Interest Rate** section, edit the existing information that was automatically completed based on your product interest rate selections:

- **Interest** - Automatically completed with the interest (or interests, for `Collection` type product interest rate) selected in the previous **Product Interest Rate** section. You can select from the drop-down list the interest to be applied for this contract. Only the interests associated to the selected banking product are displayed within the list. Penalty and overdraft interests cannot be selected here. Depending on the selected interest, other fields can be displayed to be filled in.
- **Start Date** - The interest's start date, automatically completed with the contract's activation date.
- **End Date** - The interest's end date, automatically completed with the contract's maturity date.
- **From Installment** - The first installment for which this interest is applied to the contract.
- **To Installment** - The last installment for which this interest is applied to the contract.
- **Minimum Interest Rate** - This read-only cell is automatically completed with the minimum interest rate applicable for the contract, defined at the banking product level.

- **Fixed Rate** - The fixed rate of the interest. You can only change it if the interest at the banking product level was marked as `Is Negociable`.
- **Margin** - This cell is automatically completed with the margin of the previously selected product interest. You can only change it if the interest at the banking product level was marked as `Is Negociable`. If the product interest was not selected, you can manually enter the margin.
- **Reference Rate** - This read-only cell is automatically completed with the interest type's definition's reference rate valid at the previously selected date.
- **Total Interest Rate** - This read-only cell is automatically completed with the calculated total interest rate of the previously selected product interest and any values entered for margin and reference rate. If the product interest was not selected or if the interest at the banking product level was marked as `Is Negociable`, you can manually enter the interest rate.
- **Notified** - This is a read-only checkbox. For contracts in `Version Draft` status, it shows you whether the installments range shown on this table line was already notified or not.
- **Past Unnotified** - This is read-only cell read-only checkbox. For contracts in `Version Draft` status, it shows whether there are days that already passed from the current month's not yet notified installment, days for which you can't change the interest rate.

2. After performing the desired changes, make sure that the interest rate (s) cover the entire tenor of the contract, from activation date until maturity date, and there are no overlapping intervals, otherwise an error prevents you from approving the contract.
3. Click the **Save and Reload** button.

Later, after contract approval, the contract repayment schedule is calculated taking into consideration the contract interest rates as defined in this section. For example, for multiple interest rates, the installment amounts differ depending on the interest rate applicable for those installment numbers. The picture below shows different values calculated for the

repayment schedule of a loan with multiple interest rates, where an interest rate was applied for the first 12 installments, and another interest rate was applied for the rest of the installments.

Contract Repayment Schedule Details

No.	Due Date	Remaining Value	Interest	Principal	Commission	Total Installment	IRR	Notif.No.
1	15-08-2022	200,000.00	600.00	8,082.10	10.00	8,692.10	0.00	61142
2	15-09-2022	191,917.90	639.73	8,042.37	10.00	8,692.10	0.00	61735
3	15-10-2022	183,875.53	612.92	8,069.18	10.00	8,692.10	0.00	61923
4	15-11-2022	175,806.35	586.02	8,096.08	10.00	8,692.10	0.00	62929
5	15-12-2022	167,710.27	559.03	8,123.07	10.00	8,692.10	0.00	
6	15-01-2023	159,587.20	531.96	8,150.14	10.00	8,692.10	0.00	
7	15-02-2023	151,437.06	504.79	8,177.31	10.00	8,692.10	0.00	
8	15-03-2023	143,259.75	477.53	8,204.57	10.00	8,692.10	0.00	
9	15-04-2023	135,055.18	450.18	8,231.92	10.00	8,692.10	0.00	
10	15-05-2023	126,823.26	422.74	8,259.36	10.00	8,692.10	0.00	
11	15-06-2023	118,563.90	395.21	8,286.89	10.00	8,692.10	0.00	
12	15-07-2023	110,277.01	367.59	8,314.51	10.00	8,692.10	0.00	
13	15-08-2023	101,962.50	616.87	8,212.53	10.00	8,839.40	0.00	
14	15-09-2023	93,749.97	567.19	8,262.21	10.00	8,839.40	0.00	
15	15-10-2023	85,487.76	517.20	8,312.20	10.00	8,839.40	0.00	
16	15-11-2023	77,175.56	466.91	8,362.49	10.00	8,839.40	0.00	
17	15-12-2023	68,813.07	416.32	8,413.08	10.00	8,839.40	0.00	
18	15-01-2024	60,399.99	365.42	8,463.98	10.00	8,839.40	0.00	

The following picture exemplifies another example, for different interest rates applied during the same installment, where the interest is accrued daily:

Contract Interest Rate

Interest	Start Date	End Date	From Installment	To Installment	Minim Interest ...	Fixed Rate	Margin	Reference Rate	Total Interest R...	Notified	Past Unnot
Fix Eur 5%	2023/03/07	2023/07/06	1	5	3.0000	3.0000	0.0000	0.0000	3.0000		
Fixed 4%	2023/07/07	2023/07/20	5	5	3.0000	4.0000	0.0000	0.0000	4.0000		
Fix Eur 5%	2023/07/21	2025/03/25	5	10	3.0000	3.0000	0.0000	0.0000	3.0000		
Corporate Fi...	2023/03/07	2025/03/25	11	25	3.0000	0.0000	6.0000	1.2600	7.2600		

Accruals And Provisions

Delete Export Refresh

Classification	Contract	Calculation Date	Daily Accrual In...	Accumulated In...	Daily Accrual U...	Accumulated U...	Daily Interest P...	Accumulated In...	Daily Fee Accrual	Accumulated F...	Principal Pro
Sub-standard	16350	2023/07/16	0.94637400	3.78549600					0.00000000	0.00000000	
Sub-standard	16350	2023/07/15	0.94637400	2.83912200					0.00000000	0.00000000	
Sub-standard	16350	2023/07/14	0.94637400	1.89274800					0.00000000	0.00000000	
Sub-standard	16350	2023/07/13	0.94637400	0.94637400					0.00000000	0.00000000	
Sub-standard	16350	2023/07/12	1.11304100	8.89000000					0.00000000	0.00000000	
Sub-standard	16350	2023/07/11	1.11304100	7.78000000					0.00000000	0.00000000	
Sub-standard	16350	2023/07/10	1.11304100	6.67000000					0.00000000	0.00000000	
Sub-standard	16350	2023/07/09	1.11304100	5.56000000					0.00000000	0.00000000	
Sub-standard	16350	2023/07/08	1.11304100	4.45000000					0.00000000	0.00000000	
Sub-standard	16350	2023/07/07	0.83478000	3.34000000					0.00000000	0.00000000	

NOTE
Loan Management takes into consideration and recalculates the schedule even in cases when the interest is of `Collection` type and there are multiple interest rates within a single repayment interval due to changes in the interest definition between due dates. In such cases, Loan Management keeps the interest accrued up until the change occurs and rebuilds the remaining of the repayment schedule.

NOTE
Depending on the contract interest rates, Loan Management calculates the **accrual** in the following ways:
- If there are multiple interest rates during an installment period, then, for each rate, a daily interest percent is calculated for the corresponding interest percent and the contract's remaining value
- If only one interest rate during an installment period, the daily interest is calculated using the `Next Due Date` schedule interest from the repayment schedule.
For the last day of the period, Loan Management performs an adjustment to meet the repayment schedule's `Next Due Date` interest.

To customize the information specific to each of the contract's **penalty interest rates**:

1. In the **Contract Penalty Interest Rate** section, edit the existing information that was automatically completed based on your product interest rate selections:


Contract Penalty Interest Rate

Name	Start Date	End Date	From Installment	To Installment	Fixed Rate	Margin	Total Interest Rate	Operation Item	Reference Rate
BNPL-6X Penalt...	2022/12/16	2023/11/11	1	11	24.0000	0.0000	24.0000	Loan Principal	0.0000

- **Interest Name** - Automatically completed with the penalty interests selected in the previous **Product Interest Rate** section. Only the penalty interests associated to the selected banking product are displayed within the list.
- **Start Date** and **End Date** - The penalty interest's start and end dates, automatically completed with the contract's activation, respectively maturity date.

- **From Installment** - The first installment for which this penalty interest is applied to the contract.
- **To Installment** - The last installment for which this penalty interest is applied to the contract.
- **Fixed Rate** - The fixed rate of the penalty interest. You can only change it if the interest at the banking product level was marked as `Is Negociable`.
- **Margin** - This cell is automatically completed with the margin of the previously selected product interest. You can only change it if the interest at the banking product level was defined as variable and marked as `Is Negociable`.
- **Total Interest Rate** - This read-only cell is automatically completed with the calculated total penalty interest rate as Fixed Rate or Margin + Reference Rate.
- **Operation Item** - The contract's operation item to which the penalty interest is applied.
- **Reference Rate** - This read-only cell is automatically completed with the interest type's definition's reference rate valid at the previously selected date.

2. After performing the desired changes, make sure that the penalty interest rate(s) cover the full contract period, from activation date until maturity date, and there are no overlapping intervals, otherwise an error prevents you from approving the contract.
3. Click the **Save and Reload** button.

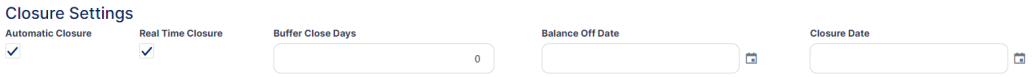
Later, after contract approval, you can change the penalty interest rate by versioning the contract and editing the information in this section. Thus, the new penalty rate becomes effective after approving the contract version, only impacting future calculations and notifications.

> **NOTE** Loan Management notifies the applicable penalty interest on a daily basis. The penalty calculation process uses the penalty interests values from the contract level.

6. Amend Closure Settings

If allowed from the product definition, you can amend the closure settings of the contract, the way Loan Management should behave once the loan is repaid and the contract can be closed. Most of the times this is not something that you have to access, but it adds extra flexibility at the contract level. This may prove useful if you suspect there may be reasons to keep a contract open for some time post recovering all amounts for instances when there may appear claims of funds (SEPA DD) or other similar cases.

The **Closure Settings** section is only displayed for contracts based on banking products having the `Closing Is Flexible = True` setting.



To amend the closure settings brought from product level here at the contract level:

1. Fill in or modify the following fields:
    - **Automatic Closure** – If selected, Loan Management automatically closes the contract once all other conditions are met. This field is automatically completed with the value defined at the banking product level, but you can modify it.
        - Select this checkbox to instruct Loan Management to close the contract automatically when the available amount becomes zero and there are no further amounts to be recovered, and after the number of days set as buffer before closure pass and `Closure Date = Current Date`.
        - Deselect it to instruct Loan Management to keep the contract open, regardless of the fulfillment of its maturity and balance criteria, waiting to be manually closed by changing its status to `Closed`.

        > **NOTE**
        >
        > Revolving loans are closed only after maturity. In

this case, the available loan amount is considered as balance.

You can perform contracts events as specified in the **Allowed Transactions** section of the banking product, plus manual closure while the contract is pending closure. Performing any other transactions displays an error message.

- **Real Time Closure** – If you select this checkbox, when the amounts become zero and the loan is not a revolving one, the contract is closed automatically. If `Real Time Closure = True`, then `Buffer Close Days = 0` and `Automatic Closure = True`. For more details about the real-time closure, see Close Contracts RealTime(CB) Job.
- **Buffer Close Days** - Enter the number of days used as buffer before automatically closing the contract. If `Buffer Close Days > 0`, then `Real Time Closure = False`. Loan Management waits the entered number of days after the contract's balances reach zero, and at the end of that day the contract is closed.
- **Balance Off Date** – This is a system maintained field and it is populated with the date on top of which Loan Management adds the Buffer Close Days to arrive to the Closure Date.
- **Closure Date** – This is a system maintained field and holds the date when the contract is closed. For automatic closure, the date is calculated by Loan Management as Balance Off Date + Buffer Close Days.

2. Click the **Save and Reload** button.

7. Check Other Details Pre-Filled Based on Product Definition

Once you defined the mandatory details, then saved and reloaded the contract, Loan Management updates some of the next sections on the page, based on product definitions:



Loan Management brings the **Contract Participants**, the Borrower being also Beneficiary of the funds and the Customer who is granted the loan. If needed, you can add other participants to the contract, like Guarantors, Co-Debtors, etc. There may be cases when some roles are mandatory for a product. Those are detailed in a separate section. If there is a mandatory role defined in the banking product definition, Loan Management displays an error on trying to approve the contract.

Contract Tranches is a section where you can implement progressive access to the funds. This is valuable in case of loans granted for investment projects where you can know upfront that there is a plan for the project and payments need to happen for each stage of the project, those stages being known from the start.

Another important section brought from the product definition is the **Fees & Commissions**. Depending on the system setup, you are allowed or not to amend fees and commissions in this section.

Contract Covenants section displays the covenants that applicants must abide by after getting the loan, configured at the product level. Such conventions are usually applicable for corporate clients that must meet certain requirements in order to continue to receive disbursements and not only: submit balance sheet every x months, have account turnover of at least x percent from average monthly turnover, provide other relevant documents from authorities. In this section, you can manage covenants for the contract. These covenants would need to be monitored procedurally; Loan Management doesn't have the logic in place to implement automated processes.

You can use the **Contract Classifications** section to capture various classifications that might be relevant for the financial institution for that loan at a moment in time. It is a placeholder for such details and there is no automated logic in place to update them. In implementation this can be used for other developments if required.

After defining the relevant details of the contract, proceed to **contract approval**.

Creating A New Secured Loan

Secured loans are business or personal loans that require some type of collateral as a condition of borrowing. A financial institution can request collateral for large loans for which the funds are used to purchase a specific asset or in cases where the customer's credit scores aren’t sufficient to qualify for an unsecured loan.

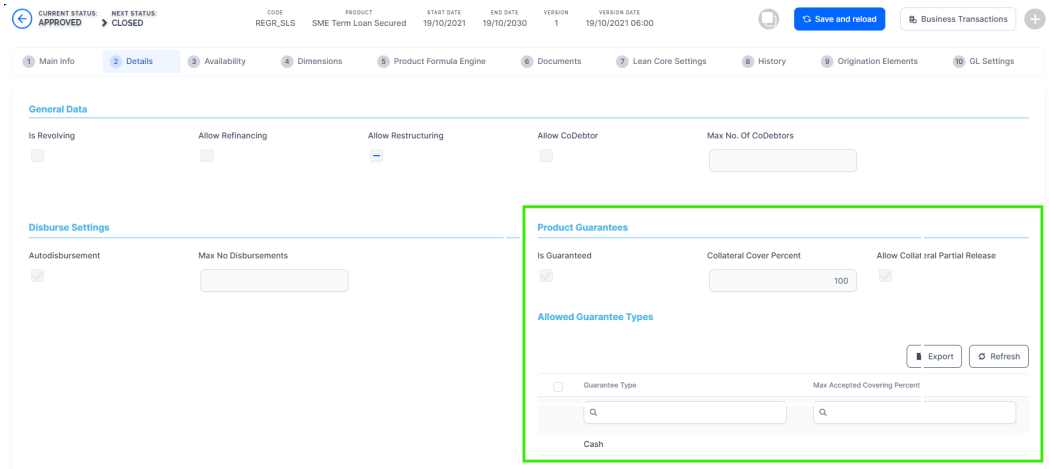
Before creating a secured term loan contract, make sure that:

- the customer is recorded in Loan Management,
- a settlement account (a current account contract for the same customer) is set up for the desired currency,
- the collateral is registered in Loan Management,
- and the limits are configured according to Loan Management's setup.

To create a new secured loan contract:

1. Follow the Same Steps as for Creating an Unsecured Loan Contract

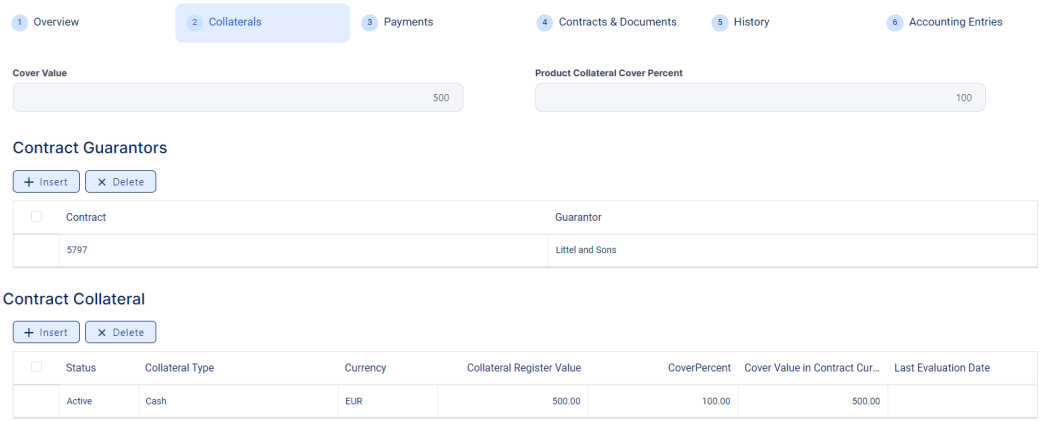
The process of creating an secured loan contract is similar to that of creating an unsecured loan contract. The difference is that you must choose a banking product that is **secured** when creating the contract.



After performing the steps described in the "Creating A New Unsecured Loan" on page 216 page, perform the actions required to link collaterals to the contract.

2. Link Collaterals to the Loan

You can set collaterals as guarantees for secured loan contracts in the **Collaterals** tab. This tab is displayed only for contracts based on **secured** products.

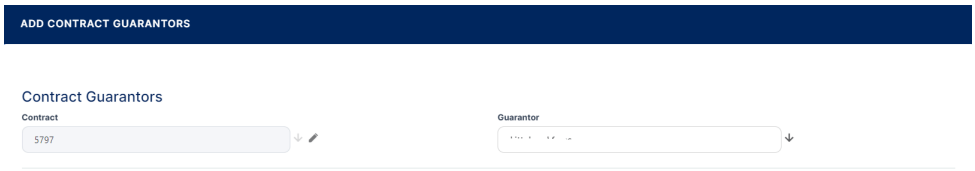


The fields **Cover Value** and **Product Collateral Cover Percent** are automatically populated from product level.

> **NOTE**
> You can send a contract for approval only if `Cover Value = Financed Amount * Product Collateral Cover Percent / 100`.

You can use collaterals not owned by the customer, but by a guarantor to cover the contract risk. To add guarantors to the contract, follow these steps:

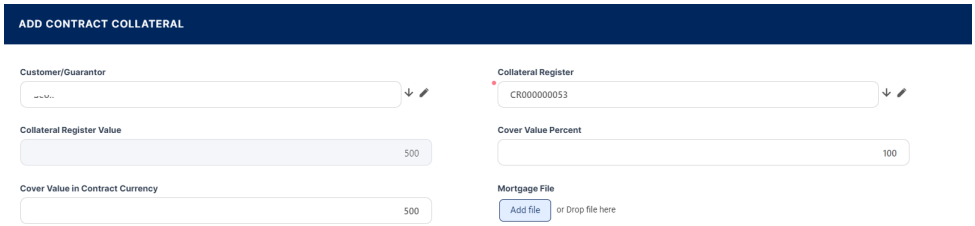
1. Click the **Insert** button within the **Contract Guarantor** section.
2. In the newly displayed **Add Contract Guarantors** page, fill in the **Guarantor** field by selecting the customer who acts as a guarantor for the contract.



3. Click the **Save and Close** button.

To link a collateral to the contract, follow these steps:

1. Click the **Insert** button from the **Contract Collateral** section.
2. In the newly displayed **Add Contract Collateral** page, fill in or modify the following fields:



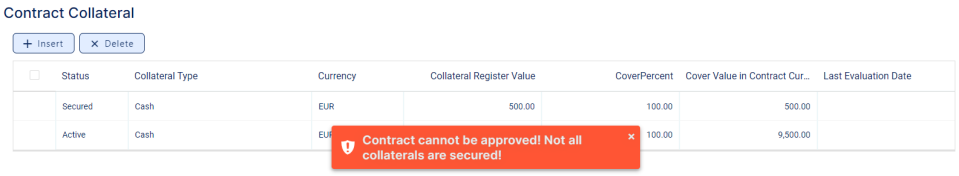
- **Customer / Guarantor** - Select the customer who acts as guarantor for the contract. The field is automatically completed with the customer selected in the contract, but you can select any of the customers already added as guarantors for this contract.
- **Collateral Register** - Select a collateral registered to the current customer. When selecting a collateral, the `Cover Register Value`, `Cover Register Value Usage` and `Cover Value in Contract Currency` fields are automatically calculated.

> **NOTE**
>
> Make sure that the collateral you are planning to use for the secured loan contract is previously registered, otherwise you can't use it for covering the contract.
> You can link a registered collateral to multiple contracts if the contracts total amount does not exceed the collateral available amount.

When selecting a **Fixed deposit** collateral, the status of the associated bank account becomes `Blocked`.
If the contract's status changes from `Owned` or `Active` into `Released`, the status of the bank account becomes `Opened`.

- **Collateral Register Value Usage (%)** - Edit the percentage to be used from the registered collateral's total value. It was set at banking product level, in the `Collateral Cover Percent` field. As a result, the `Collateral Register Value` and the `Cover Value in Contract Currency` values are automatically recalculated by Loan Management.
- **Cover Register Value** - The value is taken from collateral, expressed in the collateral's currency. If the collateral is in a different currency than the contract currency, the exchange rate is automatically applied in order to have the available amount correctly calculated. Further, a job runs daily in order to recalculate the available amount for each collateral.
- **Cover Value in Contract Currency** - This is the cover value converted in the contract's currency at the exchange rate defined in Loan Management.
- **Mortgage File** - Attach a file relevant for the mortgage.

3. Click the **Save and close** button.
4. Change the status of the collateral to **Secured** before approving the contract, otherwise Loan Management triggers an error.



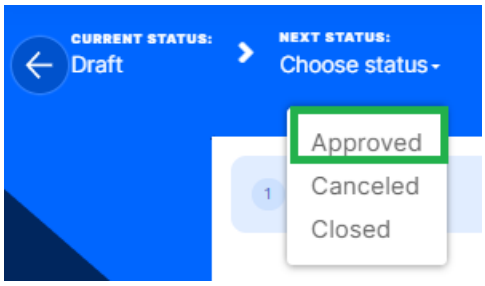
After defining the relevant details of the contract, proceed to **contract approval**.

Approving a Loan

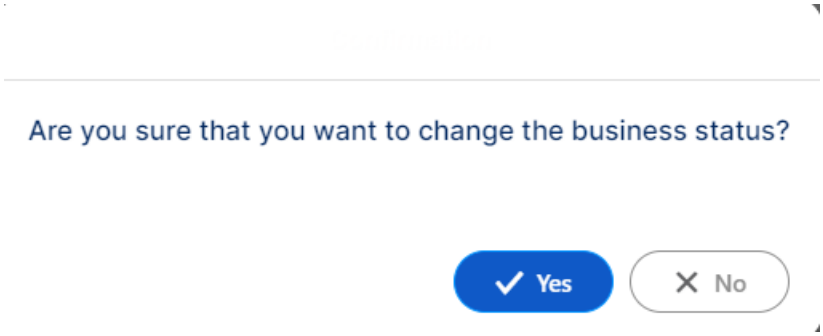
You can perform the approval either from a customer journey flow via API integration or from the Loan Management user interface.

After defining the relevant details of the contract, proceed to contract approval:

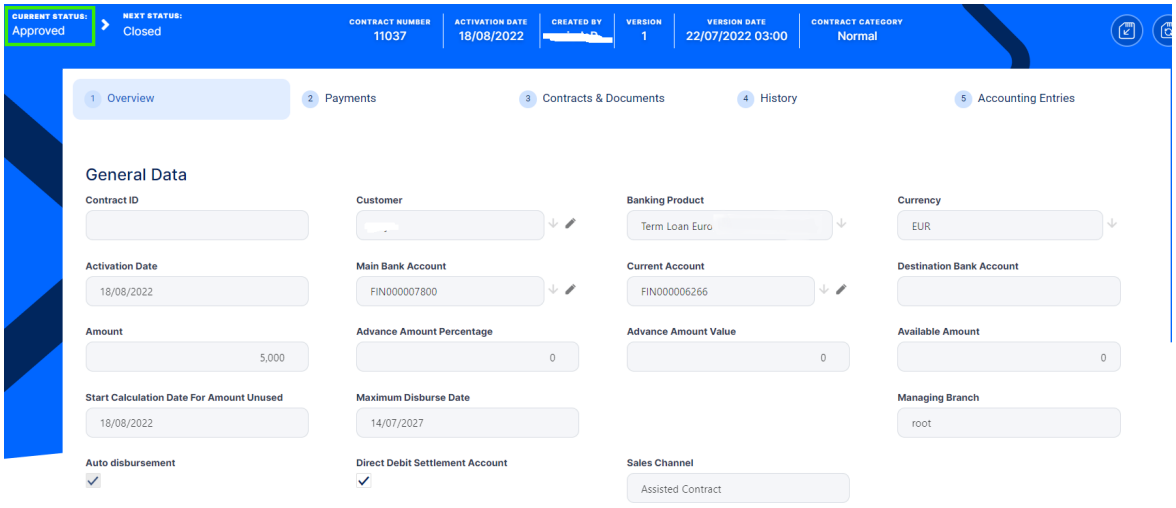
1. Select a contract in **Draft** (or **Version Draft**) status.
2. Change its status into **Approved**.



3. Click **Yes** to confirm your action.

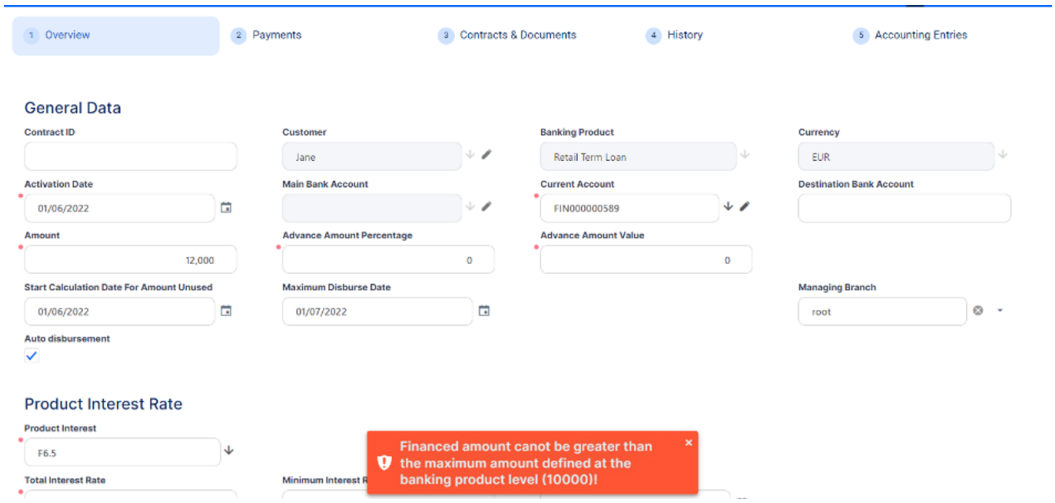


If Loan Management performs all the validations successfully, then the current status of the contract changes to **Approved**.

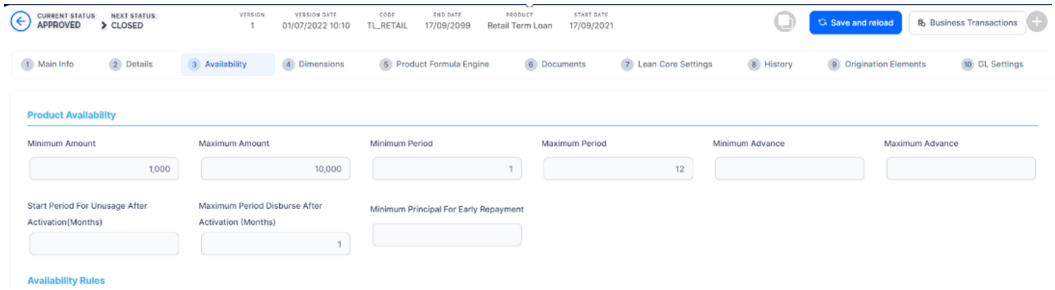


Example of error displayed by Loan Management if the validations are not met

If you try to approve a contract for the value of 12000, but the maximum amount allowed at the product level is only 10000, then Loan Management triggers an error because this value is beyond the product setup.



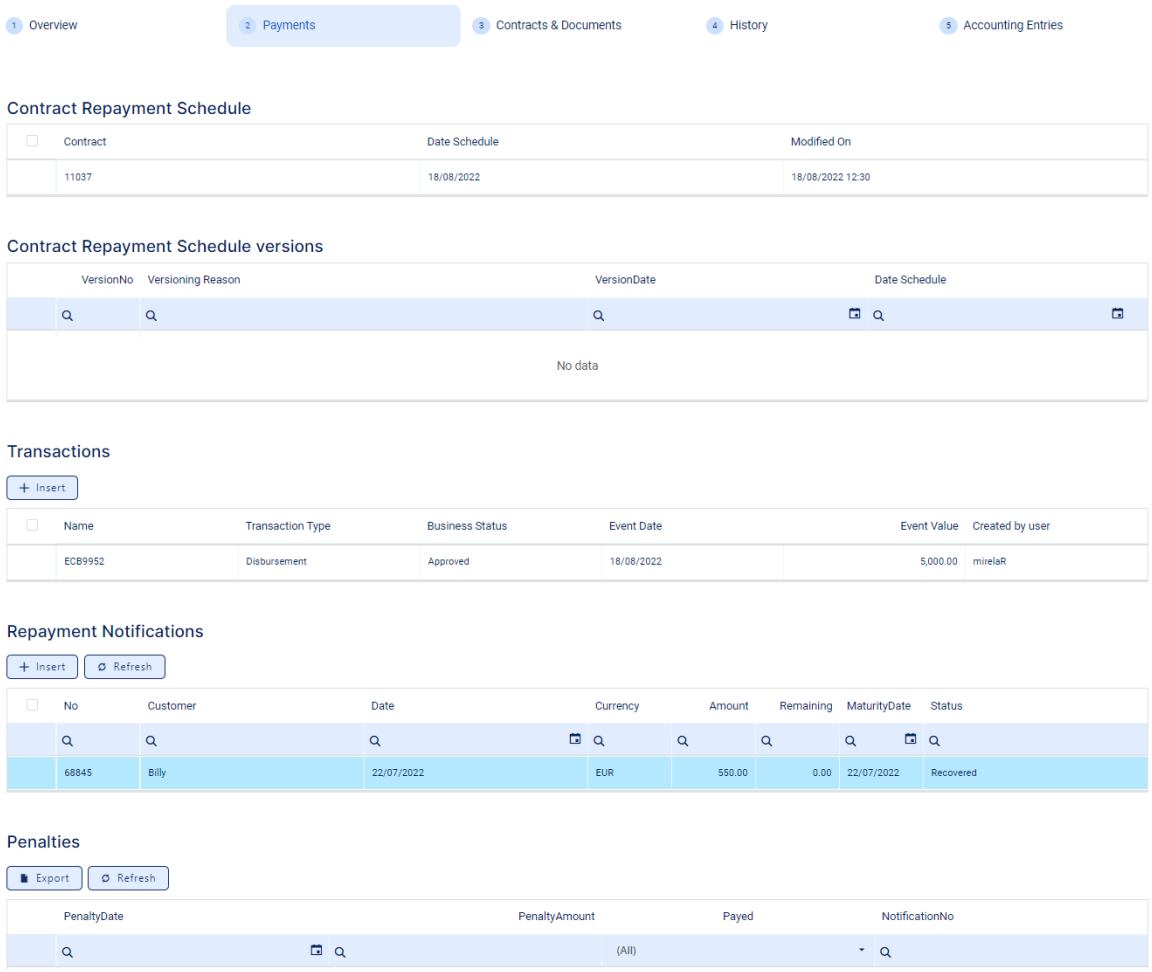
Similarly, Loan Management triggers errors if there are specific limits imposed from product level regarding minimum and maximum term, advance or amount itself. Below is how those limits are captured on product definition.



Automated Actions After Contract Approval

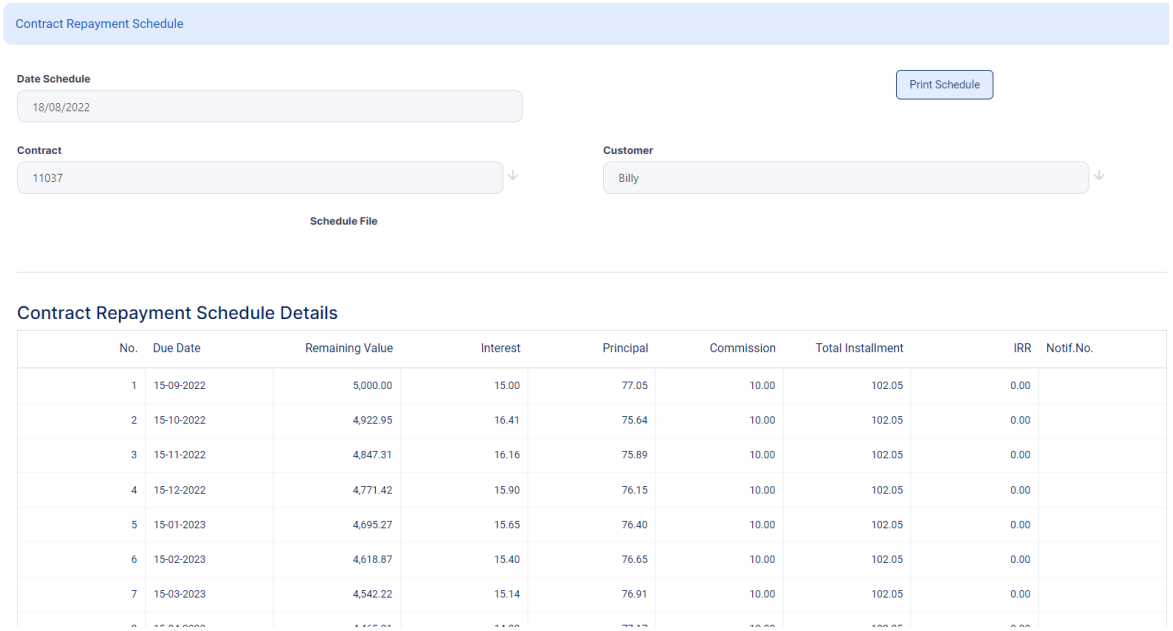
The **Main Bank Account** is created automatically for the bank defined as Main within the **Loan Management Operational > Bank** menu. In order for Loan Management to generate an account number, a rule must be defined during the implementation phase (example: branch code + incremental sequence number).

On the **Payments** tab you can see the repayment schedule that was generated following the disbursement (the auto disbursement functionality was selected), the disbursement transaction itself, the notification that was created for the flat upfront fee. Not only it got created but it is also settled from funds on the Current Account selected for settlement purposes, the same where funds were disbursed.



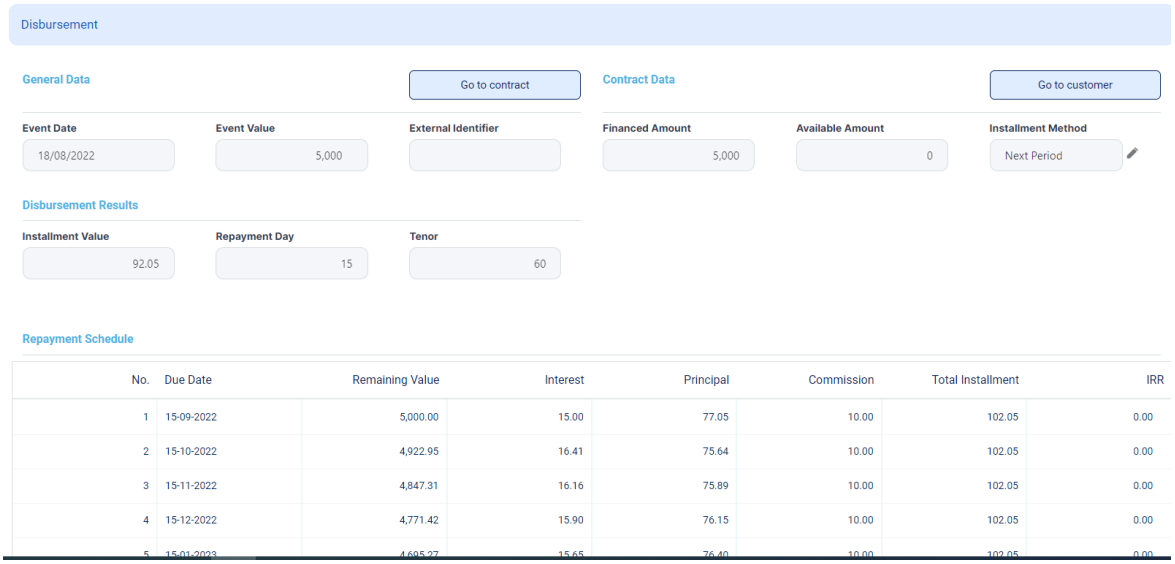
> **NOTE** The tab **Payments** has no information to display while the contract is in the **Draft** status. You must approve the contract to perform any contract event. Meaningful payment information is displayed in this tab only after performing transactions on the contract.

The **Contract Repayment Schedule** shows the equal installments, on top of which the Administration fee is added on contracts where applied.



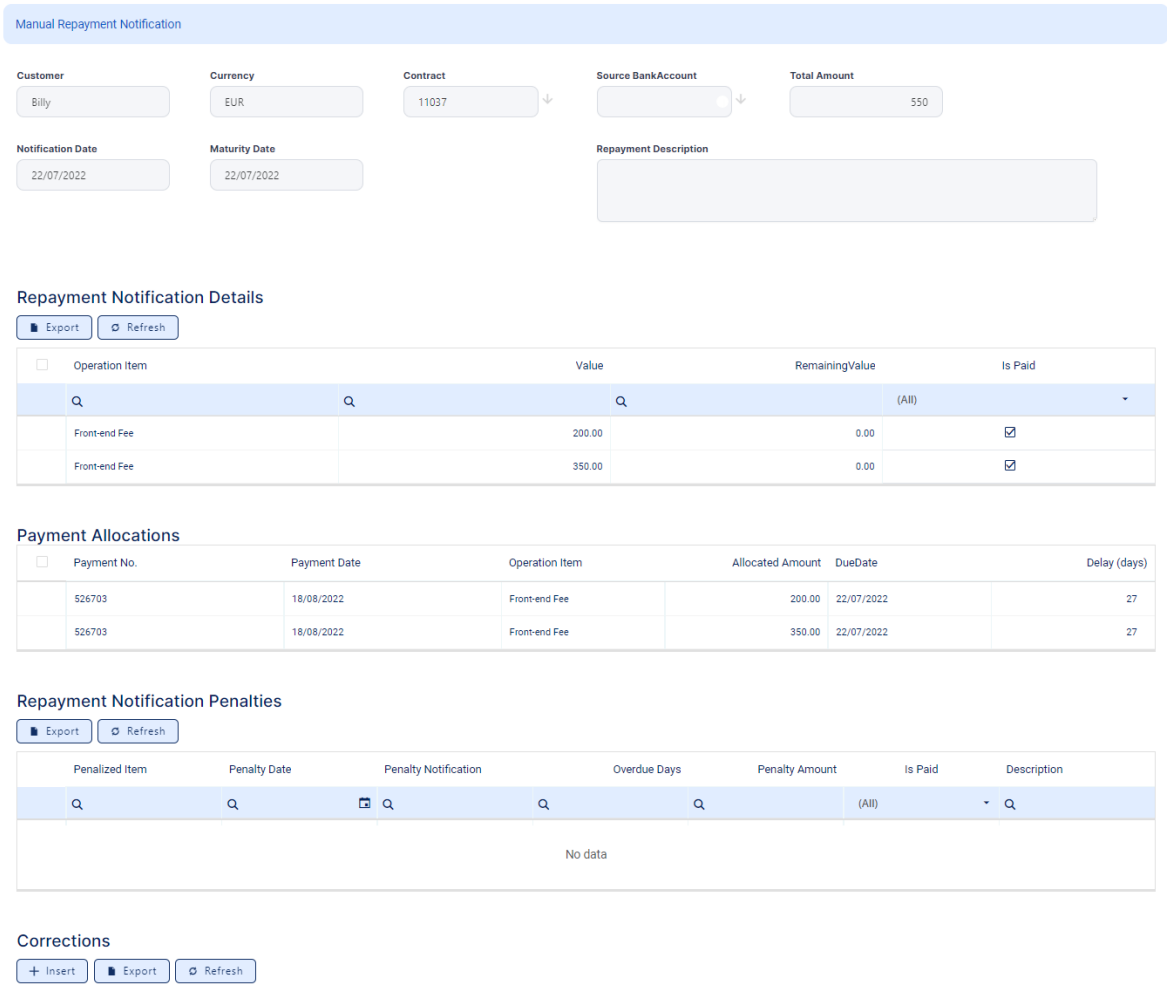
No.	Due Date	Remaining Value	Interest	Principal	Commission	Total Installment	IRR	Notif.No.
1	15-09-2022	5,000.00	15.00	77.05	10.00	102.05	0.00	
2	15-10-2022	4,922.95	16.41	75.64	10.00	102.05	0.00	
3	15-11-2022	4,847.31	16.16	75.89	10.00	102.05	0.00	
4	15-12-2022	4,771.42	15.90	76.15	10.00	102.05	0.00	
5	15-01-2023	4,695.27	15.65	76.40	10.00	102.05	0.00	
6	15-02-2023	4,618.87	15.40	76.65	10.00	102.05	0.00	
7	15-03-2023	4,542.22	15.14	76.91	10.00	102.05	0.00	

The **Disbursement** transaction includes the repayments schedule that was generated with that disbursement.

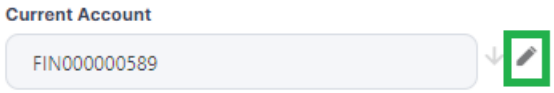


No.	Due Date	Remaining Value	Interest	Principal	Commission	Total Installment	IRR
1	15-09-2022	5,000.00	15.00	77.05	10.00	102.05	0.00
2	15-10-2022	4,922.95	16.41	75.64	10.00	102.05	0.00
3	15-11-2022	4,847.31	16.16	75.89	10.00	102.05	0.00
4	15-12-2022	4,771.42	15.90	76.15	10.00	102.05	0.00
5	15-01-2023	4,695.27	15.65	76.40	10.00	102.05	0.00

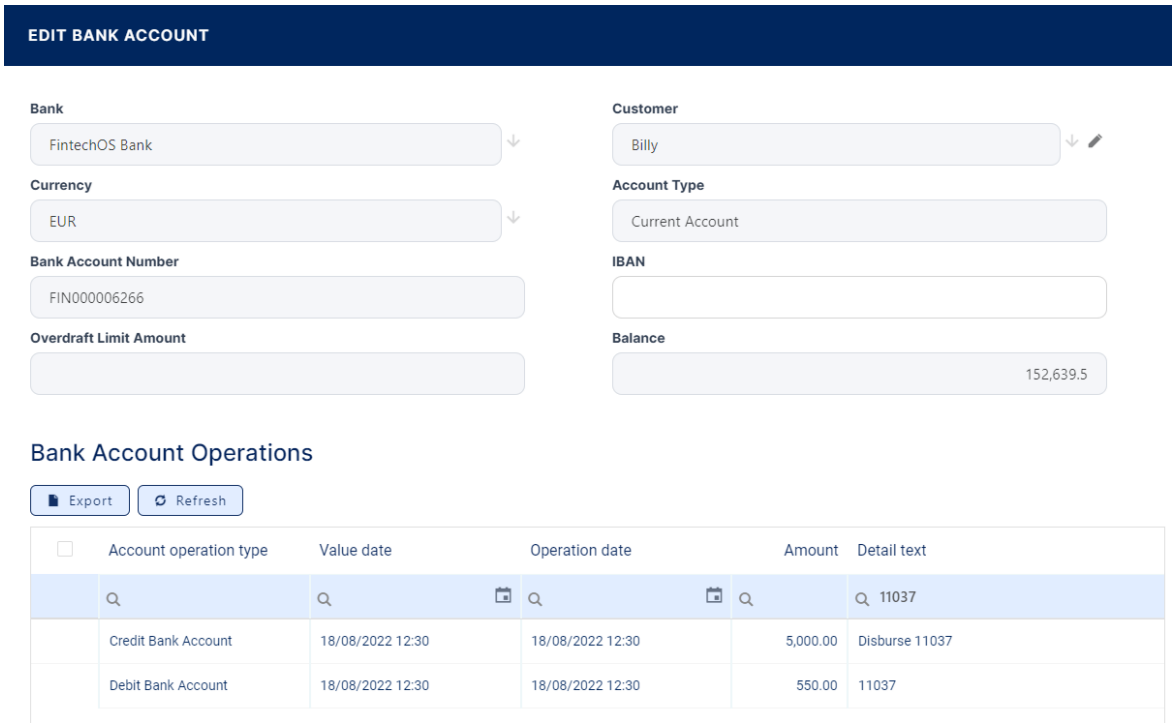
The **Repayment Notification** shows what was made due, when, how much, what type of amount and when/if amounts were recovered.



You can also check the **Current Account** transaction. On the contract's **Overview** tab, click on the pencil next to the `Current Account`:



In the displayed window, filter by your contract number and get the amounts posted to the account following our contract processing.

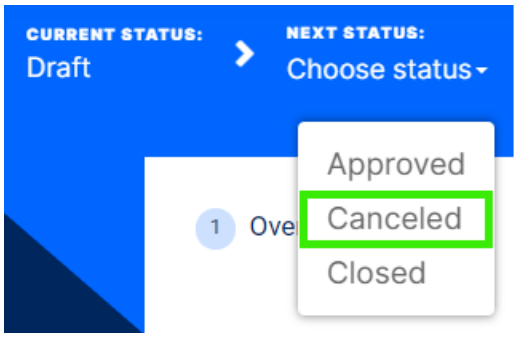


Rejecting a Loan

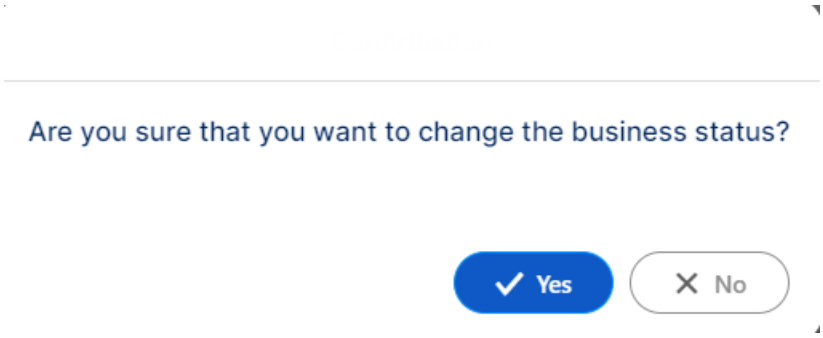
You can reject a loan, canceling it, when the deal with the customer drops. You can perform the cancellation either from a customer journey flow via API integration or from the Loan Management user interface.

Follow these steps to cancel the contract:

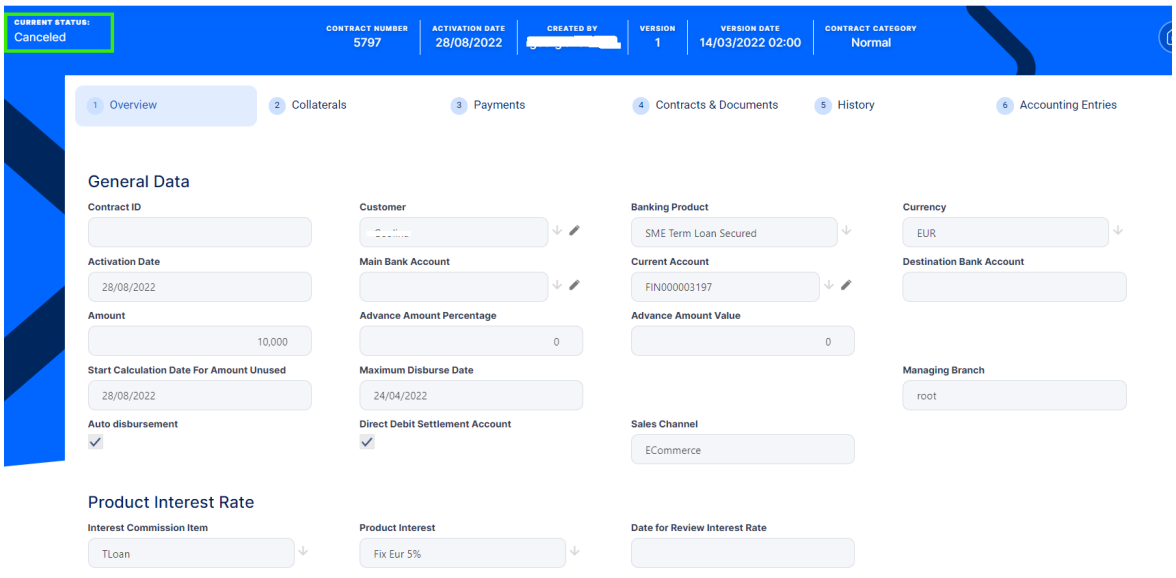
1. Select a contract in **Draft** (or **Version Draft**) status.
2. Change its status into **Canceled**.



3. Click **Yes** to confirm your action.



If Loan Management performs all the validations successfully, then the current status of the contract changes to **Canceled**.



NOTE You can't further use a canceled contract. Create a new contract, if you need to.

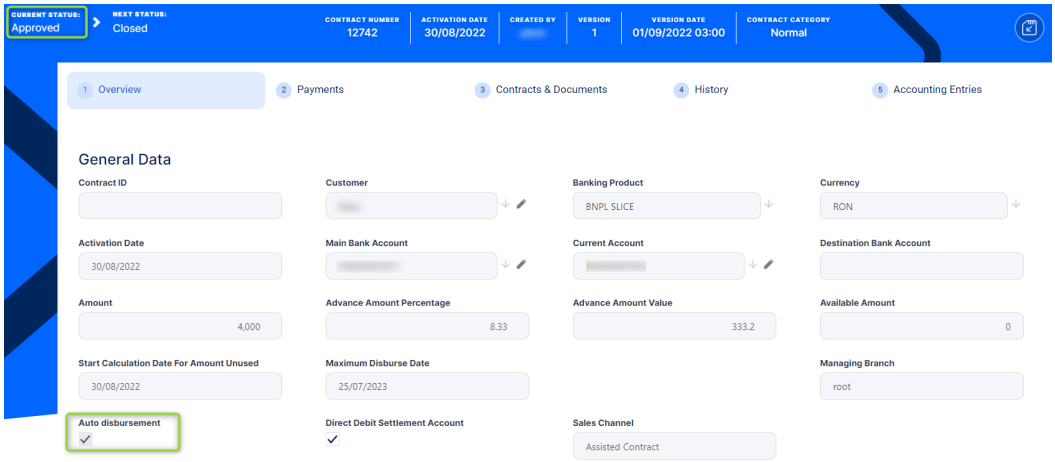
Disbursing a Loan

A disbursement represents the actual delivery of funds from a bank account to the customer. The repayment schedule of the loan contract is automatically calculated or recalculated as a result of performing a disbursement.

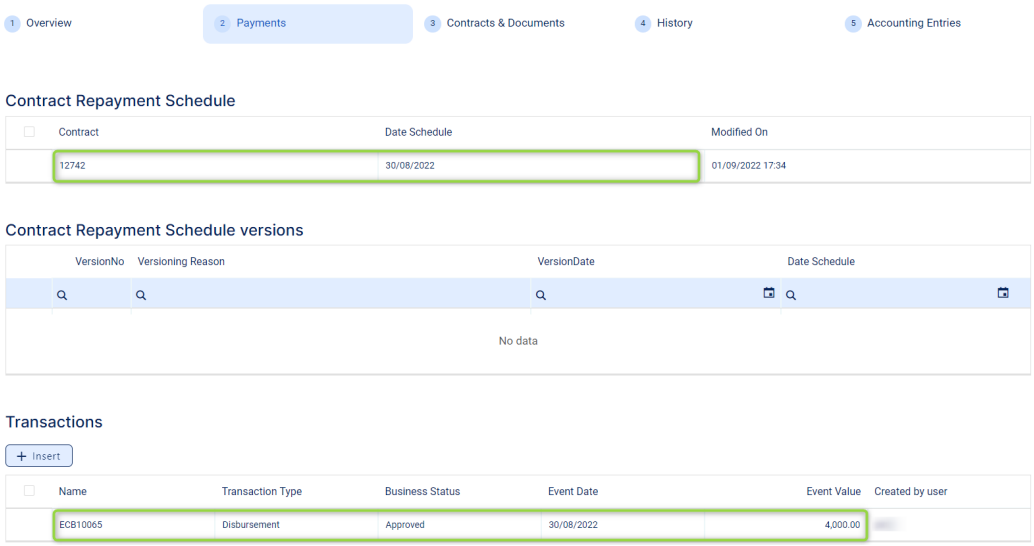
In Loan Management, you can choose between a disbursement performed automatically when the contract gets approved, or manually triggering the disbursement by performing a disbursement transaction, for cases when you need a request for disbursement from the customer, or other details are pending before you can release the funds, such as registering the collaterals with a certain authority.

Automatic Disbursement at Contract Approval

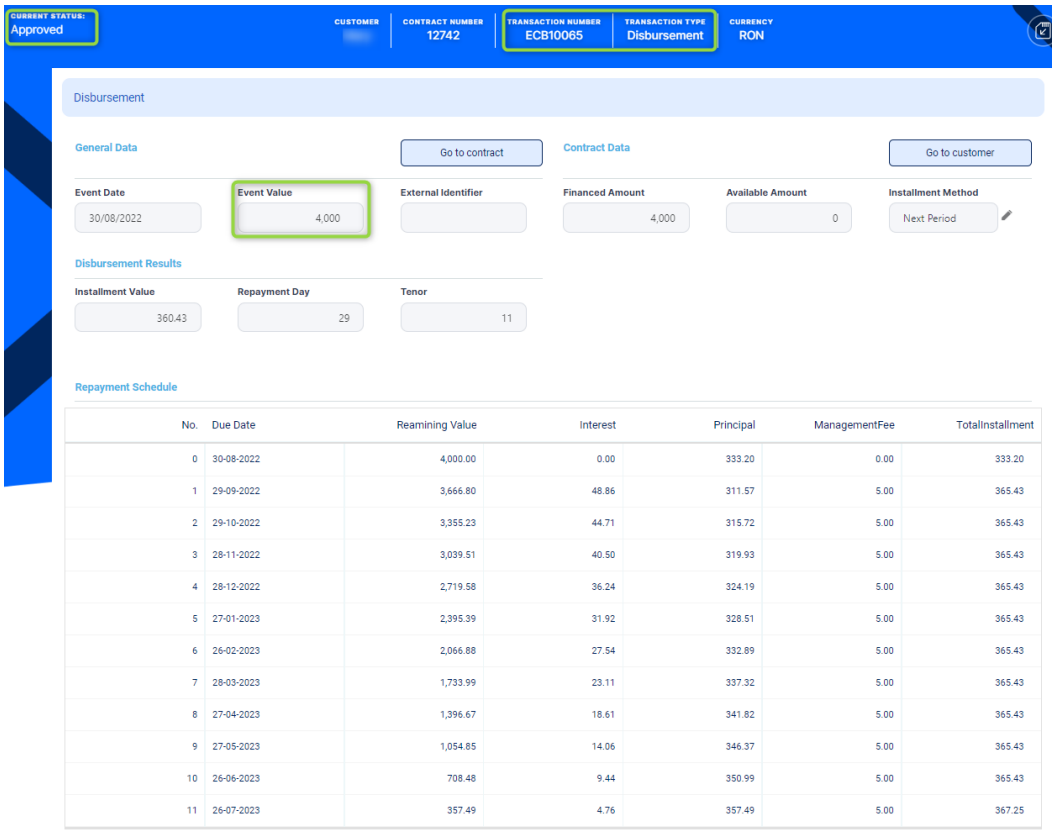
You can instruct Loan Managementto automatically perform the disbursement of funds for a loan contract during contract creation, if the auto disbursement property was set at banking product level. To do this, select the **Auto disbursement** checkbox in the contract's **Overview** tab, thus marking the financed amount to be automatically disbursed on the approval of the contract.



If selected, Loan Management performs the disbursement transaction immediately after contract approval, and the funds are moved to the settlement account or destination account as per instructions. The contract repayment schedule is also calculated:



You can see the disbursement transaction's details and the calculated repayment schedule by double-clicking the transaction.



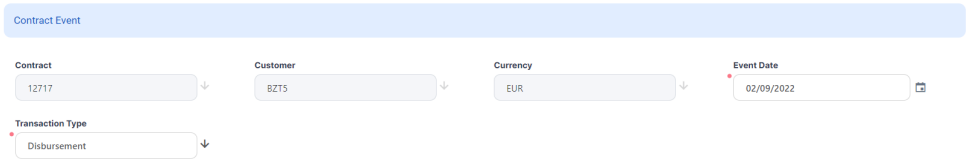
No.	Due Date	Reamining Value	Interest	Principal	ManagementFee	TotalInstallment
0	30-08-2022	4,000.00	0.00	333.20	0.00	333.20
1	29-09-2022	3,666.80	48.86	311.57	5.00	365.43
2	29-10-2022	3,355.23	44.71	315.72	5.00	365.43
3	28-11-2022	3,039.51	40.50	319.93	5.00	365.43
4	28-12-2022	2,719.58	36.24	324.19	5.00	365.43
5	27-01-2023	2,395.39	31.92	328.51	5.00	365.43
6	26-02-2023	2,066.88	27.54	332.89	5.00	365.43
7	28-03-2023	1,733.99	23.11	337.32	5.00	365.43
8	27-04-2023	1,396.67	18.61	341.82	5.00	365.43
9	27-05-2023	1,054.85	14.06	346.37	5.00	365.43
10	26-06-2023	708.48	9.44	350.99	5.00	365.43
11	26-07-2023	357.49	4.76	357.49	5.00	367.25

Adding a Disbursement Transaction To an Approved Contract

You can add disbursement transactions to an approved contract via Loan Management's user interface or through API calls, using the Loan Management endpoints. Read more about these endpoints in the Loan Management Developer Guide.

In order to add a disbursement transaction to a loan contract through the menus available in Loan Management, follow these steps:

1. In FintechOS Portal, select a contract with **Approved** status and double-click to open it.
2. Navigate to the contract's **Payments** tab and click the **Insert** button above the **Transactions** section. The **Event** page is displayed.

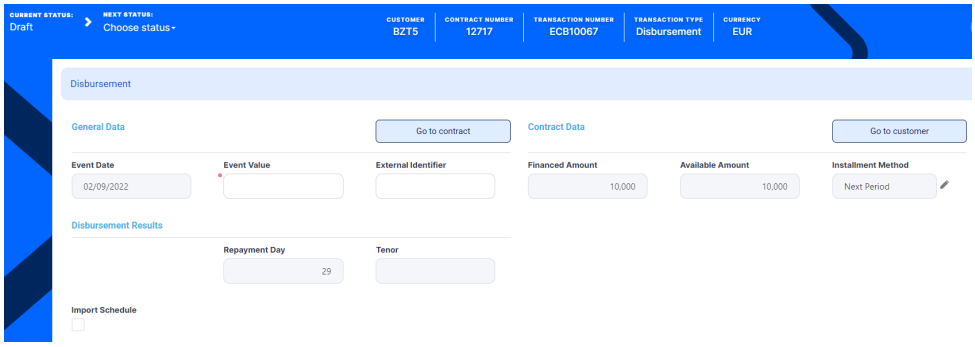


3. Fill in the following fields:

   - **Event Date** - This is pre-filled with current date.
   - **Transaction Type** - Select from the list the **Disbursement** transaction type. If you can't find it, then the transaction type is not associated with the banking product which is at the base of the contract.

   Other values are automatically completed: contract, customer, and currency.

4. Click the **Save and Reload** button.
   The event is saved in **Draft** status and a transaction number is automatically generated for it. The **Edit Contract Event** page corresponding to the selected transaction type is displayed. A series of value fields are automatically calculated, their values are displayed, and you can't edit them: the contract's financed and available amounts, the selected installment calculation method, the repayment day ant the contract's tenor.

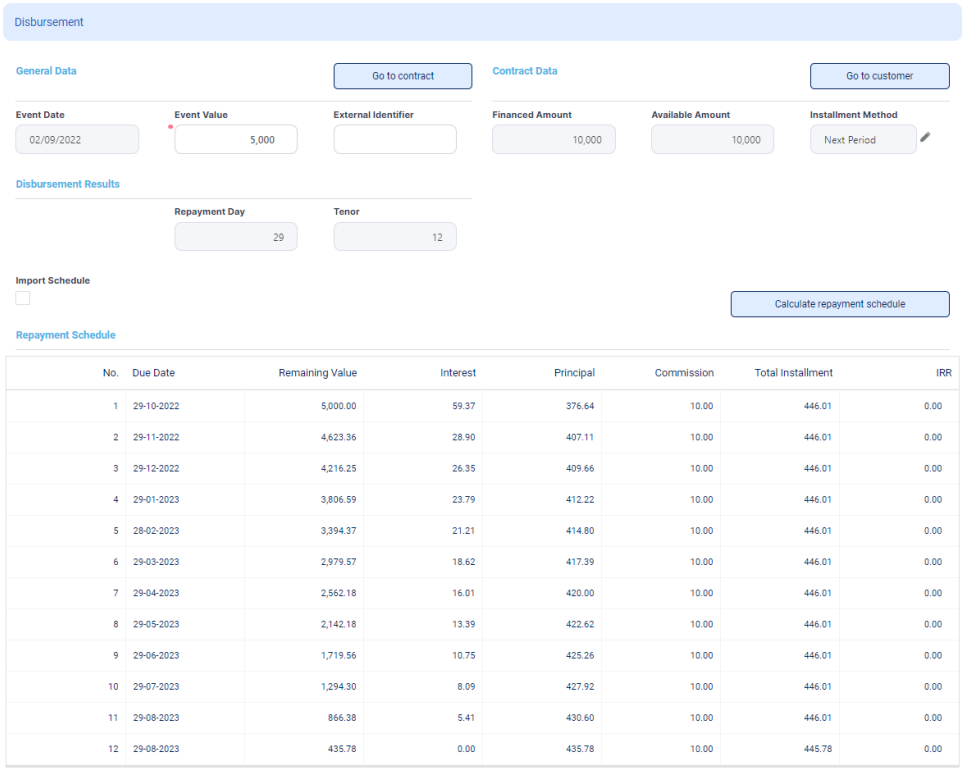


5. Fill in the **Event Value** with the value of the transaction, and enter an **external identifier** of the transaction, if available.

6. Decide whether you want to use the repayment schedule calculated by Loan Management through automatic processes, or you want to import a custom schedule. For custom schedule, select the **Import Schedule** checkbox. Read more about importing a custom schedule file in the Manually Upload Repayment Schedules section of the user guide.

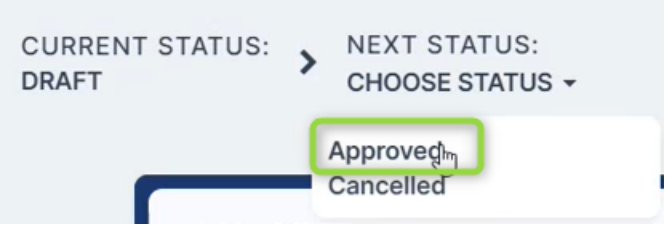
7. Click the **Save and Reload** button.
   If the event value meets the business requirements defined within Loan Management, the event is saved. Otherwise, an error message appears. Change the values as instructed in the message and try saving the event again.
   While the event is in **Draft** status, you can modify all the event's fields except **Transaction Type**. The event value is not applied to the contract while the event is still in this status.

8. Click the **Calculate repayment schedule** button to view the details of each installment of the calculated repayment schedule.



No.	Due Date	Remaining Value	Interest	Principal	Commission	Total Installment	IRR
1	29-10-2022	5,000.00	59.37	376.64	10.00	446.01	0.00
2	29-11-2022	4,623.36	28.90	407.11	10.00	446.01	0.00
3	29-12-2022	4,216.25	26.35	409.66	10.00	446.01	0.00
4	29-01-2023	3,806.59	23.79	412.22	10.00	446.01	0.00
5	28-02-2023	3,394.37	21.21	414.80	10.00	446.01	0.00
6	29-03-2023	2,979.57	18.62	417.39	10.00	446.01	0.00
7	29-04-2023	2,562.18	16.01	420.00	10.00	446.01	0.00
8	29-05-2023	2,142.18	13.39	422.62	10.00	446.01	0.00
9	29-06-2023	1,719.56	10.75	425.26	10.00	446.01	0.00
10	29-07-2023	1,294.30	8.09	427.92	10.00	446.01	0.00
11	29-08-2023	866.38	5.41	430.60	10.00	446.01	0.00
12	29-08-2023	435.78	0.00	435.78	10.00	445.78	0.00

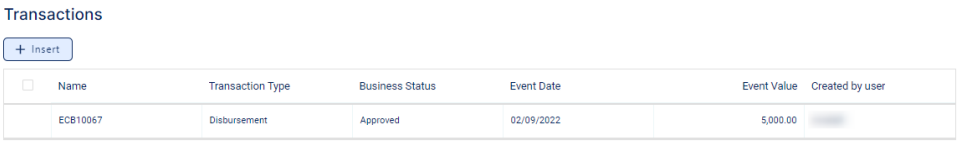
9. Approve the event by changing its status to **Approved** in the upper left corner of the **Event** page.



> **IMPORTANT!** A Disbursement event can't be approved if the current system date > the minimum between Maximum Disburse Date and Maturity Date-1.

10. Confirm the change of status in the **Confirmation** window, clicking **Yes**. The event is now in **Approved** status and Loan Management transfers the funds to the configured settlement account.

    The event value is now applied and visible in the contract's **Payments** tab -> **Transactions** section.



> **NOTE**
> All existing versions of the contract in **Contract Version Draft** status are automatically changed to **Contract Version Closed** when a payment event is approved for that contract.

Processing Loan Repayments

Based on defined product pricing and established parameters, financial institutions can manage the billing and collection process on a loan contract fully automatically. Once the disbursement is performed, Loan Management generates the repayment

schedule and sends the payment notifications to the customer on the due date. Automated collection is triggered via integration with 3rd party payments engine or from the customer's settlement account. If funds are not available, overdue amounts and days are calculated and penalties applied.

Once a loan approved and disbursed, you can check the repayment schedule built based on contract details, on the **Payments** tab, in the designated section, as described in the Viewing a Contract's Repayment Schedule topic. If there are front-end fees, they are notified automatically by Loan Management and you can check them on the **Payments** tab, in the **Repayment Notifications** section (even without disbursing the funds). Read about repayment notifications in the Managing Repayment Notifications topic.

When the system reaches the dates that appear on schedule projections, the amounts resulting are made due, and Loan Management automatically triggers the notifications. Depending on the availability of funds in the settlement account and the direct debit setup, Loan Management settles those notifications, marking them with the **Recovered** status. Any amount that is not recovered on due date stays on the notification, and when funds become available, Loan Management automatically recovers and allocates them to the pending notifications based on the **Payment Allocation Method** setup at the product level, in the Lean Core tab. When the notifications are recovered, you can see the underlying debit transactions on the settlement account – there is always such traceability of the funds. You can also see the automatic payments performed by Loan Management either at contract level or in a dedicated menu, as described in the Viewing Customer Payments topic.

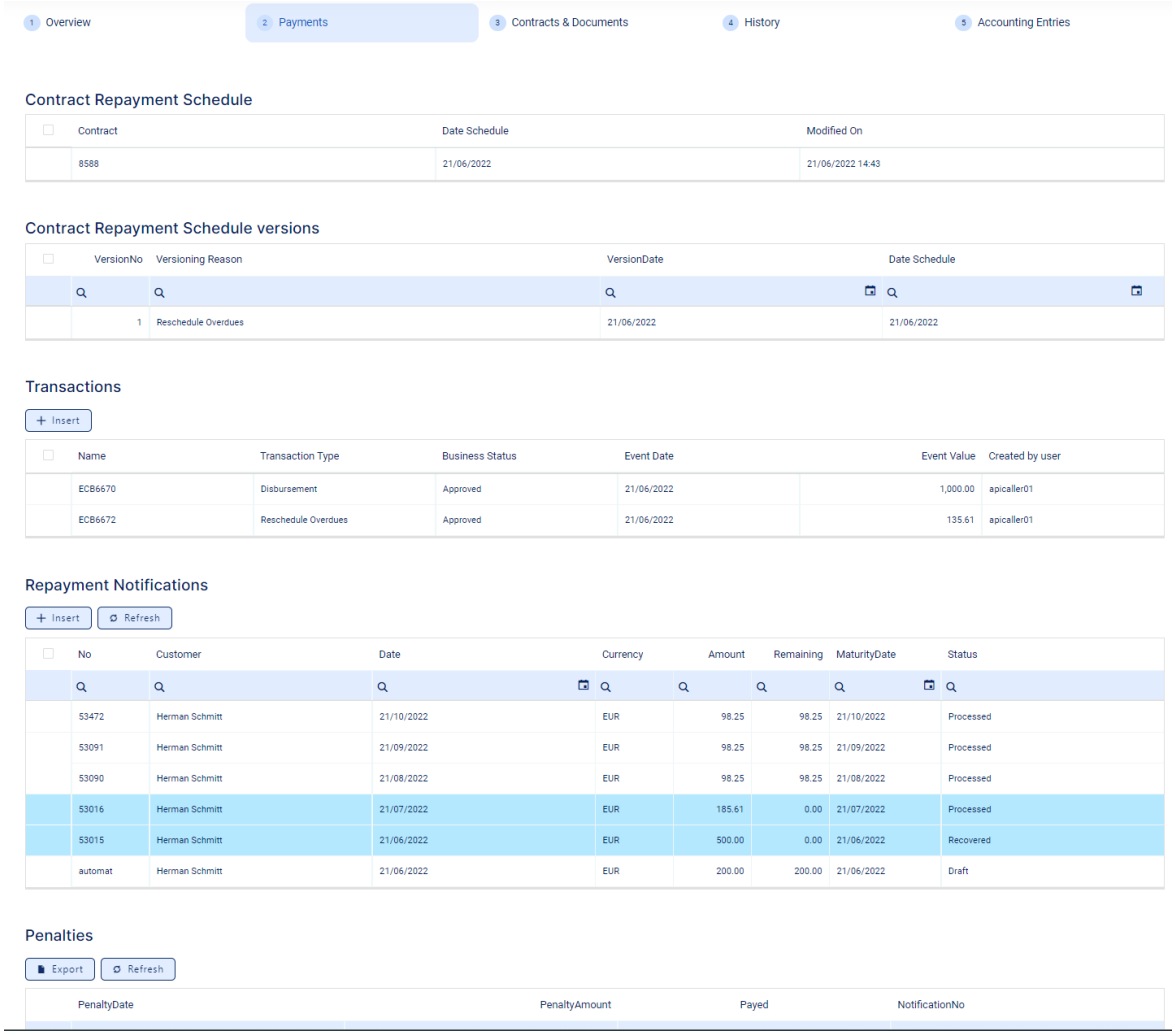
Payment Schedule Types, defined at the Banking Product Factory level, define how Loan Management handles the following:

- How the interest is calculated (day basis: 30/360, actual/ 360, etc).
- If the capital repayment is linear or annuity type (same principal for every installment or increasing principal and decreasing interest resulting in a constant amount being due for every installment).
- Fees you want to include in the repayment schedule.
- If you allow for manual installment amount to be provided and overwrite automatically calculated one.
- Frequency of the installment (monthly, every 30 days, etc).

When amounts are not available to cover notified amounts and there is a penalty interest defined for the product, the missed amounts are subject to automatic penalty calculation. Loan Management calculates and notifies the penalty interest daily. All the penalties applied by automated processes at the contract level are visible on the **Payments** tab, in the **Penalties** section, as described in the Viewing a Contract's Penalties topic. When the penalty interest is defined, there are specific Operation Items linked to it so that the system knows what types of amounts are subject to penalty: overdue principal, interest, commissions. Alternatively, the penalty interest can be applied to all missed payments. Penalty interest is defaulted from the product level and, if allowed, it can also be amended at contract level.



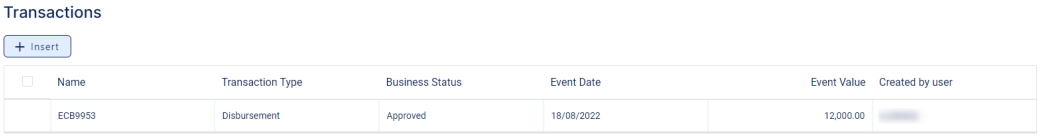
You can find all the existing transactions, payments, penalties, bank account operations, repayment schedules, schedule versions, repayment notifications for a contract on the **Payments** tab. The tab has no information to display for contracts in **Draft** status. Approve the contract to perform any transactions on the contract. Meaningful payment information is displayed here only after performing a disbursement.



Managing a Contract's Transactions

Contract transactions are events/ changes performed at the **Approved** contract's level. Such events are disbursements, reschedule overdues, early repayments, applying payment holidays, returning of amounts, and so on. Read more information about the available transaction types in the "Transaction Types Used in Loan Management" on page 80 topic.

The **Transactions** section within the **Payments** tab holds all the transactions performed at the contract level, in any status. This section only has information if the contract is in **Approved** status and transactions were already created.



Event Statuses

An event record has the following statuses, visible in the top left corner of any **Event** page:

- **Draft** - the status of a newly created event record that was not yet sent for approval. The event value is not applied to the contract while the event is still in this status. While in this status, you can edit some fields. Approve after editing all the necessary details.
- **Approved** - the status of an event record after being authorized. The event value is applied to the contract and you cannot edit any of the event's details. There is no further transition from this status.
- **Canceled** - the status of an event record after being canceled. The event value is not applied to the contract and you cannot edit any of the event's details. There is no further transition from this status.

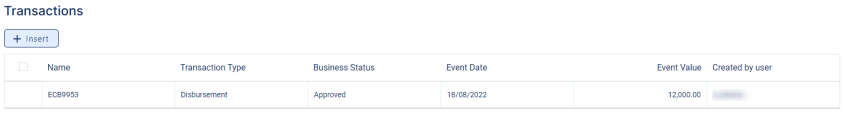
> **NOTE**
> As a best practice, new records or new versions of existing records created on a specific day should be approved on the same day.

Viewing Existing Events

To view the events on a contract, follow these steps:

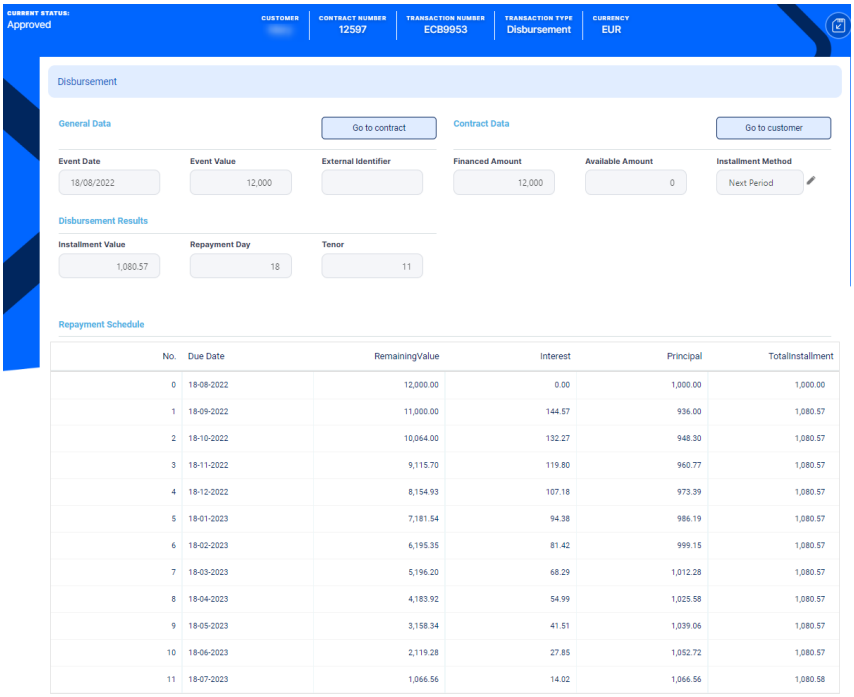
1. In FintechOS Portal, select a contract with **Approved** status and double-click to open it.

2. Navigate to the contract's **Payments** tab and view the list of events displayed in the **Transactions** section.



Here you can see only basic information about the transactions, such as event number, status, date, transaction type, value and the user who created it.

3. To view detailed information about the transaction and the repayment schedule generated for the **Approved** event, double-click the event record to open the **Event** page:



You can't edit the information displayed on this page.

4. View the following information displayed about each event, with some variations depending on the event type:

- Transaction status, contract number, customer name, transaction number, type, and currency, all displayed in the header of the page.

The following details are displayed in the body of the page:

- Event date and value. Contract events added through API integration also contain an external identifier.
- Contract's financed and available amounts, and the installment method used for calculating the schedule.
- The event's results, in this example, the results of a disbursement event: amount of the installments, repayment day and tenor of the contract in months.

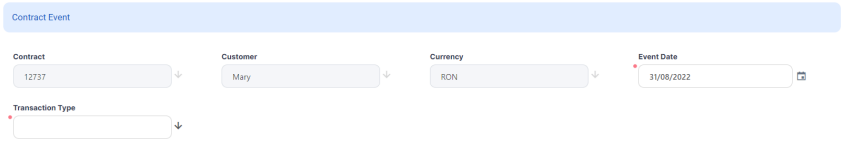
5. In the **Contract Repayment Schedule** section, view the information about each of the installments that are part of the repayment scheduled calculated as a result of performing the transaction:
    - Number of the repayment schedule version detail, date when the installment must be paid, value remaining to be repaid from the contract value at the moment of this installment, value of the interest and of the principal calculated for this installment, and total value of the installment to be paid.

Adding Events To Approved Contracts (General Steps)

You can add events to an approved contract via Loan Management's user interface or through API calls, using the Loan Management endpoints. Read more about these endpoints in the Loan Management Developer Guide.

In order to add events to a contract through the menus available in Loan Management, follow these steps:

1. In FintechOS Portal, select a contract with **Approved** status and double-click to open it.
2. Navigate to the contract's **Payments** tab and click the **Insert** button above the **Transactions** section. The **Event** page is displayed.

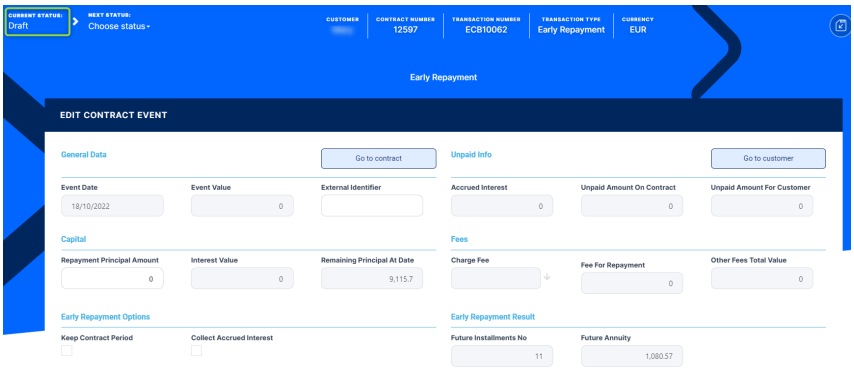


3. Fill in the following fields, common to every transaction type:
   - **Event Date** - This is pre-filled with current date.
   - **Transaction Type** - Select from the list the transaction type. Only the transaction types associated with the banking product which is at the base of the contract are displayed here.

   Other values are automatically completed: contract, customer, and currency.
4. Click the **Save and Reload** button.
   The event is saved in **Draft** status and a transaction number is automatically generated for it. The **Edit Contract Event** page corresponding to the selected transaction type is displayed. A series of value fields are automatically calculated and their values are displayed.
5. Depending on the selected transaction type, follow the instructions for each event type as defined in the corresponding topics:
   - Disbursement - described in the "Disbursing a Loan" on page 248 topic
   - Early repayment - described in the "Refinancing a Loan By Performing an Early Repayment" on page 320 topic

- Reschedule Overdues - described in the "Rescheduling the Overdue Amounts for a Contract" on page 316 topic
- Payment Holiday - described in the "Applying Payment Holiday to a Loan" on page 286 topic
- Returned Amount or Goods - described in the "Working with Returns" on page 301 topic.

6. After defining the event as described in the corresponding page, save it.
   If the event value meets the business requirements defined within Loan Management, the event is saved. Otherwise, an error message appears. Change the values as instructed in the message and try saving the event again.
   While the event is in **Draft** status, you can modify all the event's fields except **Transaction Type**. The event value is not applied to the contract while the event is still in this status.

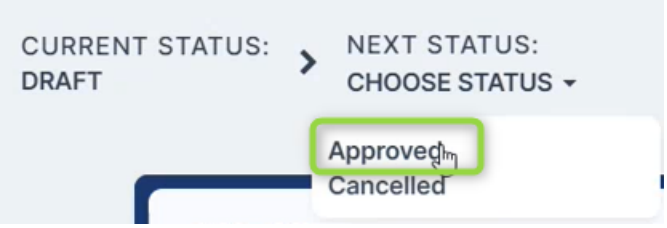


   Depending on the selected transaction type, new sections are displayed at the bottom of the page, containing the contract repayment schedule for the event and any generated notifications.

7. If it appears, click the **Calculate/ Simulate Early repayment/ Simulate repayment schedule/ Simulate reschedule overdues** buttons (the displayed button depends on the selected transaction type) to view the details of each installment.

For **Reschedule Overdue** transaction type, select from the list the overdue payment notifications that you wish to reschedule.

8. Approve the event by changing its status to **Approved** in the upper left corner of the **Event** page.



9. Confirm the change of status in the **Confirmation** window, clicking **Yes**. The event is now in **Approved** status.

   The event value is now applied and visible in the contract's **Payments** tab -> **Transactions** section.

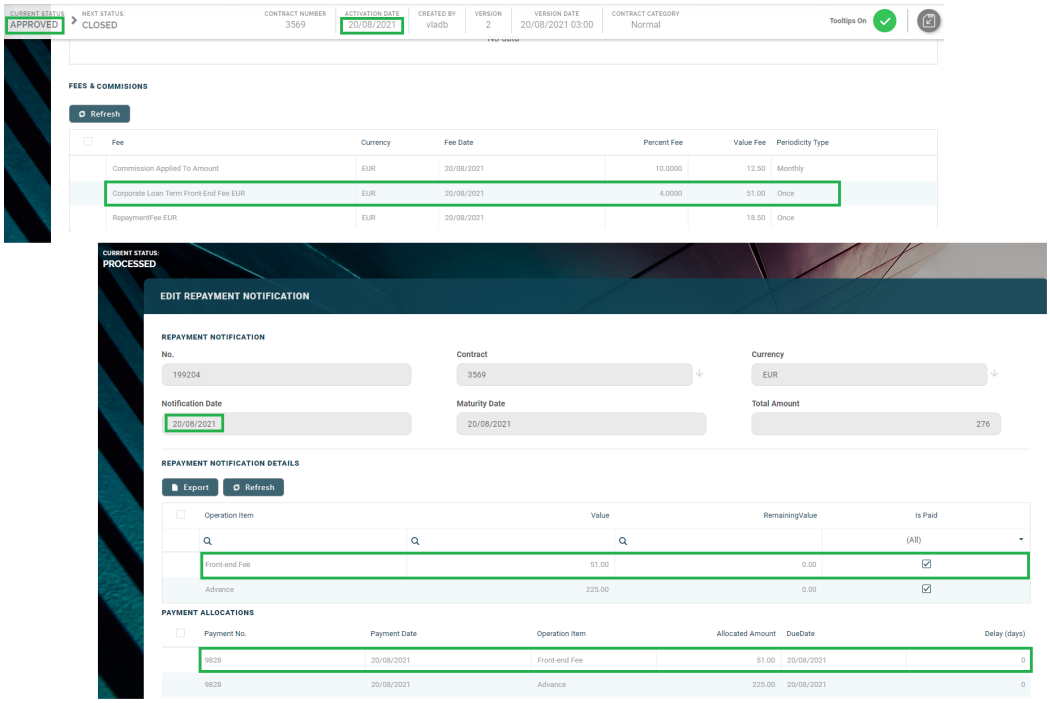
> **NOTE**
> All existing versions of the contract in **Contract Version Draft** status are automatically changed to **Contract Version Closed** when a payment event is approved for that contract.

Managing Repayment Notifications

Loan Management automatically generates notifications for each installment that has to be paid for existing contracts that disbursed various amounts to customers. There can be various types of notifications generated for fees, commissions, payment holidays, and so on. Loan Management also assists you reconcile wrongly processed cases and other situations resulted from delayed processing/ wrong updates. Thus, depending on your user rights, you can manually add notifications for an active contract based on lending product types, term loans, and mortgages, or add notifications for amounts to be recovered even if there is no active contract. Loan Management includes all such manual notifications in the recovery processes.

All the repayment notifications that are not paid in full after their maturity dates are subject to automated penalty calculation processes performed by Loan Management. Thus, for repayment notifications linked to a contract, Loan Management uses the penalty interests defined within the interests list attached at the banking product level, and for repayment notifications not linked to a contract, it uses the penalty interest list specified in a `system parameter`.

> **NOTE**
> All the `Front-End Fee` commission types with `Once` periodicity type applied to a contract are notified and must be paid when the contract is approved. The Loan Management system parameter `FrontEndFee` defines the type of commission that is automatically notified at the contract approval.

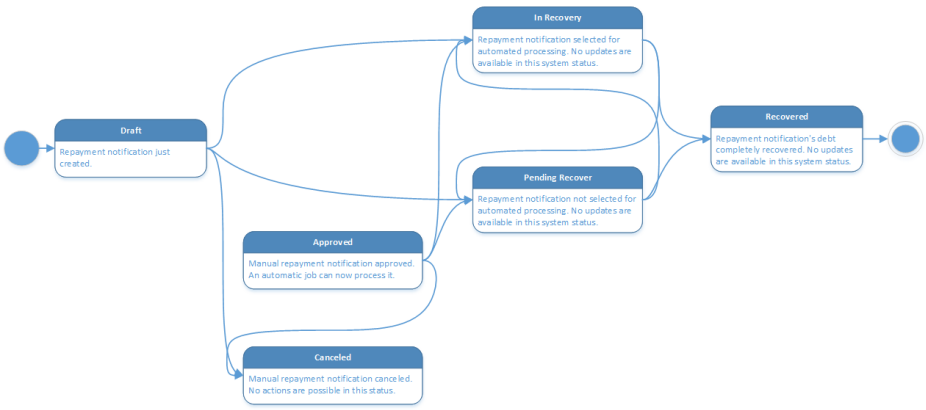


Repayment Notification Statuses

A repayment notification record has the following business workflow statuses:

- **Draft** - the status of a newly created repayment notification record, either automatic or manual.
- **Approved** - the status of a manual repayment notification record after being authorized by a user with notification approval competencies. While in this status, you cannot edit the record's details. From this status, the record is picked up by a scheduled job and its status is automatically changed, depending on the direct debit settlement settings. If the `Direct Debit Settlement Account` field at the contract level = `True`, then the manual notification's status changes to **In Recovery**, otherwise it changes to **Pending Recover**.
- **Canceled** - the status of a manual repayment notification after canceling it straight from the **Draft** status. You can only cancel a manual notification if its `Total Amount` = `Remaining Value`.
- **Pending Recover** - this is a system status applied to repayment notification when `Direct Debit Settlement Account` at the contract level is set to `False`. No updates are available in this system status.
- **In Recovery** - this is a system status applied to repayment notification when `Direct Debit Settlement Account` at the contract level is set to `True`. No updates are allowed on the record.
- **Recovered** - the last status of a repayment notification, after the complete recovery of the notification's debt. No updates are allowed on the record.

The repayment notification status transitions are illustrated below:



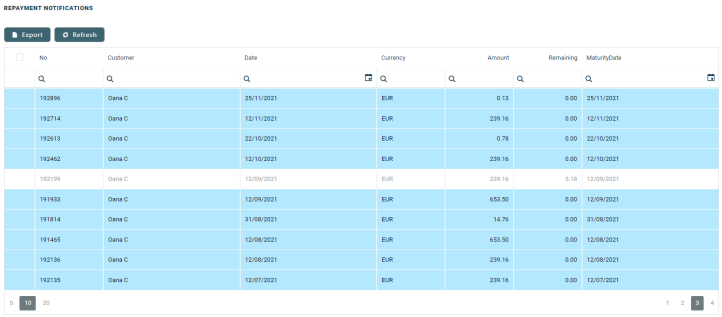
Accessing Repayment Notifications

Loan Management enables you to access notifications in several places, for your convenience.

Accessing a contract's repayment notifications

To view the notifications generated for a specific contract, follow these steps:

1. On the **Contract** page, navigate to the **Payments** tab > **Repayment Notifications** section.
2. View all the repayment notifications generated for the contract. This section only has information if the contract is in **Approved** status and disbursements were already performed.



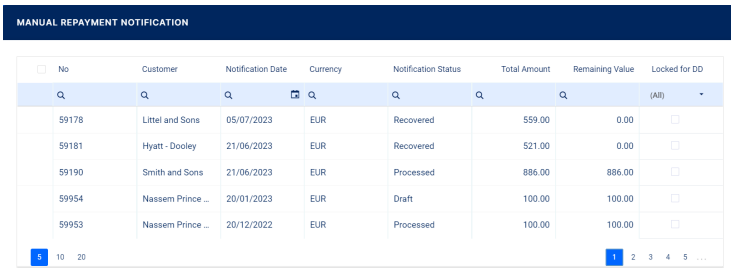
Repayment notifications highlighted in blue are already paid, allocated or closed to payment, while the ones not highlighted remain to be paid.

3. View the information is displayed about each notification:
    - Number, date, and status of the notification
    - Customer and currency of the contract
    - Amount of the installment for which the notification was generated
    - Remaining amount from the installment to be paid
    - Maturity date of the notification, automatically calculated adding the value of the `Grace period for repayment` field at the banking product level to the notification date.

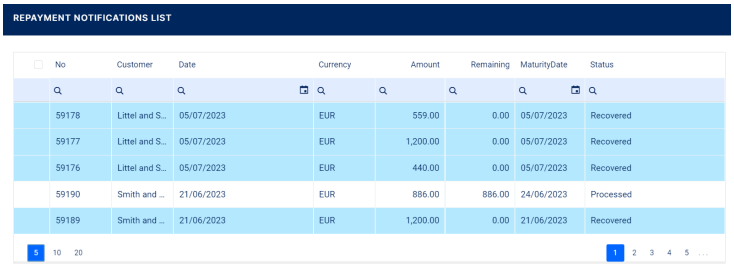
Accessing all the repayment notifications generated by Loan Management

To access all the notifications created in Loan Management, follow these steps:

1. In the FintechOS Portal, click the main menu icon and expand the **Loan Management Operational** menu.
2. To access only manually captured notifications, click **Manual Repayment Notification** menu item to open the **Manual Repayment Notifications** page.

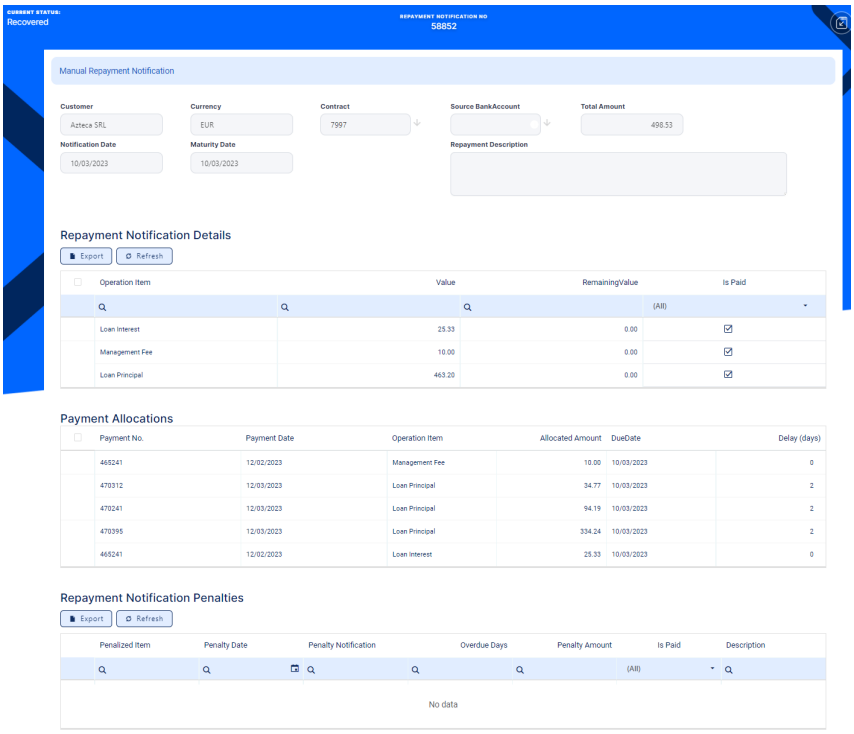


3. To access automatic and manual notifications, click **Repayment Notification** menu item to open the **Repayment Notifications List** page.



Viewing Repayment Notifications

1. To view the details of a repayment notification, double-click the desired record. The **Edit Repayment Notification** page is displayed for automatically generated notifications, or the **Edit Manual Repayment Notification** page for manual notifications, both presenting the repayment notification details.



> **NOTE**
>
> Automatically generated notifications can't be edited!
>
> You can only edit the details of manual notifications in `Draft` status.

2. View notification specific data in the **Repayment Notification** section:
    - **Repayment Notification No.** - The number of the repayment notification record.
    - **Customer** - The customer for whom the notification was generated.
    - **Currency** - The currency of the notification.

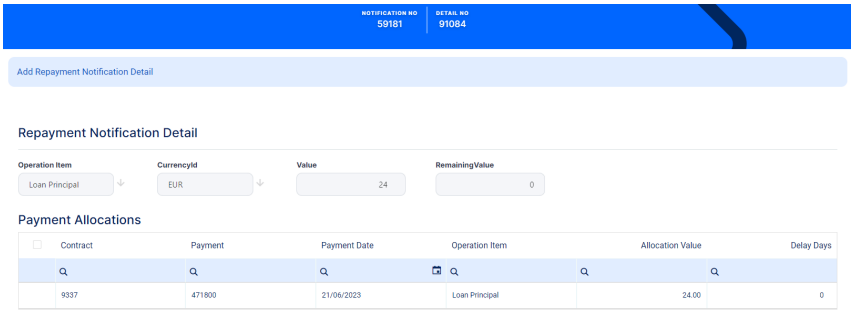
- **Contract** - The number of the contract for which the notification was generated.
- **Notification Date** - The date when the notification was generated.
- **Maturity Date** - The maturity date of the notification. This is calculated by adding the value of the `Grace period for repayment` field at the banking product level to the notification date.
- **Source Bank Account** - The bank account from where the notified amount should be allocated.
- **Total Amount** - The total amount to be paid within the notification (the sum of all the details' values).
- **Repayment Description** - A description of the manual notification.

3. View details (lines) of the notification in the **Repayment Notification Details** section:
   - **Operation Item** - The operation item for which the notification detail is generated.
   - **Value** - The value of the notification detail.
   - **Remaining Value** - The remaining value still to be paid from the notification value.
   - **Is Paid** - This checkbox is automatically marked as true when the full amount is allocated to the detail value. You cannot change this value.

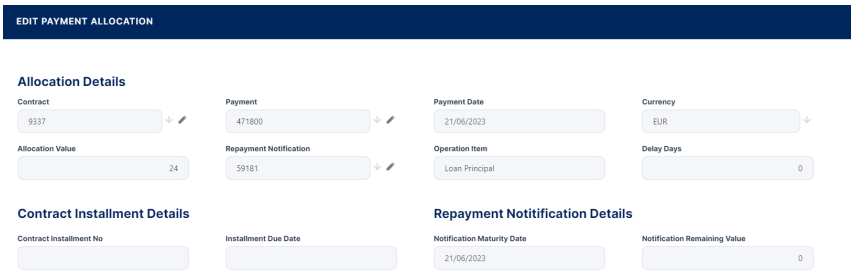
> **NOTE**
>
> Notification details are automatically marked as paid when a repayment transaction performed

and approved for the contract is allocated by the system to cover the value of the notification detail.

4. To view more information about a notification detail, double-click it to open the **Repayment Notification Details** page:



5. View information about the payments allocated for the notification details in the **Payment Allocation** section:
    - **Payment No.** - The number of the payment.
    - **Payment Date** - The date when the payment was performed.
    - **Operation Item** - The operation item from the notification for which the payment was allocated.
    - **Allocated Amount** - The amount allocated from the payment.
    - **Due Date** - The due date of the notification.
    - **Delays (days)** - The number of days passed since the notification's due date.
6. To view more information about a payment allocation, double-click it to open the **Edit Payment Allocation** page:



You cannot edit any of the fields from this page.

> **NOTE**
>
> The operation item is used in the payment allocation process. If the repayment notification is not linked to contract, then Loan Management takes the operation item value from the allocation method configured within the `ManualAllocationMethod` system parameter. If a repayment notification is created for a contract with `Closed` status, then Loan Management takes the operation item value from the allocation method selected at the banking product level.

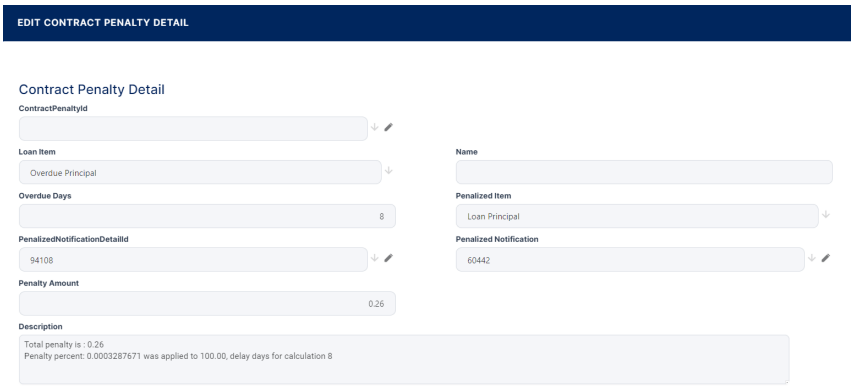
7. View information about the penalties calculated for the manual repayment notifications that were not paid in full until their maturity date in the **Viewing Notification Penalties** section:



All the repayment notifications that are not paid in full after their maturity dates are subject to automated penalty calculation processes performed by Loan Management. Thus, for repayment notifications linked to a contract, Loan Management uses the penalty interests defined within the interests list attached at the banking product level. For repayment notifications which are not linked to a contract, Loan Management uses the penalty interest list specified in the `ManualPenaltyInterestList` system parameter.

Each penalty displays information about the penalized item, the penalty date, the number of penalty notification, the number of overdue days after the repayment notification's maturity date, the penalty amount, a description, and whether the penalty was pad or not.

8. To view more information about a correction entry, double-click it to open the **Edit Contract Penalty Detail** page:

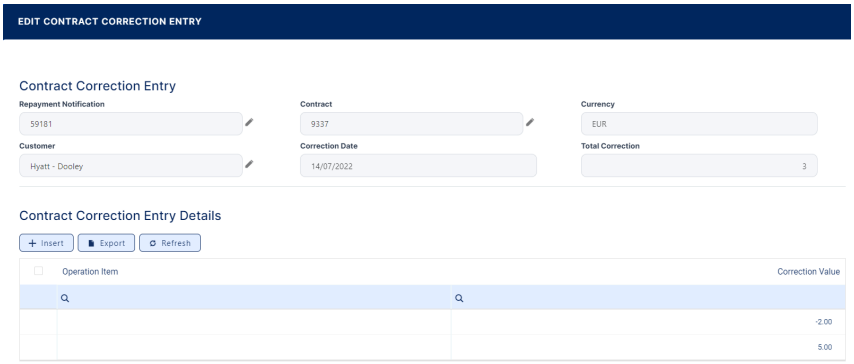


You can't edit any of the fields from this page.

9. View information about any correction entries created for the notification in the **Viewing Corrections** section. Contract correction entries are automatically generated, for notifications that are overdue, when creating a **Reschedule Overdue** transaction type contract event. Here you can see information about the customer of the contract, the date and

currency of the correction entry, and the total amount of the correction (the sum of all the correction details' values).

10. To view more information about a correction entry, double-click it to open the **Edit Contract Correction Entry** page:



- **Repayment Notification** - The repayment notification number.
- **Contract** - The contract number associated with the notification.
- **Currency** - The currency of the notification.
- **Customer** - The customer associated with the notification.
- **Correction Date** - The date when the correction was created.
- **Total Correction** - The sum of all correction entry detail records associated with the current correction entry.

11. Additionally, you can view information about each detail within the correction:

- **Operation Item** - The operation item of the transaction for which the correction entry detail was inserted.
- **Correction Value** - The value of the correction entry detail, in the correction entry's currency.

Understanding Automated Settlement of Repayment Notifications (Direct Debit Settlement Account)

The automated settlement of repayment notification, or direct debit settlement account, is the functionality whereby, if funds are available on the settlement account and the contract has repayment notifications pending for recovery, Loan Management automatically uses the available balance up to full settlement of repayment notifications.

When you have restrictions of any kind on the settlement account or the allocation simply needs to be done as per a legal authority instructions, you can turn off the automated settlement of `Installment` type repayment notifications functionality (the payment allocation) at the **contract level** using the `Direct Debit Settlement Account` checkbox. De-selecting the checkbox leads to the underlying amounts on notifications pending recovery not being retrieved automatically even if there are available funds in settlement account. Thus, financial institution can manage the contracts in case of blocked accounts and control the allocation of funds to outstanding `Installment` type notifications in case of need to impose a block on the settlement account, or manage the settlement of multiple loans from the same settlement account when short on funds and exceptional rules might apply.

This parametrization is available at product level, you can it amended at the contract level, and it is also available at customer level with a system parameter to instruct Loan Management if the customer level setup should impact underlying contracts or not. Thus, you can manage the `Direct Debit Settlement Account` setting at the **customer level**. The customer level setting takes precedence over the setting at the contract level when creating new contracts. For existing contracts, Loan Management applies the setting configured within the `CustomerToContractDirectDebitSettlementAcc` system parameter.

If the automated settlement of repayment notification functionality is turned off, the contract is pending for manual repayment. You can turn it back on as required, when required, and allow Loan Management to allocate the funds according to its automated processes, using any funds that become available in the settlement account in order to cover pending notifications. When the functionality is turned on or off, the notifications already processed remain unchanged. You can turn the automated settlement functionality on or off even after the maturity of a loan, as long as the contract is not closed.

The following validations are performed for the `Direct Debit Settlement Account` field at the contract level:

- If `Direct Debit Settlement Account` = `True` and new `Installment` type repayment notifications are generated, the system automatically tries to recover the values from Settlement Amount. When the repayment notification is fully paid, Loan Management automatically changes the `Installment` type repayment notification's status to `Recovered`.
- If `Direct Debit Settlement Account` = `True` and old unpaid `Installment` type repayment notifications already exist, the system tries to create recover debt records for the remaining amount for all unpaid `Installment` type repayment notifications, and changes their status to `In recovery`.
- If `Direct Debit Settlement Account` = `False` and new `Installment` type repayment notifications are generated, the system doesn't register any debt to recover, and changes the notification's status to `Pending Recover`.

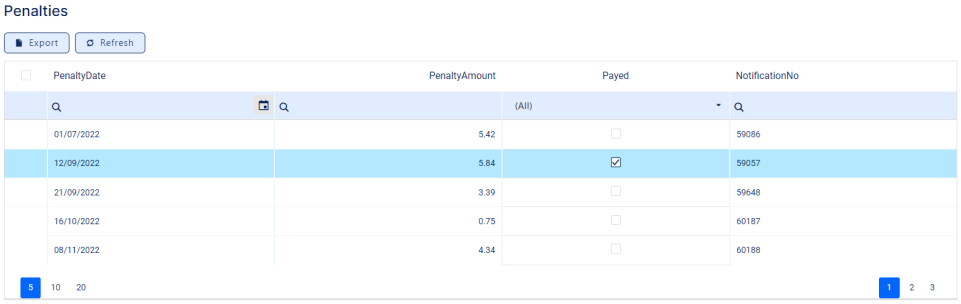
- if `Direct Debit Settlement Account` = `False` and old `Installment` type repayment notifications are generated, the system removes debts to recover from the Settlement Account, and changes the status to `Pending Recover`.

Viewing a Contract's Penalties

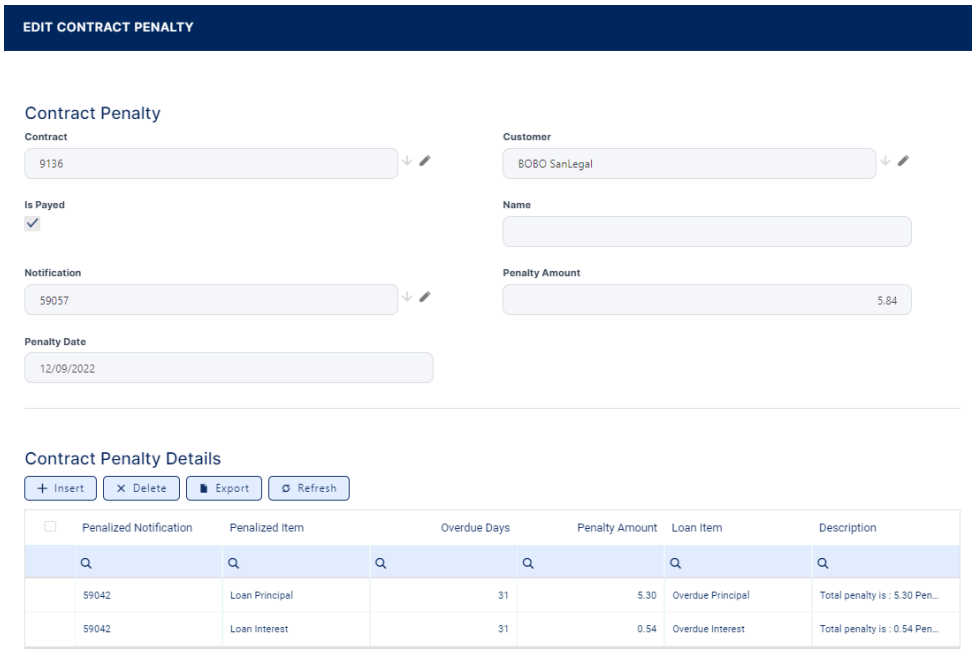
You can view the penalty interest already notified for the contract in the **Penalties** section of the **Payments** tab. These penalties are automatically calculated by Loan Management for an approved contract based on all the interests with selected **Is Penalty** checkbox that are applied to this contract.

To view the penalties applied to a contract, follow these steps:

1. On an approved contract's **Payments** tab, navigate to the **Penalties** section. If any penalty interest was calculated for the contract, they are displayed here:



2. View basic information about the penalties in the list, such as penalty date, amount, notification number and whether it was paid or not. Payed penalties are also highlighted in blue, for your convenience.
3. To see detailed information about one of the applied penalties, double-click on the desired penalty record. The **Contract Penalty** page is displayed with the selected penalty's details:

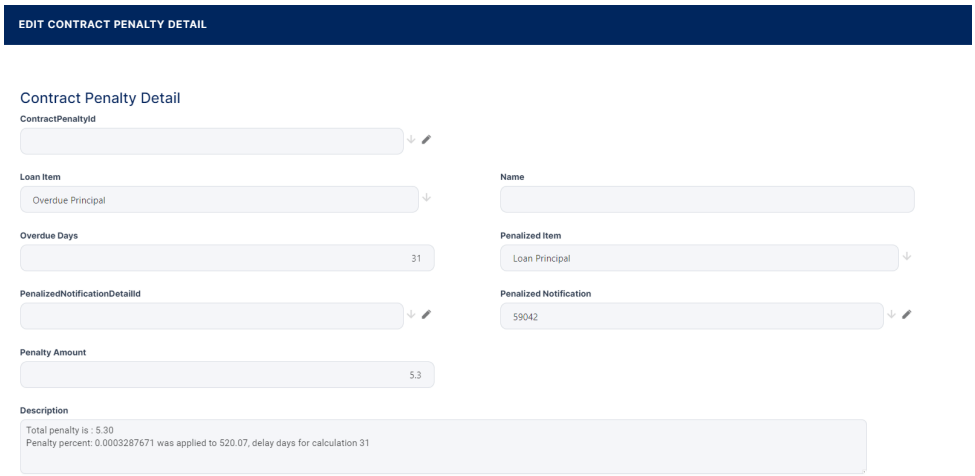


You can't edit the information displayed on this page.

4. View the information in the **Contract Penalty** section, as displayed:
    - **Contract** - The number of the contract for which the penalty is applied.
    - **Customer** - The customer for whom the contract was created.
    - **Is Paid** - A checkbox indicating whether the penalty was already paid through a payment allocation or not.
    - **Name** - The name of the penalty.
    - **Notification** - The number of the notification where the penalty is included.
    - **Penalty Amount** - The amount of the penalty expressed in the contract's currency.
    - **Penalty Date** - The date when the penalty was calculated.
5. View the information in the **Contract Penalty Details** section, as displayed:

- **Penalized Notification** - The notification which was not paid in time and for which the penalty is calculated.
- **Penalized Item** - The item to which the penalty interest was applied.
- **Overdue Days** - The number of days since the notification was overdue for payment.
- **Penalty Amount** - The calculated amount of the penalty.
- **Loan Item** - The loan item which is used to calculate the penalty interest.
- **Description** - The description of the contract penalty detail. It contains the total penalty value, the penalty percent or value applied to the number of overdue, and the delay days for calculation.

6. Double-click a detail record to view the details of the penalty on a separate page, **Edit Contract Penalty Detail**:



You can't edit the information displayed on this page.

Viewing Customer Payments

Loan Management processes payments automatically, so you can't insert payment records manually. Payment information is displayed at the contract level, within the **Payments** tab. For convenience, you can also view each

performed payment, along with its allocation details, on the **Customer Payments** page.

Payment Statuses

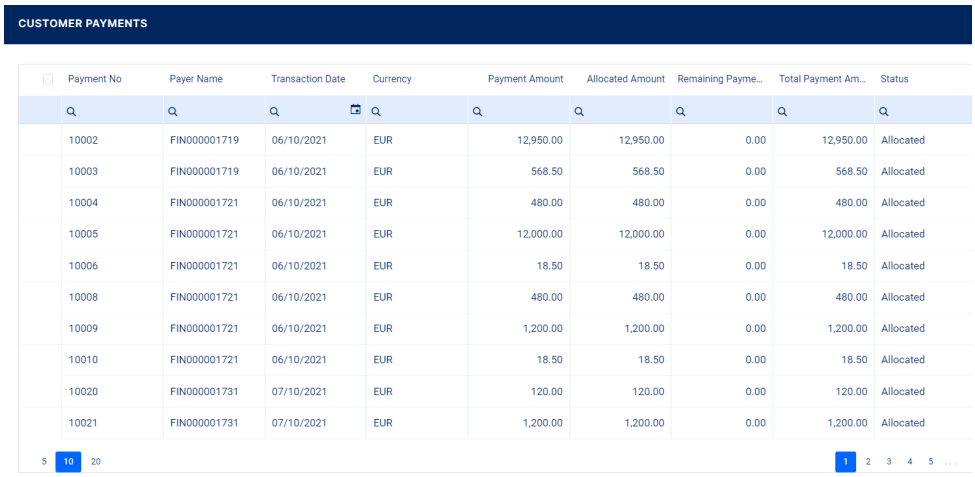
A payment record has the following statuses, visible next to the payment number on the **Payment** page:

- **Draft** - the status of a newly created record that was not yet sent for allocation. The payment amount is not allocated yet as repayment for the contract's repayment notification. You can delete records with this status.
- **Unallocated** - the status of a payment record before its amount gets allocated as payment for a repayment notification.
- **Partially Allocated** - the status of a record after some of its amount gets allocated as payment for a repayment notification.
- **Allocated** - the status of a record after its entire amount gets allocated as payment for a repayment notification. There is no further transition from this status.

> **IMPORTANT!**
> Only payments with **Allocated** status operate changes at the contract repayment notification level.

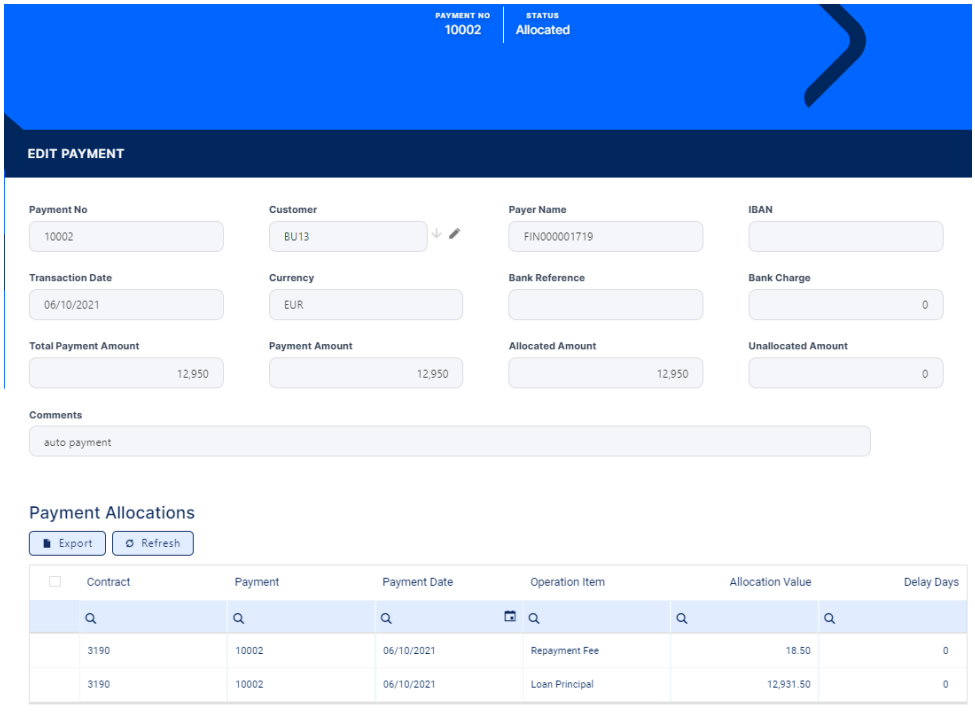
To view a specific payment, follow these steps:

1. In FintechOS Portal, click the main menu icon and expand the **Loan Management Operational** menu.
2. Click **Customer Payments** menu item to open the **Customer Payments** page.



On the **Customer Payments** page, you can search for records, delete a payment in **Draft** status, or open a specific a payment for viewing.

3. Double-click the desired payment on the **Customer Payments** page. The **Edit Payment** page is displayed.



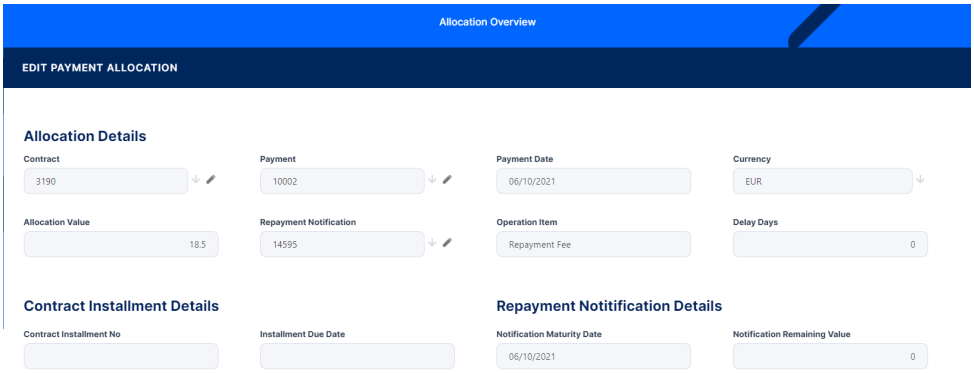
You can only view a payment record, but you can't edit any of the fields of a payment.

4. View the following information about the selected payment:
    - **Payment No** - The number of the payment, as generated by Loan Management.
    - **Customer** - The name of the customer associated with the payment.
    - **Payer Name** - The bank account number from where the payment was performed.
    - **IBAN** - The IBAN of the account where the money is being paid.
    - **Transaction Date** - The date of the payment transaction.
    - **Currency** - The currency of the payment.
    - **Bank Reference** - The bank reference for the payment.
    - **Bank Charge** - The amount charged by the bank for performing this transaction.
    - **Total Payment Amount** - The sum of the payment amount and the bank charge value.
    - **Payment Amount** - The amount of the payment.
    - **Allocated Amount** - The amount that was already allocated as a contract's repayment for a notification for the selected customer.
    - **Unallocated Amount** - The amount that remains to be allocated as a contract's repayment for a notification for the selected customer.
    - **Comments** - Any comments referring to the payment.
5. View the payment allocation information, displayed in the **Payment Allocation** section.



This section is empty for payments in **Draft** or **Unallocated** status. The payment allocations for due repayment notifications are automatically calculated by Loan Management and they are displayed after the record reaches the **Allocated** or **Partially Allocated** status.

6. View more details about each allocation by double-clicking it. he payment allocation record opens in the **Allocation Overview** page, displaying the following information:



- **Contract Id** - The contract for which the payment was allocated.
- **Payment** - The number of the payment.
- **Payment Date** - The date of the payment.
- **Currency Id** - The currency of the payment.
- **Allocation Value** - The value of the allocation.
- **Repayment Notification** - The repayment notification for which the payment was allocated.
- **Operation Item** - The operation item which was repaid with this allocation.
- **Delay Days** - The days passed since the due date.
- **Contract Installment No** - The number of the contract installment.
- **Installment Due Date** - The due date of the installment.

- **Notification Maturity Date** - The notification's maturity date.
- **Notification Remaining Value** - The notification's remaining value to be paid.

> **NOTE**
>
> You can't edit any of the fields of a payment allocation record.

Working with Overdue Loans

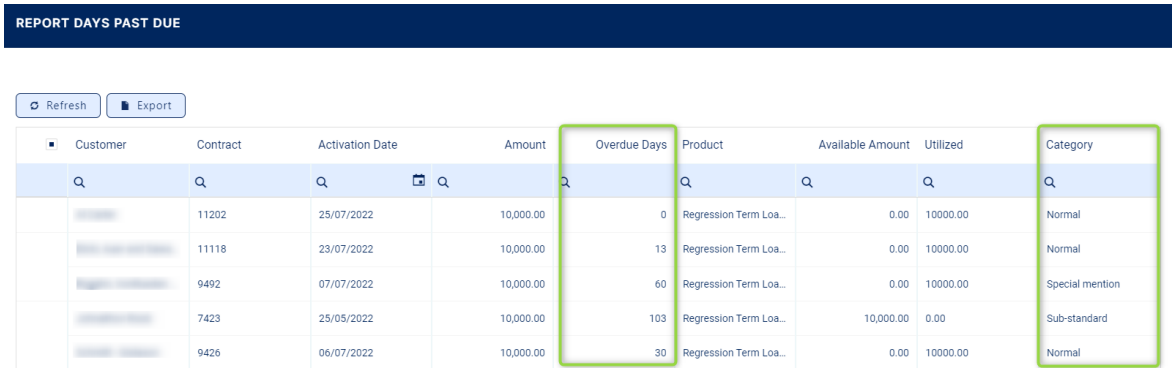
Financial institutions classify their existing loan contracts based upon the days past due (DPD), the number of days passed since repayment due date without fully repaying the due amount for the oldest unpaid repayment notification. In order to comply with the risk method calculation, the DPD (days past due) value is calculated as the number of days between the contract's due date and the current system date of Loan Management. The financial institutions can apply different provision percentages for principal or for interest for each contract, based on this classification: the higher the delay period, the higher the provision percentage applicable and the risk classification.

In Loan Management, the loan classification works by risk contamination at the customer and the group levels. This means that if a loan contract belonging to a customer is classified as one of a higher risk due to delays in the repayment process, all the other loans of the customer and of the group where the customer is a member are further classified into that high-risk classification. The risk classification of loan contracts is automatically performed by the `Update Loan Classification (CB)` scheduled job based on the loan classification records' definition. Read about managing loan classification records in the Loan Classification topic.

Loan Management uses two system parameters that help you manage contracts with DPD:

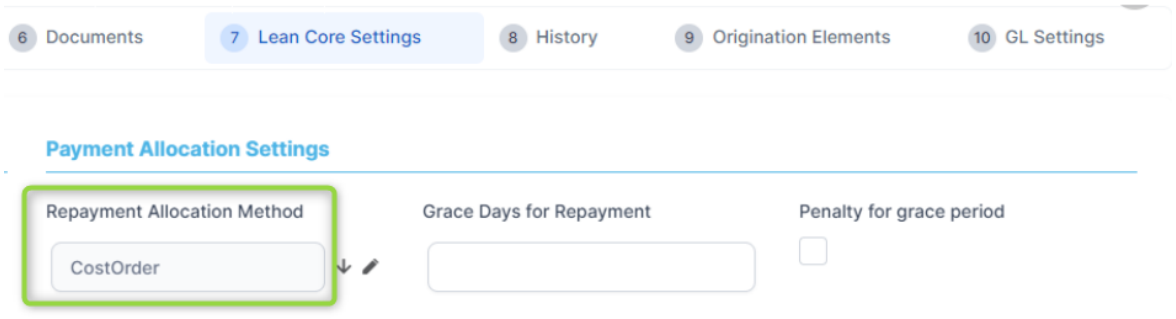
- `UseContaminationForDPDCategory` - this parameter specifies whether Loan Management should use the risk contamination for loan classification or not;
- `DelayDaysForBlockNewContractApproval` - this parameter controls the default number of delay days for blocking the approval of new loan contracts for customers who have overdue payments.

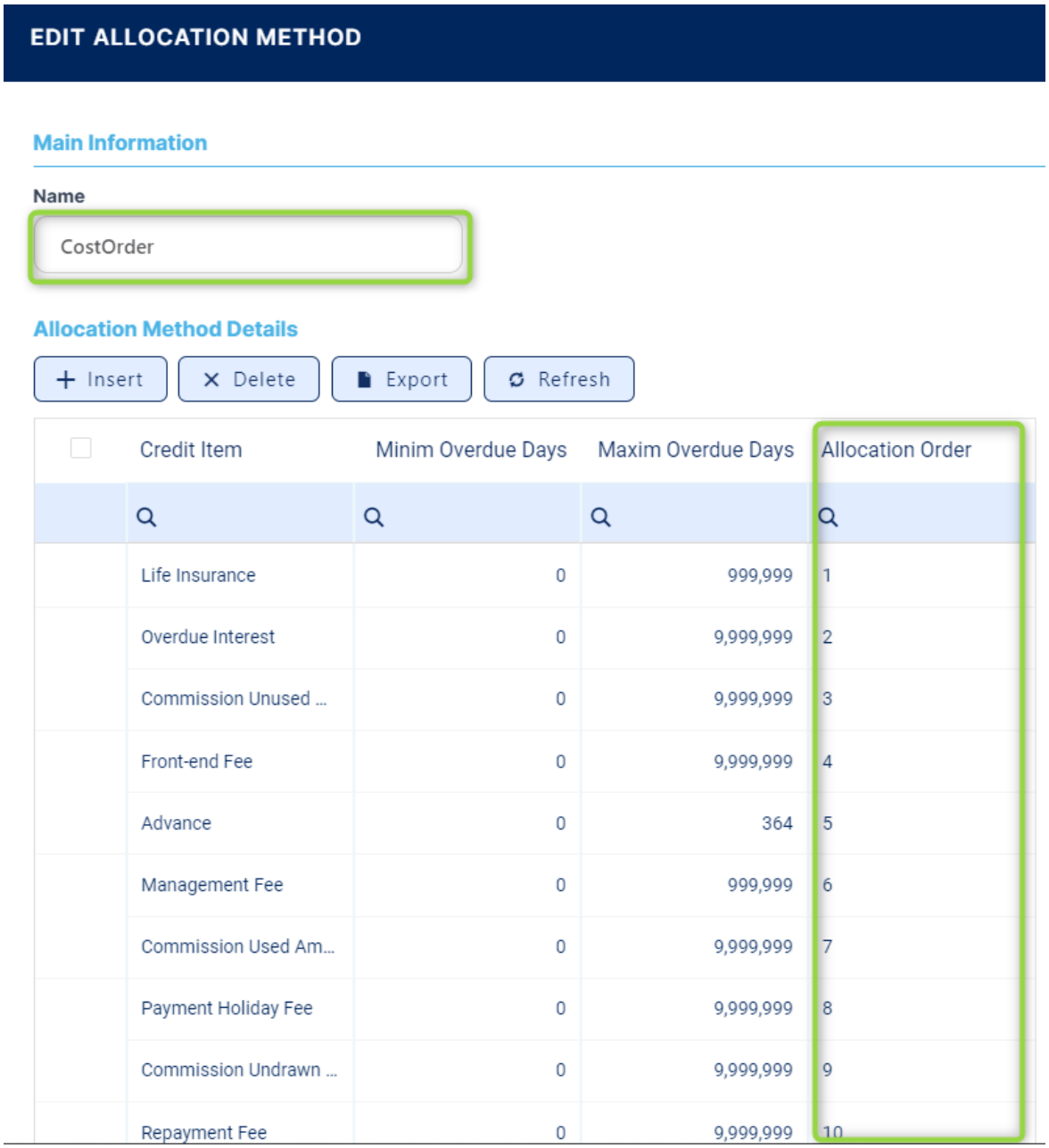
You can view the contracts with DPD in the dedicated **Days Past Due** report, accessible through Loan Management's **Reports** menu. Double-click any of the contracts from the report to open it for editing. The report displays the contracts with overdue repayment notifications, along with information about the number of overdue days and the contract's classification based on the DPD:



You can also extract the information about overdue repayment notifications through API integration, using the `GetDataSourcePastDueInstallmentsReport` endpoint.

The allocation of funds for repayment notification is performed according to the cost allocation method defined at the banking product definition level, in the product's **Lean Core Settings** tab -> **Payment Allocation Settings** section, as described in the Banking Product Factory user guide:





NOTE

In order to avoid having to deal with overdues, you can perform payment holiday transactions. If you already have overdues, then perform reschedule overdues transactions on the contracts. Both transactions are usually part of the risk management/ collection departments' policies and can be proactively implemented by the bank, or on the customer's demand.

Applying Payment Holiday to a Loan

The payment holiday represents taking a break of any number of installments for the generated repayment schedule of a loan. You can apply payment holiday for the principal alone or for both interest and principal of a loan contract. This has been a functionality in demand lately during the COVID pandemic, being a regulatory requirement for financial institutions. After applying a payment holiday transaction to a loan contract, Loan Management recalculates the repayment schedule, rebuilding the schedule projection automatically based on the instructions you provided.

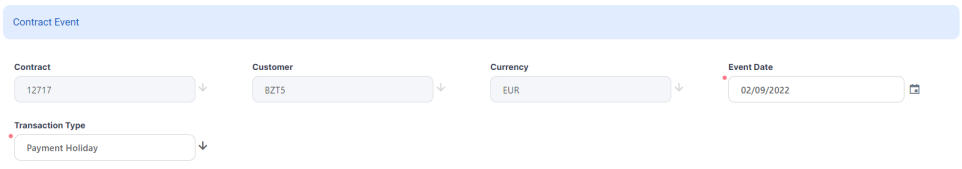
> **NOTE** If you are referring to payment holiday as a grace period, then you should read the "Working with Grace" on page 290 topic. For that sort of grace, after the initial granting of the loan, Loan Management has dedicated a transaction that can be enabled via product definition. It does pretty much the same just during the life of the loan, not at the very beginning of it.

Adding a Payment Holiday Transaction To an Approved Contract

You can add payment holiday transactions to an approved contract via Loan Management's user interface or through API calls, using the Loan Management endpoints. Read more about these endpoints in the Loan Management Developer Guide.

In order to add a payment holiday transaction to a loan contract through the menus available in Loan Management, follow these steps:

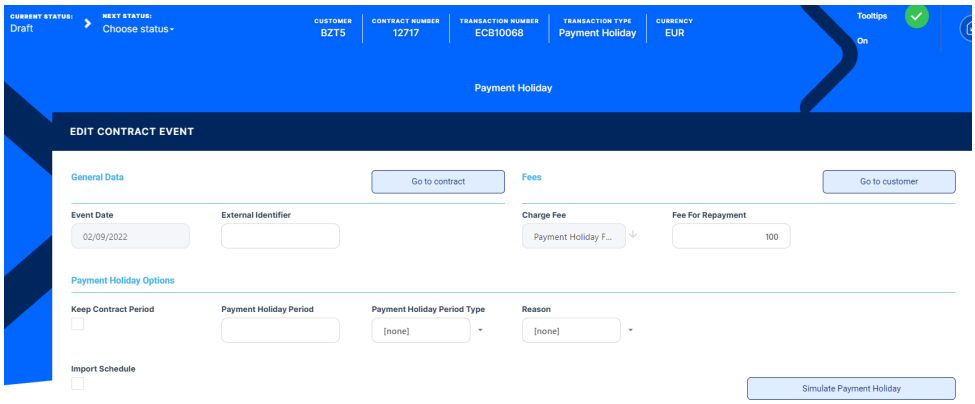
1. In FintechOS Portal, select a contract with **Approved** status and double-click to open it.
2. Navigate to the contract's **Payments** tab and click the **Insert** button above the **Transactions** section. The **Event** page is displayed.



3. Fill in the following fields:
   - **Event Date** - This is pre-filled with current date.
   - **Transaction Type** - Select from the list the **Payment Holiday** transaction type. If you can't find it, then the transaction type is not associated with the banking product which is at the base of the contract.

   Other values are automatically completed: contract, customer, and currency.

4. Click the **Save and Reload** button.
   The event is saved in **Draft** status and a transaction number is automatically generated for it. The **Edit Contract Event** page corresponding to the selected transaction type is displayed. A series of value fields are automatically calculated, their values are displayed, and you can't edit them: the contract's financed and available amounts, the selected installment calculation method, the repayment day ant the contract's tenor.

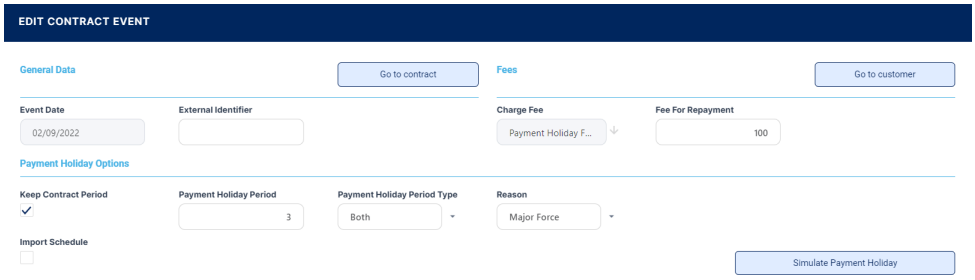


5. Fill in the **external identifier** of the transaction, if available.

6. View the **fee for repayment** or the **repayment fee percent**, depending on which is displayed. The fee value or the percentage are pre-filled by Loan Management according to the **Charge Fee** defined for this transaction type. Depending on the `ManualRepaymentFee`Loan Management system parameter's value, the system may allow you to change the fee or the percentage. See Transaction Fees for more

details.

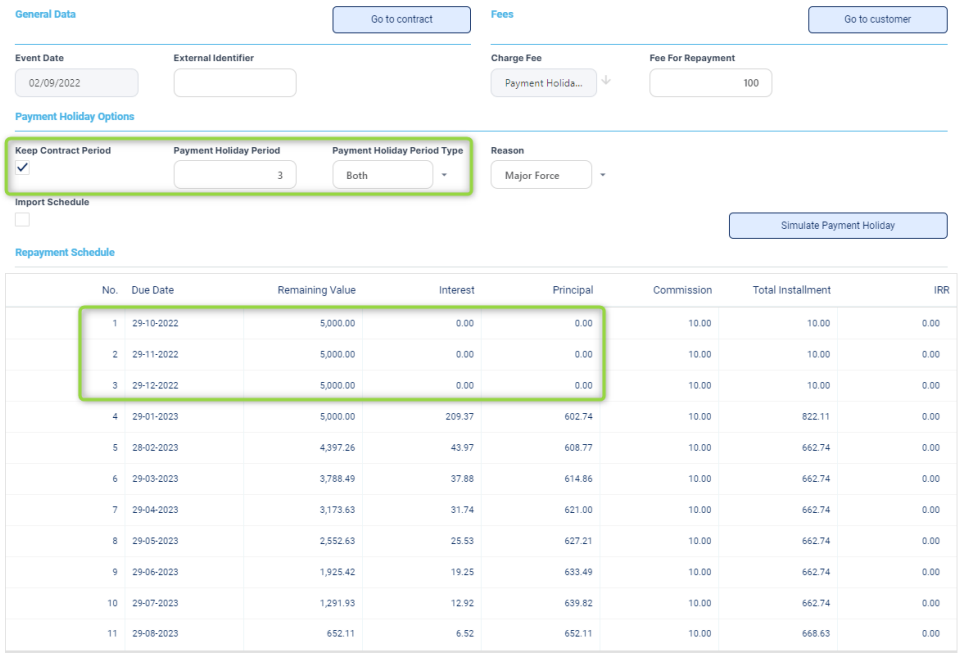
7. Set the payment holiday options:
    - **Keep Contract Period** - Select this checkbox to instruct Loan Management to keep the period of the contract. If left unselected, the contract period is recalculated.
    - **Payment Holiday Period** - Enter the number of months for which you request a break from the payments.
    - **Payment Holiday Period Type** - Select the type of payment holiday to be applied for the contract:
        - `Both` - take a break from paying the principal and interest amounts of the installments.
        - `Principal` - take a break from paying the principal amount of the installments.
    - **Reason** - Select the reason for requesting the event: activity suspended, major force, or financial restructuring.



8. Decide whether you want to use the repayment schedule calculated by Loan Management through automatic processes, or you want to import a custom schedule. For custom schedule, select the **Import Schedule** checkbox. Read more about importing a custom schedule file in the Manually Upload Repayment Schedules section of the user guide.
9. Click the **Save and Reload** button.
   If the event value meets the business requirements defined within Loan Management, the event is saved. Otherwise, an error message appears. Change the values as instructed in the message and try saving the event again.
   While the event is in **Draft** status, you can modify all the event's fields

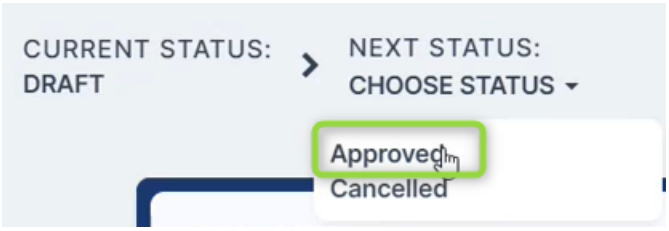
except **Transaction Type**. The event value is not applied to the contract while the event is still in this status.

10. Click the **Simulate Payment Holiday** button to view the details of each installment of the calculated repayment schedule.



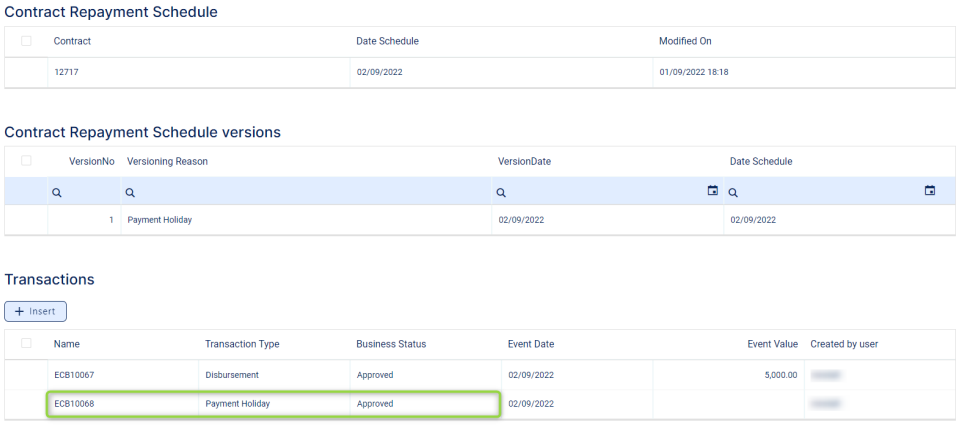
No.	Due Date	Remaining Value	Interest	Principal	Commission	Total Installment	IRR
1	29-10-2022	5,000.00	0.00	0.00	10.00	10.00	0.00
2	29-11-2022	5,000.00	0.00	0.00	10.00	10.00	0.00
3	29-12-2022	5,000.00	0.00	0.00	10.00	10.00	0.00
4	29-01-2023	5,000.00	209.37	602.74	10.00	822.11	0.00
5	28-02-2023	4,397.26	43.97	608.77	10.00	662.74	0.00
6	29-03-2023	3,788.49	37.88	614.86	10.00	662.74	0.00
7	29-04-2023	3,173.63	31.74	621.00	10.00	662.74	0.00
8	29-05-2023	2,552.63	25.53	627.21	10.00	662.74	0.00
9	29-06-2023	1,925.42	19.25	633.49	10.00	662.74	0.00
10	29-07-2023	1,291.93	12.92	639.82	10.00	662.74	0.00
11	29-08-2023	652.11	6.52	652.11	10.00	668.63	0.00

11. Approve the event by changing its status to **Approved** in the upper left corner of the **Event** page.



12. Confirm the change of status in the **Confirmation** window, clicking **Yes**. The event is now in **Approved** status and Loan Management applies the recalculated repayment schedule to the contract, displaying the previous schedule version in the **Contract Repayment Schedule Versions** section of the contract's **Payments** tab.

    The transaction is visible in the **Transactions** section.



> **NOTE**
> All existing versions of the contract in **Contract Version Draft** status are automatically changed to **Contract Version Closed** when a payment event is approved for that contract.

Working with Grace

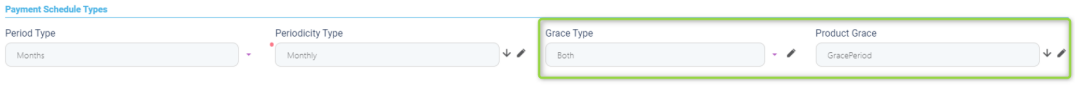
You can apply a grace period to a loan contract, so that the customer starts repaying the loan later, effectively moving the due installments in the future with a number of months (or as per periodicity). You can set the grace period to apply for the loan's principal, the loan's interest, or both.

> **NOTE** If you wish to apply a grace period during the life of the contract, then you should read the "Applying Payment Holiday to a Loan" on page 286 topic.

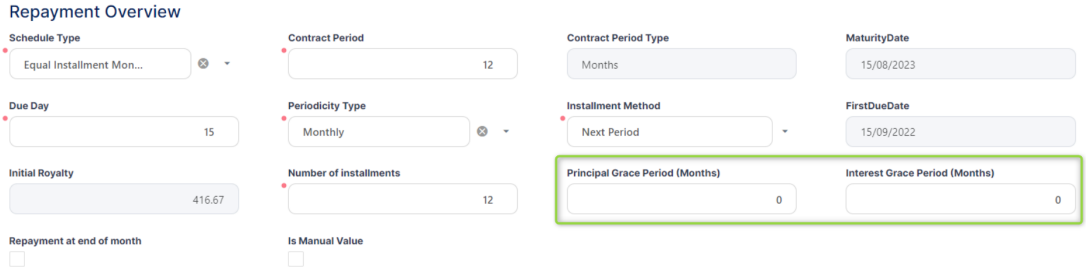
Loan Management uses the grace concept in two different contexts, at the loan level or the installment level. The grace period settings must be applied at the banking product definition level, as follows:

- **Loan grace** - This is negotiated as part of the approval of a loan contract and is granted at the beginning of the loan, for a given number of installments during which the borrower pays only interest or nothing at all. This one has an impact on the schedule projection. Set the loan grace at the banking product definition

level, in the **Details** tab's **Payment Schedule Types** section, as described in the Banking Product Factory user guide.



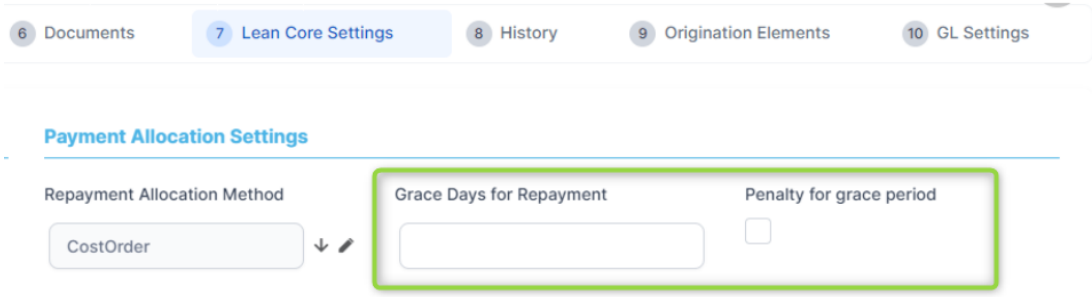
The loan grace setting are exposed at contract definition level for you to customize according to the customer's needs, within the Repayment Overview section:



> **NOTE**
>
> You can change the grace period only until the first installment is notified.

- **Installment grace** - This is when you set a number of days allowed for the borrower to settle their notified amounts. Suppose the contract's schedule has its due date every month on the 15th and you want to allow for 5 days for the amounts to be recovered. If the amounts are not recovered in the 5 days, you also have the option to recalculate the penalty interest, if such interests exist on the contract, starting from the initial due date of 15th, or start applying it starting from 20th of the month. This one does not have an impact on the schedule projection, but on how and when to consider the amounts as overdue, and it relates to the notifications processing. These settings must be set at the banking product level, within the **Lean Core Settings**' tab **Payment Allocation**

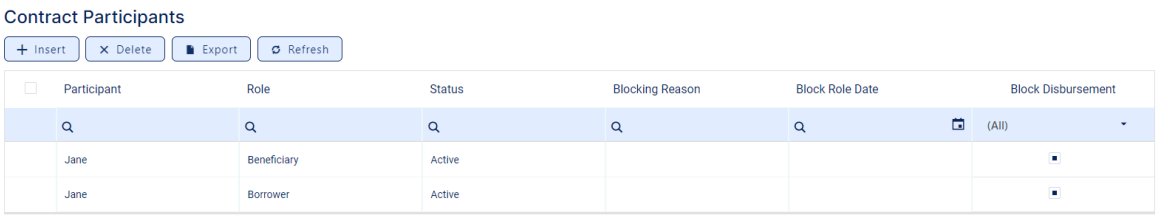
Settings section, as described in the in the Banking Product Factory user guide.



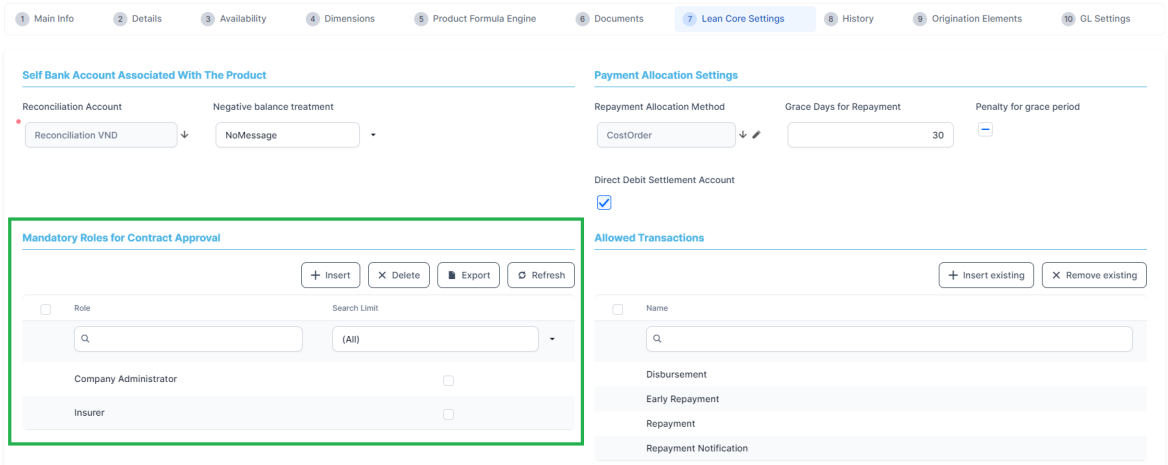
Working with Participants

The participants to a contract are those legal or individual persons who have a role to play during the life-cycle of the contract. They can be the person who borrows the funds, the actual beneficiary of the funds, the company administrator of the legal person, a notary, and so on. Another example are the agents, brokers, insurers, or merchants who participate in contracts as third-party entities, and they may get commissions according to third-party agreements. They must be recorded in contracts as contract participant with the specified role in order to qualify for the commissions stated in the agreement pricing records added to an agreement.

While creating a contract, Loan Management automatically populates the **Contract Participants** section within the **Overview** tab of the contract with the customer's information as both Borrower and Beneficiary of the funds, for loan contracts. If the customer is a legal entity, all the company's already entered legal representatives such as administrators, affiliates, owners, or other key contact persons are displayed in this list. In the **Contract Participants** section, you can add other participants to the contract, like Guarantors, Co-Debtors, etc, even after approval, delete, block, or export customers who participate in a contract.

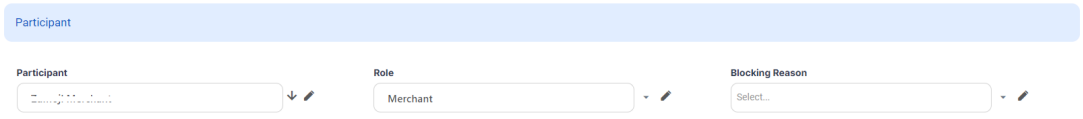


There may be cases when some roles are mandatory for a product. If there is a mandatory role defined in the banking product definition, Loan Management displays an error on trying to approve the contract without a customer mentioned in the contract with that specific role.



Adding Participants

1. To add a participant, click **Insert** in the **Contract Participants** section of contract in `Draft, Version Draft`, or `Approved` status.
2. On the newly displayed **Participant** page, fill in the following fields:



   - **Participant** - Select from the list the name of the customer who can access the contract.
   - **Role** - Select from the list the role in the contract of the previously selected customer.
   - **Blocking Reason** - Leave this empty if you don't want to limit the customer's access to the contract.
3. Click the **Save and Close** button.

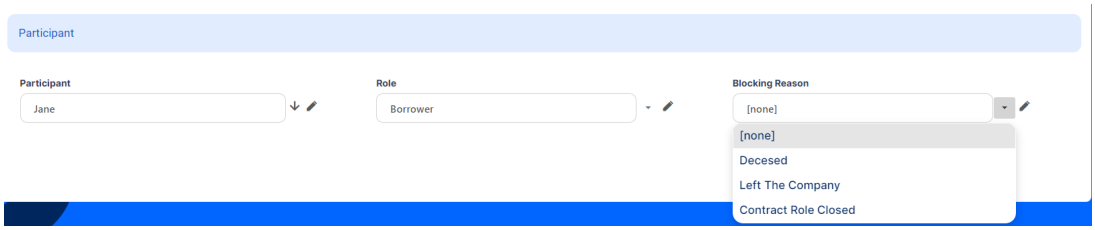
IMPORTANT!
For legal entity customers, add the participant with the `Company Administrator`

role, otherwise, the loan contracts cannot be approved. This is not the case for current account contracts.

Blocking Participants

If you need to block an existing participant's access to the contract for various reasons, such as the person left the company who is the beneficiary of the contract, follow these steps:

1. Double-click an existing participant in the **Contract Participants** section of contract in `Draft, Version Draft`, or `Approved` status.
2. On the displayed **Participant** page, in the **Blocking Reason** field, choose the reason for blocking the selected participant from accessing the contract.

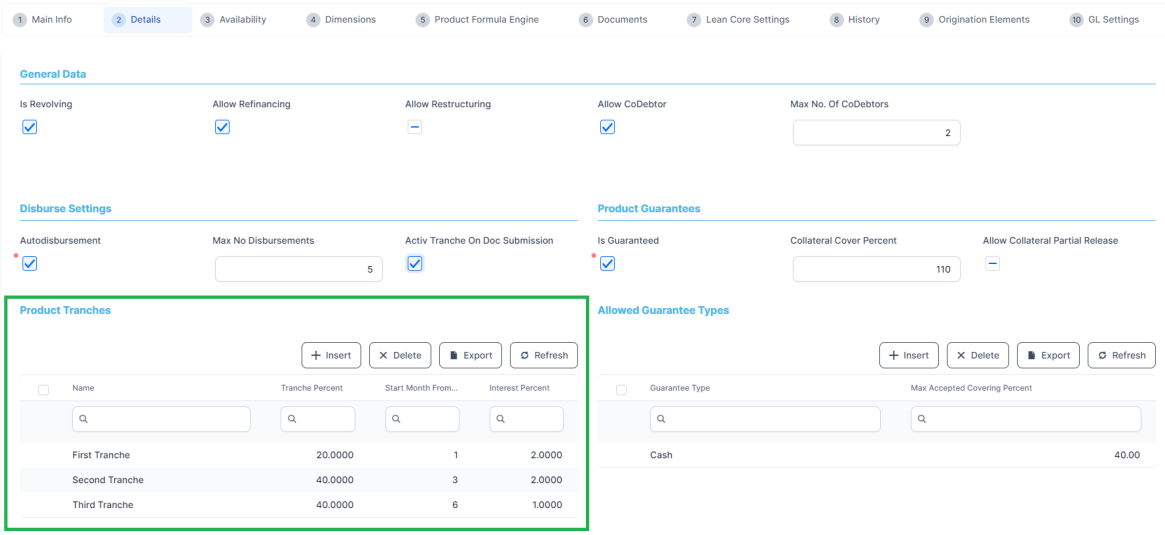


3. In the **Block Role Date** field, select the starting date for blocking the participant's access to the contract.
4. Select the **Block Disbursement** checkbox to instruct Loan Management to stop disbursements on the contract, if needed.
5. Click the **Save and Close** button.

Working with Tranches

Tranches represent multiple disbursements performed from a loan's financed amount, which allow you to implement progressive access to the funds. This is valuable in case of loans granted for investment projects where you can know upfront that there is a plan for the project and payments need to happen for each stage of the project, those stages being known from the start. Tranches are usually used for corporate loans, such dividend payments or cash outflows are not done in equal amounts and usually trigger a recalculation in terms of interest.

The disbursement tranches are configured at the product level.

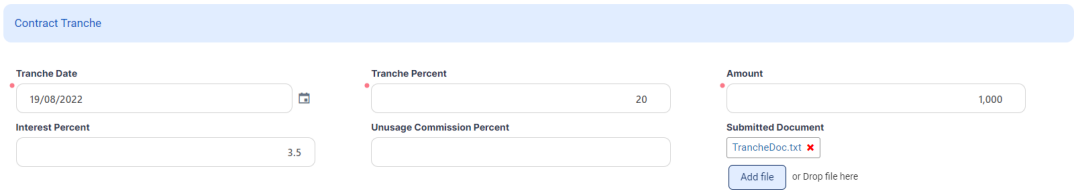


At contract creation, Loan Management brings the tranche information from the product level. In the **Contract Tranches** section of the contract's **Overview** tab, you can also insert, delete or export disbursement tranches for the contract.



Adding Tranches to a Contract

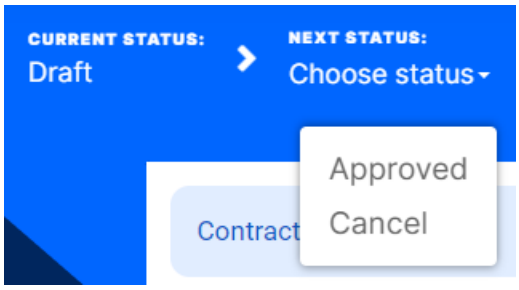
1. To add a tranche in a contract based on a product that has a disbursement matrix set up, click **Insert** and fill in the following fields:



- **Tranche Date** - Select the date of the disbursement tranche.
- **Tranche Percent** - Enter the percentage from the contract value that has to be disbursed with this tranche, or allow Loan Management to calculate it based on the **Amount** value.

- **Amount** - Enter the amount from the contract value that has to be disbursed with this tranche, or allow Loan Management to calculate it based on the **Tranche Percent** value.
- **Interest Percent** - Enter the interest percent applicable for this tranche if it must be different from the interest rate applicable for the entire contract.
- **Unusage Commission Percent** - Enter the commission percent applicable for the unused loan amount from this tranche.
- **Submitted Document** - Upload the documents related to the tranche disbursement.

2. Click the **Save and Reload** button. If the product has no disbursement matrix, then Loan Management triggers an error. If Loan Management performs all validations successfully, then the tranche record is saved in **Draft** status.
3. To activate the tranche, change the tranche's status to **Approved**. Note that you can't modify the details of an approved tranche.



4. Click the **Save and Close** button.

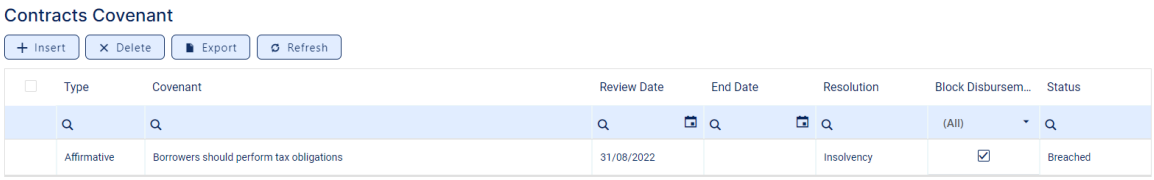
   Each day, Loan Management runs a specialized job to disburse Approved tranches, and the amount recorded in the tranche is disbursed in the contract's destination account. Alternatively, if you want to process the disbursement right-away, you can run the `Activate Tranche (CB)` job.

Working with Covenants

The covenants are conventions that applicants must abide by after the approval of a contract. Such conventions are usually applicable for corporate clients that must meet certain requirements in order to continue to receive disbursements and not only:

submit balance sheet every x months, have account turnover of at least x percent from average monthly turnover, provide other relevant documents from authorities. Covenants are configured at the product level.

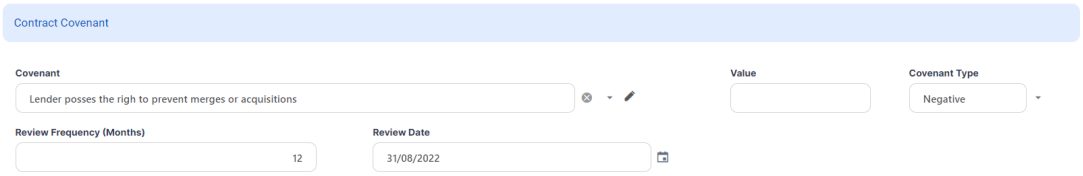
While creating a contract, Loan Management brings the covenants to the contract level, in the **Contract Covenant** section of the **Overview** tab. There you also add, delete or export covenants for the contract.



Upon adding a covenant to a contract, you must activate it. After approving the contract, when it reaches the covenant's review date, you must perform the review of the covenant. If the conditions are not met, then you can mark the covenant for blocking further disbursements of the contract. Further implementation is needed if you want automatic processes to take care of contracts with breached covenants.

Adding & Activating Covenants

1. To add a covenant to a contract, click **Insert** in the **Contracts Covenant** section of a contract in `Draft` or `Version Draft` status.
2. On the newly displayed **Contract Covenant** page, fill in the following fields:



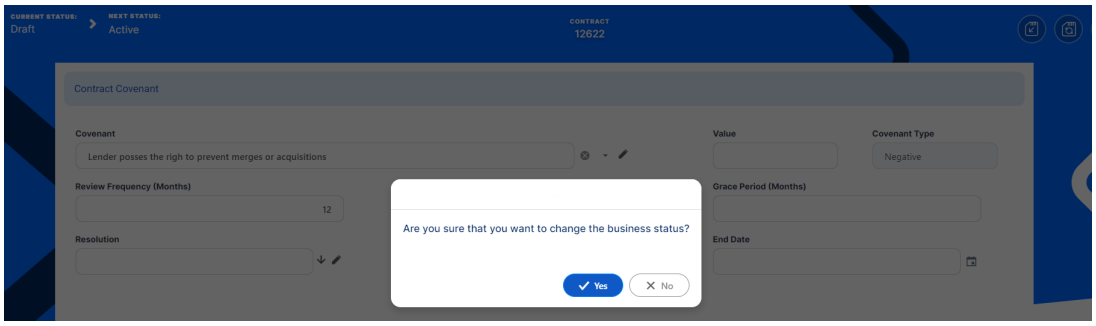
- **Covenant** - Select the desired covenant from the list of possible values:
  - `Borrowers should perform tax obligations`: the lenders expect the borrowers to perform their tax obligations to both the business and towards their employees. This covenant is of type affirmative.

- `Lender can monitor borrower's current ratio`: the lender may continuously monitor the borrower's current ratio to ensure it stays relatively attractive and promising. This covenant is of type financial.
- `Lender posses the right to prevent merges or acquisitions`: a clear stipulation that the lender possesses the right to prevent merges of acquisitions without proper notification or full knowledge of the process. This covenant is of type negative.

  Loan Management automatically fills in the covenant type.

- **Value** - Enter the numeric value of the covenant, if applicable.
- **Review Frequency (Months)** - Enter the number of months after which the covenant has to be reviewed.
- **Review Date** - Enter the date when the covenant has to be reviewed.

3. Click the **Save and Reload** button. The covenant is displayed in the list of covenants in the **Contracts Covenant** section, in `Draft` status.
4. Activate the covenant record by changing its status to **Active** and confirming your action.



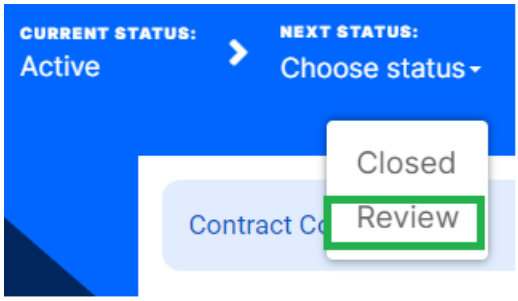
5. Click the **Save and Close** button. The covenant's status changes to `Active`.

Reviewing Covenants

Loan Management allows you to add details about the process of reviewing a covenant for an approved contract.

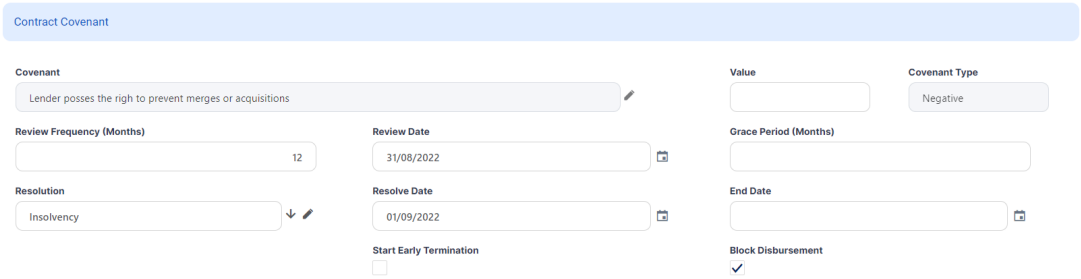
1. To review an active covenant for an approved contract, double-click the desired covenant in the **Contracts Covenant** section of the contract's Overview tab.

2. On the newly displayed **Contract Covenant** page, change the covenant's status to **Review** and confirm your action.



The covenant's status changes to Review and the page reloads with new fields.

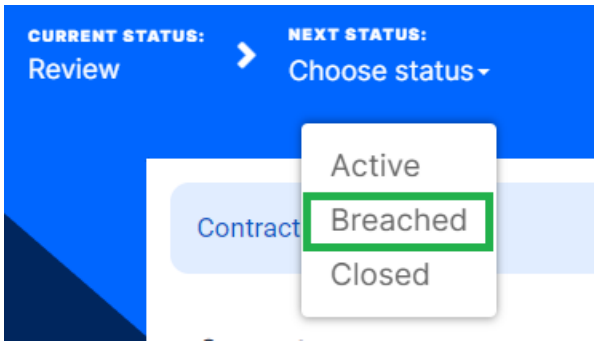
3. Fill in the following fields with the results of the covenant review process:



- **Grace Period (Months)** - Enter a grace period in month for the fulfillment of the covenant, if needed.
- **Resolution** - Select from the list the actual resolution of the covenant. Add a new covenant resolution, if you can't find a match in the list.
- **Resolve Date** - Enter the date when the covenant is considered as resolved.
- **End Date** - Enter an end date for the covenant, if needed.
- **Start Early Termination** - If the covenant's terms are not met, then you can check this field to mark the covenant for contract early termination.
- **Block Disbursement** - If the covenant's terms are not met, then you can check this field to mark the covenant for blocking further disbursements of the contract.

4. Click the **Save and Reload** button.

5. If the covenant's terms are met, change the covenant's status to **Active** and confirm your action.
   If the covenant's terms are not met, change the covenant's status to **Breached** and confirm your action.



6. Click the **Save and Close** button. The covenant's status changes to `Active` or `Breached`, according to your previous choice.

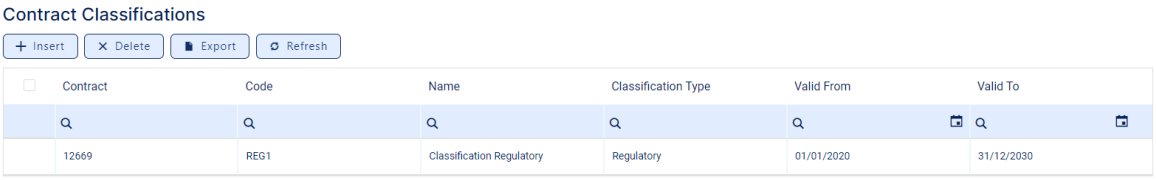
> **NOTE** Further implementations are needed in order for Loan Management to manage contracts with breached covenants if you need actions enforced at the contract level.

Working with Contract Classification

Financial institutions may classify their contracts for organization purposes, or to mark some contracts as to belonging to a specific category or another. Loan Management brings the classifications defined at the product level to the contract level when creating a contract.

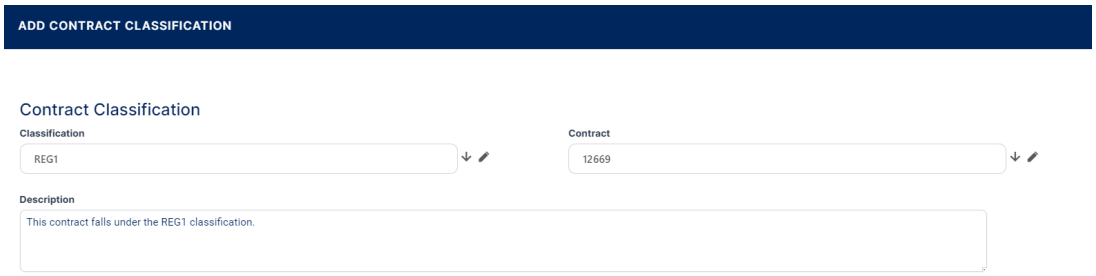
> **NOTE**
> For information about the **automatic loan classification** performed by Loan Management based on DPD, please read the "Loan Classification" on page 49 topic.

You can manage a contract's classification within the **Contract Classifications** section on the **Overview** tab. Here you can insert, delete or export classifications for the contract.



Adding Classifications to a Contract

1. To add a classification to a contract, click **Insert** in the **Contract Classifications** section of a contract.
2. On the newly displayed **Add Contract Classification** page, fill in the following fields:



   - **Classification** - Select the desired classification for the contract from the list of classifications associated with the banking product.
   - **Description** - Enter a description for the contract classification.

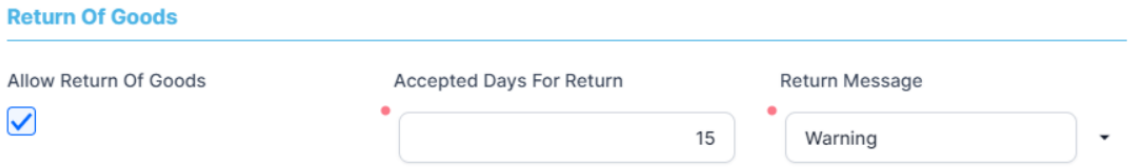
3. Click the **Save and Close** button.

Working with Returns

To manage merchandise return in a contract based on a BNPL-type banking product or even a return of funds in a different kind of loan, no matter if the return is partial or full, Loan Management allows you to perform an early repayment and decide if the repayment amount is excluded from interest calculation, as well as decide how to treat potentially claimed interest at the moment of performing the early repayment. This feature was created in context of BNPL where you can return goods but can also apply for mortgages when the deal drops and the solicitor returns the funds.

The Returned Amounts or Goods transaction triggers an early repay and reconciles/ gives back any interest if collected for that specific amount so far, as well as all or part of the upfront fee. The recalculation of the repayment schedule covers the recalculation of interest and the mitigation of potentially already charged/ notified interest amounts. Loan Management can capture the return for any amount no matter if the disbursement of the loan was done in one or multiple transactions. Even if there are overdue payments on the contract, the principal can be decreased and also the overdue notifications overdue are adjusted to reflect the early repayment, if the date of processing is before the notification date. The transaction only accepts `Return Fee` commission types. Upon transaction approval, a new contract version is automatically created.

Whether a contract allows or not Returned Amounts or Goods transactions must be defined at the banking product level, within the **Lean Core Setting**s' tab **Return of Goods** section, as described in the Banking Product Factory user guide.

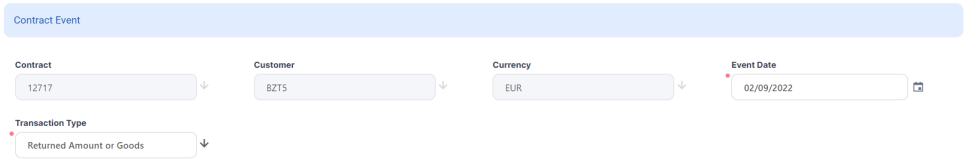


Adding a Returned Amount or Goods Transaction To an Approved Contract

You can add Returned Amounts or Goods transactions to an approved and disbursed contract via Loan Management's user interface or through API calls, using the Loan Management endpoints. Read more about these endpoints in the Loan Management Developer Guide.

In order to add a Returned Amounts or Goods transaction to a loan contract through the menus available in Loan Management, follow these steps:

1. In FintechOS Portal, select a contract with **Approved** status and double-click to open it.
2. Navigate to the contract's **Payments** tab and click the **Insert** button above the **Transactions** section. The **Event** page is displayed.

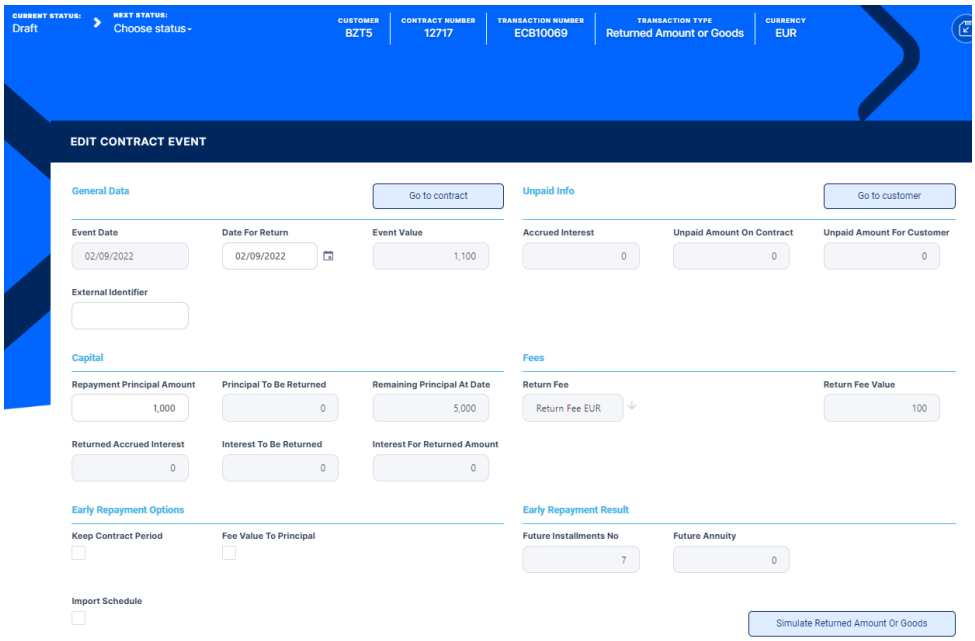


3. Fill in the following fields:

   - **Event Date** - This is pre-filled with current date.
   - **Transaction Type** - Select from the list the **Returned Amounts or Goods** transaction type. If you can't find it, then the transaction type is not associated with the banking product which is at the base of the contract.

   Other values are automatically completed: contract, customer, and currency.

4. Click the **Save and Reload** button.
   The event is saved in **Draft** status and a transaction number is automatically generated for it. The **Edit Contract Event** page corresponding to the selected transaction type is displayed. A series of value fields are automatically calculated, their values are displayed, and you can't edit them: the contract's financed and available amounts, the selected installment calculation method, the repayment day ant the contract's tenor.

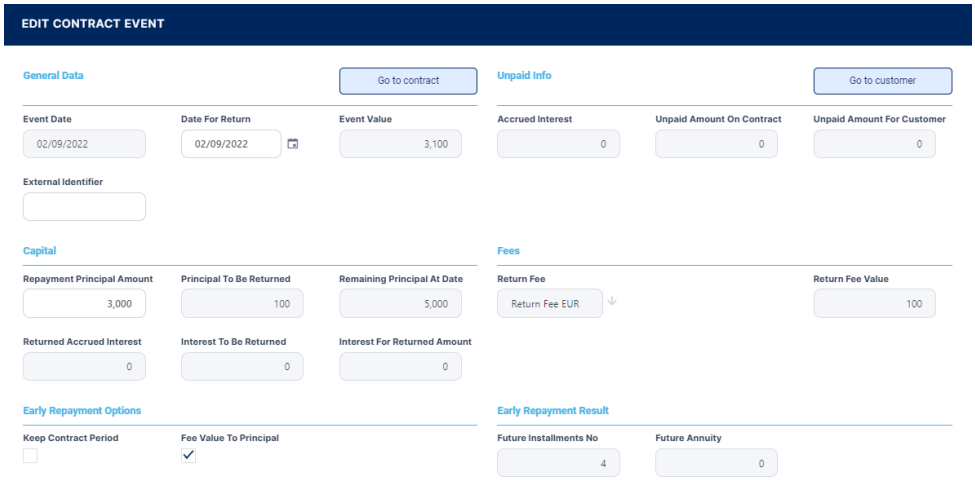


5. In the **General Data** section, fill in the **external identifier** of the transaction, if available.
   The **date for return** is pre-filled and it can't be before `Activation Date` and after `Current Date`.
   The **event value** represents value of the transaction, calculated and displayed after saving the record, based on the information filled in a series of other fields: `Event Value = Repayment Principal Amount + Commission Value + Interest To Be Returned`.
6. In the **Unpaid Info** section, you can only view the information about:
   - **Accrued Interest** - The interest accrued up until the event date for an early repayment contract event.
   - **Unpaid Amount On Contract** - The value of the unpaid amount still on the contract.
   - **Unpaid Amount For Customer** - The value of the unpaid amount for the customer.
7. In the **General Data** section, fill in the **Repayment Principal Amount** with the amount from the Principal that the customer wishes to return. You can only view the following information:

- **Remaining Principal At Date** - The remaining value of the principal at the current date.
- **Returned Accrued Interest** - The returned accrued interest.
- **Interest For Returned Amount** - The interest for the returned amount.
- **Interest To Be Returned** - The interest to be returned with this event.

8. In the **Fees** section, you can only view the information about:
   - **Return Fee** - The transaction fee applicable for a Returned Amount or Goods transaction on this contract. See Transaction Fees for more details.
   - **Return Fee Percent** - The return fee percentage applicable for the contract, if the return fee is set up as a percentage.
   - **Return Fee Value** - The return fee value applicable for the contract.
9. In the **Early Repayment Options** section, select the **Keep Contract Period** checkbox if Loan Management should keep the period of the contract.
10. Select the **Fee Value To Principal** checkbox to indicate that the value of the return fee should be applied to the principal, using its value to include it in the repayment amount and diminishing the outstanding principal.
11. In the **Early Repayment Result** section, you can only view the information about:
    - **Future Installments No** - the number of installments to be paid in the future. This depends on whether you opted to keep the contract period as it was or not.
    - **Future Annuity** - the future value of the installment as recalculated after this payment.

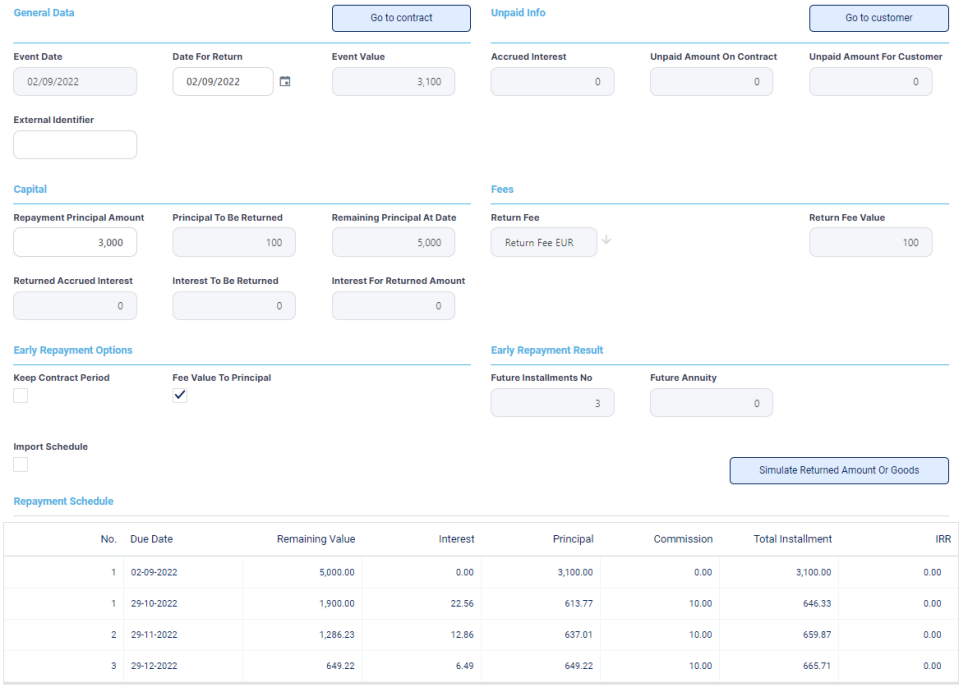
- **Future Principal For Installment** - the future value of the principal as recalculated after this payment.



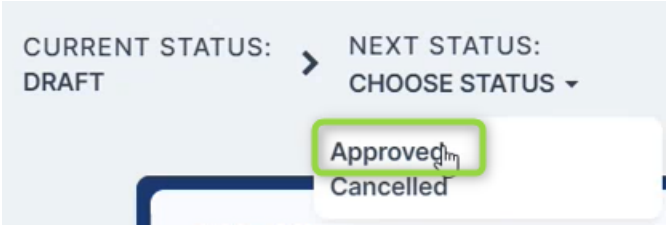
12. Decide whether you want to use the repayment schedule calculated by Loan Management through automatic processes, or you want to import a custom schedule. For custom schedule, select the **Import Schedule** checkbox. Read more about importing a custom schedule file in the Manually Upload Repayment Schedules section of the user guide.

13. Click the **Save and Reload** button.
If the event value meets the business requirements defined within Loan Management, the event is saved. Otherwise, an error message appears. Change the values as instructed in the message and try saving the event again.
While the event is in **Draft** status, you can modify all the event's fields except **Transaction Type**. The event value is not applied to the contract while the event is still in this status.

14. Click the **Simulate Returned Amounts or Goods** button to view the details of each installment of the calculated repayment schedule.

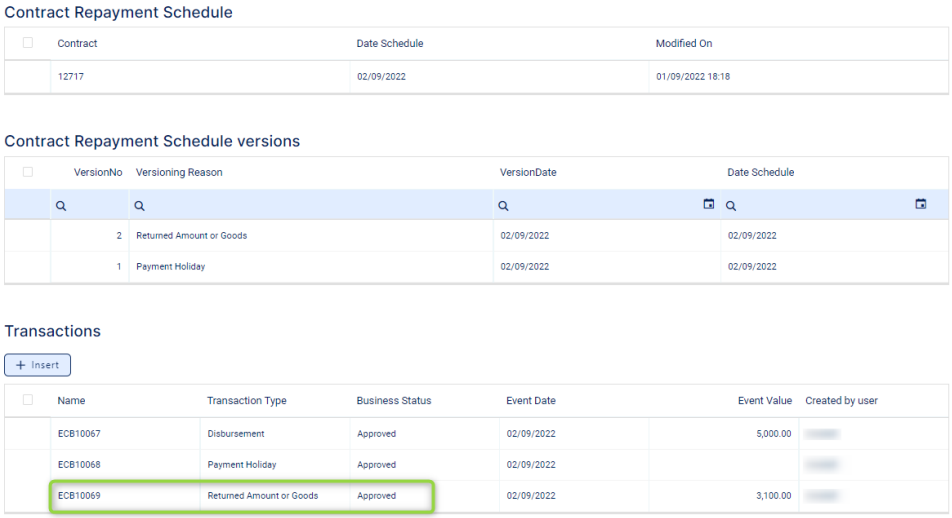


15. Approve the event by changing its status to **Approved** in the upper left corner of the **Event** page.



16. Confirm the change of status in the **Confirmation** window, clicking **Yes**. The event is now in **Approved** status and Loan Management applies the recalculated repayment schedule to the contract, displaying the previous schedule version in the **Contract Repayment Schedule Versions** section of the contract's **Payments** tab.

    The transaction is visible in the **Transactions** section.



NOTE
All existing versions of the contract in **Contract Version Draft** status are automatically changed to **Contract Version Closed** when a payment event is approved for that contract.

Automatic Calculations of Value Fields

Legend:

- *RA* = Returned amount/ Returned value of the goods
- *IFRV* = Interest for remaining value after returning the amount/ returning the value of the goods
- *IFRA* = Interest for returned amount starting from return date
- *RVA* = Remaining value after returned amount/ returned value of the good
- *RVBA* = Remaining value before returned amount/ return value of the good

`Principal To Be Returned` = if *RA > RVBA => RA - RVBA*, else 0.

`Interest To Be Returned` = if *IFRA > IFRV=>IFRA -IFRV*, else 0.

Interest on the first installment generated by the transaction: if *IFRV > IFRA => IFRV - IFRA*, else 0.

Returned Amount Or Goods Event Validations

On event approval, Loan Management verifies if `Event Date` respects the formula: `Activation Date + Accepted Days For Return >= Current Date`. If the formula is not respected, Loan Management returns:

- an error message, if `Return Message = Error` on the banking product definition;
- a warning message, if `Return Message = Warning` on the banking product definition;
- nothing, if `Return Message = NoMessage` on the banking product definition.

Loan Management also checks on event approval if the `Return Fee` is not greater that the sum of commissions with type `Front-End Fee` with `Is Returnable = True` on the contract level.

There is no validation of the event amount on this transaction related to balance of current account.

After Event Validation

After event validation, Loan Management creates a new version of the contract. The contract's current account (customer casa account) is topped-up TOP Up with the `Event Value = Amount + Return Fee`. The return fee is added to the contract's **Fees & Commissions** section with a negative value that can't be modified.

In the Contract Repayment Schedule, Loan Management creates an installment with all columns = 0, except Principal, Remaining Value and Total Installment, with `Is Early Repayment = True` and `Is Return = True`. The Principal amount is:

- the principal amount saved on the transaction if `Fee Value To Principal = False`;
- the event amount saved on transaction if `Fee Value To Principal = True`.

Loan Management generates a notification for this installment and performs the allocation. The allocation process affects the main bank account balance as it is now, all the limits with `Is Revolving = True` and `On Repayment = True` affected by this contract, and the available amount on the contract if `Is Revolving = True` at the product level.

Applying Fees and Commissions

The financial institutions take commissions and fees for offering a product or service such as opening an account, for cash withdrawals, for transfers, for making payments in certain countries, for exchanging currencies, for emitting debit cards, for handling documents etc. These commissions are set at the product level and vary from institution to institution, based on their policy.

In the **Fees & Commissions** section within the **Overview** tab of the contract, you can view all the fees and commissions configured at the product level that have the `Automatic Load on Contract` checkbox set to `True`. After the first saving operation, Loan Management display all the fees that are defined as values. The fees defined as percentages are displayed after completing all the values of the contract. Read more about the commissions automatically inserted and calculated in the below section. You can also add, delete or export fees and commissions for the contract.



Automatic Insertion and Calculation of Commissions

Loan Management automatically inserts/ updates commissions in the **Fees & Commissions** section depending on the life cycle and status of the contract:

- **Creating a new contract**: Loan Management automatically inserts active commissions associated to the banking product, within their defined validity period, with `Automatically load on contract = True`, with `Is For Unusage = False`, and `Commission value is percentage = False`.
  If `Commission value is percentage = True`, then the commission is only inserted if the amount value was previously inserted.
- **Updating a contract in `Draft` status**: Loan Management automatically inserts active commissions associated to the banking product, within their defined validity period, with `Automatically load on contract = True`, with `Is For Unusage = False`. If a commission with `Commission value is percentage = True` was already inserted, then the commission's value is updated according to the contract's financed amount. If the value of a commission with `Commission value is percentage = True` was manually modified (for negotiable commissions), then the new value is calculated based on the modified percentage.
- **Creating a new version for a contract**: Loan Management automatically inserts all the commissions already present in the contract. Additionally, all commissions specifically created for contract version (`Is For Contract Version = True`) are added as well.

> **NOTE**
>
> If a version for a contract is created more than once on the same day, then all commissions with `Is For Contract Version = True` that were not notified yet for each previous version are deleted. At the end of the day, there is only one commission for the latest version.

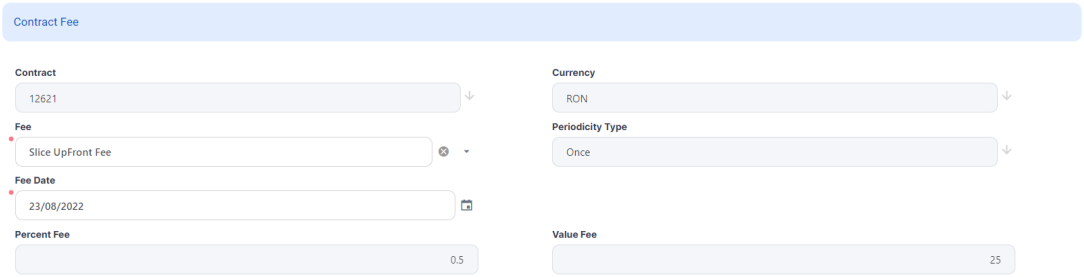
- **Updating a contract in `Contract Version Draft` status**: Loan Management only updates the percentage commissions that are not already notified.

For percentage commissions (with `Commission value is percentage = True`), the financed amount of the contract is used to calculate the commission value based on the percentage. The calculation method differs depending on the contract type:

- For contracts based on **Term Loan,Mortgage** or **Overdraft** banking products:
  - If the commission is applied to amount, then the financed amount = amount due;
  - If the commission is applied to financed amount, then the financed amount = amount due - advance amount;
  - If the commission is applied to remaining value and the contract is in `Contract Version Draft` status, then financed amount = (-1) * main bank account balance. If the result is a negative value, then financed amount = null. In all the other cases, financed amount = null, which is the default value.
- For contracts based on **Bank Account with Overdraft** banking products:
  - If the commission is applied to overdraft limit amount, then the financed amount = overdraft limit amount;
  - If the commission is applied to used amount and the commission's period type is `Once`, then the financed amount = overdraft limit amount - available amount for overdraft. In all the other cases, financed amount = null, which is the default value.

Adding Fees

1. To add a fee for this contract, click **Insert** in the **Fees & Commissions** section of a contract in `Draft` or `Version Draft` status.
2. On the newly displayed **Contract Fee** page, fill in the following fields:



- **Fee** - Select a commission from the list of commissions defined for the banking product used when creating the contract.
- **Fee Date** - Specify which value of the commission is to be used by selecting the date of the commission.

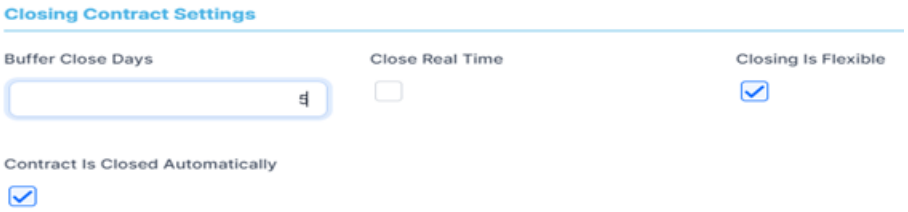
3. Optionally, check the rest of the fields, automatically filled in by Loan Management: contract number, currency, periodicity type of the selected fee, the fee percentage or value applicable for the selected date. You can't change these values.
4. Click the **Save and Close** button.

Closing a Loan With All Obligations Met

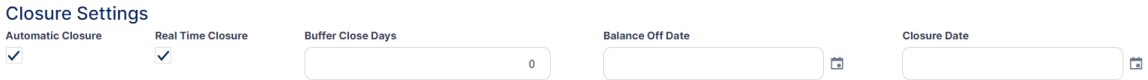
Loan contracts with all their financial obligations met can be closed. Loan Management enables you to close these contracts automatically through scheduled jobs or manually, according to a series of settings defined at the banking product and at the contract level.

There are cases when you might expect the loan to get closed once all amounts recovered and the loan is not revolving, or you might want such contracts to be closed after a certain number of days, allowing for possible reconciliations, or even leave them to be manually closed or with a localized job. All this is enabled from product level and, if set as negotiable, you can also change the default at contract level. You might need such settings if you work with direct debit and need to allow for the number of days the direct debit can bounce to pass before you really close the deal.

You can configure the closure settings during product definition, in the **Lean Contract Settings**' tab -> **Closing Contract Settings** section, as described in the Banking Product Factory user guide:



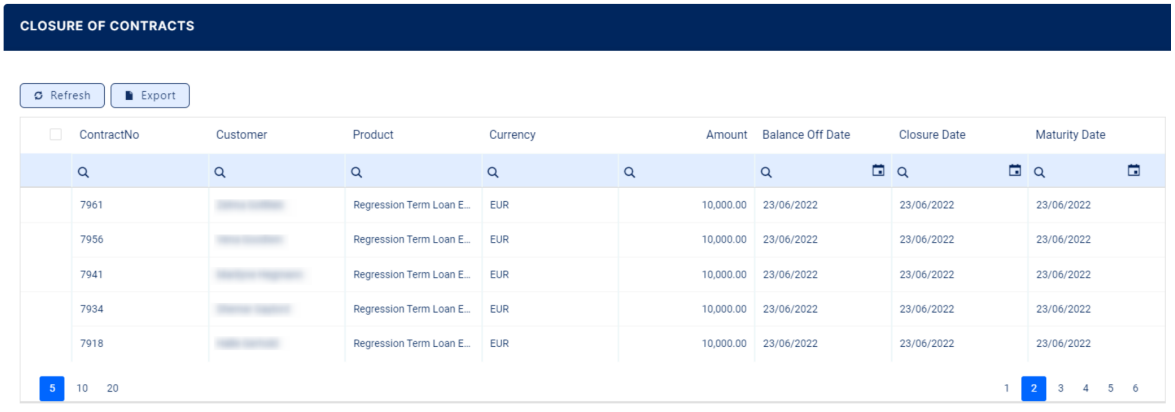
If allowed from the product definition, you can amend the closure settings of the contract, the way Loan Management should behave once the loan is repaid and the contract can be closed. Perform these configurations in the **Closure Settings** section of the **Overview** tab, during contract creation, for contracts based on banking products having the `Closing Is Flexible = True` setting:



Depending on the real time closure setting, Loan Management uses the one following scheduled jobs to close the contracts automatically:

- `Close Contracts (CB) Job` - this job closes automatically all contracts with `Automatic Closure = True` and `Real Time Closure = False`, with zero available amount and with no further amounts to be recovered, that have `Balance Off Date` filled in and `Closure Date = Current Date`.
- `Close Contracts RealTime(CB) Job` - this job closes automatically all contracts with `Automatic Closure = True` and `Real Time Closure = True`, with zero available amount and with no further amounts to be recovered.

You can see the list of contracts that are ready to be closed in the **Closure of Contracts** report:

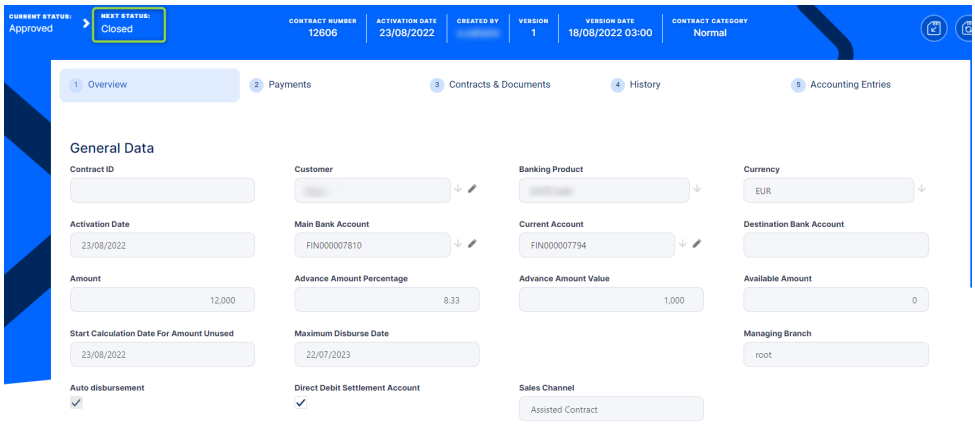


You can also use the `GetClosureOfContracts` endpoint to fetch the same information within your own API integration.

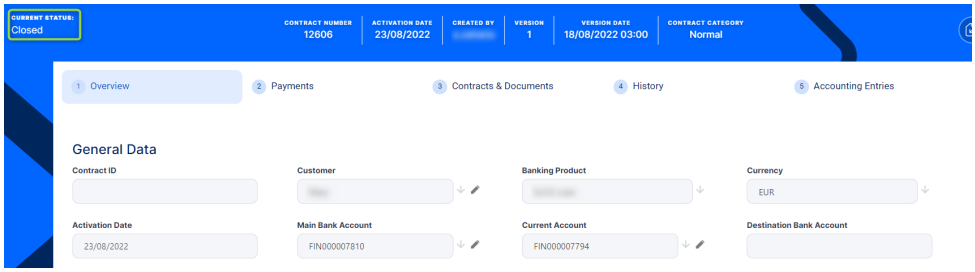
Manually Closing a Contract

If you opted to close a contract with all the obligations met manually, and not automatically, then follow these steps:

1. Double-click an approved contract with zero amounts to be recovered, opening it for editing.
2. Change the contract's **Next Status** into **Closed**.

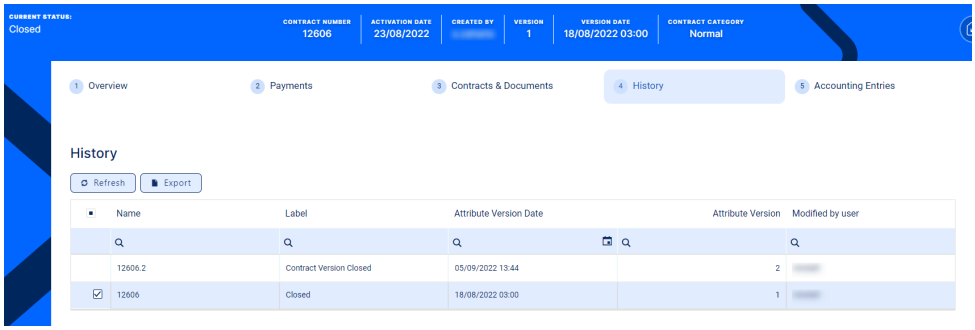


If Loan Management performs all the validations and finds that the financial obligations are met and there are no more amount to be recovered, then the contract's status becomes **Closed**. You can't perform any other operations on this contract.



Any existing versions of the contract are also automatically closed, as

you can see in the **History** tab.



Rescheduling and Refinancing Loans

Loan Management allows you to reschedule the overdue amounts of a loan contract, or to perform an early repayment of the due amounts, effectively refinancing the loan. This page covers the steps you must follow when performing these transactions on a loan contract.

Rescheduling the Overdue Amounts for a Contract

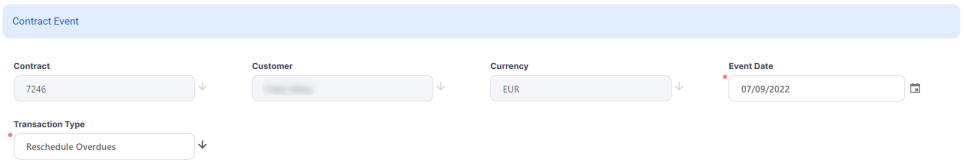
A reschedule overdues transaction represents an operation where overdue installments are merged to the following installments, turning the remaining amounts on the notifications into capital, and they are no longer collecting penalties. Loan Management also recalculates the repayments schedule.

You can add reschedule overdues transactions to an approved contract via Loan Management's user interface or through API calls, using the Loan Management endpoints. Read more about these endpoints in the Loan Management Developer Guide.

In order to add a reschedule overdues transaction to a loan contract through the menus available in Loan Management, follow these steps:

1. In FintechOS Portal, select a contract with **Approved** status with notified overdue amounts and double-click to open it.

2. Navigate to the contract's **Payments** tab and click the **Insert** button above the **Transactions** section. The **Event** page is displayed.

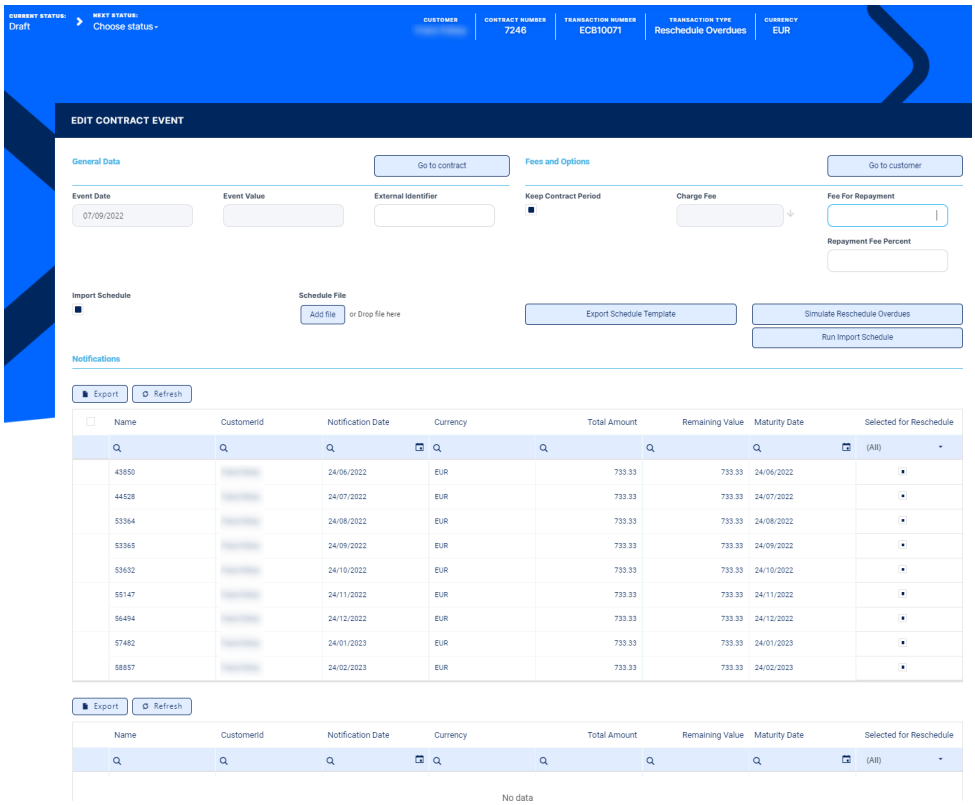


3. Fill in the following fields:

   - **Event Date** - This is pre-filled with current date.
   - **Transaction Type** - Select from the list the **Reschedule Overdues** transaction type. If you can't find it, then the transaction type is not associated with the banking product which is at the base of the contract.

   Other values are automatically completed: contract, customer, and currency.

4. Click the **Save and Reload** button.
   The event is saved in **Draft** status and a transaction number is automatically generated for it. The **Edit Contract Event** page corresponding to the selected transaction type is displayed.



5. Fill in the **external identifier** of the transaction, if available.
6. Decide whether to keep the period of the contract or recalculate it. Select the **Keep Contract Period** checkbox to keep the period.
7. View the transaction fee applicable for the transaction, if it was defined, displayed in the **Charge Fee** field.
8. View the **fee for repayment** or the **repayment fee percent**, depending on which is displayed. The fee value or the percentage are pre-filled by Loan Management according to the **Charge Fee** defined for this transaction type. Depending on the `ManualRepaymentFee`Loan Management system parameter's value, the system may allow you to change the fee or the percentage. See Transaction Fees for more details.
9. Decide whether you want to use the repayment schedule calculated by Loan Management through automatic processes, or you want to import a custom schedule. For custom schedule, select the **Import Schedule**

checkbox. Read more about importing a custom schedule file in the Manually Upload Repayment Schedules section of the user guide.

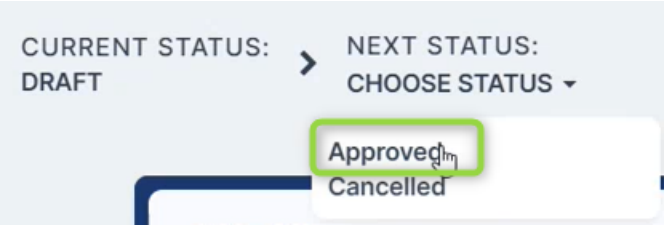
10. Select from the **Notifications** list the overdue payment notifications that you wish to reschedule.



11. Click the **Save and Reload** button.
If the event value meets the business requirements defined within Loan Management, the event is saved. Otherwise, an error message appears. Change the values as instructed in the message and try saving the event again.
While the event is in **Draft** status, you can modify all the event's fields except **Transaction Type**. The event value is not applied to the contract while the event is still in this status.

12. Click the **Simulate Reschedule Overdues** button to view the details of each installment of the calculated repayment schedule.

13. Approve the event by changing its status to **Approved** in the upper left corner of the **Event** page.



14. Confirm the change of status in the **Confirmation** window, clicking **Yes**. The event is now in **Approved** status and Loan Management applies the recalculated repayment schedule to the contract, displaying the previous schedule version in the **Contract Repayment Schedule Versions** section of the contract's **Payments** tab.

    The transaction becomes visible in the **Transactions** section.

> **NOTE**
> All existing versions of the contract in **Contract Version Draft** status are automatically changed to **Contract Version Closed** when a payment event is approved for that contract.

Refinancing a Loan By Performing an Early Repayment

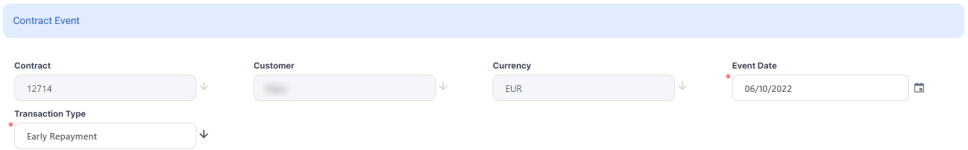
An early repayment transaction represents the early return of funds previously borrowed from a lender. Loan Management also recalculates the repayments schedule. Early repayments can result in a decrease of term while keeping the monthly installment, or a decrease of installment amount while keeping the term. You can also perform early repayments with the collection of interest accrued to date or leaving it until the next regular due date. When you insert the early repayment principal amount, make sure that you also have available funds for the interest accrued to date in case you want to collect it, as well as any transactional fee, for cases when you have an early repayment fee.

> **NOTE**
> Various activities can be orchestrated according to the internal procedures of the financial institution. If the refinancing takes place in another bank, then it's possible that the amounts are released only after the proof of closing the other loan. Some financial institutions may refinance Principal + Costs, others only Principal and so the customer must cover from their own funds the interests and potential fees.
> All these must be orchestrated during the implementation process, before actually starting to use Loan Management, if they must be automated, otherwise they can be implemented as internal procedures.

You can add early repayment transactions to an approved contract via Loan Management's user interface or through API calls, using the Loan Management endpoints. Read more about these endpoints in the Loan Management Developer Guide.

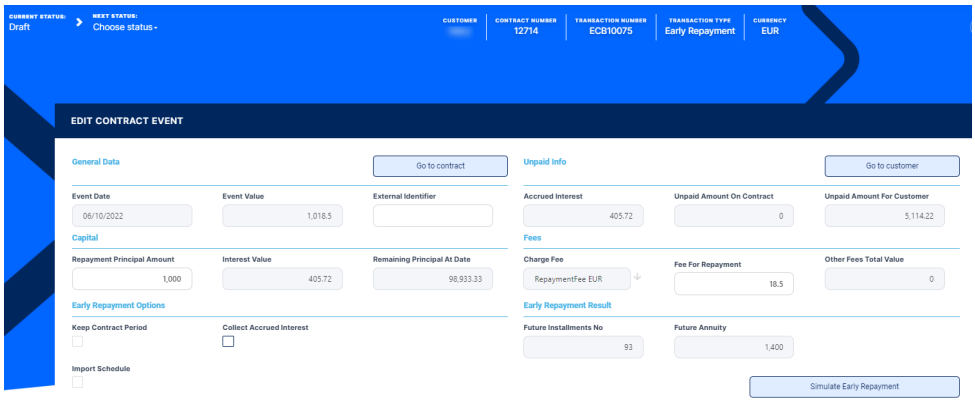
In order to add an early repayment transaction to a loan contract through the menus available in Loan Management, follow these steps:

1. In FintechOS Portal, select a contract with **Approved** status, already disbursed, and double-click to open it.
2. Navigate to the contract's **Payments** tab and click the **Insert** button above the **Transactions** section. The **Event** page is displayed.



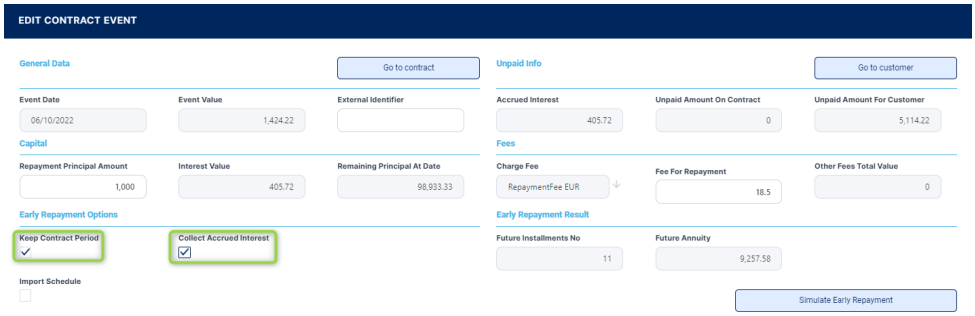
3. Fill in the following fields:
   - **Event Date** - This is pre-filled with current date.
   - **Transaction Type** - Select from the list the **Early Repayment** transaction type. If you can't find it, then the transaction type is not associated with the banking product which is at the base of the contract.

   Other values are automatically completed: contract, customer, and currency.
4. Click the **Save and Reload** button.
   The event is saved in **Draft** status and a transaction number is automatically generated for it. The **Edit Contract Event** page corresponding to the selected transaction type is displayed. A series of value fields are automatically calculated, their values are displayed, and you can't edit them: the contract's unpaid information such as accrued interest, unpaid amount on contract or on customer, interest value for capital, remaining principal at date, future installments number and future annuity.



5. Fill in the **external identifier** of the transaction, if available.
6. View the **interest accrued** up until the event date, the **unpaid amount on the contract** , and the **unpaid amount for the customer**. The values are pre-filled by Loan Management and you can't change them.
7. Edit the principal amount to be repaid with this event, in the **Repayment Principal Amount** field.
8. View the **interest value** applicable to the payment and the **remaining value of the principal** at the current date, as calculated by Loan Management.
9. View the transaction fee applicable for the transaction, if it was defined, displayed in the **Charge Fee** field.
10. View the **fee for repayment** or the **repayment fee percent**, depending on which is displayed. The fee value or the percentage are pre-filled by Loan Management according to the **Charge Fee** defined for this transaction type. Depending on the `ManualRepaymentFeeLoan` Management system parameter's value, the system may allow you to change the fee or the percentage. See Transaction Fees for more details.
11. View the **total value of other fees** applicable for the transaction, if any.
12. Set the early repayment options:

- **Keep Contract Period** - Select this checkbox to instruct Loan Management to keep the period of the contract. If left unselected, the contract period is recalculated.
- **Collect accrued interest** - Select this checkbox to instruct Loan Management to collect the interest accrued up to date. If the checkbox is not selected, then the `Accrued Interest` = 0.00, if it's selected, then `Accrued Interest` = the value of accrued interest until the event's date. The payment schedule projection takes into consideration both the payment schedule type defined at the banking product level, and whether to collect the interest accrued on the contract until the date of an early repayment event, thus repaying the accrued interest, or not, thus adding the accrued interest to the next installment due after the early repayment event.
  The `CalculateAccrualEarlyRepayment` Loan Management system parameter specifies whether the accrual and provision should be calculated for early repayments with the event value equal to a part of contract's unpaid amount (partial early repayments) or only for full early repayments.
- **Accrued interest** - View the interest accrued up until the event date, as calculated by Loan Management.



13. View the early repayment results, with the **number of installments** to be paid in the future, and either the future value of the installment as recalculated after this payment in the **Future Annuity** field, or the future value of the principal in the **Future Principal For Installment** field.

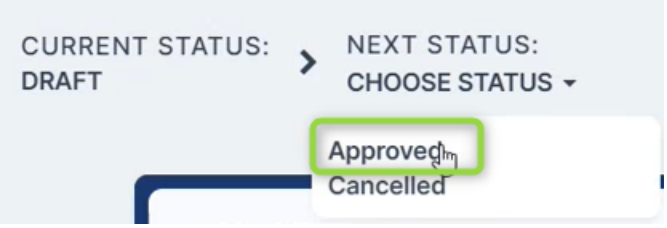
14. Decide whether you want to use the repayment schedule calculated by Loan Management through automatic processes, or you want to import a custom schedule. For custom schedule, select the **Import Schedule** checkbox. Read more about importing a custom schedule file in the Manually Upload Repayment Schedules section of the user guide.

15. Click the **Save and Reload** button.
If the event value meets the business requirements defined within Loan Management, the event is saved. Otherwise, an error message appears. Change the values as instructed in the message and try saving the event again.
While the event is in **Draft** status, you can modify all the event's fields except **Transaction Type**. The event value is not applied to the contract while the event is still in this status.

16. Click the **Simulate Early Repayment** button to view the details of each installment of the calculated repayment schedule.

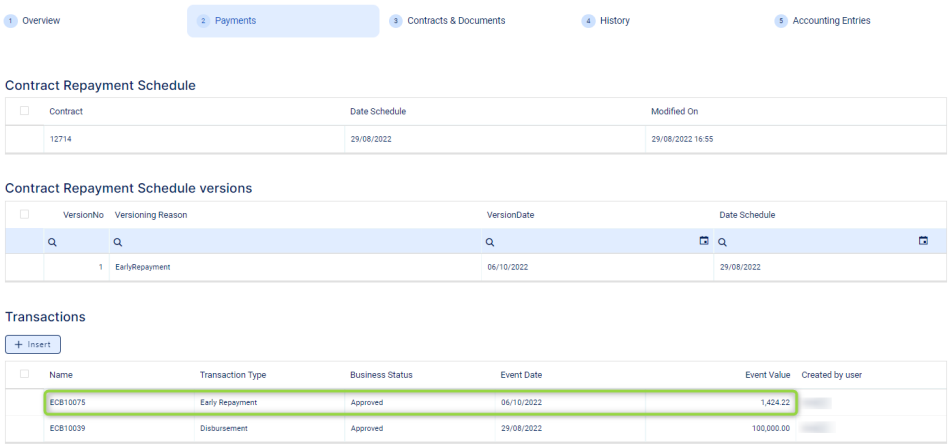


17. Approve the event by changing its status to **Approved** in the upper left corner of the **Event** page.



18. Confirm the change of status in the **Confirmation** window, clicking **Yes**. The event is now in **Approved** status and Loan Management applies the recalculated repayment schedule to the contract, displaying the previous schedule version in the **Contract Repayment Schedule Versions** section of the contract's **Payments** tab.

    The transaction is visible in the **Transactions** section.



> **NOTE**
> All existing versions of the contract in **Contract Version Draft** status are automatically changed to **Contract Version Closed** when a payment event is approved for that contract.

Writing Off a Loan

A write-off is an accounting action that reduces the value of an asset while simultaneously debiting a liabilities account. It is primarily used by financial institutions seeking to account for unpaid loan obligations. The write-off is the last action taken in the attempt of recovering a non-performing loan. If there are collaterals linked to the loan, those are turned into cash to be used to recover the loan's remaining notified and unpaid amounts. Usually before writing off a loan, it passes a `Doubtful` loan classification at 90 days overdue, and thereafter, depending on the internal risk procedures, the financial institution can write it off or sell the loan to collection agencies. Regardless of the financial institution's decision, such a process involves a loss.

In Loan Management, the write-off functionality is intended for non-performing loans, allowing you to "delete" any or all notified overdue amounts from the loan contracts if your financial institution approves this action and assumes the loss. Using the specific settings defined at the banking product level, the system can automatically write off non-performing loan contracts, using the `Process Contracts For Auto WriteOff` scheduled job, after their maturity day passes and the remaining notified and unpaid value on the contract is under the specified threshold. Loan Management also allows you to manually perform write-off transactions, even if the remaining notified and overdue amounts on the contract are above the defined threshold. In this case, you can perform the write-off either for a partial or the full amount.

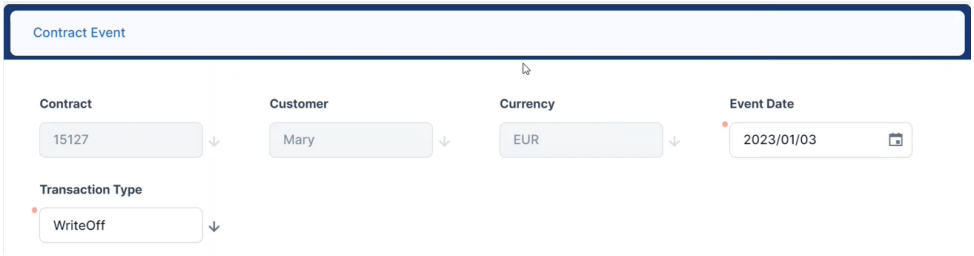
Regardless of how the write-off transaction is performed, manually or automatically, it results in taking the write-off amount from the specified internal write-off account and crediting the loan's main bank account for the loan principal, respectively the product's self bank account for all the other operation items. The `DefaultWriteOffReconciliationAccountBankingProduct` system parameter holds the internal write-off account defined for the system's default currency and automatically populates this field, but you can change the account according to your needs.

Adding a Write-Off Transaction To an Approved Contract

You can add write-off transactions to an approved contract via Loan Management's user interface or through API calls using the `FTOS_CB_AddUpdateWriteOff` and `FTOS_CB_ApproveWriteOff` endpoints. Read more about these endpoints in the Loan Management Developer Guide.

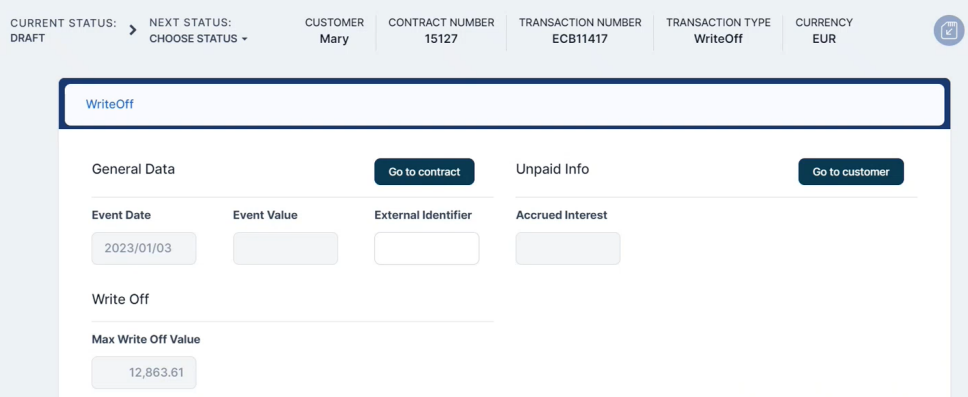
In order to add a write-off transaction to a loan contract through the menus available in Loan Management, follow these steps:

1. In FintechOS Portal, select a contract with **Approved** status and double-click to open it.
2. Navigate to the contract's **Payments** tab and click the **Insert** button above the **Transactions** section. The **Event** page is displayed.

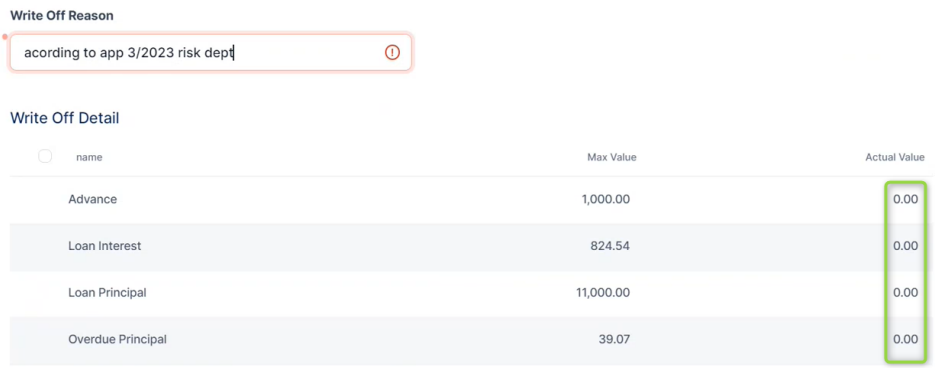


3. Fill in the following fields:
   - **Transaction Type** - Select from the list the **WriteOff** transaction type. If you can't find it, then the transaction type is not associated with the banking product which is at the base of the contract.
   - **Event Date** - This is pre-filled with current date.

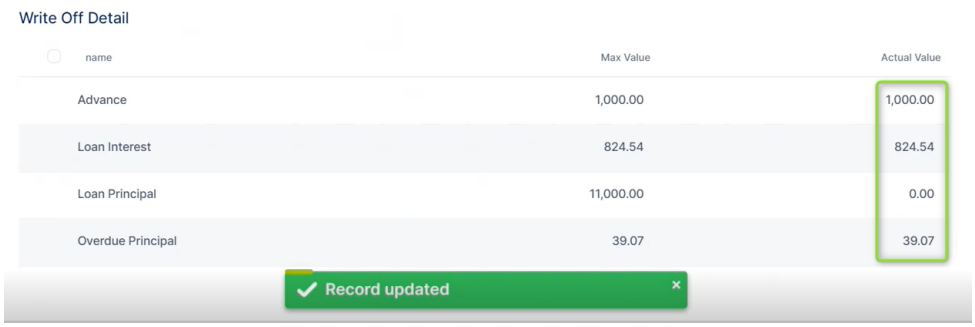
   Other values are automatically completed: contract, customer, and currency.
4. Click the **Save and Reload** button.
   The event is saved in **Draft** status and a transaction number is automatically generated for it. The **Edit Contract Event** page corresponding to the selected transaction type is displayed. A series of value fields are automatically calculated, their values are displayed, and you can't edit them: the contract's maximum possible write-off value and its unpaid accrued interest up to the date of this transaction.



5. Fill in the **external identifier** of the transaction, if available.
6. In the **Write Off** section, fill in the **Write Off Reason**, entering the reason for performing the write-off transaction, according to your financial institution's internal procedures.
7. View the **Max Write Off Value** field, displaying the contract's maximum possible value for the write-off transaction, automatically calculated as `Remaining Principal (absolute main balance) + Accrued Interest + unpaid values from Repayment Notifications (interest, commissions, penalties, or insurances)`.
8. Adjust the actual write-off amount, displayed in the **Event Value** field, changing the values for each operation item displayed in the **Write Off Detail** section. Initially, the actual values for the write-off transaction are defaulted as zero, as pictured below:

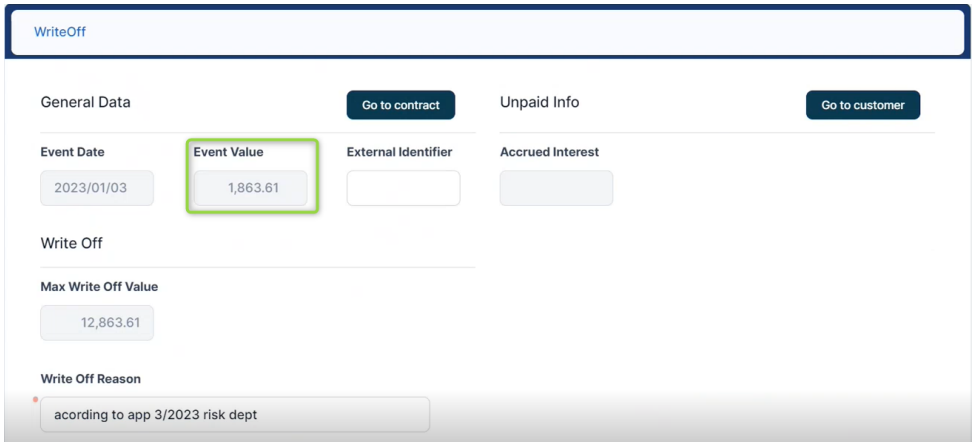


For example, you can specify to perform a write-off transaction for the maximum values for the advance, the loan interest, and the overdue principal, leaving the loan principal unchanged:

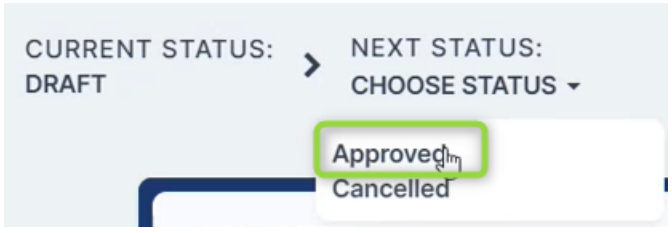


For each write-off detail, you can't specify a value greater than the maximum value.

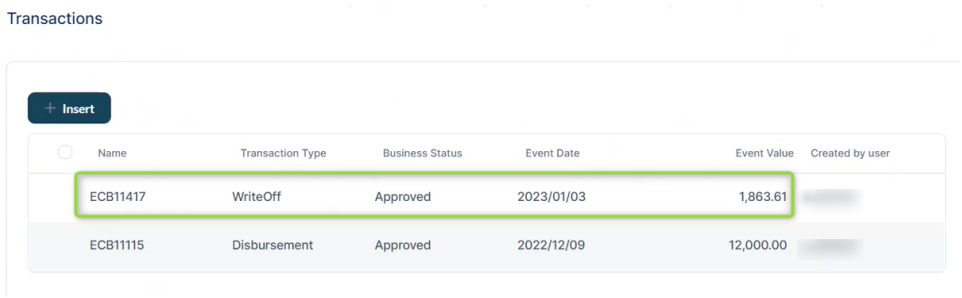
The **Event Value** is automatically recalculated after changing the actual values of the operation items.



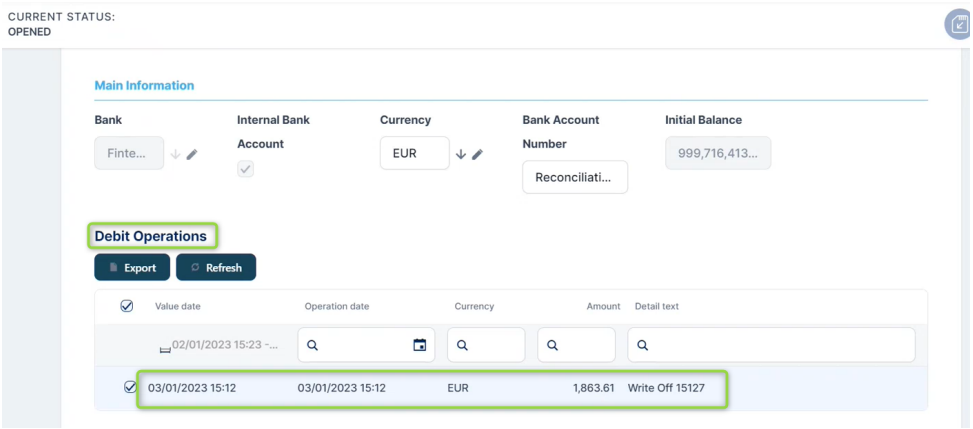
9. Approve the transaction by changing its status to **Approved** in the upper left corner of the **Event** page.



10. Confirm the change of status in the **Confirmation** window, clicking **Yes**. The transaction is now in **Approved** status, visible in the **Transactions** section.

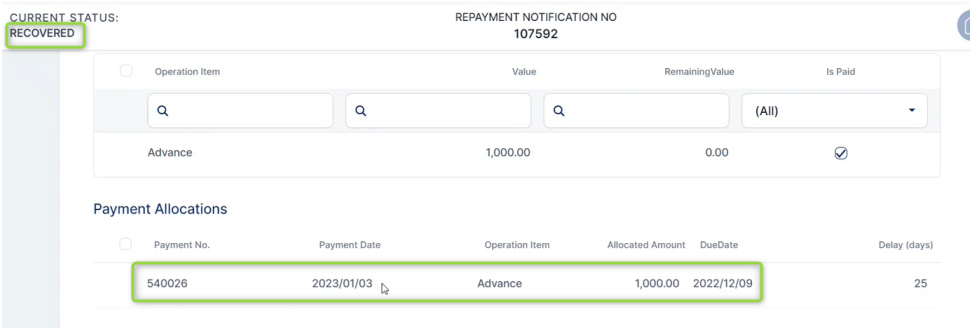


Loan Management performs the write-off transaction for each operation item's actual value. It takes the amounts from the internal write-off account defined at the banking product level and credits the loan's main bank account for the loan principal, respectively the product's self bank account for all the other operation items. The example below shows the debit operation performed on the write-off account:



> **NOTE**
>
> In case of partial write-off, Loan Management performs the allocation of amounts according to the allocation method, closing the oldest unpaid values first.

The affected repayment notifications are processed. If the write-off amount covers the repayment notification's amount, then the notification's status becomes **Recovered**, otherwise it remains in the same status, but its amount is lower. The example below shows the repayment notification containing the advance of 1000, fully recovered after performing the write-off:

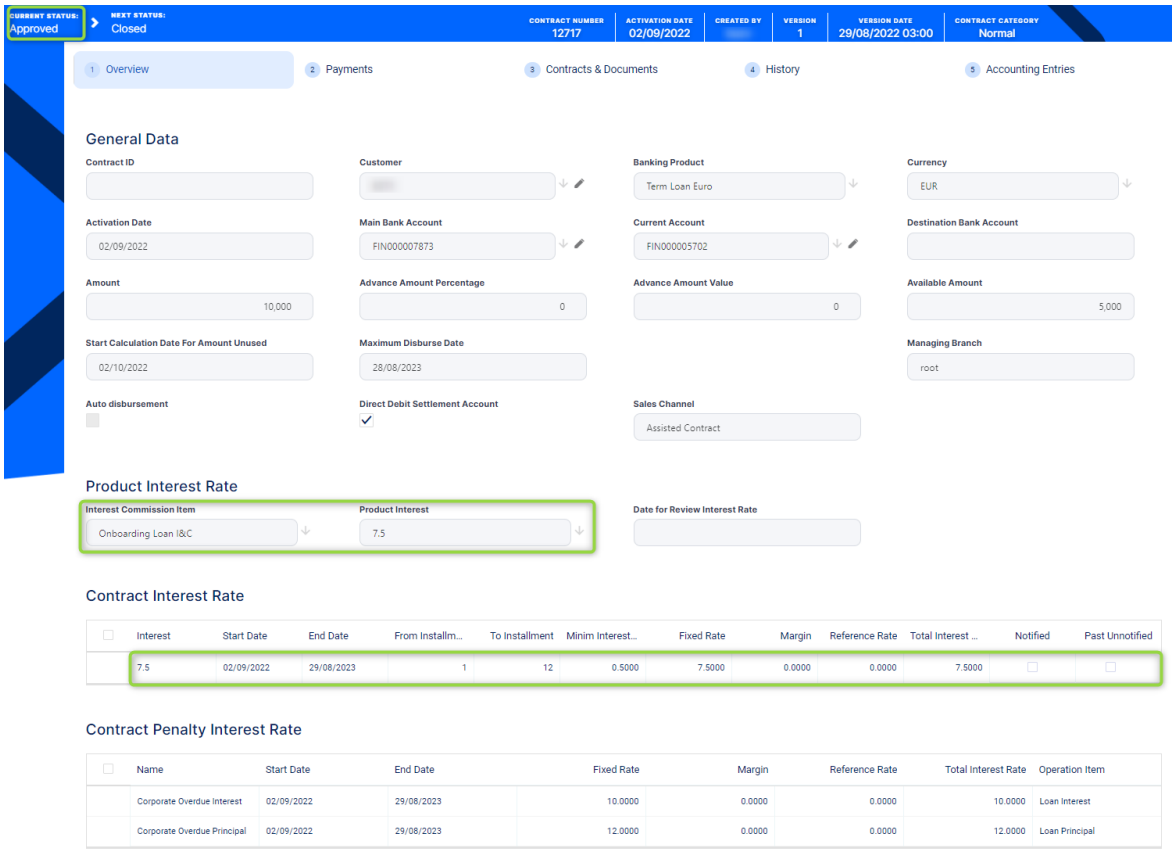


The contract is automatically closed after write-off as required, as long as there are no pending amounts on the contract, the loan is not revolving, and if the contract closure settings are in place.

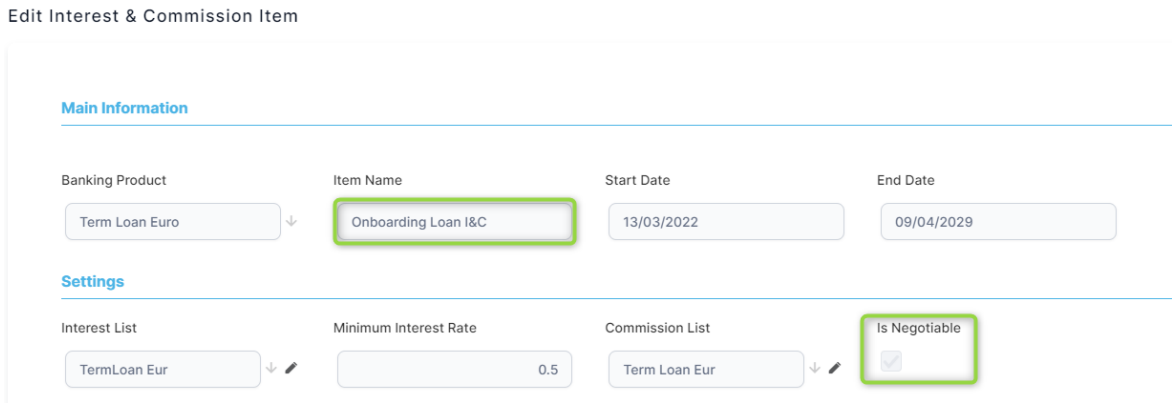
IMPORTANT!
Loan Management allows you to add write-off transactions to approved loan contracts that have notified and overdue amounts regardless of their classification, as long as you perform a partial write-off. You could use partial write-off transactions as a method of adjusting existing repayment notifications.

Changing the Interest Rate

During the life-cycle of a contract, there may be situations when you need to change the interest rates applicable to the contract.

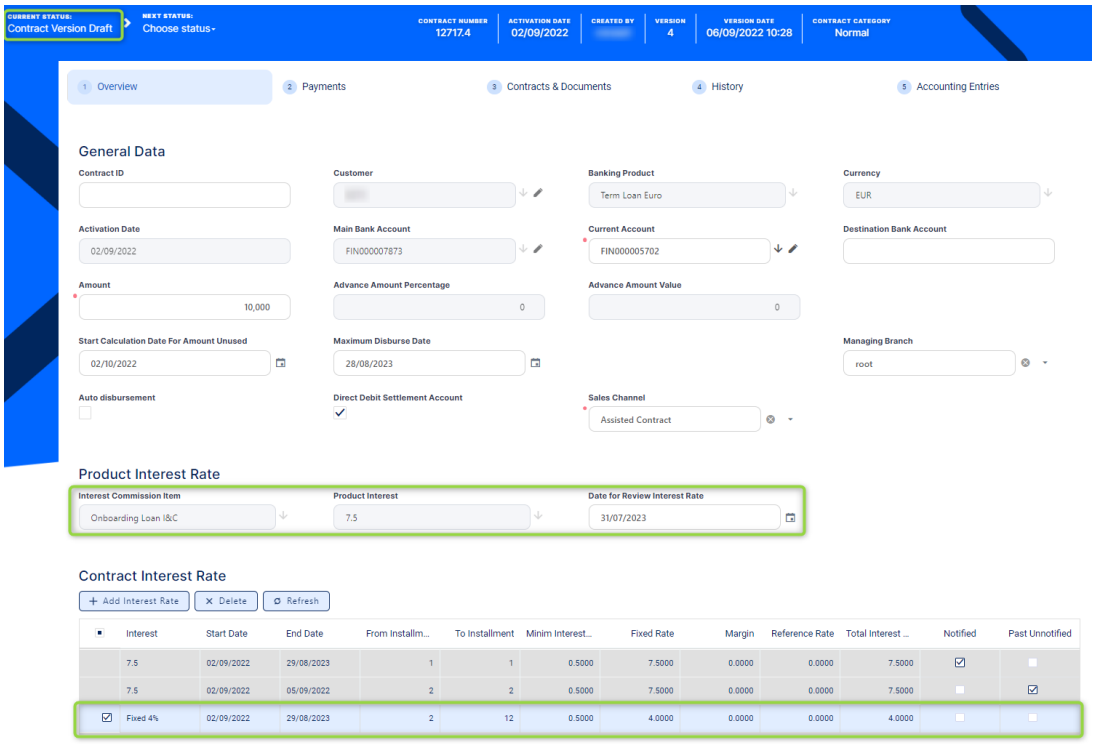


You can do this if the interest is defined as negotiable:



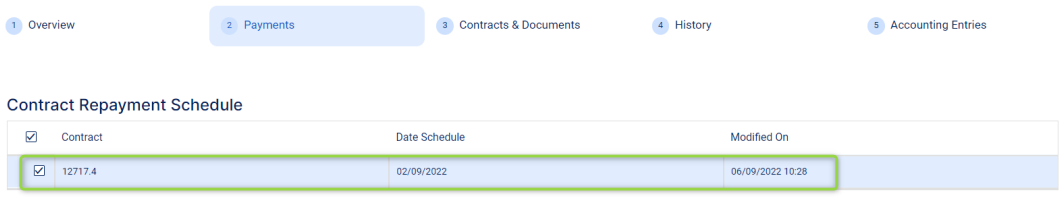
1. To modify an approved contract's interest, you must first create a new version as described in the dedicated topic.

2. In the contract with `Version Draft` status, you can change the interest rates either using another definition of interest from those listed in product setup, or modifying the fixed rate, margin, or minimum interest rate, in the **Overview** tab's **Product Interest Rate** and **Contract Interest Rate** sections.

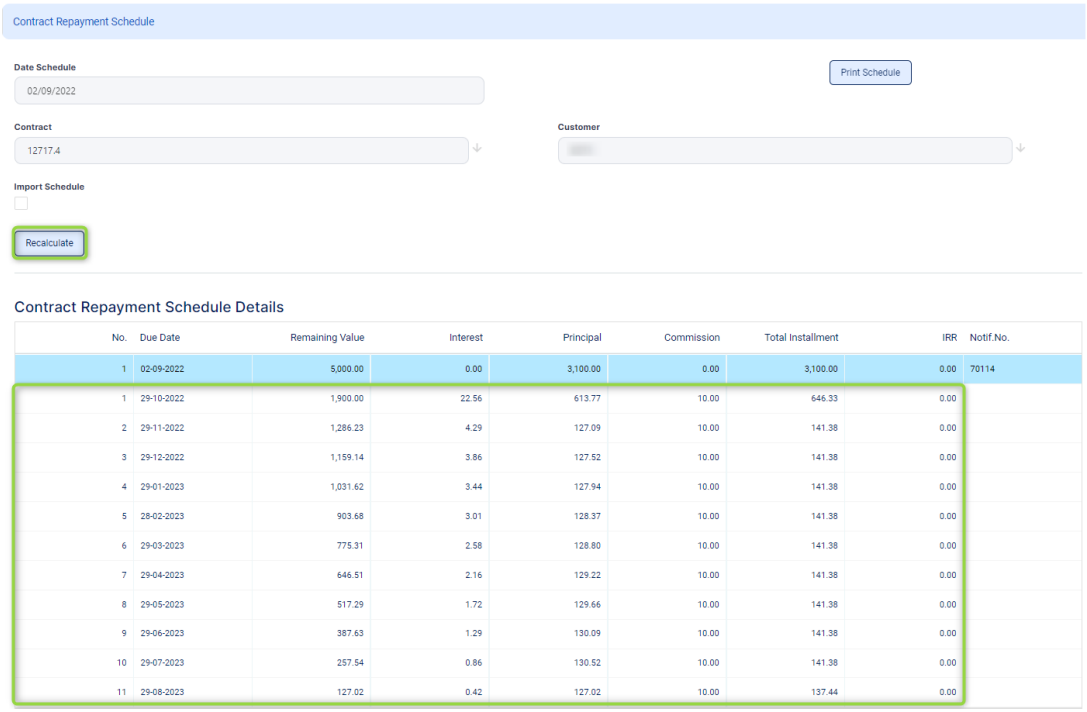


For contracts in `Version Draft` status, you can't perform any changes to the contract interest rates for notified installments or for days that have elapsed already from the current month's installment (if either the **Notified** or the **Past Unnotified** checkboxes are selected).

3. Remember to recalculate the repayment schedule before approving a contract in `Version Draft` status for which you performed interest rate changes, otherwise an error prevents you from approving the contract!
Navigate to the **Payments** tab -> **Contract Repayment Schedule** section and double-click the repayment schedule generated for contract version.

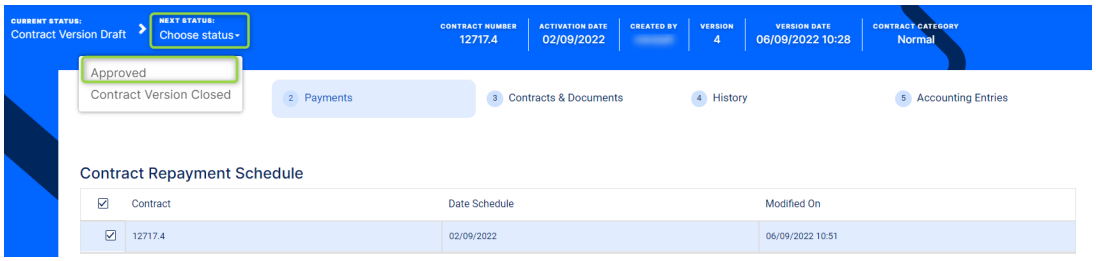


4. On the displayed **Contract Repayment Schedule** page, click **Recalculate**. Loan Management recalculates the repayment schedule for the remaining installments, using the changed interest rates.

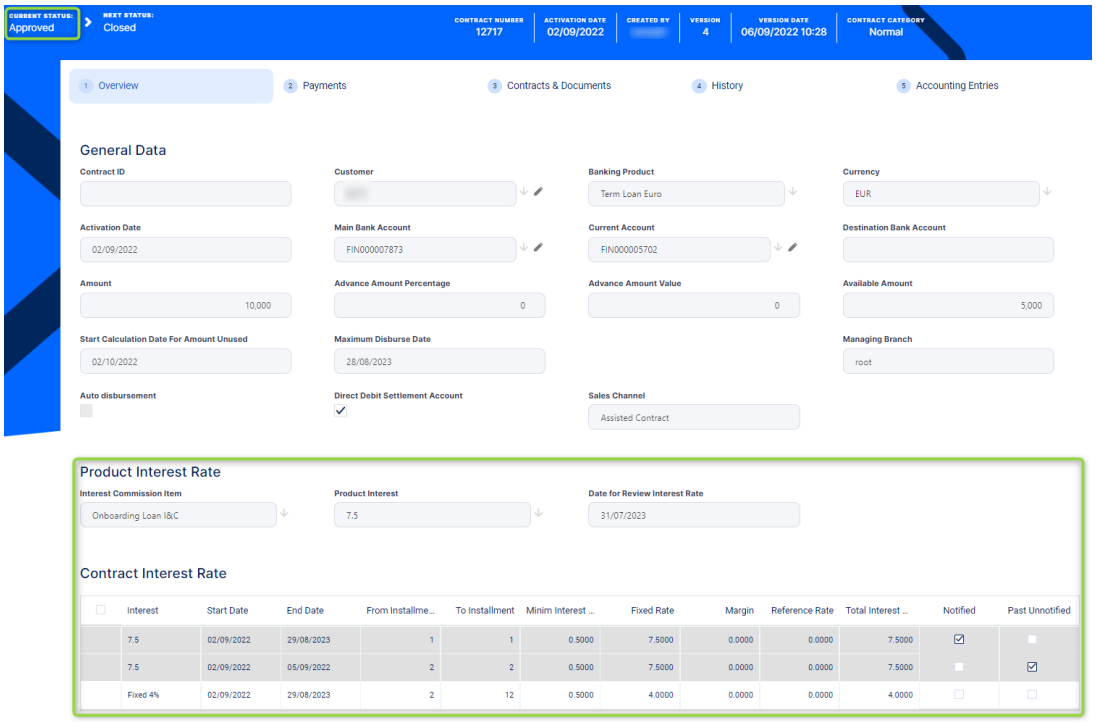


5. Click the **Save and Close** button.

6. Approve the contract in `Version Draft` status, changing its status to `Approved` and then confirming your action. Thus, Loan Management applies the new recalculated repayment schedule to the contract.

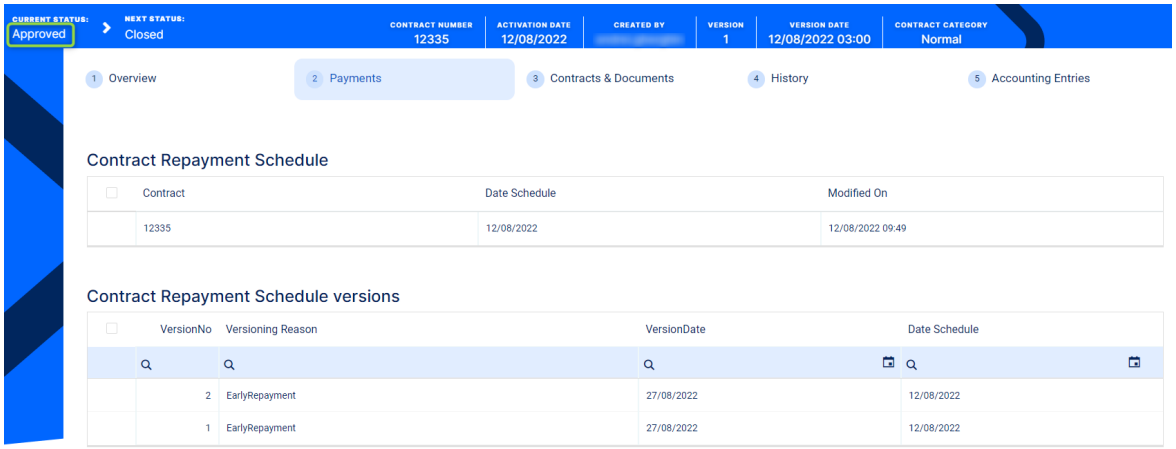


7. View the approved contract's changed interest rates in the **Overview** tab.

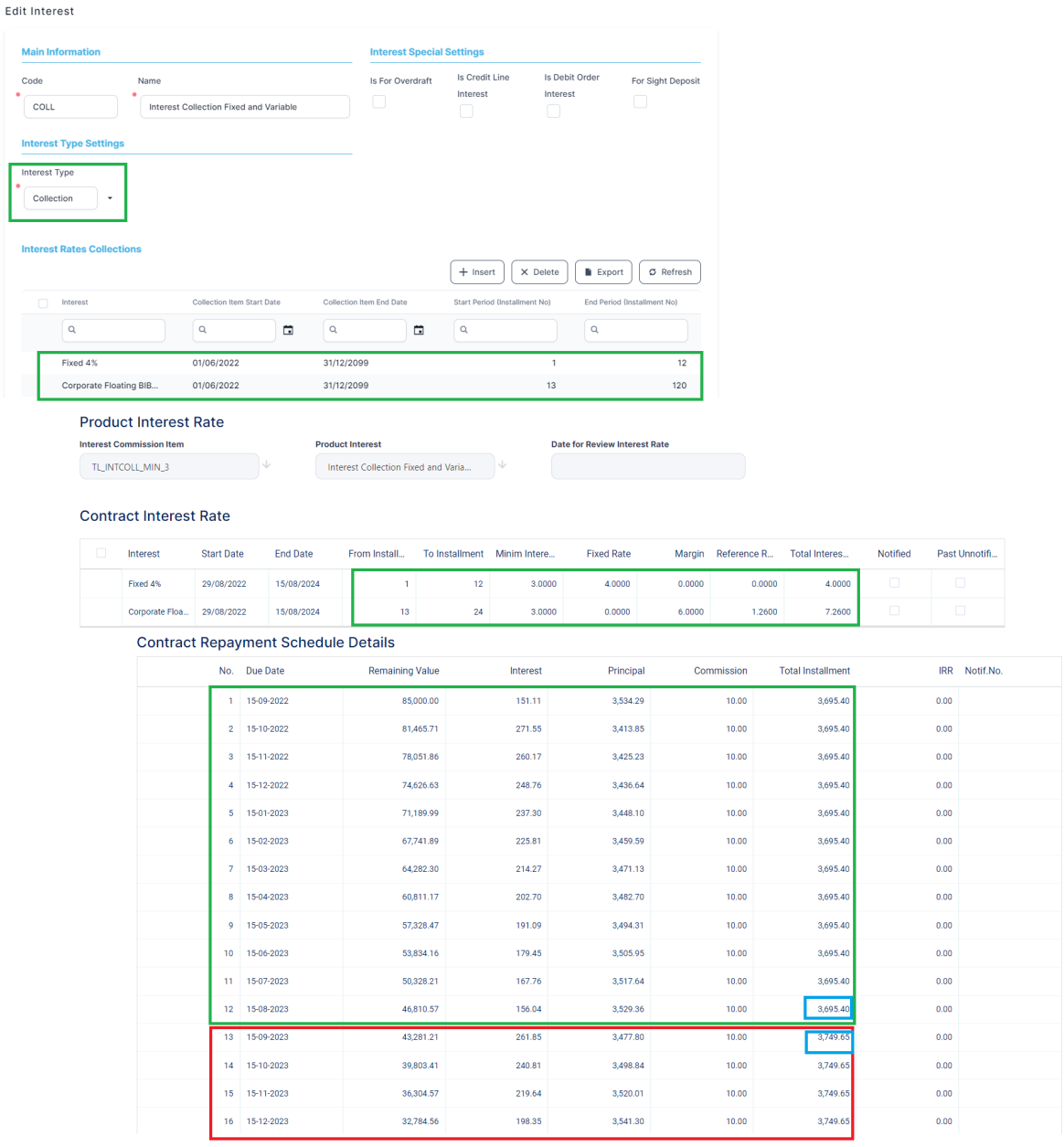


Editing and Customizing Repayment Schedules

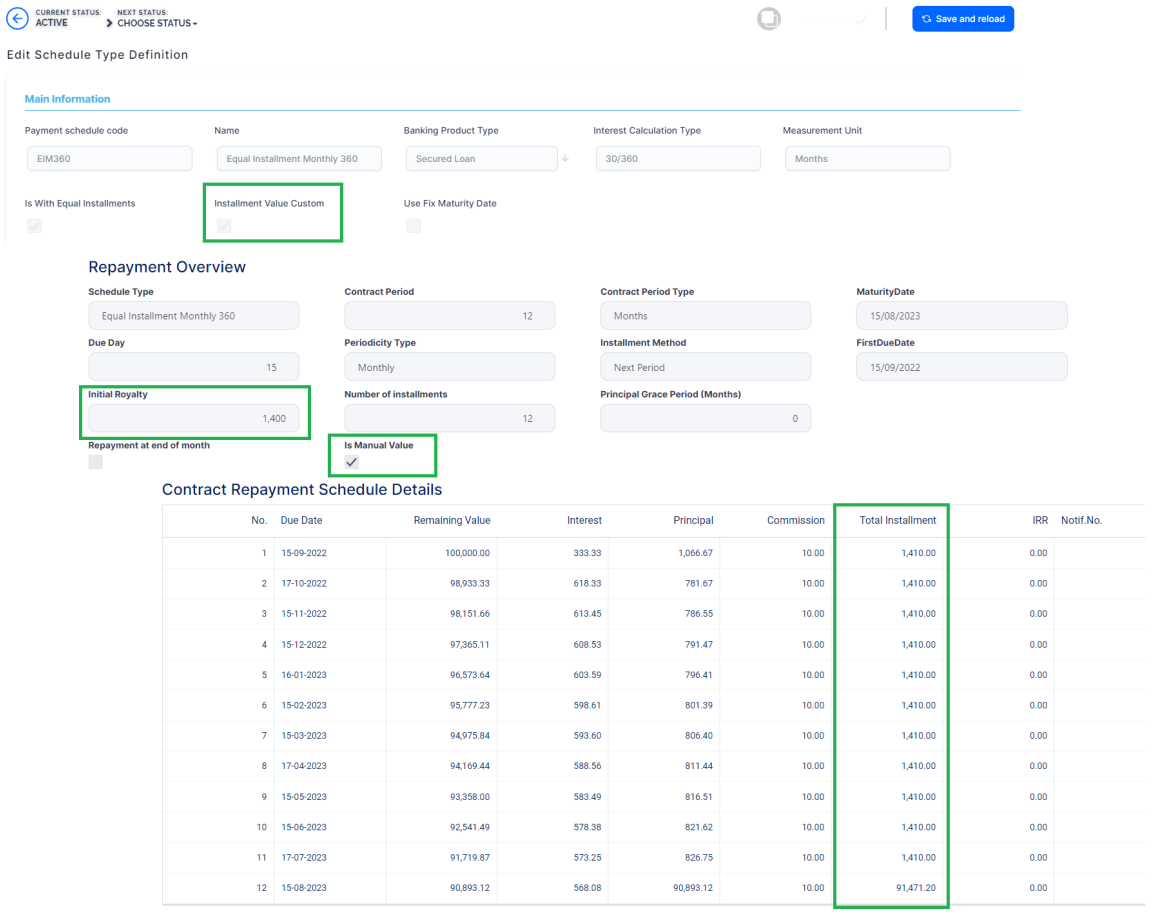
Once a loan approved and disbursed, you can check the repayment schedule built based on the contract's details in the **Payments** tab. You can also view the versions of the repayment schedules automatically generated by Loan Management when the schedule is recalculated due to specific contract events.



If you need to have a schedule to reflect a product that has a fixed rate for the first x years and moving to variable afterward, use a **Collection** interest type at product definition. When creating the contract, Loan Management automatically builds the schedule projection with the 2 different rates, while the changes in amounts to be collected are visible upfront. This is sometimes referred to as fixed to variable loan, especially valid for mortgages. Read more in the Manage Contract Level Interest & Penalty Interest Rate and in the Applying Fees and Commissions sections.

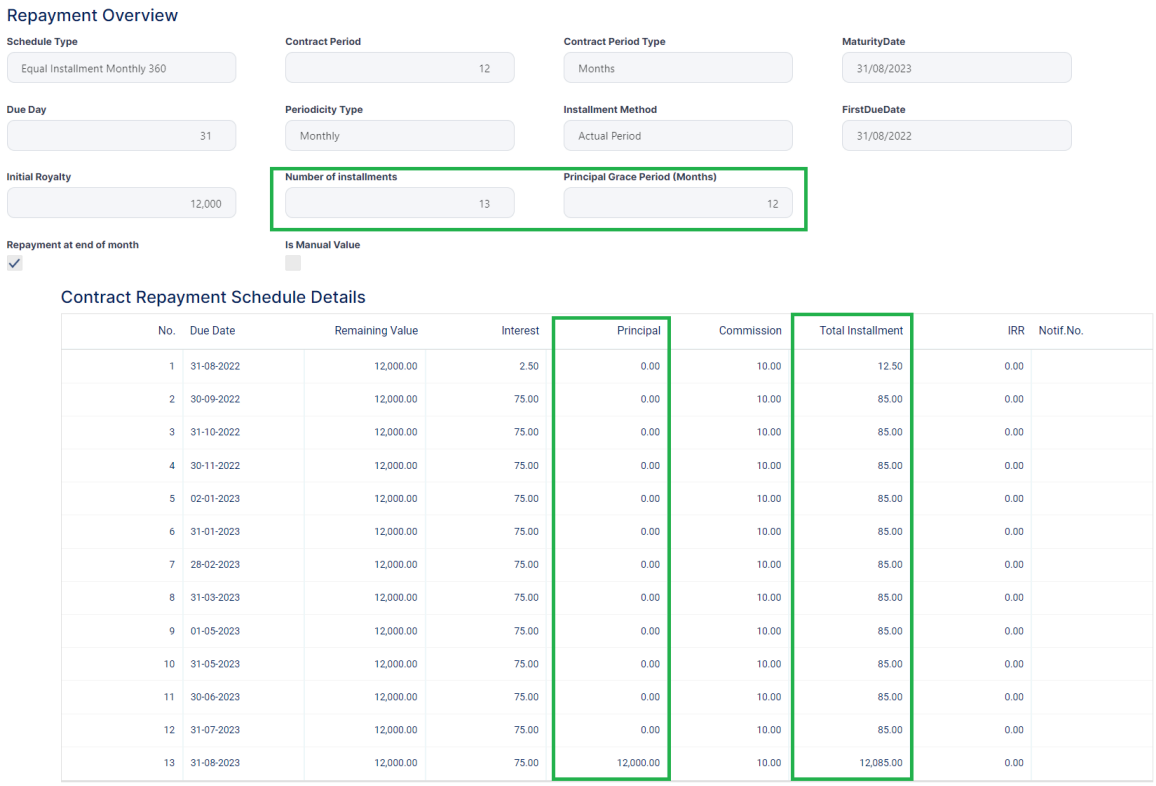


When you need to collect amounts during the life of the loan, but also leave a balloon (residual) payment, you can use the functionality enabled via the **Is Manual Value** option on the contract, if **Installment Value Custom** is set on the schedule type definition used within the product. This allows you to overwrite the repayment amount in the **Initial Royalty** or **Initial Principal Value** field and provide a lower one, thus any capital not made due until maturity results in a balloon payment. In practice, you can also manage in this way fixed to variable loans where the business is to fix the mortgage for the x years, and you are expected either to close by the time it moves to variable, or re-fix again. Read more about setting the repayment information on contract level in the Enter Repayment Information for the Contract section.



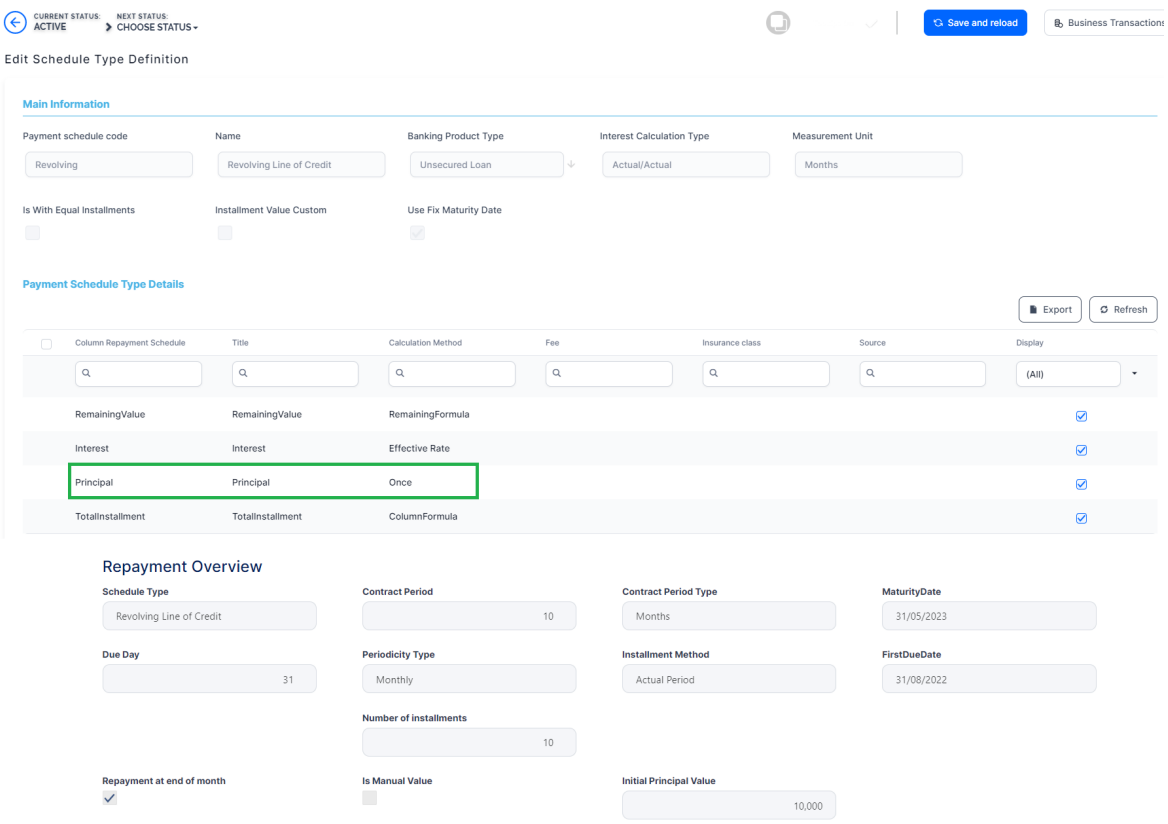
A revolving line of credit might be a case where you need to enable a loan for which you collect interest and potentially some fees monthly, at end of month, and the principal at maturity. Such loan would also be a revolving one. This sort of setup can be achieved with a schedule definition that includes the fee. On the contract, you select the due date as end of month, plus a grace for principal for the number of

installments of the full loan minus one, for example the loan has 13 installments, the **Principal Grace Period (Months)** has 12 installments, and thus all principal is expected to be repaid on maturity. When the funds are available, you can perform an early repayment transaction and reuse those funds again when needed since it is a revolving setup. In practice, such approvals may be extended and if this is the case you need to make sure you perform the extension one day in advance of the maturity, otherwise the jobs set at the end of the day also pick up and process such maturity. Read more about setting the repayment information on contract level in the Enter Repayment Information for the Contract section.



No.	Due Date	Remaining Value	Interest	Principal	Commission	Total Installment	IRR	Notif.No.
1	31-08-2022	12,000.00	2.50	0.00	10.00	12.50	0.00	
2	30-09-2022	12,000.00	75.00	0.00	10.00	85.00	0.00	
3	31-10-2022	12,000.00	75.00	0.00	10.00	85.00	0.00	
4	30-11-2022	12,000.00	75.00	0.00	10.00	85.00	0.00	
5	02-01-2023	12,000.00	75.00	0.00	10.00	85.00	0.00	
6	31-01-2023	12,000.00	75.00	0.00	10.00	85.00	0.00	
7	28-02-2023	12,000.00	75.00	0.00	10.00	85.00	0.00	
8	31-03-2023	12,000.00	75.00	0.00	10.00	85.00	0.00	
9	01-05-2023	12,000.00	75.00	0.00	10.00	85.00	0.00	
10	31-05-2023	12,000.00	75.00	0.00	10.00	85.00	0.00	
11	30-06-2023	12,000.00	75.00	0.00	10.00	85.00	0.00	
12	31-07-2023	12,000.00	75.00	0.00	10.00	85.00	0.00	
13	31-08-2023	12,000.00	75.00	12,000.00	10.00	12,085.00	0.00	

Below is an example of an out-of-the-box setting for a repayment schedule setup, to obtain the same approach with full capital repaid on maturity. Make sure you have Once for **Principal** definition, as pictured here:



Apart from the standardized regular repayment schedules, there may be cases when for a specific customer or for a specific business area you need custom repayment frequencies or even amounts. For such cases, you can manually upload a repayment schedule from an excel template. There are some validations and transformation requirements, but at the end you have a repayment schedule that does not follow any of the classical definitions.

Calculation Algorithm of a Repayment Schedule with Equal Installments

Loan Management can use a series of methods to calculate the repayment schedule. For example, it uses the following mathematical formula to generate equal monthly installments:

$$Installment = PV * \prod_{i=1}^{n}(1 + E_i)/F_n$$

where:

$$E_i = OutstandingAmount_i * InterestRate * NumberOfDaysBetween2Installments/NumberOfDaysInYear$$

$$OutstandingAmount_i = OutstandingAmount_{i-1} - InstallmentValue + E_i$$

$$F_n = 1 + F_{n-1}(1 + E_n)$$

$$F_{n-1} = 1 + F_{n-2}(1 + E_{n-1})$$

...

$$F_2 = 1 + F_1(1 + E_2)$$

$$F_1 = 1$$

$$PV = PresentValue$$

$$n = TotalNumberOfInstallments$$

Depending on other definitions within the banking product, Loan Management uses business logic to factor in the instructed number of days between the installments' due dates, as defined in the repayment schedule's `Interest Calculation Type`: 30/360, 30/365, Actual/360, etc. These elements, together with the amount, interest rate, and period, lead to the equal installments.

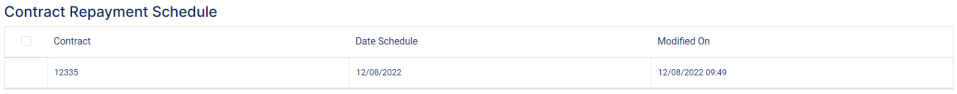
For instance, for Actual/Actual, the installment is calculated as a recursive calculation to equalize the payment flow and result in equal installments, although the interest amount is calculated based on the exact number of days between two installments and within a year. In this case, Loan Management uses the same formula, but with an adjusted (actual) number of days:

$$E_i = OutstandingAmount_i * InterestRate * ActualNumberOfDaysBetween2Installments / ActualNumberOfDaysInYear$$

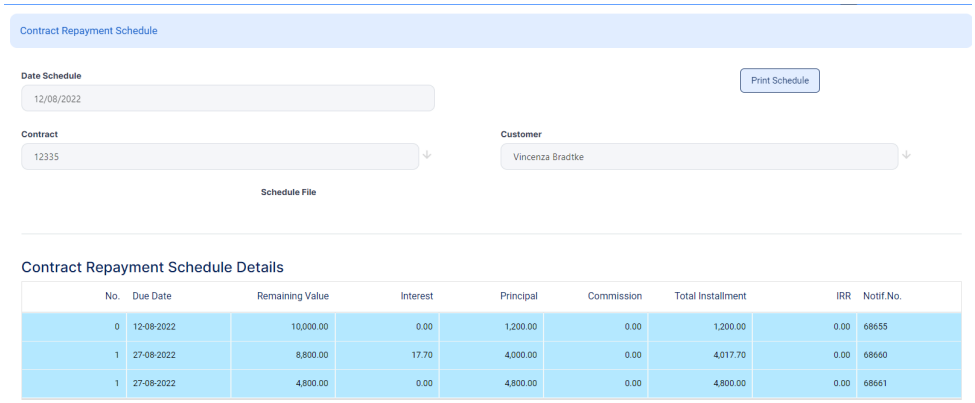
Viewing a Contract's Repayment Schedule

Follow these steps to view the repayment schedule automatically generated by Loan Management for the approved contract, after performing a disbursement:

1. In the contract's **Payments** tab, go to the **Contract Repayment Schedule** section. This section displays only basic information about the generated schedule, such as contract number, schedule date and last modification date and time.



2. To see detailed information and the actual list of the installments, double-click the schedule. The **Contract Repayment Schedule** page is displayed with the selected schedule and a list with every schedule detail:



You can't edit the information displayed on this page, but for contracts in **Version Draft** status, you can import a custom repayment plan, if you don't want the contract to use the repayment plan automatically generated by Loan Management.

> **NOTE** Following an early repayment event, for contracts based on banking products with the **Is Revolving** field set to `False`, when the installments number recalculated after such an event is lower than the previous installments number, the maturity date and the contract period are updated along with the number of installments.

3. View the information displayed about each schedule detail (installment): number, due date of the installment, interest, principal and commissions calculated for this installment, the value of the repayment notification generated for this schedule detail, and its number, if the repayment notification was already generated.

   Within the list, the schedule details are color coded as follows:

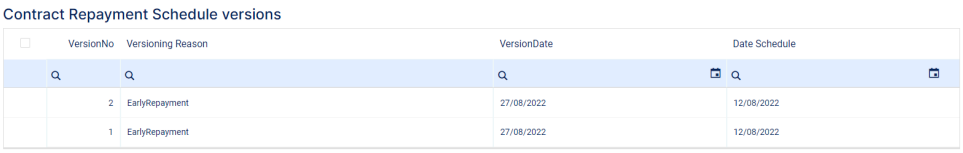
   - Schedule details highlighted in blue are already paid, allocated or closed to payment.
   - Schedule details not highlighted (displayed on a white background) remain to be paid.

4. To export the schedule in a `.pdf` file, click the **Print Schedule** button. Your browser automatically downloads the `PaymentScheduleFile` file, with all the information displayed within the **Contract Repayment Schedule** page.

Viewing a Contract's Repayment Schedule Versions

Follow these steps to view the versions of the repayment schedules automatically generated by Loan Management each time when a contract event that changed either the maturity date or the amount of a repayment was performed on the contract:

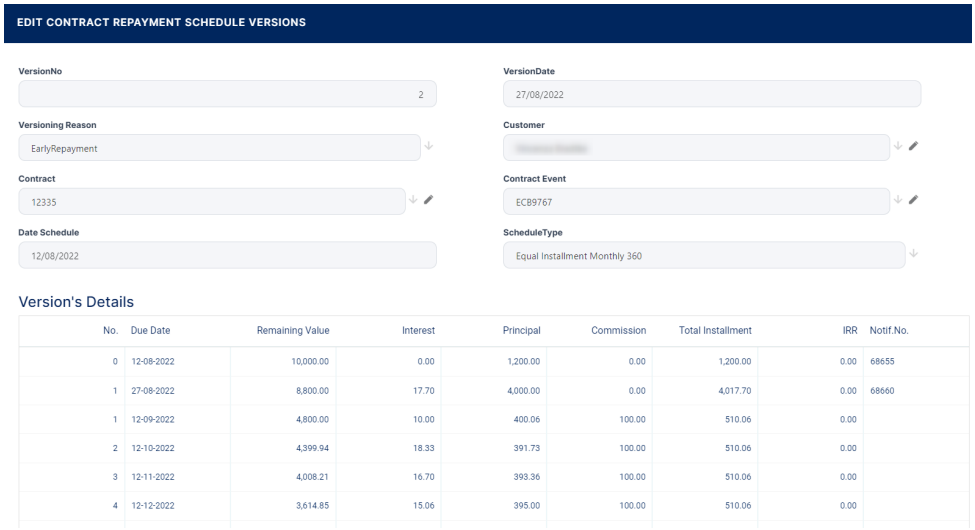
1. In the contract's **Payments** tab, go to the **Contract Repayment Schedule Versions** section. The section displays only basic information about the generated versions, such as version number, versioning reason, version creation date and the date of the previously active schedule.



> **NOTE**
>
> The most recent repayment schedule version is considered active

by Loan Management, while the previous schedule records are kept for historical purposes.

2. To see detailed information about the repayment schedule version, double-click on the desired record. The **Edit Contract Repayment Schedule Version** page is displayed with the selected schedule and a list with every schedule detail:



You can't edit the information displayed on this page.

3. View the information displayed about each schedule version: its number and date, the versioning reason specifying the type of contract event that triggered the generation of the version, the customer, contract and event information, as well as the date of the previous repayment schedule and the type of schedule selected in the contract and used to generate the repayment schedule.

4. View the information displayed about each schedule detail (installment): number, due date of the installment, remaining value to be repaid from the contract value at the moment of this installment, interest, principal and commissions calculated for this installment, the value of the repayment notification generated for this schedule detail, and its number, if the repayment notification was already generated.

Manually Upload Repayment Schedules

To accommodate cases when a contract's repayment plan can't be built using any of the existing Loan Management automated logic, you can import a custom-built schedule on a contract. You can either perform the import at a contract event level (when adding a **Disbursement**, **Early Repayment**, **Payment Holiday**, or **Reschedule Overdues** transaction), or at a contract level within the **Contract Repayment Schedule** page, for contracts in **Version Draft** status. After importing a custom repayment plan on a contract through a `.xlsx` file whose template must be first exported, Loan Management validates that the total outstanding amount is matched by the principal on schedule. On due dates, the due amounts are processed properly and the notifications are generated as per the imported repayment plan.

For example, you could use such imported schedules for contracts that require seasonal repayment plans. For seasonal credits in agriculture, you could have due dates only in April or September. You could build a schedule to have installments only in these months, to accommodate such situations.

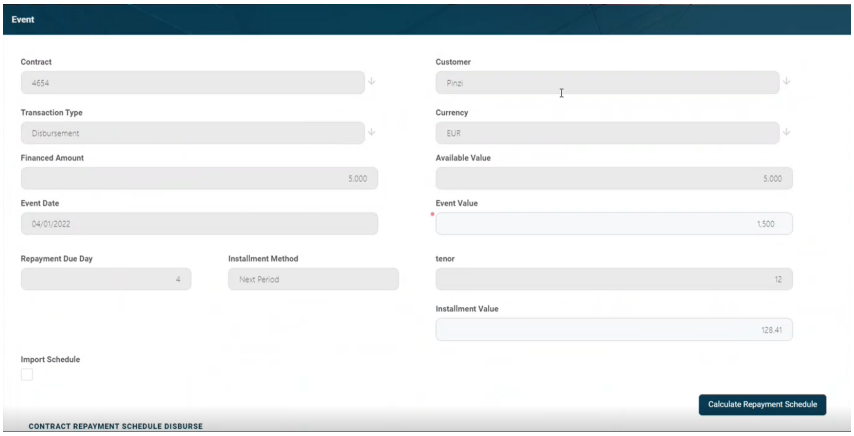
Another example is that you could use the import functionality for contracts with atypical schedules that have to be migrated into Loan Management. First, you could import the schedules as they are, then you could decide to adopt the Loan Management logic for schedule calculation, maybe change the interest rate, so you version the contract to have a repayment plan automatically calculated by the system.

IMPORTANT!
After importing a repayment plan on a contract in **Version Draft** status, the `Auto disbursement` field on that contract version is automatically set to `False` and any future contract events must have imported repayment plans!

Importing Repayment Plans at Contract Event Level

To import a custom repayment plan at a contract event level (when performing a transaction), for contract events in **Approved** status, perform the following steps:

1. On the **Contracts** page or in the **Contracts Dashboard**, select a contract with **Approved** status.
2. Double-click the contract to open it for editing.
3. Navigate to the contract's **Payments** tab.
4. Double-click a **Disbursement**, **Early Repayment**, **Payment Holiday**, **Returned Amount Or Goods**, or **Reschedule Overdues** transaction in **Draft** status.
   OR
   Create a new **Disbursement**, **Early Repayment**, **Payment Holiday**, **Returned Amount Or Goods**, or **Reschedule Overdues** transaction, up until saving the event in **Draft** status.



5. Select the **Import Schedule** checkbox. The **Schedule File** field is displayed, with the **Select file** button. The **Export Schedule Template** button is also displayed. A warning message informs you that "*The previous repayment schedule was generated by the system*".



6. Click the **Export Schedule Template** button to download a `.xslx` file with the schedule template for this specific contract.
7. Open the downloaded schedule template `.xlsx` file exported from Loan Management and make it editable.
8. Format the **Due Date** column as `Text`
9. Fill in the lines of the template file with the data needed in the contract's custom repayment plan.
   Fill in each installment's type according to the event type.

> **NOTE** Make sure you don't change the data formats within the columns, except for the **Due Date** column which needs to be `Text`. Do not change the headings. Validations are performed for the file after uploading it to Loan Management, when the record is saved, and incorrect formats or data are not accepted, then error messages are shown for such files.

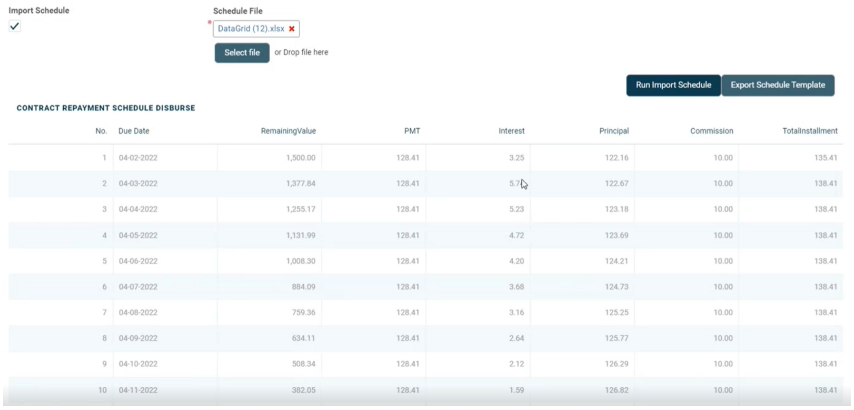
> **IMPORTANT!** For contracts with existing repayment notifications, these notifications are already present in the downloaded template file. Do not modify the existing notifications' data. Compose your custom schedule taking into account the already existing repayment notifications.

Read more about the repayment plan template file and how to fill it in its dedicated section.

10. Save the file. If the file needs to be labeled according to your company's information protection policy, label it as `Public`, otherwise, Loan Management can't import it.
11. Back in Loan Management's **Contract Repayment Schedule** page, click **Select File**.
12. In the newly displayed Explorer window, browse for the `.xlsx` file that contains the custom schedule you've filled in with the contract's repayment plan, then click **Open**.
The selected file's name is displayed under the **Schedule File** field.



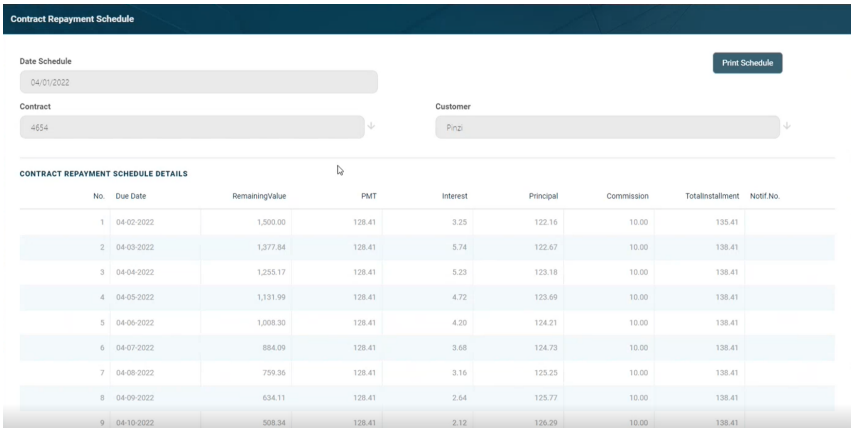
13. Click the **Save and Reload** button.Loan Management performs the validations, making sure that the uploaded file meets all the criteria for a correct, functional schedule.
14. Click the **Run Import Schedule** button to perform the import of the new custom schedule to the contract. This button can be clicked only if a file was selected in the **Schedule File** field.
The **Contract Repayment Schedule Details** section now displays data contained in your custom schedule.



If you click the **Run Import Schedule** button one more time, the values within the **Contract Repayment Schedule Details**

section are deleted and reinserted.

15. Approve the event by changing its status to **Approved** in the upper left corner of the **Event** page.
The contract's repayment schedule, accessible through the Contract Repayment Schedule page, now displays the custom repayment plan uploaded through the approved event.



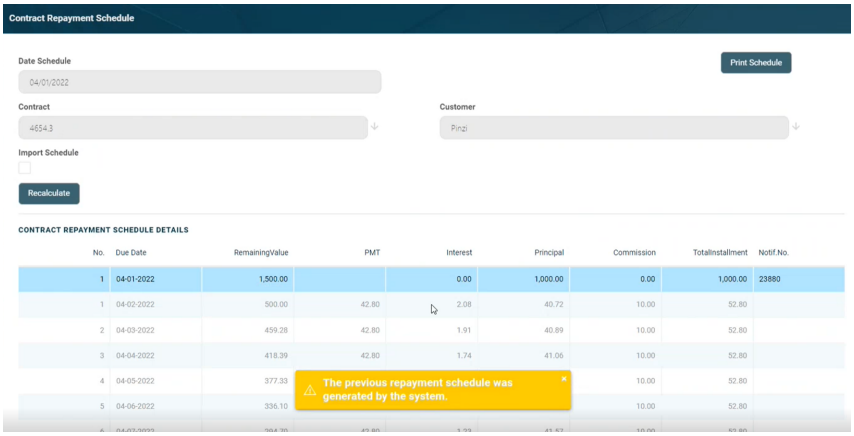
> **NOTE**
> When versioning a contract that has an imported repayment plan, the schedule recalculation is not mandatory. You can choose between recalculating and importing an updated repayment plan.

Importing Repayment Plans at Contract Level

To import a custom repayment plan directly at a contract level, for contracts in **Version Draft** status, perform the following steps:

1. On the **Contracts** page or in the **Contracts Dashboard**, select the contract for which you wish to change its schedule with a custom repayment plan. Make sure that the contract is in **Version Draft** status.

2. On the displayed contract's page, navigate to the **Payments** tab and double-click the contract displayed in the **Contract Repayment Schedule** section. The **Contract Repayment Schedule** page is displayed.



A warning message lets you know that "*The previous repayment schedule was generated by the system*". You can either recalculate the schedule using the Loan Management logic, or you can import a custom repayment plan which you build for this contract version.

3. On the newly opened **Contract Repayment Schedule** page, select the **Import Schedule** checkbox. The **Schedule File** field is displayed, with the **Select file** button. The **Export Schedule Template** button is also displayed. A warning message informs you that "*The previous repayment schedule was generated by the system*".

4. Click the **Export Schedule Template** button to download a `.xslx` file with the schedule template for this specific contract.

5. Open the downloaded schedule template `.xlsx` file exported from Loan Management and make it editable.

6. Format the **Due Date** column as `Text`

7. Fill in the lines of the template file with the data needed in the contract's custom repayment plan.
   Fill in each installment's type according to the event type.

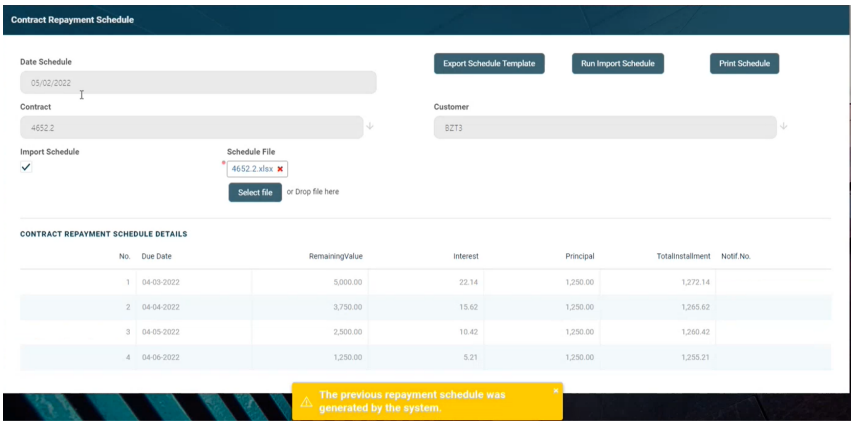
> **NOTE**
>
> Make sure you don't change the data formats within the columns, except for the **Due Date** column which needs to be `Text`. Do not change the headings. Validations are performed for the file after uploading it to Loan Management, when the record is saved, and incorrect formats or data are not accepted, then error messages are shown for such files.

> **IMPORTANT!**
>
> For contracts with existing repayment notifications, these notifications are already present in the downloaded template file. Do not modify the existing notifications' data. Compose your custom schedule taking into account the already existing repayment notifications.

   Read more about the repayment plan template file and how to fill it in its dedicated section.

8. Save the file. If the file needs to be labeled according to your company's information protection policy, label it as `Public`, otherwise, Loan Management can't import it.

9. Back in Loan Management's **Contract Repayment Schedule** page, click **Select File**.
10. In the newly displayed Explorer window, browse for the `.xlsx` file that contains the custom schedule you've filled in with the contract's repayment plan, then click **Open**.
 The selected file's name is displayed under the **Schedule File** field.
11. Click the **Save and Reload** button.Loan Management performs the validations, making sure that the uploaded file meets all the criteria for a correct, functional schedule.
12. Click the **Run Import Schedule** button to perform the import of the new custom schedule to the contract. This button can be clicked only if a file was selected in the **Schedule File** field. The **Contract Repayment Schedule Details** section now displays data contained in your custom schedule.



If you click the **Run Import Schedule** button one more time, the values within the **Contract Repayment Schedule Details** section are deleted and reinserted.

> **NOTE**
> When versioning a contract that has an imported repayment plan, the schedule recalculation is not mandatory. You can choose between recalculating and importing an updated repayment plan.

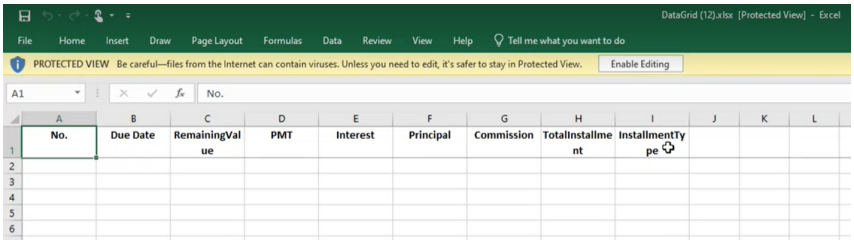
Working with a Repayment Plan Template File

Each time you need to import a custom schedule to a contract, you must export its template file first, make the needed changes, then save the file. Follow these steps to work with repayment schedule template files:

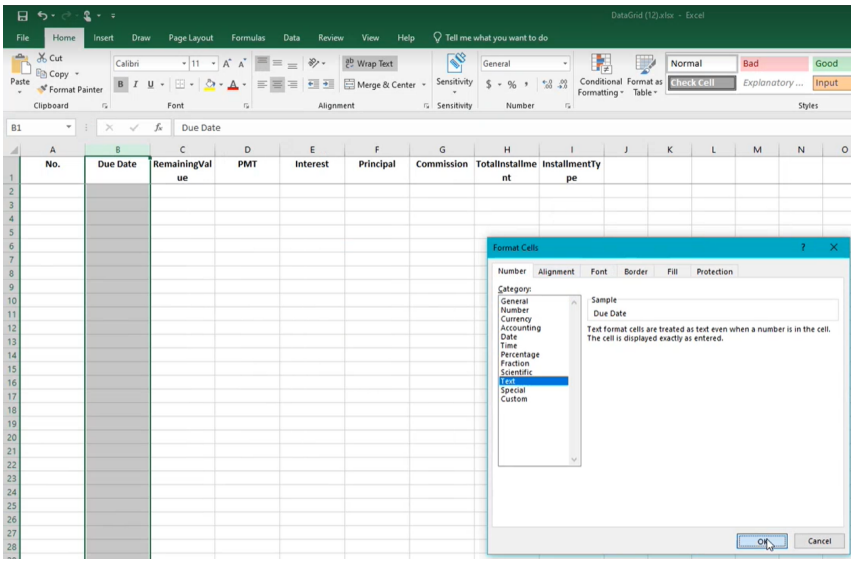
1. On the **Contract Repayment Schedule** page, or on the **Contract Event** page, click the **Export Schedule Template** button.
   A file with `.xlsx` format is downloaded to your computer. The file contains the columns that must be filled in based on the contract's schedule type definition.

   > **NOTE**
   >
   > Each contract type may have a different schedule type definition, so make sure you download the template!

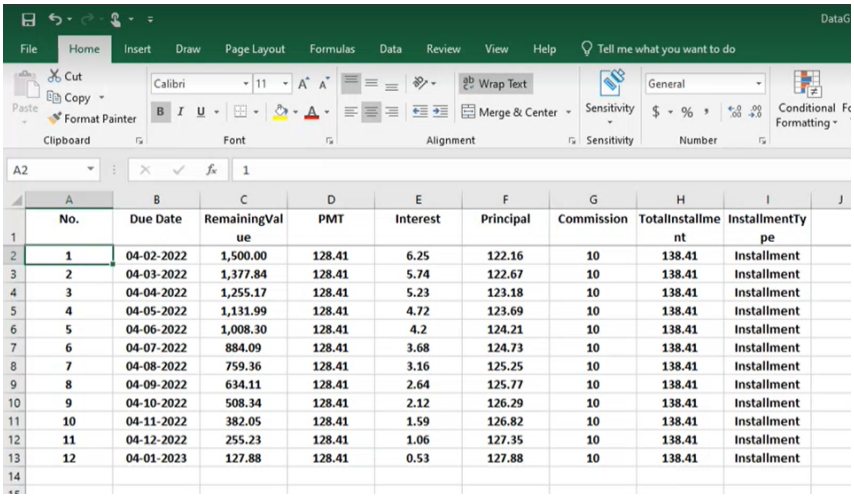
   This is an example of an exported repayment plan template file:



2. After opening the downloaded schedule template `.xlsx` file, make it editable so that you can perform changes within its cells.
3. Format the **Due Date** column as `Text`.



4. Fill in the lines of the template file with the data needed in the contract's custom repayment plan. Fill in each installment's type according to the event type. The possible values are `Installment`, `EarlyRepayment`, `DownPayment`, `PaymentHoliday`, `ReturnAmount`, or `RescheduleOverdue`.



	A	B	C	D	E	F	G	H	I
1	No.	Due Date	RemainingValue	PMT	Interest	Principal	Commission	TotalInstallment	InstallmentType
2	1	04-02-2022	1,500.00	128.41	6.25	122.16	10	138.41	Installment
3	2	04-03-2022	1,377.84	128.41	5.74	122.67	10	138.41	Installment
4	3	04-04-2022	1,255.17	128.41	5.23	123.18	10	138.41	Installment
5	4	04-05-2022	1,131.99	128.41	4.72	123.69	10	138.41	Installment
6	5	04-06-2022	1,008.30	128.41	4.2	124.21	10	138.41	Installment
7	6	04-07-2022	884.09	128.41	3.68	124.73	10	138.41	Installment
8	7	04-08-2022	759.36	128.41	3.16	125.25	10	138.41	Installment
9	8	04-09-2022	634.11	128.41	2.64	125.77	10	138.41	Installment
10	9	04-10-2022	508.34	128.41	2.12	126.29	10	138.41	Installment
11	10	04-11-2022	382.05	128.41	1.59	126.82	10	138.41	Installment
12	11	04-12-2022	255.23	128.41	1.06	127.35	10	138.41	Installment
13	12	04-01-2023	127.88	128.41	0.53	127.88	10	138.41	Installment
14									

NOTE

Make sure you don't change the data formats within the columns, except for the **Due Date** column which needs to be `Text`. Do not change the headings. Validations are performed for the file after uploading it to Loan Management, when the record is saved, and incorrect formats or data are not accepted, then error messages are shown for such files.

IMPORTANT!

For contracts with existing repayment notifications, these notifications are already present in the downloaded template file. Do not modify the existing notifications' data. Compose your custom schedule taking into account the already existing repayment notifications.

5. Save the file after filling in the template file with the desired schedule information. If the file needs to be labeled according to your company's information protection policy, label it as `Public`, otherwise, Loan Management can't import it.

Automatic Validations Performed by Loan Management

A series of checks and validations are performed for the file after uploading it to Loan Management, when the record is saved, before being imported:

- Due Date must be of the correct `Text` format: `dd-mm-yyyy`;
- Remaining Value = previous Remaining Value - previous Principal;
- Sum(Principal) = eventValue for `Disbursement` event type;
- Sum(Principal) = outstanding amount (sum of principal for installments not notified yet) in all other cases;
- Values are not negative;
- Due Date >= Activation Date if installment type = `DownPayment`;
- Due Date >= First Due Date if installment type = `Installment`;
- Due Date <= Maturity Date;
- Total Installment respects the formula from the contract's schedule type;
- Max(InstallmentNo) <= Installment Number from the contract **Overview** tab;
- Due Date and InstallmentNo values must be different and consecutive, except for `Early Repayment` event type, where you can have 2 installments of the same number on the same day;
- If the installment is of `Early Repayment` event type, it can only be the first installment in the imported schedule;
- If the import is performed on a contract in **Version Draft** status and there are installments notified, those should not be modified and the checks should

be done over installments that were not notified yet, so sum(principal) for installments to be imported = outstanding amount. In this case, Due Date should be >= than the last notified installment and Installment Number > the last notified installment.

If any of these checks fail, then you are presented with error messages and cannot continue with the import process. For example, an error appears when you try to change an installment's Due Date (within the contract's Overview section) starting with the second disbursement.

Warning messages notify you if the previous repayment schedule was generated by the system or it was imported. These warnings are for your information and do not affect the import process. For example, whenever you opt for using the automatically generated repayment plan on a contract that already had an imported schedule, a warning message informs you that the standard system method overwrites the schedule that was imported in a previous version/ event.

IMPORTANT!

For contracts with imported schedules where the second disbursement with repayment contract was automatically generated by Loan Management, the workaround is to:

1. Create a new version of the contract.
2. Recalculate the repayment plan.
3. Approve the contract version.
4. Create a new disbursement.

Manually Capture Notifications

Apart from the notifications automatically generated by Loan Management for each installment that has to be paid for existing contracts that disbursed various amounts to customers, Loan Management also assists you reconcile wrongly processed cases and other situations resulted from delayed processing/ wrong updates. Thus, depending on your user rights, you can manually add notifications for an active contract based on lending product types, term loans, and mortgages, or add notifications for amounts to be recovered even if there is no active contract. Loan Management includes all such manual notifications in the recovery processes.

> **NOTE**
> Loan Management enables you to manage manual repayment notification via the user interface or via integration through APIs. For information about the available endpoints, please visit the Loan Management Developer Guide.
> For information about managing manual repayment notification via the user interface, continue reading this page.

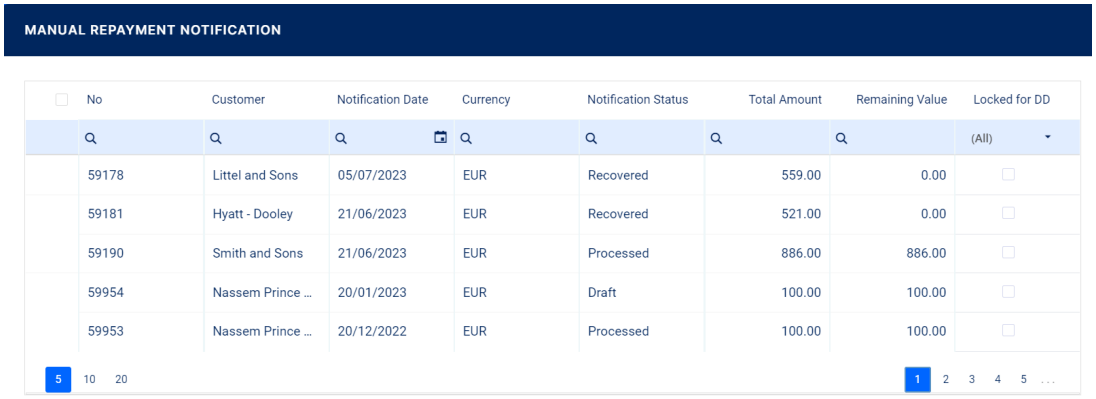
> **NOTE**
> You need one of the **Corporate Credit Officer**, **Retail Credit Officer**, or **Loan Admin Officer** security roles to view, create, delete, and update manual repayment notifications.
> You need the **Loan Admin Officer** security role to update their status to Approved.

Adding Repayment Notifications

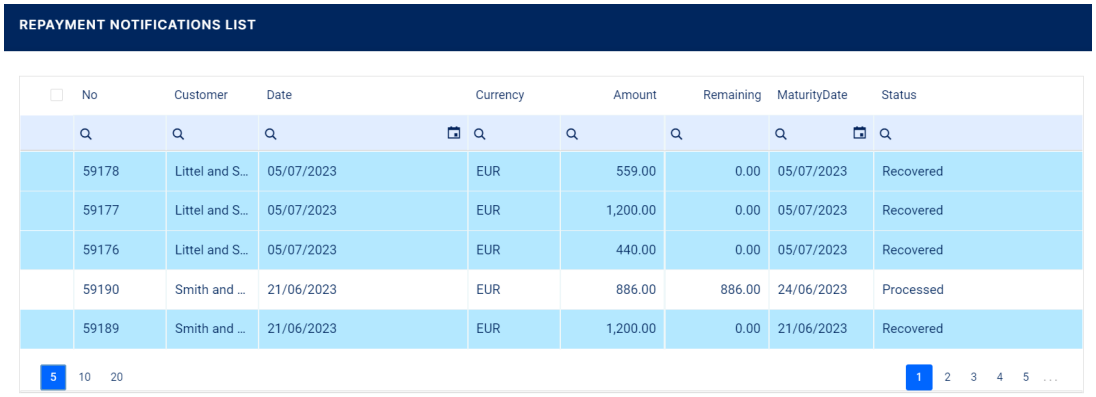
Follow these steps to manually add a repayment notification:

1. In the FintechOS Portal, click the main menu icon and expand the **Loan Management Operational** menu.

2. Click **Manual Repayment Notification** menu item to open the **Manual Repayment Notifications** page.



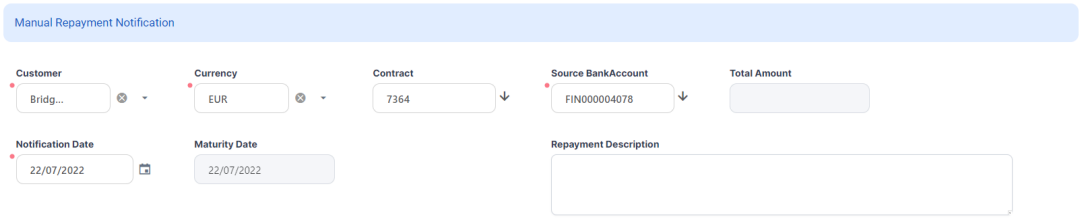
Or, click **Repayment Notification** menu item to open the **Repayment Notifications List** page.



Within the list, the notifications highlighted in blue are already paid, allocated, or closed to payment, while notifications not highlighted (displayed on a white background) remain to be paid.

3. On the **Manual Repayment Notifications** page, click **Insert** to open the **Add Manual Repayment Notification** page.

4. Fill in the following details regarding the notification:

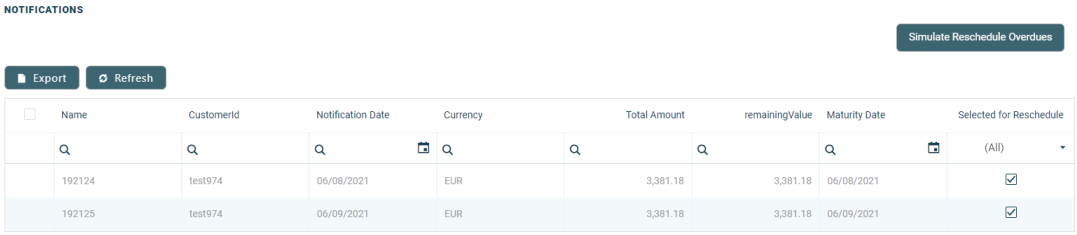


- **Customer** - Select the customer for whom the notification is created.
- **Contract** - Select the number of the contract for which the notification is generated. You can choose from the approved and closed contracts of the selected customer. The currency and the source bank account are automatically filled in using the values from the selected contract. If the notification is not linked to an active contract, you must select a source bank account.
- **Source Bank Account** - Automatically filled in if the contract was selected. Select the bank account from where the notified amount should be allocated. After selecting a source bank account, the currency is changed with the bank account's currency.
- **Notification Date** - Select the date when the notification is created.

> **NOTE**
>
> You can also add manual repayment notification from the contract level's **Payments** tab, clicking **Insert** within the **Repayment Notification** section. In the displayed **Add Manual Repayment Notification** page, some of the fields are automatically filled in based on the contract's information and can't be modified.

5. Optionally, view or edit the following details:
   - **Currency** - Automatically filled in with the currency of the notification, if the contract or the source bank account was selected.
   - **Total Amount** - This read-only field holds the total amount to be paid within the notification, calculated as the sum of all the details' values.
   - **Maturity Date** - This field is automatically filled in with the maturity date of the notification, calculated by adding the value of the `Grace period for repayment` field at the banking product level to the notification date. If no contract is selected, hence there is grace period to consider from the banking product level, then the `ManualGraceRepayment` Loan Management system parameter is used for maturity date calculation.
   - **Repayment Description** - Enter a description for the manual notification.

6. Click the **Save and Reload** button. The manual notification is saved by Loan Management in `Draft` status. You can now continue by adding repayment notification details to it.
   You can view the notifications generated for a specific contract on the **Contract** page, in the **Payments** tab > **Repayment Notifications** section:

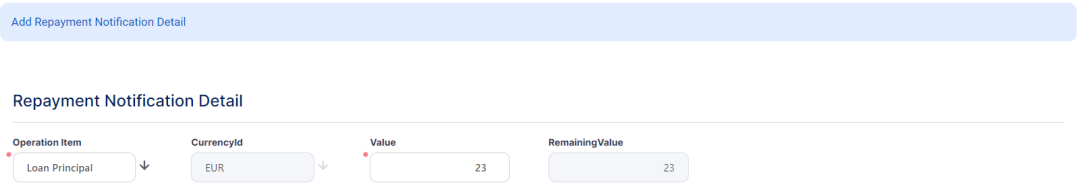


> **NOTE**
> Once the repayment notification is in `Draft` status, you can edit the currency and the source bank account only if there are no notification details created for it.

Adding Repayment Notification Details

Follow these steps to manually add a repayment notification detail:

1. On the **Edit Manual Repayment Notifications** page, click **Insert** to open the **Add Manual Repayment Notification Detail** page. The page already has the currency of the notification and the remaining value still to be paid from the notification value completed. When you create the notification detail, `Remaining Value = Value`.
2. Fill in the following details regarding the notification detail:



- **Operation Item** - Select the operation item for which the notification detail is created.
  The operation item is used in the payment allocation process. If you select an operation item that is not included in the allocation method used for manual notifications (stored in the `ManualAllocationMethod` system parameter), then Loan

Management displays a warning message.

- **Value** - Enter the value of the notification detail. It must be must be greater than 0.

3. Click the **Save and Close** button. The notification detail is saved by Loan Management. You can add as many details as needed to a manual repayment notification in `Draft` status.

> **NOTE**
> For the payment allocation job to process the details, you must first approve the manual repayment notification record.

Approving Manual Repayment Notifications

After adding all the details you need to a manual repayment notification, make sure you approve it by changing its status to **Approved**. Otherwise, the payment allocation automated jobs don't process it.

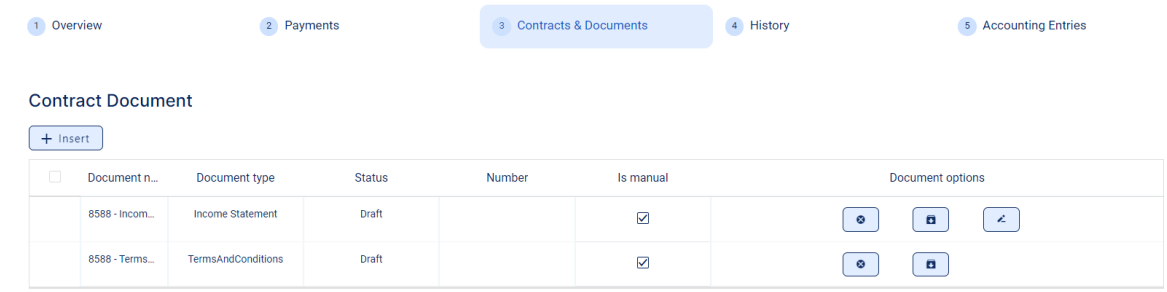
Loan Management performs the following validations before approving a manual repayment notification:

- The `Total Amount` of the repayment notification must be greater than 0;
- The `Value` fields at the details level must be greater than 0;
- The operation items selected at details level must be included in the allocation method used for manual notifications (stored in the `ManualAllocationMethod` system parameter).

After approval, Loan Management automatically transitions manual repayment notifications from the **Approved** status into **Pending Recover** or **In Recovery** statuses, using the `Auto Process Manual Repayment Notifications` scheduled job. Further, the automated settlement of repayment notification takes the notification and processes it, allocating funds from the source bank account to settle the debt.

Working with Documents

Loan Management allows you to manage all the documents related to a contract in one place, in the contract's **Contracts & Documents** tab. The tab is meant to be the electronic folder of the contract. It displays a list of the document records for the current contract, with details such as document name, type, status, number, whether the record was added through the user interface (`Is manual = True`) or through API integration (`Is manual = False`), and download options for the attached files. Contract documents have a dedicated business workflow, thus you can transition them through a series of statuses.



In the **Contracts & Documents** tab, you can: add a new contract document record, edit or delete a record in **Draft** status, view the details for records in **Signed** or **Canceled** status, or download the initial or the signed document, if it exists, by clicking the **Initial document**, respectively the **Signed document** button next to the record. Open the downloaded file to view its content.

> **NOTE**
> Users with the associated predefined security roles of Corporate Credit Officer and Retail Credit Officer can perform contract document-related operations such as adding, updating, and deleting records or changing their statuses.

Contract Document Statuses

A contract document record has the following statuses, visible in the top left corner of the **Add Contract Document** page, after saving the record:

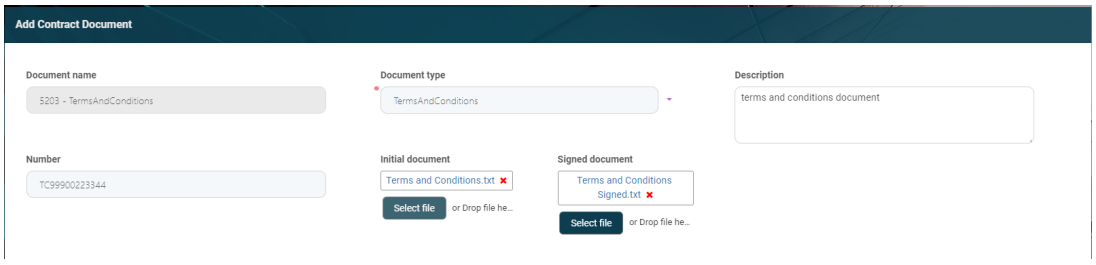
- **Draft** - the status of a newly created contract document record that was not yet authorized (marked as **Signed**). While in this status, you can edit some fields and you can delete the uploaded documents. Change its status to **Signed** after editing all the necessary details and uploading the `Signed Document` file. Change its status to **Canceled** if the document is not to be used within the contract.
- **Signed** - the status of a contract document record after being authorized. You cannot edit any of the record's details. You can change the status of the record to **Canceled**, if needed.
- **Canceled** - the status of a contract document after being canceled. Once **Signed**, a contract document should be canceled if the document is not to be used within the contract. You cannot edit any of the record's details. There is no further transition from this status. Contract document records created through integration (having their `Is manual` field = `False`) can't be canceled.

Adding Contract Documents

1. To insert a document to the contract, click the **Insert** button in the **Contract Document** section.
   The **Add Contract Document** page is displayed, with the **Document Name** field automatically completed with the name of the document.

   > **NOTE**
   >
   > You can't add documents to contracts in **Contract Closed** or **Contract Version Closed** statuses.

2. Fill in the following fields:



- **Document type** - Select the type of the document.
- **Description** - Enter the description of the document.
- **Number** - Enter the number of the document, if the document has an external identifier number.
- **Initial document** - Insert the file containing the initial, unsigned document.
  Click the **Select file** button under this field, navigate to the desired file, select it and click **Open**. The selected file's name is displayed here. You can delete it by clicking the **x** next to the file name.
- **Signed Document** - Insert the file containing the final, signed document, if available.
  Click the **Select file** button under this field, navigate to the desired file, select it and click **Open**. The selected file's name is displayed here. You can delete it by clicking the **x** next to the file name.

> **NOTE**
>
> To change the status of the contract document record to **Signed**, a signed document file must exist within the record.

3. Click the **Save and Close** button.

> **NOTE**
> You can also add, update, and approve contract document records through API integration, using the `AddUpdateContractDocument` and `ApproveContractDocument` endpoints. Read more details in the Loan Management Developer Guide.
> Contract document files added through integration cannot be deleted and those records can't be canceled!

Automatic Contract Document Validations

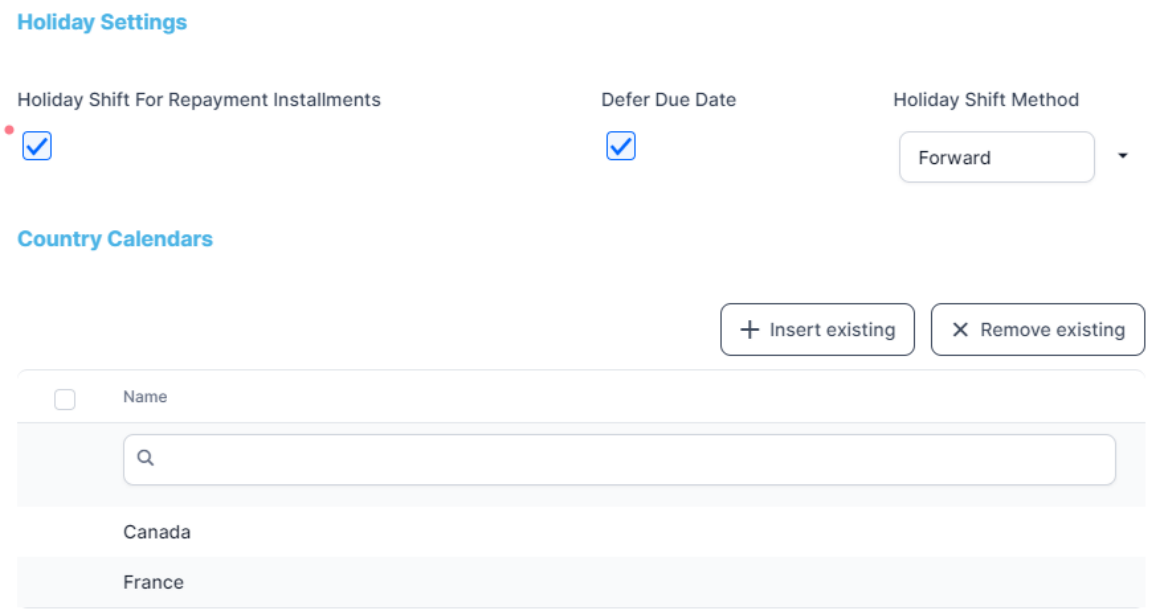
Loan Management performs the following validations for contract document records:

- The uploaded files' specifications follow FintechOS Platform's settings and restrictions regarding size and format, allowing .pdf,.doc,.docx,.els,.jpg,.jpeg,.xlsx,.dll,.ppt,.pptx,.txt,.png,.ttf,.xml file formats.
- If the contract document record is in **Signed** status, the record can't be deleted or updated, nor can its files be deleted.
- The name of the contract document record is unique, automatically generated by Loan Management. The naming convention is "the contract name + '-' + the selected document type + '-' + a unique document increment". For example, `5203 - Income Statement - 60`.
- The names of the selected files are not validated for uniqueness.

Treatment of Non Working Days for Schedule

Loan Management takes into consideration the defined non-working days when creating the repayment schedules. The calendar is the default holiday treatment for the loan contracts. In order to have flexibility on how to treat non-working days or public holidays, Loan Management uses a series of attributes and underlying logic. Thus, if the regular due date for loan repayment is defined as the 15th of every month and it falls on a non-working day according to the country calendar followed for that product, Loan Management can shift the due date backward or forward to the first working day, and then continue with the regular 15th of every month.

Manage the holiday settings during banking product definition, in the product's **Details** tab -> **Holiday Settings** and **Country Calendars** sections, as described in the Banking Product Factory user guide:



Furthermore, you also have a defer option that allows you to shift the due date without changing the amounts being collected.

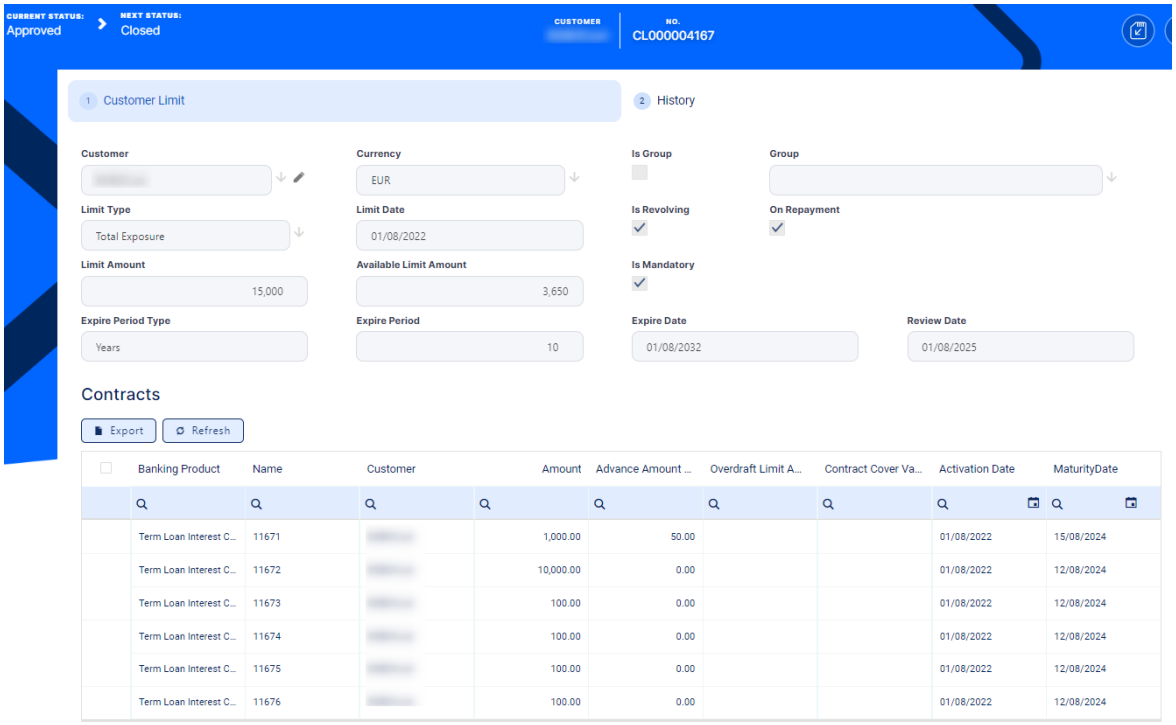
Working with Limits

Limits are used in order to have control over risk exposure. You may approve for a particular customer to take up to certain maximum amount across multiple products or you might want to approve types of products or even have a limit for a specific product alone. The granularity of control you can enforce is up to each financial institution's way of driving the business. Limits capability supports financial institutions to properly manage the company exposure, giving control over their exposure in the market and alignment with their market strategy.

IMPORTANT!
Setting up limits is mandatory before creating loan contracts. Approved and disbursed loan contracts affect the available limit amounts, so make sure you've configured the limits settings according to your financial institution's needs.

The idea behind working with limits is that when you insert the loan contract, Loan Management performs a validation against the available limit and, if the amount exceeds the available limit, you can decide to increase the limit or decrease the loan.

Once you create a limit record, Loan Management automatically identifies it and links the relevant contracts to it, you can see them on the **Customer Limit** page's **Contracts** section.

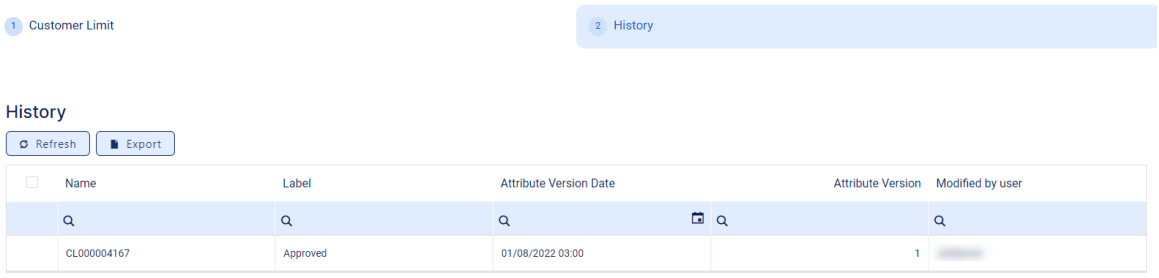


Some contracts can be linked to multiple limits if you have a complex limit structure approved: `Total Exposure` includes all contracts, then `Product Type Exposure` has also contracts that may be found under `Product Exposure`. You can only have one limit of each type of exposure valid in the system. Each exposure is limited in its own and by the higher level. If a customer that already has approved contracts becomes a member of a group, all its active limits are suspended. The same applies when excluding a customer from a group. Read more about group and customer exposure types.

Apart from the limit types available out-of-the-box, Loan Management allows you to add your own limit types based on roles associated to contract participants specific to your business, and use them throughout Loan Management with all the functionality of any other default limit type. Read more about managing limit types and role-based limits.

Similar to contracts, the limits can be revolving or not. For revolving limits, after performing a repayment, the amounts become available on the limit after closing the loan contract or after each repayment transaction.

As for contracts, the **History** tab allows you to you see when and who created each version of the limit and access the history version to spot the differences.



When versioning a limit, you can change certain details, while other can no longer be amended. Usually, you would change the amount and term/ review date for the limits.

You can have the limit in one currency and the underlying contracts in other currencies, unless you have product specific limits. Loan Management uses the available Exchange Rate to translate the amount into the limit currency and impact the usage and available figures.

Jobs, System Parameters, and Reports for Limits

Loan Management uses the following jobs to recalculate limits:

- `Start Of Day (SOD) Job` with the following services:
  - `Set Limit Available Amount Due To FX Change` - The service recalculates the available amount on limits depending on the currency's exchange rate on a specific day
  - `Set Contract Amount (Overdraft) Due To Plan Due Date Reached (Increase/ Decrease)` - The service increases/ decreases the limit amount on overdraft contracts that reached their reevaluation plan due date.
  - `Set Credit Facility Amount Due To Plan DueDate Reached (Increase/ Decrease)` - The service increases/

decreases the limit amount on credit facilities that reached their reevaluation plan due date.

- `End Of Day (EOD) Job` with the following services:
  - `Set Limit Expired` - The service sets the limits which are about to expire in the current day as Expired.
  - `Set Limit Available Amount Due To FX Changes` - The service sets the limit amounts available to all contracts due to exchange rates changes.

The system parameters used for limits management are listed below:

- `LimitMandatoryForIndividuals` - specifies whether Loan Management should validate the limits for individual customers or only validate them for legal entity customers.
- `DefaultIntervalLimitsReport` - represents the default number of months considered when running the reports within the **Limit Report** dashboard.
- `CreditFacilityLimitPercent` - represents the default limit of credit facility records.

Here are the reports that help you view the limits in Loan Management:

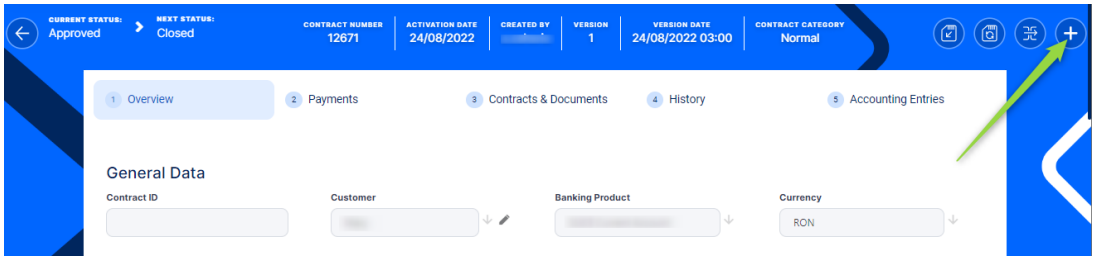
- **Customer Limits** - displays a list of the existing customer limit records, a list of the customer limit approval requests, and a button for adding new customer limits.
- **Limit Report** - displays different sections for expired limits, limits with available amount lower than 0, limits about to expire and limits to be reviewed, the latest two with the option to select the desired interval of dates.

Creating New Versions of Existing Loan Contracts

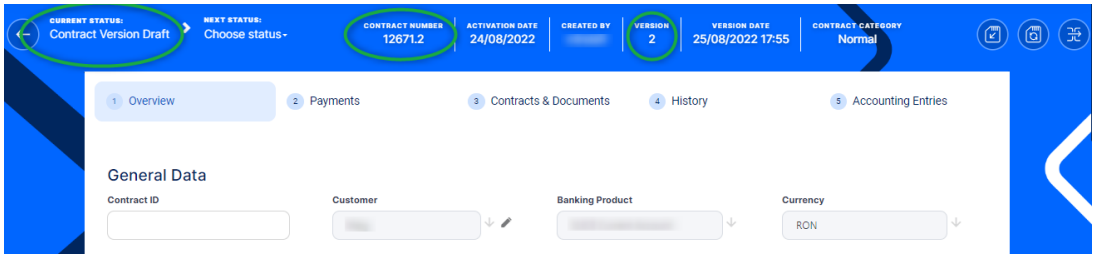
In Loan Management, the contracts are set up for versioning. Thus, if you want to update the details of an approved contract, then you must create a new version of the record.

To create a new version for a record with the **Approved** status, follow these steps:

1. While in the **Contract** page of the record selected for updates, click the **New Version** button.



2. View the new version of the contract created by Loan Management, with **Contract Version Draft** status.



3. Edit the desired fields in the **Overview** tab. You can only edit a set of fields for contracts based on specific banking products.
4. Click the **Save and Reload** button.

If you approve the contract in **Contract Version Draft** status, then the original record transitions into the **Contract Version Closed** status and the secondary version becomes the **Approved** currently active contract record.

Read more details about versioning a record on the How to Version an Entity Record page.

> **NOTE**
> Loan Management automatically creates a new version of a contract when the payment schedule is modified.
> When versioning a contract that has an imported repayment plan, the schedule recalculation is not mandatory. You can choose between recalculating and importing an updated repayment plan.

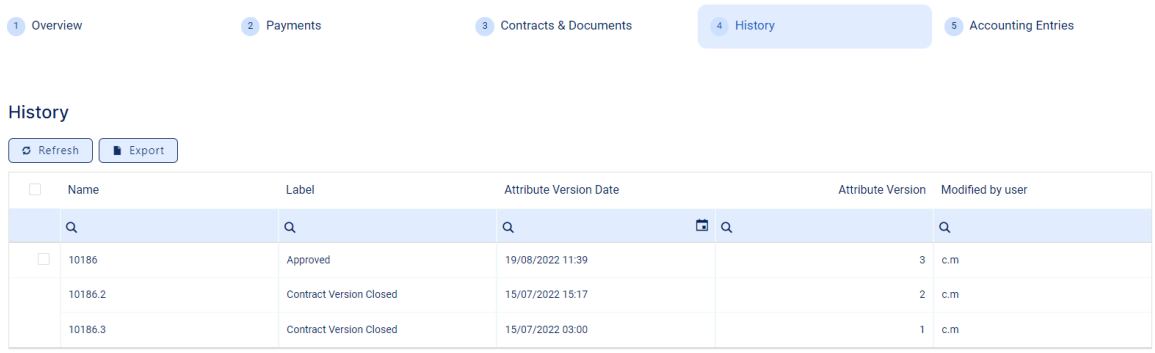
Possible Changes on New Loan Contract Versions

- The **Financed Amount** value can either be increased or decreased. The amount can be decreased with a number smaller than or equal to the **Available amount**. Financed amount can be increased up to the maximum value specified at banking product level.
- The **Current Account** attached to the loan contract can be changed to any other active account belonging to the customer.
- **Product Interest** can be changed to any other type set at banking product level.
- **Schedule Type** can be changed with any other type set at banking product level.
- **Contract Period** cannot exceed the maximum period set at banking product level.
- **Repayment Due Date** can be changed with any value between 1-31.
- **Grace Period** can be changed up to the maximum number of months set at banking product level.

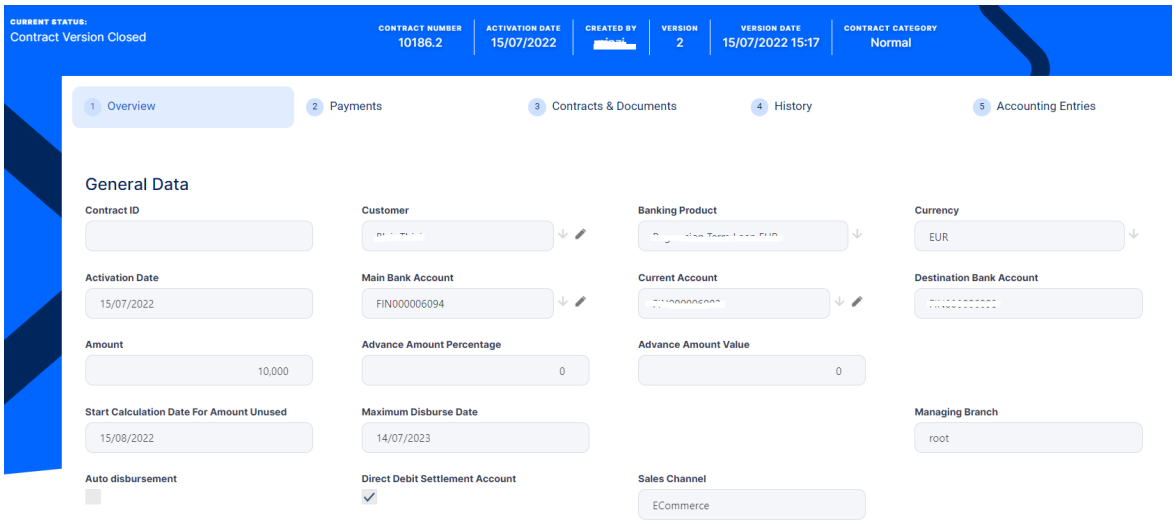
After any of the above changes, in order to approve the new version of contract, the **Contract Repayment Schedule** must be recalculated.

Viewing a Contract's History

You can view the versions of the contract, along with workflow status and the user who modified the record, in the contract's **History** tab.



A contract can have only one **Draft** version, one **Current** version, but it may have multiple **History** versions, which are displayed in this section. Here you can track the contract's life cycle and view older versions that are no longer active. Double-click a version in the list to view its details.



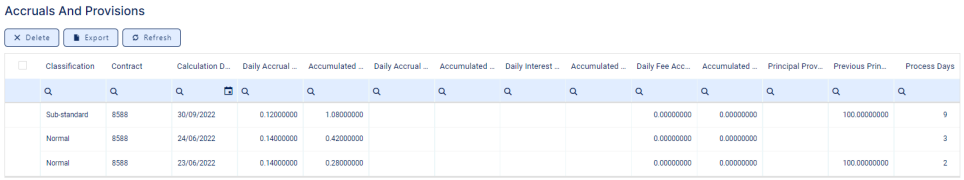
Viewing a Contract's Accounting Entries

You can view all the accounting entries, accounting totals, and accruals and provisions recorded for a contract within the **Accounting Entries** tab of the contract. These records are automatically generated by the system, after performing transactions for an approved contract.

View Accruals and Provisions

To view the records containing daily accrual and provisions, generated automatically by the system respecting the definition of the contract, product dimensions, system parameters and jobs, follow these steps:

1. Navigate to the contract's **Accounting Entries** tab > **Accruals and Provisions** section.

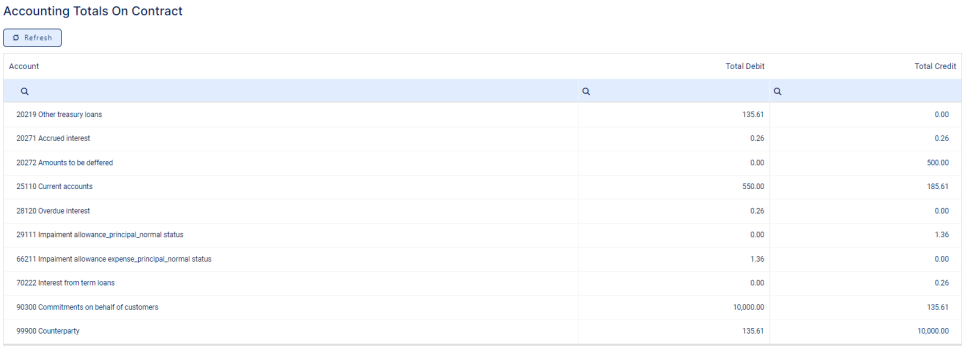


2. View the information displayed for each accrual and provision entry:
    - **Classification** - The classification of the accrual and provision entry. The classification is determined based on the records created in the **Loan Classification** menu. These records classify transactions based on the number of days since a repayment notification is overdue.
    - **Contract** - The number of the current contract.
    - **Calculation Date** - The date when the accrual and provision calculation was performed.
    - **Daily Accrual Interest** - The amount of interest accrued on that day.
    - **Accumulated Interest Accrual** - The total amount of interest accrued until that day.
    - **Daily Interest Provision** - The amount of interest provisioned on that day.
    - **Accumulated Interest Provision** - The total amount of interest provisioned until that day.
    - **Daily Fee Accrual** - The amount of fees and commissions accrued on that day.
    - **Accumulated Fee Accrual** - The total amount of fees and commissions accrued until that day.
    - **Principal Provision** - The amount of principal provisioned.
    - **Previous Principal Provision** - The previous amount of principal provisioned.
    - **Process Days** - The number of days processed.

View Accounting Totals on Contract

To view an overview of the total amounts specified in accounting records generated by the **Generate Accounting Entries** service in the Core Banking END OF DAY (CB) daily job, follow these steps:

1. Navigate to the contract's **Accounting Entries** tab > **Accounting Totals on Contract** section.


Accounting Totals On Contract

Refresh

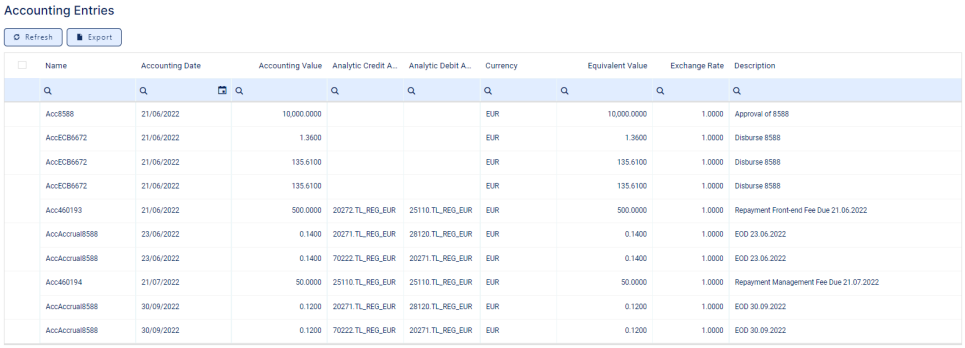
Account	Total Debit	Total Credit
20219 Other treasury loans	135.61	0.00
20271 Accrued interest	0.26	0.26
20272 Amounts to be deffered	0.00	500.00
25110 Current accounts	550.00	185.61
28120 Overdue interest	0.26	0.00
29111 Impairment allowance_principal_normal status	0.00	1.36
66211 Impairment allowance expense_principal_normal status	1.36	0.00
70222 Interest from term loans	0.00	0.26
90300 Commitments on behalf of customers	10,000.00	135.61
99900 Counterparty	135.61	10,000.00

2. View the information displayed for each total amount:
   - **Account** - The account where the operation was performed.
   - **Total Debit** - The amount which was debited from the account.
   - **Total Credit** - The amount which was credited to the account.

View Accounting Entries

To view the accounting for the transactions related to the loan contract generated by the **Generate Accounting Entries** service in the Core Banking END OF DAY (CB) daily job, follow these steps:

1. Navigate to the contract's **Accounting Entries** tab > **Accounting Entries** section.


Accounting Entries

Refresh Export

Name	Accounting Date	Accounting Value	Analytic Credit A...	Analytic Debit A...	Currency	Equivalent Value	Exchange Rate	Description
Acc8588	21/06/2022	10,000.0000			EUR	10,000.0000	1.0000	Approval of 8588
AccECB6672	21/06/2022	1.3600			EUR	1.3600	1.0000	Disburse 8588
AccECB6672	21/06/2022	135.6100			EUR	135.6100	1.0000	Disburse 8588
AccECB6672	21/06/2022	135.6100			EUR	135.6100	1.0000	Disburse 8588
Acc460193	21/06/2022	500.0000	20272.TL_REG_EUR	25110.TL_REG_EUR	EUR	500.0000	1.0000	Repayment Front-end Fee Due 21.06.2022
AccAccrual8588	23/06/2022	0.1400	20271.TL_REG_EUR	28120.TL_REG_EUR	EUR	0.1400	1.0000	EOD 23.06.2022
AccAccrual8588	23/06/2022	0.1400	70222.TL_REG_EUR	20271.TL_REG_EUR	EUR	0.1400	1.0000	EOD 23.06.2022
Acc460194	21/07/2022	50.0000	25110.TL_REG_EUR	25110.TL_REG_EUR	EUR	50.0000	1.0000	Repayment Management Fee Due 21.07.2022
AccAccrual8588	30/09/2022	0.1200	20271.TL_REG_EUR	28120.TL_REG_EUR	EUR	0.1200	1.0000	EOD 30.09.2022
AccAccrual8588	30/09/2022	0.1200	70222.TL_REG_EUR	20271.TL_REG_EUR	EUR	0.1200	1.0000	EOD 30.09.2022

2. View the information displayed for each accounting entry:
    - **Name** - The id of the accounting entry.
    - **Accounting Date** - The date when the entry was generated.
    - **Accounting Value** - The value of the accounting entry.
    - **Analytic Credit Account Code** - The code of the analytic credit account.
    - **Analytic Debit Account Code** - The code of the analytic debit account.
    - **Currency** - The currency of the accounting entry.
    - **Equivalent Value** - The equivalent value of the accounting entry expressed in the contract's currency.
    - **Exchange Rate** - The exchange rate between the accounting entry currency and the contract currency.
    - **Description** - The description of the accounting operation.

Changing the Borrower of a Loan

In specific circumstances, such as takeover or merged acquisition for SME customers, or divorce or inheritance for retail customers, the financial institution might have to perform a credit novation or cession. In other words, this means that an active loan contract should be altered by changing the borrower of that contract, effectively transferring to the new owner all benefits and burdens of the loan.

IMPORTANT!
Your financial institution most likely has an elaborated novation approval internal process. Make sure you follow exactly that process, acquire all the necessary documents and approvals, and implement the process as defined!

Loan Management allows you to perform such changes to contracts, either through the user interface, or by orchestrating these actions through API integration. In a few words, you can do this by creating a loan for the new owner and use its amount to repay the previous borrower's loan, which is then closed.

Follow the steps below to perform the changes needed in order to alter the owner in an active contract:

1. Check if the new owner is recorded as a customer in the system.
   If not, then create a new customer record for the new owner.
2. Verify if the new owner has at least one settlement account in the same currency as the existing loan contract.
   If not, create a new current account for them in the same currency, to be used as a settlement account in the new loan contract. Remember to approve the contract.
3. Make sure that the new owner has an available limit with an amount at least equal to the remaining amount of the existing loan contract for non-revolving loans, or at least equal to the total amount for revolving loans.
   If not, create and approve a limit for the new owner to cover the existing loan contract's needed amount.
4. Create a new loan contract for the new owner for the approved amount.
5. Disburse the required amount into the settlement account, if it didn't happen during contract approval (for auto-disbursed loans).
6. Use the amount from the new owner's settlement account to perform a withdrawal transaction.
7. Use the withdrawn amount to perform a top-up transaction into the settlement account of the original loan.
8. If there are overdue amounts on the previous loan, make sure the internal direct debit settlement account setting is active, so the missed notifications are automatically covered.

9. If needed, perform an early repayment to cover any outstanding notifications.
10. Close the loan contract belonging to the previous borrower.
11. If the previous borrower has no other active loans, you can close their available limits or decrease it.
12. Close the settlement account of the previous borrower.
13. In special circumstances such as bankruptcy, takeover and so on, you may consider retiring the previous borrower's customer record by changing its status to `Blocked` or `Closed`.

That's it! You've created a new loan contract for another borrower, and with the funds made available you've paid back all remaining amounts of the old loan. In the process, you've tied all the loose ends, closing settlement accounts, limits and even customer records. In the future, the new borrower has to repay the due amounts as specified in the new loan contract.

> **NOTE**
> For improved traceability purposes, during the Loan Management implementation process, your financial institution should consider extending the contract to catch the id of the closed contract and the reason for closure.
> Also consider creating a special report for the accounting notes to show any future accounting entries. The report could be useful if you need record continuity from the old to the new contract, when you might want to copy the early repayment/ disbursement transactions and rename them as repay/novation and disburse/novation.

Current Accounts

A current account is a bank account where you can store and withdraw money, enable it for debit/ credit transactions internally (disburse/ repay loan, transfer between accounts) or via integrated solutions for card management or payments. Loan Management enables you to create contracts based on current account products and to manage such contracts.

In Loan Management, current accounts are mainly used as **settlement accounts** - a customer' bank account opened for the purpose of transferring the funds granted through a loan, to repay the due installments, and to perform deposit funding and liquidation. They are also used as **guarantees** - a collateral offered by the customer to the financial institution for borrowing larger amounts. In this case, the financial institution may request the customer to maintain a specific balance in the current account used as guarantee.

Read about the operations that you can perform for the current accounts with or without the overdraft functionality.

Current Account Contract Life Cycle and States 378
Creating a Current Account 382
Approving a Current Account 400
Rejecting a Current Account 403
Working with Covenants 404
Working with Participants 408
Working with Contract Classification 410
Applying Fees and Commissions 412
Working with Documents 415
Performing Transactions on Current Accounts 418
Processing Overdraft Repayments 431
Manually Capture Notifications 452
Closing a Current Account 457
Creating New Versions of Existing Current Account Contracts 462
Viewing a Contract's History 464
Viewing a Contract's Accounting Entries 465

Current Account Contract Life Cycle and States

The four-eyes principle is applicable for all contracts in FintechOSLoan Management, meaning that a record should be approved by a second financial institution employee, with higher authorization rights. This is enabled via approval task FintechOS Platform

capabilities and thus it is also a financial institution's responsibility to set proper security roles and access rights to its users, in order to make sure that the same user can't insert and also authorize the same record.

A contract record has the following business workflow statuses:

- **Draft** - the status of a newly created contract record that was not yet sent for approval. While in this status, you can edit some fields, but you can't add events (payments) to it. Send the record to approval after editing all the necessary details.
- **Pending** - this is a system status applied to contracts sent for approval, but not yet approved. No updates are available in this system status.
- **Approved** - the status of a contract record after being authorized by a user with contract approval competencies. While in this status, you cannot edit the record's details, but you can add events to it within the **Payments** tab. If you need to alter the contract's details, create a new version based on the current contract.

  > **NOTE**
  >
  > Each event must also be approved by a user with contract approval competencies, otherwise, the transaction is not performed by the system.
  >
  > New contract approval is blocked by Loan Management if the customer has overdue days >= the value of the `DelayDaysForBlockNewContractApproval` parameter.

- **Closed** - the last status of a contract, after manually closing it or after creating a new version based on the current version. No updates are allowed on the record.
- **Canceled** - the status of a contract after manually canceling it straight from the **Draft** status. No updates are allowed on the record.

> **NOTE**
>
> Change the contract's status to **Approved** so that the customer can use the contract and in order to apply transactions to it.

Contract Versioning

Loan Management allows you to create new versions for an existing contract if you need to modify an existing approved contract. New versions are automatically created when the payment schedule is modified - that implies any increase/ decrease, change of costs, reschedule or payment holiday transactions.

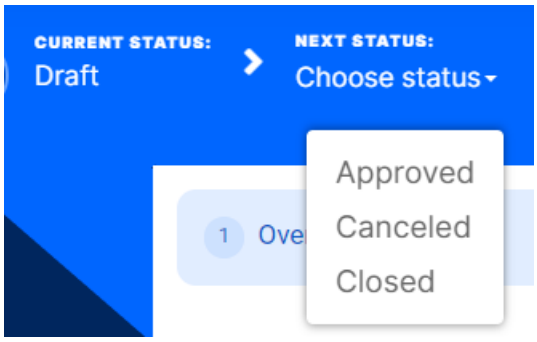
A contract version can have the following statuses:

- **Contract Version Draft** - the status of a newly created contract version record that was not yet sent for approval. While in this status, you can edit some fields, but you can't add events (payments) to it. Send the record to approval after editing all the necessary details.
- **Approved** - the status of a contract version record after being authorized by a user with contract approval competencies. While in this status, you cannot edit the record's details, but you can add events to it within the **Payments** tab.
- **Contract Version Closed** - the last status of a contract version, after manually closing it or after creating another new version based on the current version. No updates are allowed on the record.

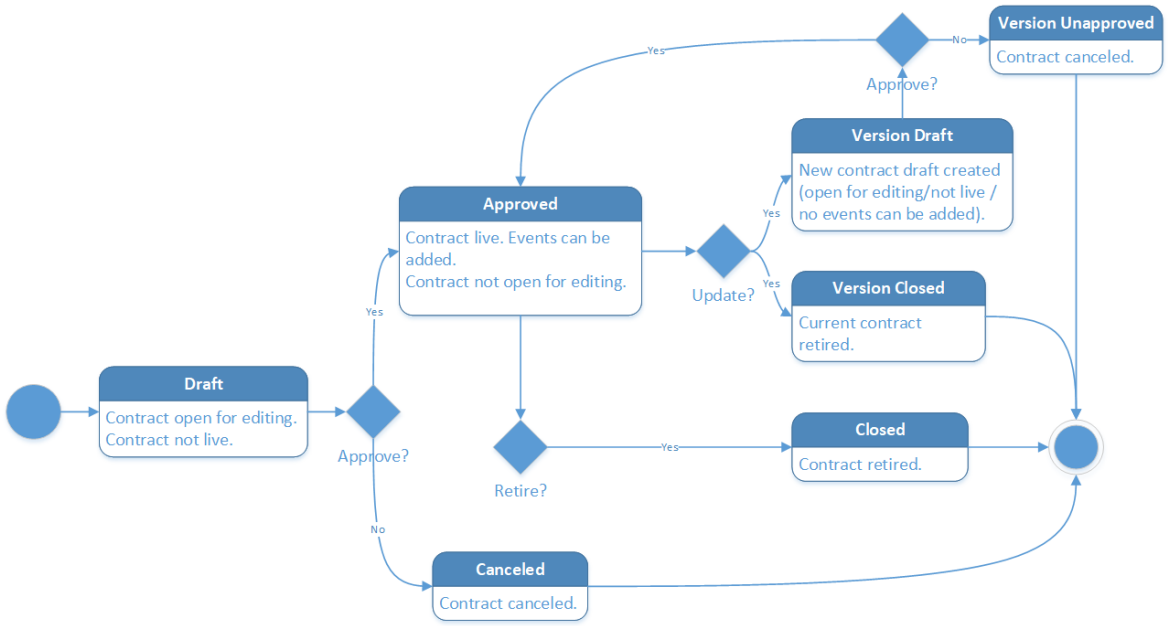
> **NOTE**
> All existing versions of the contract in **Contract Version Draft** status are automatically changed to **Contract Version Closed** when a payment event outside regular schedule is approved for that contract.

Changing Contract Statuses

You can manage a contract's life cycle by changing its status from the top right corner of the screen.



The contract status transitions are illustrated below:



Note that:

- Once a record is live, its settings can no longer be modified.
- If you want to update the details of a live contract, you must create a new contract version.
- When you create a new contract version, the current version is retired and moved to history; no updates are allowed on the retired version.
- Every contract version starts in a draft state and must go through an approval process before going live.
- Only one version of a contract can be live at one time.

> **NOTE**
> As a best practice, new records or new versions of existing records created on a specific day should be approved on the same day.

Creating a Current Account

A current account is a bank account where you can store and withdraw money, enable it for debit/ credit transactions internally (disburse/ repay loan, transfer between accounts) or via integrated solution for card management or payments.

A current account with an attached overdraft functionality allows customers to withdraw funds from the account even if the available balance goes below zero. If there is a prior agreement with the account provider for an overdraft, and the amount overdrawn is within the authorized overdraft limit, then interest is normally charged at the agreed rate. If the negative balance exceeds the agreed terms, then additional fees may be charged and higher interest rates may apply. The customer may use the account beyond their available balance (credit balance) and may have a debit balance as low as the approved overdraft/ limit. The overdraft can be added on top of existing current accounts.

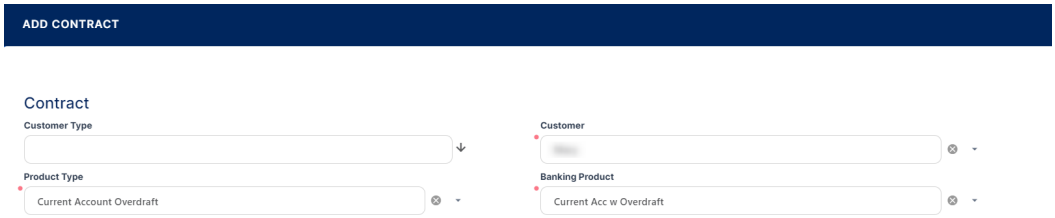
The overdraft feature of the current account may expire, in which case the contract continues to function as a current account. If needed, the overdraft feature can be reactivated in the future within the same contract by editing the contract and creating a new version. If the overdraft feature is extended by creating a new version of the contract, the repayment schedule is also updated. In cases when the feature is extended after a period of expiry, then the repayment schedule has a missing period equivalent with the period when overdraft was expired. All overdue amounts from previous overdraft notifications generated for the same current account contract are covered automatically, decreasing the balance and available limit amount of the new overdraft. The overdraft amount can fluctuate seasonally or be reduced or increased according to a schedule, defined in the Contract Reevaluation Plans section of the contract.

> **NOTE**
> Before creating a current account contract, make sure that the customer is recorded in Loan Management.

To create a new contract:

1. Add Minimum Contract Data

1. Open the **Contracts** page as described in the Managing Contracts section.
2. Click the **Insert** button to display the **Add Contract** page is displayed, the initial page when you insert any type of contract.



3. Fill in the following fields:

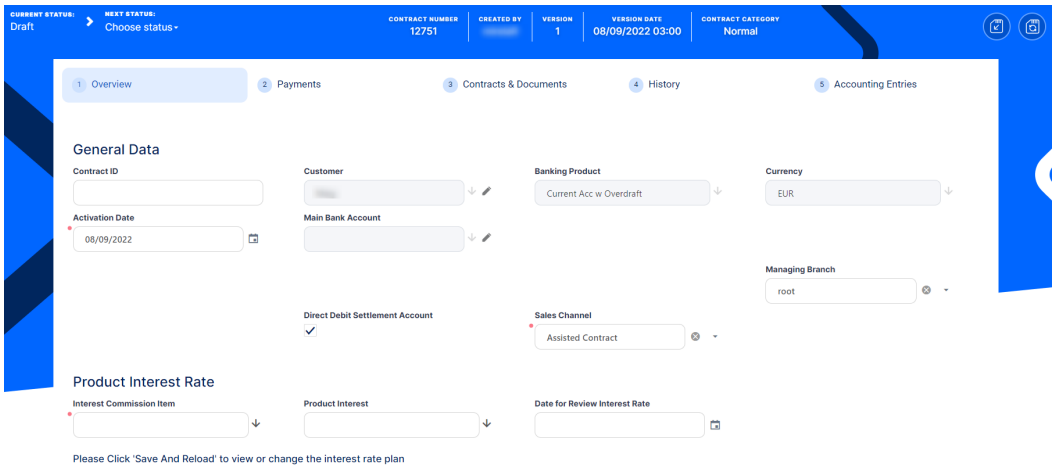
- **Customer Type** - Optionally, select the type of the customer for the contract, to filter the displayed customers in the next field.
- **Customer** - Select from the list the customer for whom you are creating a contract.
- **Product Type** - Select from the list the product type to filter the list of banking products accordingly.
- **Banking Product** - Select from the list the desired banking product.

> **NOTE**
> Be careful when choosing the values for the previously mentioned fields because you can't modify them after saving the contract!
> Make sure that you select `Current Account` in the Product Type field and a `Current Account` banking product in the Banking Product field. If you want to use the overdraft functionality now or sometime in the future for this current account contract, select `Current Account Overdraft` in the Product Type field and a `Current Account with Overdraft` banking product in the Banking Product field.

4. Click the **Save and Reload** button.

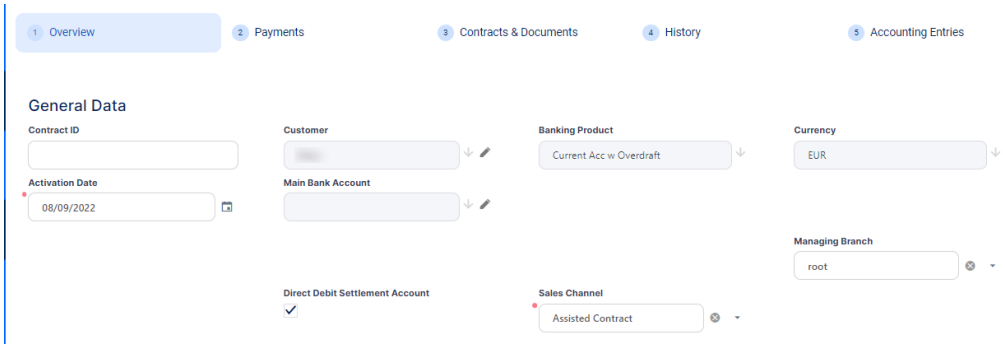
Loan Management saves the contract in **Draft** status, with minimum default information, such as an auto-generated contract number, created by, version

and version number. The previously provided details are kept on screen in the **General Data** section, but they are no longer available for update. The **Currency** has been updated from the banking product level.



Proceed to the next steps where the details about the contract are captured and validated against the underlying product, setting the basic elements for the creation of a contract such as customer, banking product, activation date, interest rate, fees, and contract classification, within the newly displayed **Overview** tab.

2. Add General Data to the Contract



1. Fill in or modify, or view the following information:
    - **Contract ID** - Enter a contract ID other than the contract number generated automatically by Loan Management when you saved the contract.
    - **Main Bank Account** - When the contract is approved, this account is created automatically for the bank defined as `Main` in Loan Management, within the **Loan Management Operational** > **Bank** menu. In order to generate an account number, a rule must be defined during the Loan Management implementation phase (example: branch code + incremental sequence number). Until contract approval, no information is displayed here, and you can't edit the field.
    - **Managing Branch** - This represents the branch of the organization where the contract was created. Suppose you work in a branch or credit center, and you need cases to be linked to a specific location so that you can properly allocate them for further actions. It is automatically completed at contract saving time, but you can select another branch from the list.
    - **Direct Debit Settlement Account** - Select this checkbox if the automated settlement of repayment notifications (the direct debit settlement account) functionality is turned on at the contract level. The value of the checkbox was set at the banking product level, but it can be modified at the contract level. The checkbox can be edited in all the statuses of a contract except `Version Closed`, `Closed`, and `Canceled`.

      > **NOTE** The `Direct Debit Settlement Account` setting at the customer level takes precedence over the setting at the contract level when creating new contracts. For existing contracts, Loan Management applies the setting configured within the

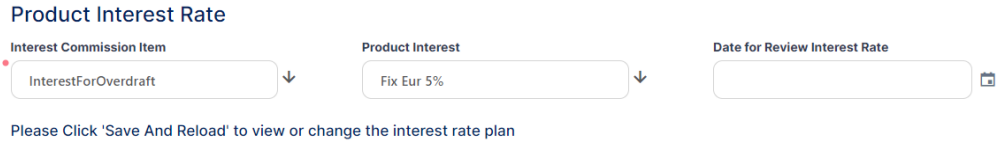
> `CustomerToContractDirectDebitSettlementAcc` system parameter.

- **Sales Channel** - Select the channel through which the contract is created.

2. Click the **Save and Reload** button.

3. Manage Product Interest Rate for the Contract

Enter the details about the Product Interest Rate applied to the contract. Depending on the product definition again, you have a list of interest definitions that you can bring along to the contract.



To manage the product interest rate as it must be applied to this contract:

1. Fill in or modify the following fields:

- **Interest Commission Item** - This field is automatically completed with the interest & commission item defined at the product level, if only one item is found at the product level. If the selected product has more items, you must select one from the list.
- **Product Interest** - Select from the interest to be applied for this contract. Only the interests associated to the selected banking product are displayed within the list. Penalty interests cannot be selected here.
- **Date for Review Interest Rate** - Enter the date for reviewing the interest rate applicable. For variable interest, this field is automatically completed with the `Reference Rate Date` + `Reference Interest Period` of the underlying interest definition, from the base type interest attached to variable interest. You can edit this field. For

months where the date is overlapped, the last day of the month is used for the calculation.

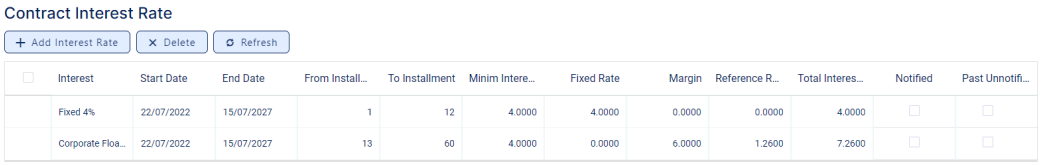
2. Click the **Save and Reload** button.

> **NOTE**
> Fill in any other mandatory fields from the **General Data** and **Repayment Overview** sections, otherwise you can't successfully save the contract.

4. Manage Contract Level Interest & Penalty Interest Rates

Define the information about the contract interest rate (or rates, if you selected a `Collection` type interest rate in the previous **Product Interest Rate** section) in a table format, in the section **Contract Interest Rate** section, which appears only after saving the selected product interest rates.

You can edit the tables cells, so you can customize the interest rates selected at the product level, if the interest and commission list was defined as `negotiable`, to obtain the desired interest rates configuration at the contract level. You can also add or delete interest rates, using the **Add Interest Rate**, respectively the **Delete** buttons above the tables. Thus, the tables enables you to work with multiple interest rates at the contract level.



> **NOTE**
> The information disappears if you change the product interest, tenor, first due date, maturity date, contract period, or activation date. In this case, save the contract again to display the updated information.

To customize the information specific to each of the contract's **interest rates**:

1. In the **Contract Interest Rate** section, edit the existing information that was automatically completed based on your product interest rate selections:

- **Interest** - Automatically completed with the interest (or interests, for `Collection` type product interest rate) selected in the previous **Product Interest Rate** section. You can select from the drop-down list the interest to be applied for this contract. Only the interests associated to the selected banking product are displayed within the list. Penalty and overdraft interests cannot be selected here. Depending on the selected interest, other fields can be displayed to be filled in.
- **Start Date** - The interest's start date, automatically completed with the contract's activation date.
- **End Date** - The interest's end date, automatically completed with the contract's maturity date.
- **From Installment** - The first installment for which this interest is applied to the contract.
- **To Installment** - The last installment for which this interest is applied to the contract.
- **Minimum Interest Rate** - This read-only cell is automatically completed with the minimum interest rate applicable for the contract, defined at the banking product level.
- **Fixed Rate** - The fixed rate of the interest. You can only change it if the interest at the banking product level was marked as `Is Negociable`.
- **Margin** - This cell is automatically completed with the margin of the previously selected product interest. You can only change it if the interest at the banking product level was marked as `Is Negociable`. If the product interest was not selected, you can manually enter the margin.
- **Reference Rate** - This read-only cell is automatically completed with the interest type's definition's reference rate valid at the previously selected date.

- **Total Interest Rate** - This read-only cell is automatically completed with the calculated total interest rate of the previously selected product interest and any values entered for margin and reference rate. If the product interest was not selected or if the interest at the banking product level was marked as `Is Negociable`, you can manually enter the interest rate.
- **Notified** - This is a read-only checkbox. For contracts in `Version Draft` status, it shows you whether the installments range shown on this table line was already notified or not.
- **Past Unnotified** - This is read-only cell read-only checkbox. For contracts in `Version Draft` status, it shows whether there are days that already passed from the current month's not yet notified installment, days for which you can't change the interest rate.

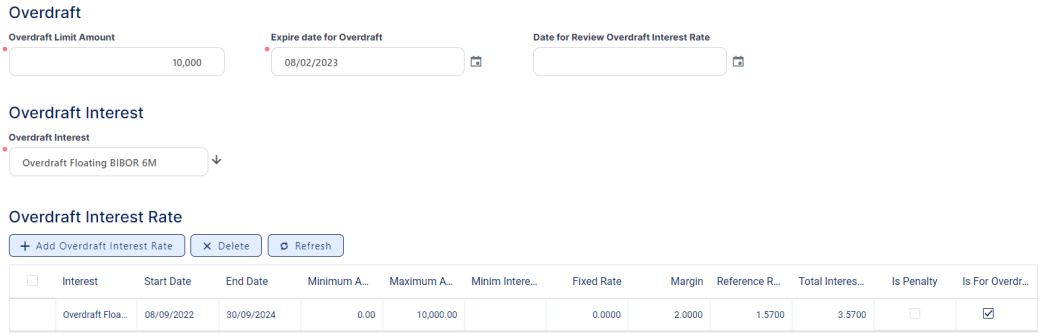
2. After performing the desired changes, make sure that the interest rate (s) cover the entire tenor of the contract, from activation date until maturity date, and there are no overlapping intervals, otherwise an error prevents you from approving the contract.
3. Click the **Save and Reload** button.

> **NOTE**
> Loan Management takes into consideration and recalculates the schedule even in cases when the interest is of `Collection` type and there are multiple interest rates within a single repayment interval due to changes in the interest definition between due dates. In such cases, Loan Management keeps the interest accrued up until the change occurs and rebuilds the remaining of the repayment schedule.

5. (Only for Current Accounts with Overdraft) Enter Overdraft Information for the Contract

For current accounts with overdraft functionality, in the **Overdraft** section you should enter the overdraft limit amount and expiation date, in the **Overdraft Interest** section select an interest applicable for the overdraft amount, and then in the **Overdraft Interest Rate** section amend the overdraft interest rates applicable for this contract, so that Loan

Management can properly build the repayment schedule.

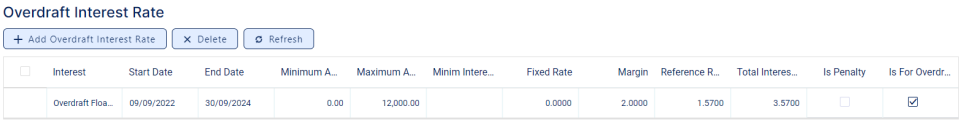


1. In the **Overdraft** section, fill in or modify the following information specific to the contract's overdraft functionality:
    - **Overdraft Limit Amount** - Enter the limit for the overdraft amount applicable for this contract. If the overdraft limit amount is greater than 0, then all the fields related to the overdraft interest rate are mandatory to be completed, as well as a newly displayed field, **Start Calculation Date For Amount Unused**, back in the **General data** section.
    - **Expire Date for Overdraft** - Enter the date until when the overdraft functionality is active for this contract. On the expiry of the overdraft limit, any used amount and the underlying interest becomes due. A repayment notification is generated in case the current account does not hold enough balance to cover the interest.

      > **NOTE**
      >
      > The due dates of all installments within the repayment schedule of current account with overdraft contracts fall before the expire date for overdraft.

    - **Date for Review Overdraft Interest Rate** - Optionally, enter the date for reviewing the interest rate applicable for the overdraft.

2. In the **Overdraft Interest** section, select from the list the overdraft interest to be applied for the overdraft amount of this contract. Only the interests marked as `Is For Overdraft` are displayed within the list.
3. Click **Save And Reload** to view or change the overdraft interest rate plan.
4. In the **Overdraft Interest Rate** section, configure the contract's overdraft interest rate in a table format, similar to the one in the **Contract Penalty Interest Rate** section.



> **NOTE**
>
> The information disappears if you change the overdraft interest.
>
> In this case, save the contract again to display the updated information.

You can edit the tables cells, so you can customize the overdraft interest rates, if the interest and commission list was defined as `negotiable`. You can also add or delete overdraft interest rates, using the **Add Overdraft Interest Rate**, respectively the **Delete** buttons above the tables. Thus, the tables enables you to work with multiple overdraft interest rates at the contract level.

To customize the information specific to each of the contract's **overdraft interest rates**, edit the existing information that was automatically completed based on your overdraft interest rate selection:

- **Interest** - Automatically completed with the interest (or interests, for `Collection` type product interest rate) selected in the previous **Overdraft Interest** section. You can select from the drop-down list the

overdraft interest to be applied for this contract. Only the overdraft interests associated to the selected banking product are displayed within the list.

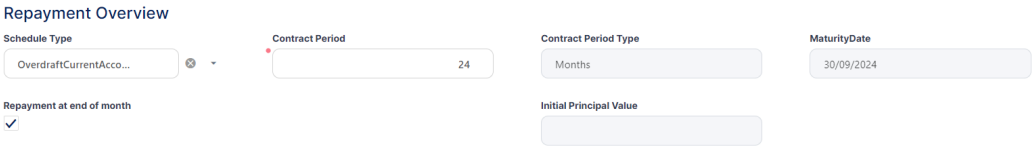
- **Start Date** and **End Date** - The overdraft interest's start and end date, automatically completed with the contract's activation, respectively maturity date.
- **Minimum** and **Maximum Amount** - The overdraft interval to which the overdraft interest is applicable.
- **Minimum Interest Rate** - This read-only cell is automatically completed with the minimum interest rate applicable for overdraft amounts for this contract, defined at the banking product level.
- **Fixed Rate** - The fixed rate of the overdraft interest. You can only change it if the interest at the banking product level was marked as `Is Negociable`.
- **Margin** - This cell is automatically completed with the margin of the previously selected overdraft interest. You can only change it if the overdraft interest at the banking product level was marked as `Is Negociable`. If the overdraft interest was not selected, you can manually enter the margin.
- **Reference Rate** - This read-only cell is automatically completed with the overdraft interest type's definition's reference rate valid at the previously selected date.
- **Total Interest Rate** - This read-only cell is automatically completed with the calculated total overdraft interest rate of the previously selected overdraft interest and any values entered for margin and reference rate. If the overdraft interest was not selected or if the overdraft interest at the banking product level was marked as `Is Negociable`, you can manually enter the overdraft interest rate.

After performing the desired changes, make sure that the overdraft interest rate(s) cover the entire tenor of the contract, and there are no overlapping intervals, otherwise an error prevents you from approving the contract.

5. Back up in the **General Data** section, in the newly displayed **Start Calculation Date For Amount Unused** field, select the date when the amount not drawn from the overdraft limit amount starts to be calculated. This field is displayed and required to be filled in only when the `Overdraft Limit Amount` value is `>0`.
6. Click the **Save and Reload** button.

6. Enter Repayment Information for the Contract

In the **Repayment Overview** section you should enter term and schedule type so that Loan Management can properly build the repayment schedule.



1. Fill in or modify the following information specific to the contract's repayment schedule:
   - **Schedule Type** - Select the payment schedule type to be used to calculate the installments of this contract. You can select one of the payment schedule types associated to the underlying banking product in the **Details** tab > **Associated Payment Schedule Types** list. Loan Management uses the schedule type to build the repayment plan with equal installments or linear payments, include fees on the schedule and arrive to the day basis to be used for interest calculation (30/360).
   - **Contract Period** - Edit the term of the contract that was automatically completed with the number defined at banking product level, according to your needs. The contract period is used together with Contract Period Type and Periodicity Type.

They all need to be in sync and also in sync with the schedule definition itself, and if there are multiple definitions allowed on the product, make sure to pick those working together.

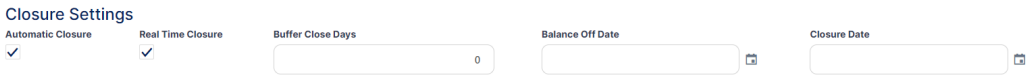
- **Contract Period Type** - This field is automatically completed with the contract period type as it was defined at banking product level. You can't edit this value.
- **Maturity Date** - This field is automatically completed with the calculated contract maturity date.
- **Initial Principal Value** - This field is automatically completed with the value of the principal within an installment. The field is displayed and can be filled in if the selected schedule type is of type `Equal Principal`. You can edit this value. If at the selected payment schedule type's level the `Installment Value Custom` field is `False`, then the `Initial Principal Value` field at the contract level is read only.
- **Repayment at end of month** - If you select this checkbox, then the due day of the contract is automatically set to the last day of the month, and the repayment schedule is calculated with an installment in the last day of month.

2. Click the **Save and Reload** button.

7. Amend Closure Settings

If allowed from the product definition, you can amend the closure settings of the contract, the way Loan Management should behave once any due amounts are repaid and the contract can be closed. Most of the times this is not something that you have to access, but it adds extra flexibility at the contract level. This may prove useful if you suspect there may be reasons to keep a contract open for some time post recovering all amounts for instances when there may appear claims of funds (SEPA DD) or other similar cases.

The **Closure Settings** section is only displayed for contracts based on banking products having the `Closing Is Flexible = True` setting.



To amend the closure settings brought from product level here at the contract level:

1. Fill in or modify the following fields:
    - **Automatic Closure** – If selected, Loan Management automatically closes the contract once all other conditions are met. This field is automatically completed with the value defined at the banking product level, but you can modify it.
        - Select this checkbox to instruct Loan Management to close the contract automatically when the available amount becomes zero and there are no further amounts to be recovered, and after the number of days set as buffer before closure pass and `Closure Date = Current Date`.
        - Deselect it to instruct Loan Management to keep the contract open, regardless of the fulfillment of its maturity and balance criteria, waiting to be manually closed by changing its status to `Closed`.

        > **NOTE**
        >
        > You can perform contracts events as specified in the **Allowed Transactions** section of the banking product, plus manual closure while the contract is pending closure. Performing any other transactions displays an error message.

    - **Real Time Closure** – If you select this checkbox, when the amounts become zero and the product is not a revolving one, the contract is closed automatically. If `Real Time Closure = True`, then `Buffer Close Days = 0` and `Automatic`

`Closure = True`. For more details about the real-time closure, see Close Contracts RealTime(CB) Job.

- **Buffer Close Days** - Enter the number of days used as buffer before automatically closing the contract. If `Buffer Close Days > 0`, then `Real Time Closure = False`. Loan Management waits the entered number of days after the contract's balances reach zero, and at the end of that day the contract is closed.
- **Balance Off Date** – This is a system maintained field and it is populated with the date on top of which Loan Management adds the Buffer Close Days to arrive to the Closure Date.
- **Closure Date** – This is a system maintained field and holds the date when the contract is closed. For automatic closure, the date is calculated by Loan Management as Balance Off Date + Buffer Close Days.

2. Click the **Save and Reload** button.

8. Check Other Details Pre-Filled Based on Product Definition

Once you defined the mandatory details, then saved and reloaded the contract, Loan Management updates some of the next sections on the page, based on product definitions:



Loan Management brings the **Contract Participants**, where you can add participants to the contract, like Guarantors, Co-Debtors, etc. There may be cases when some roles are mandatory for a product. Those are detailed in a separate section. If there is a mandatory role defined in the banking product definition, Loan Management displays an error on trying to approve the contract.

Contract Tranches is a section where you can implement progressive access to the funds. This is valuable in case of loans granted for investment projects where you can know upfront that there is a plan for the project and payments need to happen for each stage of the project, those stages being known from the start.

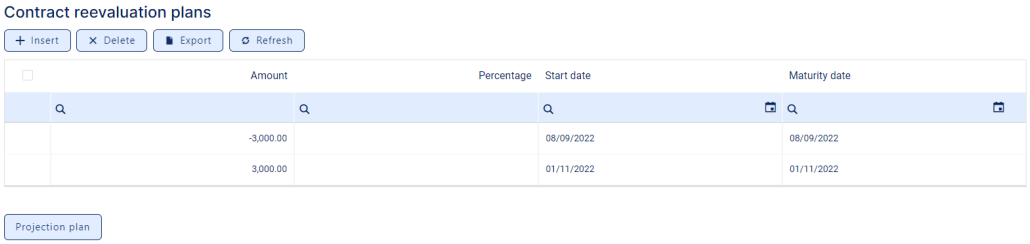
Another important section brought from the product definition is the **Fees & Commissions**. Depending on the system setup, you are allowed or not to amend fees and commissions in this section.

Contract Covenants section displays the covenants that applicants must abide by after getting the contract, configured at the product level. Such conventions are usually applicable for corporate customers that must meet certain requirements in order to continue to receive disbursements and not only: submit balance sheet every x months, have account turnover of at least x percent from average monthly turnover, provide other relevant documents from authorities. In this section, you can manage covenants for the contract. These covenants would need to be monitored procedurally; Loan Management doesn't have the logic in place to implement automated processes.

You can use the **Contract Classifications** section to capture various classifications that might be relevant for the financial institution for that loan at a moment in time. It is a placeholder for such details and there is no automated logic in place to update them. In implementation this can be used for other developments if required.

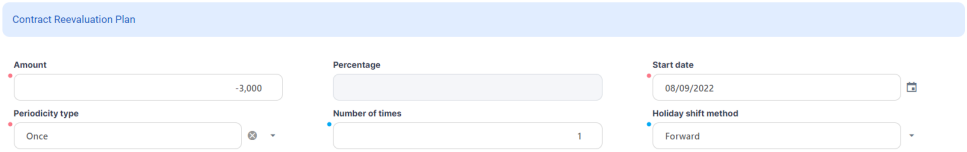
9. (Only for Current Accounts with Overdraft) Define Overdraft Limit Reevaluation Plans

For current account contracts with overdraft functionality, in the **Contract Reevaluation Plans** section, you can define the increase or decrease of the overdraft limit amount and of the customer limit attached to the contract, according to the customer's seasonal needs. For example, for an approved overdraft limit for 12 months, after 6 months the limit has to be decreased because the customer does not need the entire limit and wants to reduce the costs of the overdraft functionality. Similarly, the overdraft limit can be increased, for example for a working capital requirement, the bank can grant the limit in January, but the company works in tourism and the cash flow needs are higher in summer, thus the bank needs to increase the overdraft.



To add a reevaluation plan to a contract based on a current account with overview, follow these steps:

1. Click **Insert** to open the **Contract Reevaluation Plan** page.
2. Fill in the following fields:



- **Amount** - Enter the amount with which the overdraft limit has to increase or decrease. Use negative values for decrease and positive values for increase. This field cannot be filled in if the `Percentage` field was completed.
- **Percentage** - Enter the percentage with which the overdraft limit has to increase or decrease. Use negative values for decrease and positive values for increase. This field cannot be filled in if the `Amount` field was completed.
- **Start date** - Enter the start date when the overdraft limit increase or decrease should be performed.
- **Periodicity type** - Select the periodicity type for applying the overdraft limit increase or decrease. The possible values are: Once, Weekly, Monthly, Bimonthly, 4 Weeks, Trimestrial, Semestrial and Annual.
- **Number of times** - Enter how many times the overdraft limit increase or decrease should be performed.
- **Holiday shift method** - Select the desired holiday shift method from the possible values: Forward or Backward.

3. Click the **Save and Close** button.

   When the utilized amount is greater that the resulted available limit, a new repayment schedule detail is added to the repayment notification, displaying the difference between utilized amounts – available limit amount on the Principal column of the notification. The repayment notification notifies the principal amount and tries to collect it.

4. Click the **Projection Plan** button to view the projection of the overdraft limit reevaluation plan. A `.pdf` file containing a detailed overview of the overdraft limit fluctuations in time is automatically downloaded by your browser.

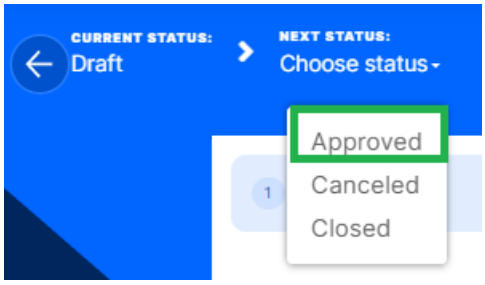
After defining the relevant details of the contract, proceed to **contract approval**.

Approving a Current Account

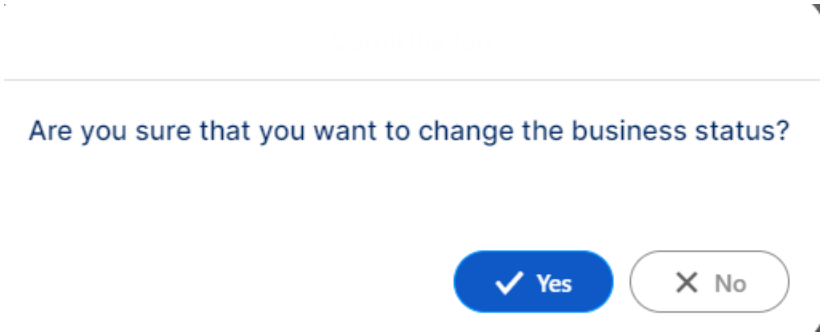
You can perform the approval either from a customer journey flow via API integration or from the Loan Management user interface.

After defining the relevant details of the contract, proceed to contract approval:

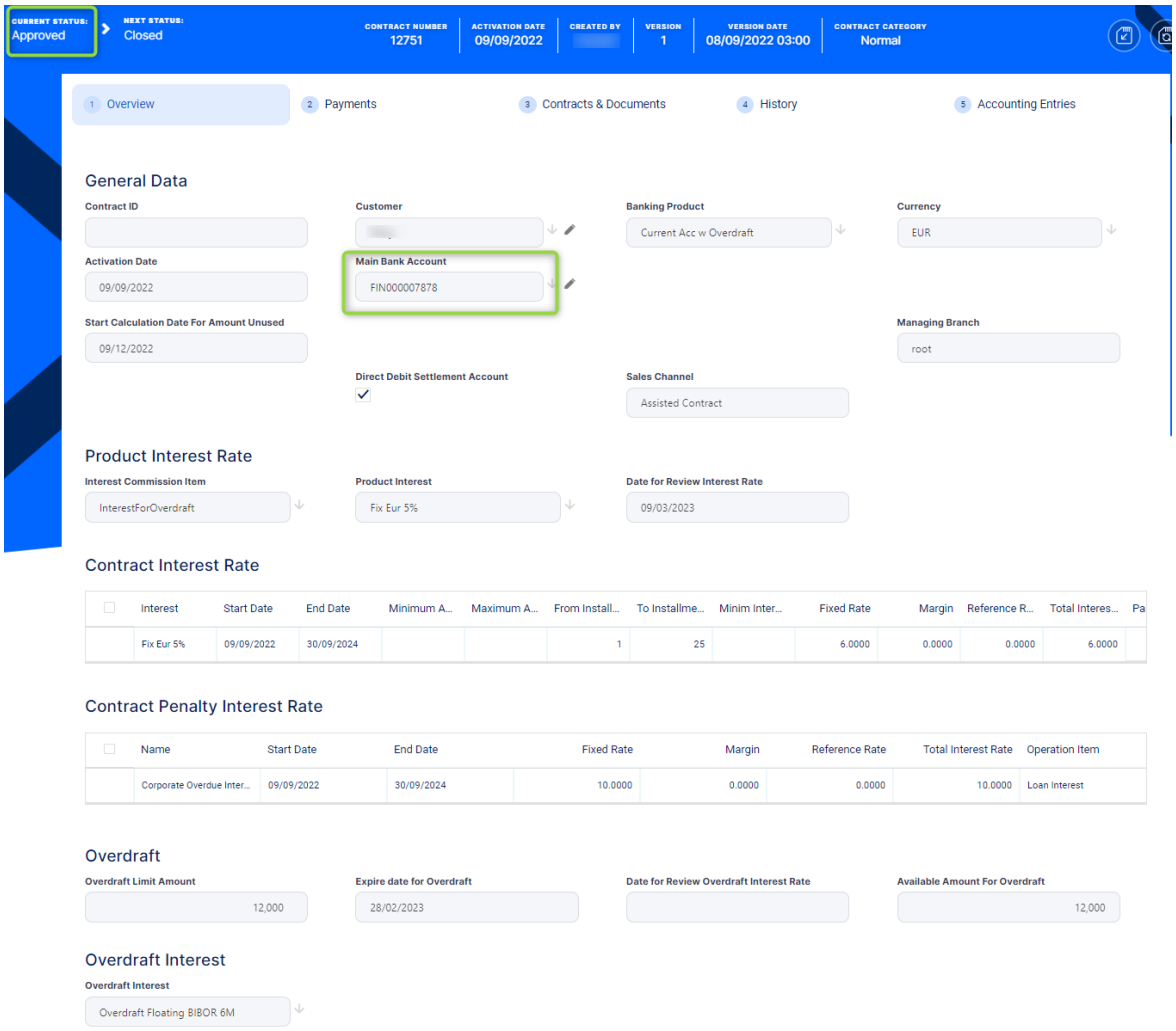
1. Select a contract in **Draft** (or **Version Draft**) status.
2. Change its status into **Approved**.



3. Click **Yes** to confirm your action.

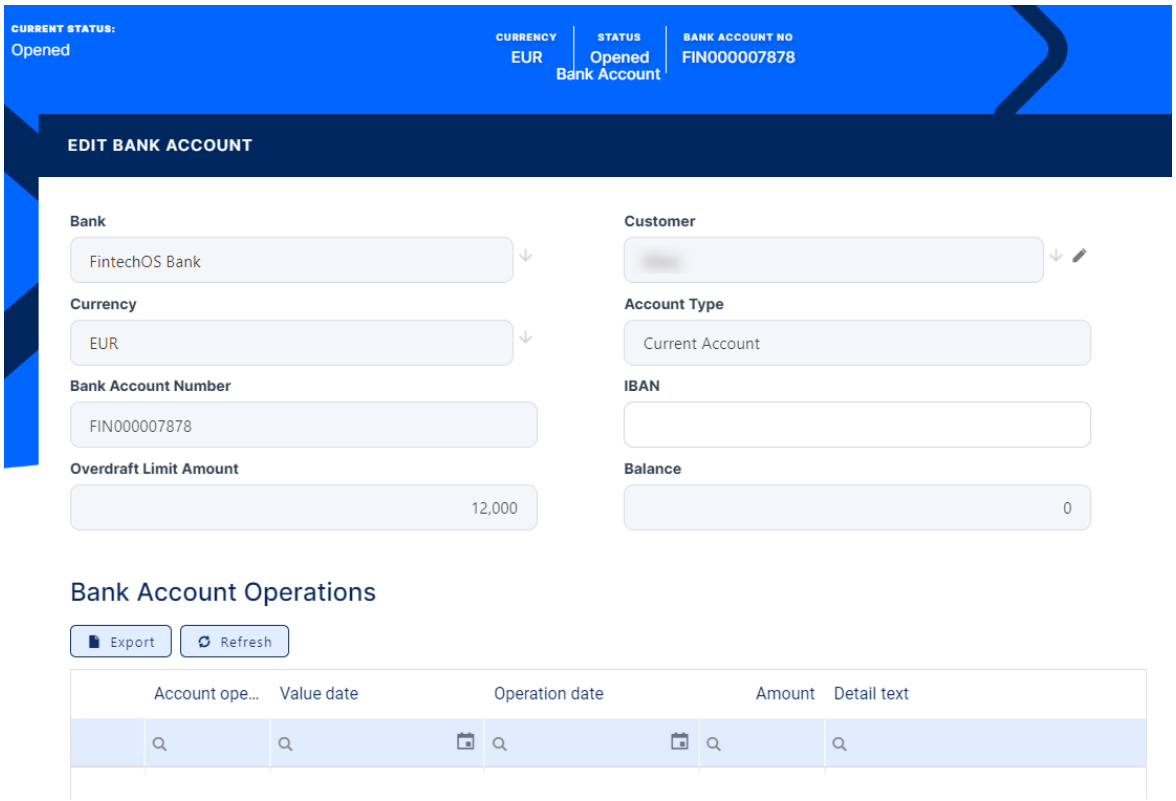


If Loan Management performs all the validations successfully, then the current status of the contract changes to **Approved** and Loan Management automatically generates the current account number, displaying it in the **Main Bank Account** field.



Automated Actions After Contract Approval

The **Main Bank Account** is created automatically for the bank defined as Main within the **Loan Management Operational > Bank** menu. In order for Loan Management to generate an account number, a rule must be defined during the implementation phase (example: branch code + incremental sequence number).



On the **Payments** tab you can see the repayment schedule that was generated based on the commissions with `Commission Undrawn Amount (overdraft)` type, if any.

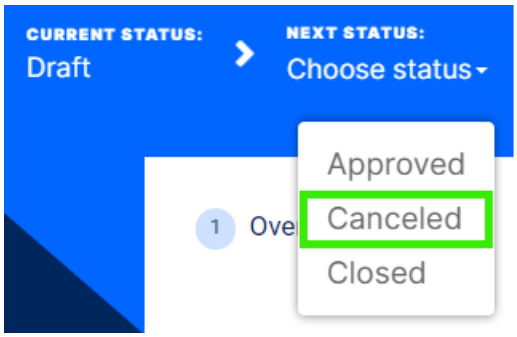
> **NOTE** The tab **Payments** has no information to display while the contract is in the **Draft** status. You must approve the contract to perform any contract event. Meaningful payment information is displayed in this tab only after performing transactions on the contract.

Rejecting a Current Account

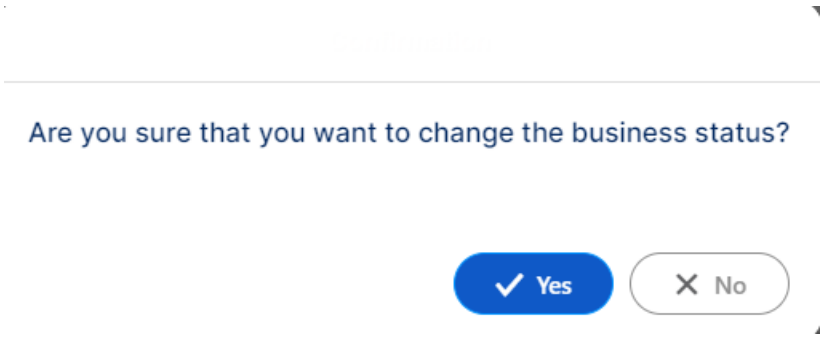
You can reject a current account contract, canceling it, when the deal with the customer drops. You can perform the cancellation either from a customer journey flow via API integration or from the Loan Management user interface.

Follow these steps to cancel the contract:

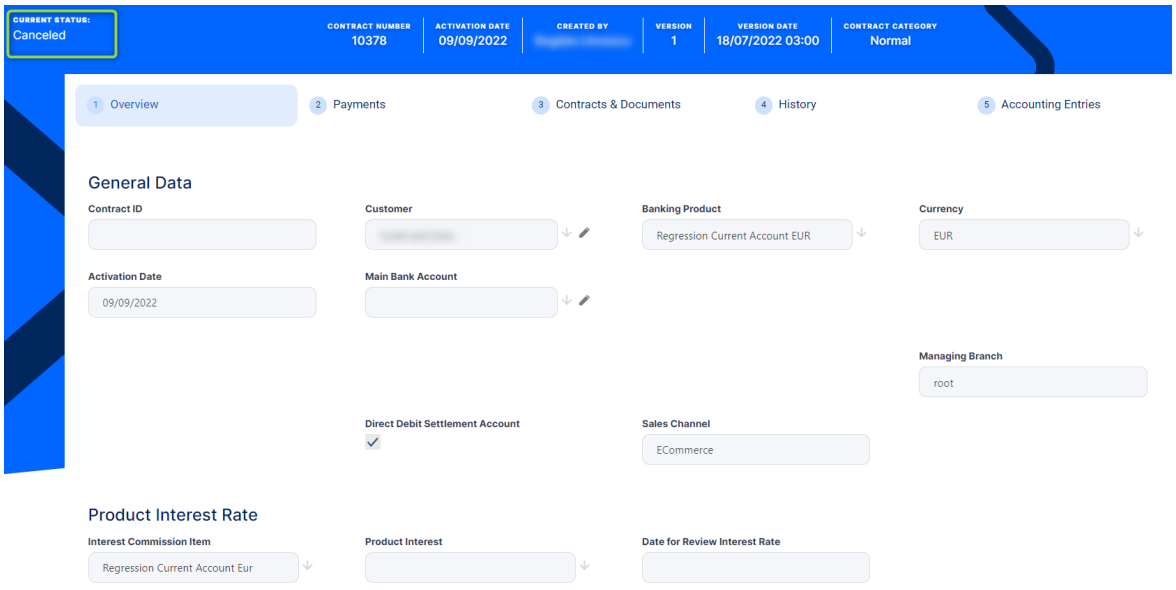
1. Select a contract in **Draft** (or **Version Draft**) status.
2. Change its status into **Canceled**.



3. Click **Yes** to confirm your action.



If Loan Management performs all the validations successfully, then the current status of the contract changes to **Canceled**.

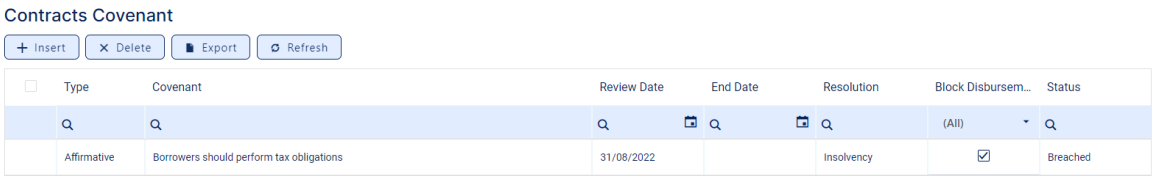


NOTE You can't further use a canceled contract. Create a new contract, if you need to.

Working with Covenants

The covenants are conventions that applicants must abide by after the approval of a contract. Such conventions are usually applicable for corporate clients that must meet certain requirements in order to continue to receive disbursements and not only: submit balance sheet every x months, have account turnover of at least x percent from average monthly turnover, provide other relevant documents from authorities. Covenants are configured at the product level.

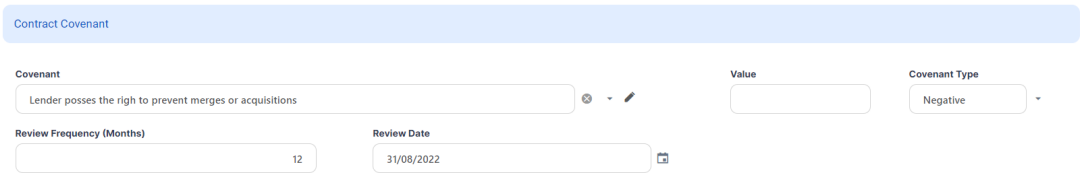
While creating a contract, Loan Management brings the covenants to the contract level, in the **Contract Covenant** section of the **Overview** tab. There you also add, delete or export covenants for the contract.



Upon adding a covenant to a contract, you must activate it. After approving the contract, when it reaches the covenant's review date, you must perform the review of the covenant. If the conditions are not met, then you can mark the covenant for blocking further disbursements of the contract. Further implementation is needed if you want automatic processes to take care of contracts with breached covenants.

Adding & Activating Covenants

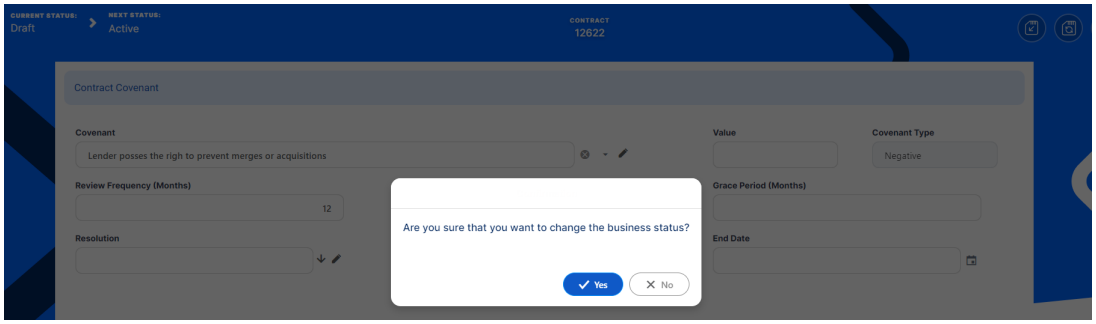
1. To add a covenant to a contract, click **Insert** in the **Contracts Covenant** section of a contract in `Draft` or `Version Draft` status.
2. On the newly displayed **Contract Covenant** page, fill in the following fields:



- **Covenant** - Select the desired covenant from the list of possible values:
  - `Borrowers should perform tax obligations`: the lenders expect the borrowers to perform their tax obligations to both the business and towards their employees. This covenant is of type affirmative.
  - `Lender can monitor borrower's current ratio`: the lender may continuously monitor the borrower's current ratio to ensure it stays relatively attractive and promising. This covenant is of type financial.
  - `Lender posses the right to prevent merges or acquisitions`: a clear stipulation that the lender possesses the right to prevent merges of acquisitions without proper notification or full knowledge of the process. This covenant is of type negative.

    Loan Management automatically fills in the covenant type.
- **Value** - Enter the numeric value of the covenant, if applicable.
- **Review Frequency (Months)** - Enter the number of months after which the covenant has to be reviewed.
- **Review Date** - Enter the date when the covenant has to be reviewed.

3. Click the **Save and Reload** button. The covenant is displayed in the list of covenants in the **Contracts Covenant** section, in `Draft` status.
4. Activate the covenant record by changing its status to **Active** and confirming your action.

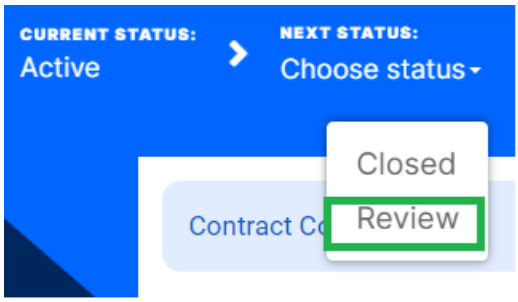


5. Click the **Save and Close** button. The covenant's status changes to `Active`.

Reviewing Covenants

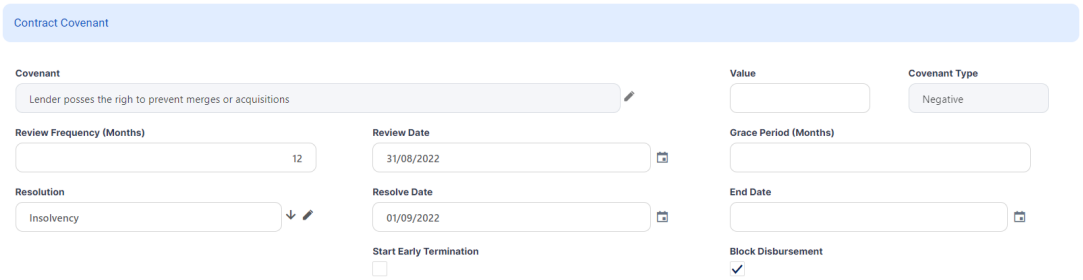
Loan Management allows you to add details about the process of reviewing a covenant for an approved contract.

1. To review an active covenant for an approved contract, double-click the desired covenant in the **Contracts Covenant** section of the contract's Overview tab.
2. On the newly displayed **Contract Covenant** page, change the covenant's status to **Review** and confirm your action.



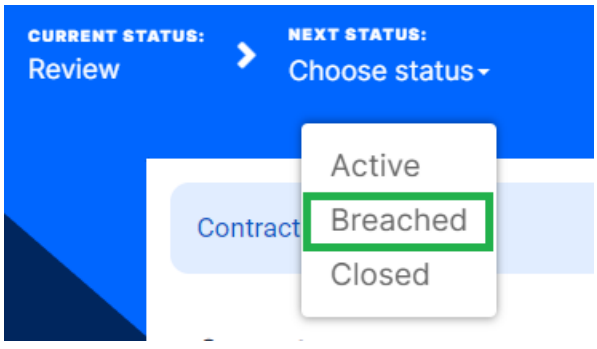
The covenant's status changes to Review and the page reloads with new fields.

3. Fill in the following fields with the results of the covenant review process:



- **Grace Period (Months)** - Enter a grace period in month for the fulfillment of the covenant, if needed.
- **Resolution** - Select from the list the actual resolution of the covenant. Add a new covenant resolution, if you can't find a match in the list.
- **Resolve Date** - Enter the date when the covenant is considered as resolved.
- **End Date** - Enter an end date for the covenant, if needed.
- **Start Early Termination** - If the covenant's terms are not met, then you can check this field to mark the covenant for contract early termination.
- **Block Disbursement** - If the covenant's terms are not met, then you can check this field to mark the covenant for blocking further disbursements of the contract.

4. Click the **Save and Reload** button.
5. If the covenant's terms are met, change the covenant's status to **Active** and confirm your action.
   If the covenant's terms are not met, change the covenant's status to **Breached** and confirm your action.



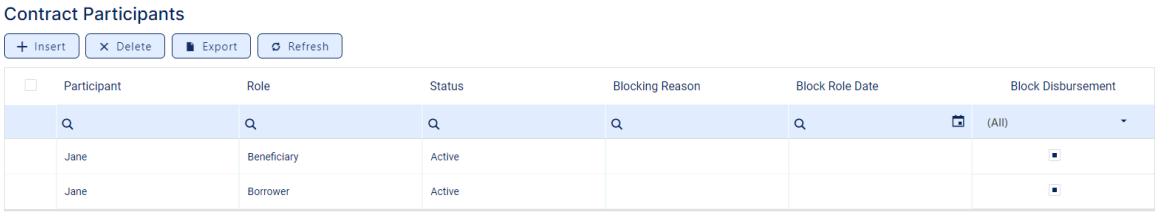
6. Click the **Save and Close** button. The covenant's status changes to `Active` or `Breached`, according to your previous choice.

> **NOTE** Further implementations are needed in order for Loan Management to manage contracts with breached covenants if you need actions enforced at the contract level.

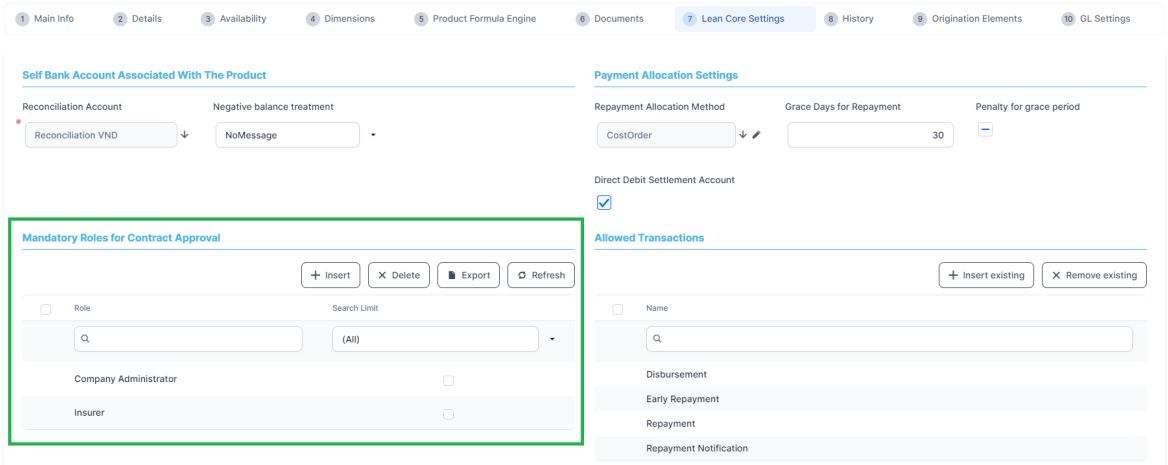
Working with Participants

The participants to a contract are those legal or individual persons who have a role to play during the life-cycle of the contract. They can be the person who borrows the funds, the actual beneficiary of the funds, the company administrator of the legal person, a notary, and so on. Another example are the agents, brokers, insurers, or merchants who participate in contracts as third-party entities, and they may get commissions according to third-party agreements. They must be recorded in contracts as contract participant with the specified role in order to qualify for the commissions stated in the agreement pricing records added to an agreement.

While creating a contract, Loan Management automatically populates the **Contract Participants** section within the **Overview** tab of the contract with the customer's information as both Borrower and Beneficiary of the funds, for loan contracts. If the customer is a legal entity, all the company's already entered legal representatives such as administrators, affiliates, owners, or other key contact persons are displayed in this list. In the **Contract Participants** section, you can add other participants to the contract, like Guarantors, Co-Debtors, etc, even after approval, delete, block, or export customers who participate in a contract.

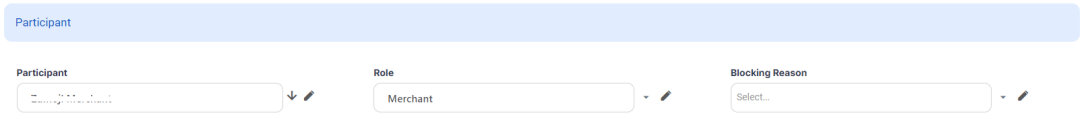


There may be cases when some roles are mandatory for a product. If there is a mandatory role defined in the banking product definition, Loan Management displays an error on trying to approve the contract without a customer mentioned in the contract with that specific role.



Adding Participants

1. To add a participant, click **Insert** in the **Contract Participants** section of contract in `Draft, Version Draft`, or `Approved` status.
2. On the newly displayed **Participant** page, fill in the following fields:



   - **Participant** - Select from the list the name of the customer who can access the contract.
   - **Role** - Select from the list the role in the contract of the previously selected customer.
   - **Blocking Reason** - Leave this empty if you don't want to limit the customer's access to the contract.
3. Click the **Save and Close** button.

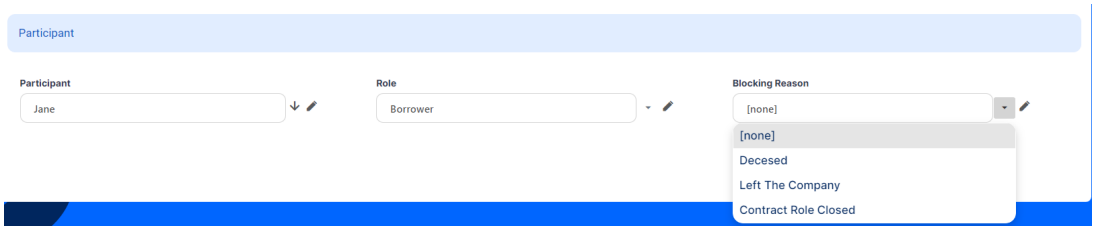
IMPORTANT!
For legal entity customers, add the participant with the `Company Administrator`

role, otherwise, the loan contracts cannot be approved. This is not the case for current account contracts.

Blocking Participants

If you need to block an existing participant's access to the contract for various reasons, such as the person left the company who is the beneficiary of the contract, follow these steps:

1. Double-click an existing participant in the **Contract Participants** section of contract in `Draft, Version Draft`, or `Approved` status.
2. On the displayed **Participant** page, in the **Blocking Reason** field, choose the reason for blocking the selected participant from accessing the contract.



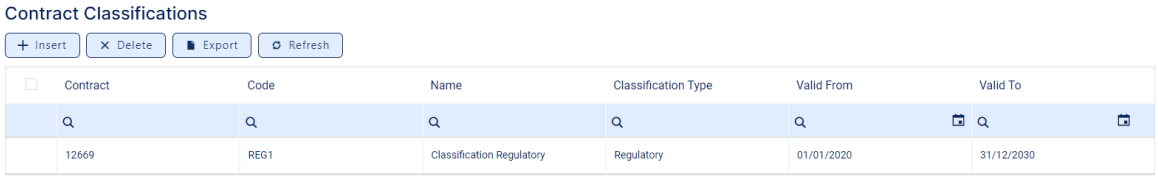
3. In the **Block Role Date** field, select the starting date for blocking the participant's access to the contract.
4. Select the **Block Disbursement** checkbox to instruct Loan Management to stop disbursements on the contract, if needed.
5. Click the **Save and Close** button.

Working with Contract Classification

Financial institutions may classify their contracts for organization purposes, or to mark some contracts as to belonging to a specific category or another. Loan Management brings the classifications defined at the product level to the contract level when creating a contract.

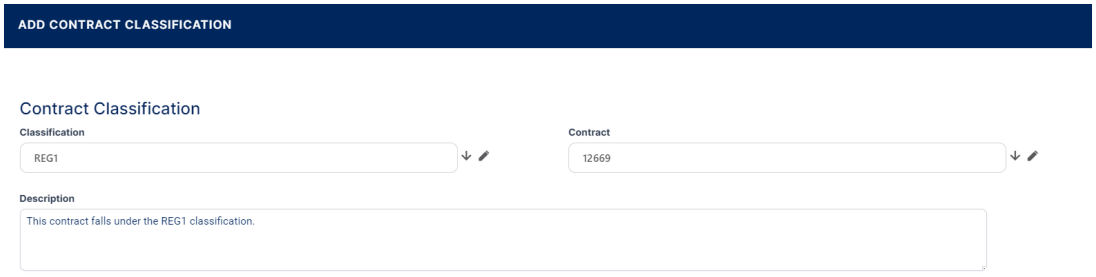
> **NOTE**
> For information about the **automatic loan classification** performed by Loan Management based on DPD, please read the "Loan Classification" on page 49 topic.

You can manage a contract's classification within the **Contract Classifications** section on the **Overview** tab. Here you can insert, delete or export classifications for the contract.



Adding Classifications to a Contract

1. To add a classification to a contract, click **Insert** in the **Contract Classifications** section of a contract.
2. On the newly displayed **Add Contract Classification** page, fill in the following fields:



   - **Classification** - Select the desired classification for the contract from the list of classifications associated with the banking product.
   - **Description** - Enter a description for the contract classification.

3. Click the **Save and Close** button.

Applying Fees and Commissions

The financial institutions take commissions and fees for offering a product or service such as opening an account, for cash withdrawals, for transfers, for making payments in certain countries, for exchanging currencies, for emitting debit cards, for handling documents etc. These commissions are set at the product level and vary from institution to institution, based on their policy.

In the **Fees & Commissions** section within the **Overview** tab of the contract, you can view all the fees and commissions configured at the product level that have the `Automatic Load on Contract` checkbox set to `True`. After the first saving operation, Loan Management display all the fees that are defined as values. The fees defined as percentages are displayed after completing all the values of the contract. Read more about the commissions automatically inserted and calculated in the below section. You can also add, delete or export fees and commissions for the contract.



Automatic Insertion and Calculation of Commissions

Loan Management automatically inserts/ updates commissions in the **Fees & Commissions** section depending on the life cycle and status of the contract:

- **Creating a new contract**: Loan Management automatically inserts active commissions associated to the banking product, within their defined validity period, with `Automatically load on contract = True`, with `Is For Unusage = False`, and `Commission value is percentage = False`.
  If `Commission value is percentage = True`, then the commission is only inserted if the amount value was previously inserted.

- **Updating a contract in `Draft` status**: Loan Management automatically inserts active commissions associated to the banking product, within their defined validity period, with `Automatically load on contract = True`, with `Is For Unusage = False`. If a commission with `Commission value is percentage = True` was already inserted, then the commission's value is updated according to the contract's financed amount. If the value of a commission with `Commission value is percentage = True` was manually modified (for negotiable commissions), then the new value is calculated based on the modified percentage.
- **Creating a new version for a contract**: Loan Management automatically inserts all the commissions already present in the contract. Additionally, all commissions specifically created for contract version (`Is For Contract Version = True`) are added as well.

  > **NOTE**
  >
  > If a version for a contract is created more than once on the same day, then all commissions with `Is For Contract Version = True` that were not notified yet for each previous version are deleted. At the end of the day, there is only one commission for the latest version.

- **Updating a contract in `Contract Version Draft` status**: Loan Management only updates the percentage commissions that are not already notified.

For percentage commissions (with `Commission value is percentage = True`), the financed amount of the contract is used to calculate the commission value based on the percentage. The calculation method differs depending on the contract type:

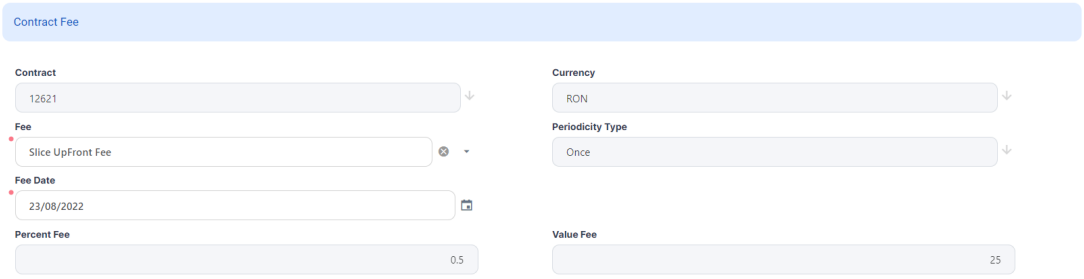
- For contracts based on **Term Loan,Mortgage** or **Overdraft** banking products:

- If the commission is applied to amount, then the financed amount = amount due;
- If the commission is applied to financed amount, then the financed amount = amount due - advance amount;
- If the commission is applied to remaining value and the contract is in `Contract Version Draft` status, then financed amount = (-1) * main bank account balance. If the result is a negative value, then financed amount = null. In all the other cases, financed amount = null, which is the default value.

- For contracts based on **Bank Account with Overdraft** banking products:
  - If the commission is applied to overdraft limit amount, then the financed amount = overdraft limit amount;
  - If the commission is applied to used amount and the commission's period type is `Once`, then the financed amount = overdraft limit amount - available amount for overdraft. In all the other cases, financed amount = null, which is the default value.

Adding Fees

1. To add a fee for this contract, click **Insert** in the **Fees & Commissions** section of a contract in `Draft` or `Version Draft` status.
2. On the newly displayed **Contract Fee** page, fill in the following fields:

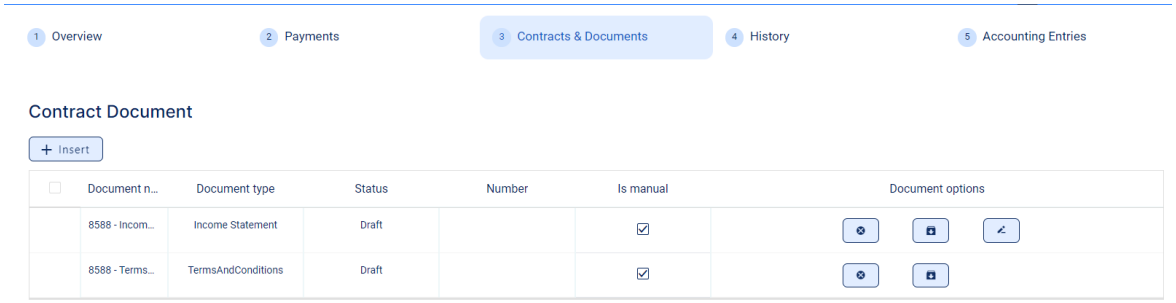


- **Fee** - Select a commission from the list of commissions defined for the banking product used when creating the contract.
- **Fee Date** - Specify which value of the commission is to be used by selecting the date of the commission.

3. Optionally, check the rest of the fields, automatically filled in by Loan Management: contract number, currency, periodicity type of the selected fee, the fee percentage or value applicable for the selected date. You can't change these values.
4. Click the **Save and Close** button.

Working with Documents

Loan Management allows you to manage all the documents related to a contract in one place, in the contract's **Contracts & Documents** tab. The tab is meant to be the electronic folder of the contract. It displays a list of the document records for the current contract, with details such as document name, type, status, number, whether the record was added through the user interface (`Is manual = True`) or through API integration (`Is manual = False`), and download options for the attached files. Contract documents have a dedicated business workflow, thus you can transition them through a series of statuses.



In the **Contracts & Documents** tab, you can: add a new contract document record, edit or delete a record in **Draft** status, view the details for records in **Signed** or **Canceled** status, or download the initial or the signed document, if it exists, by clicking the **Initial document**, respectively the **Signed document** button next to the record. Open the downloaded file to view its content.

> **NOTE**
> Users with the associated predefined security roles of Corporate Credit Officer and Retail Credit Officer can perform contract document-related operations such as adding, updating, and deleting records or changing their statuses.

Contract Document Statuses

A contract document record has the following statuses, visible in the top left corner of the **Add Contract Document** page, after saving the record:

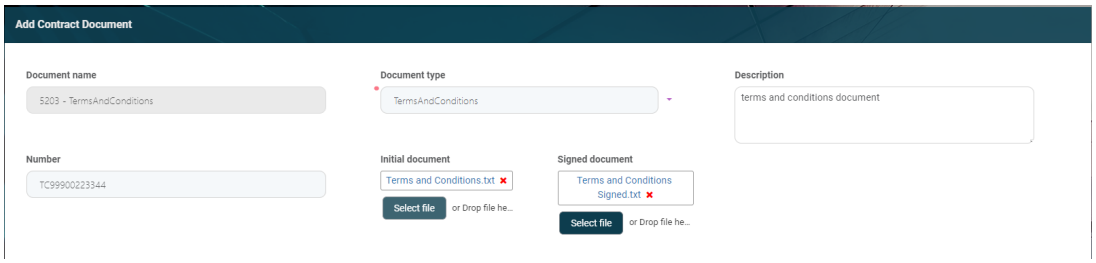
- **Draft** - the status of a newly created contract document record that was not yet authorized (marked as **Signed**). While in this status, you can edit some fields and you can delete the uploaded documents. Change its status to **Signed** after editing all the necessary details and uploading the `Signed Document` file. Change its status to **Canceled** if the document is not to be used within the contract.
- **Signed** - the status of a contract document record after being authorized. You cannot edit any of the record's details. You can change the status of the record to **Canceled**, if needed.
- **Canceled** - the status of a contract document after being canceled. Once **Signed**, a contract document should be canceled if the document is not to be used within the contract. You cannot edit any of the record's details. There is no further transition from this status. Contract document records created through integration (having their `Is manual` field = `False`) can't be canceled.

Adding Contract Documents

1. To insert a document to the contract, click the **Insert** button in the **Contract Document** section.
   The **Add Contract Document** page is displayed, with the **Document Name** field automatically completed with the name of the document.

   > **NOTE**
   >
   > You can't add documents to contracts in **Contract Closed** or **Contract Version Closed** statuses.

2. Fill in the following fields:



- **Document type** - Select the type of the document.
- **Description** - Enter the description of the document.
- **Number** - Enter the number of the document, if the document has an external identifier number.
- **Initial document** - Insert the file containing the initial, unsigned document.
  Click the **Select file** button under this field, navigate to the desired file, select it and click **Open**. The selected file's name is displayed here. You can delete it by clicking the **x** next to the file name.
- **Signed Document** - Insert the file containing the final, signed document, if available.
  Click the **Select file** button under this field, navigate to the desired file, select it and click **Open**. The selected file's name is displayed here. You can delete it by clicking the **x** next to the file name.

> **NOTE**
>
> To change the status of the contract document record to **Signed**, a signed document file must exist within the record.

3. Click the **Save and Close** button.

> **NOTE**
> You can also add, update, and approve contract document records through API integration, using the `AddUpdateContractDocument` and `ApproveContractDocument` endpoints. Read more details in the Loan Management Developer Guide.

Contract document files added through integration cannot be deleted and those records can't be canceled!

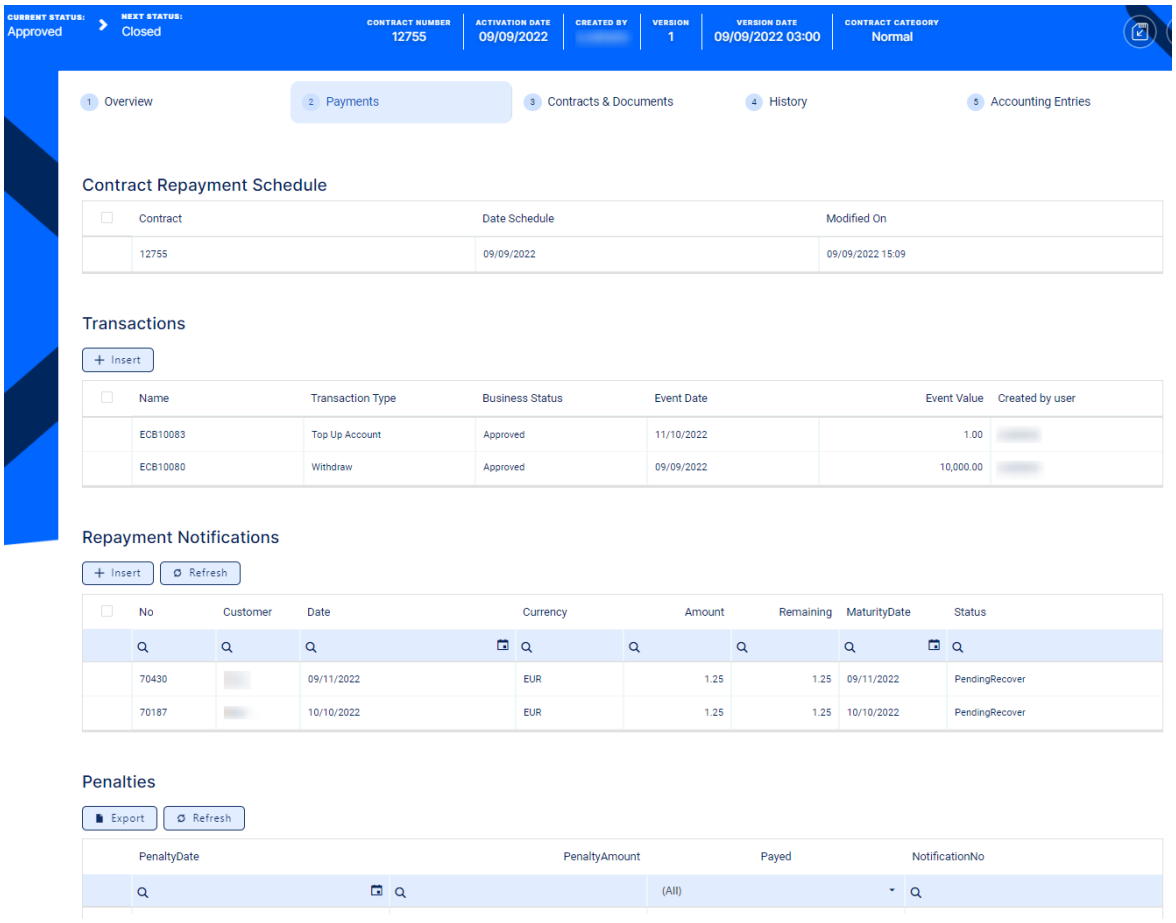
Automatic Contract Document Validations

Loan Management performs the following validations for contract document records:

- The uploaded files' specifications follow FintechOS Platform's settings and restrictions regarding size and format, allowing .pdf,.doc,.docx,.els,.jpg,.jpeg,.xlsx,.dll,.ppt,.pptx,.txt,.png,.ttf,.xml file formats.
- If the contract document record is in **Signed** status, the record can't be deleted or updated, nor can its files be deleted.
- The name of the contract document record is unique, automatically generated by Loan Management. The naming convention is "the contract name + '-' + the selected document type + '-' + a unique document increment". For example, `5203 - Income Statement - 60`.
- The names of the selected files are not validated for uniqueness.

Performing Transactions on Current Accounts

You can find all the existing transactions, payments, penalties, bank account operations, repayment schedules, schedule versions, repayment notifications for a contract on the **Payments** tab. The tab has no information to display for contracts in **Draft** status. Approve the contract to perform any transactions on the contract.



The following sections show you how to perform the usual transactions available on current account contracts:

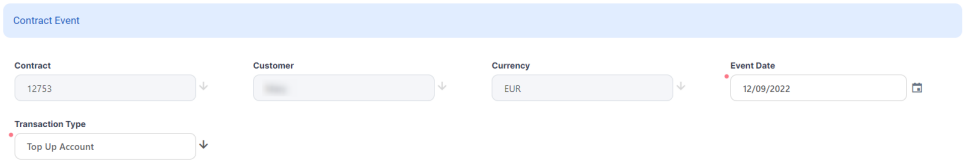
Topping Up an Approved Contract

A top-up transaction represents adding amounts to the account before the value drains down to zero.

You can add top-up transactions to an approved contract via Loan Management's user interface or through API calls, using the Loan Management endpoints. Read more about these endpoints in the Loan Management Developer Guide.

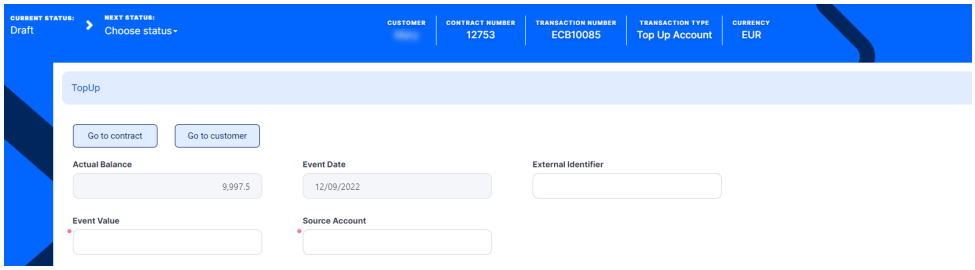
In order to add a top-up transaction to a contract through the menus available in Loan Management, follow these steps:

1. In FintechOS Portal, select a contract with **Approved** status and double-click to open it.
2. Navigate to the contract's **Payments** tab and click the **Insert** button above the **Transactions** section. The **Event** page is displayed.



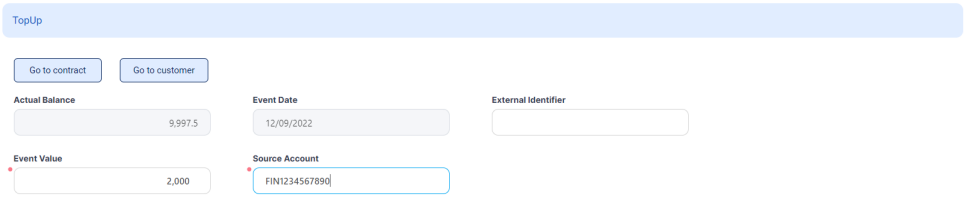
3. Fill in the following fields:
   - **Event Date** - This is pre-filled with the current date.
   - **Transaction Type** - Select from the list the **Top Up Account** transaction type. If you can't find it, then the transaction type is not associated with the banking product which is at the base of the contract.

   Other values are automatically completed: contract, customer, and currency.
4. Click the **Save and Reload** button.
   The event is saved in **Draft** status and a transaction number is automatically generated for it. The **Edit Contract Event** page corresponding to the selected transaction type is displayed. The account's actual balance is automatically calculated, and you can't edit it.



5. Fill in the **external identifier** of the transaction, if available.
6. In the **Event Value** field, enter the amount that is added to the account.

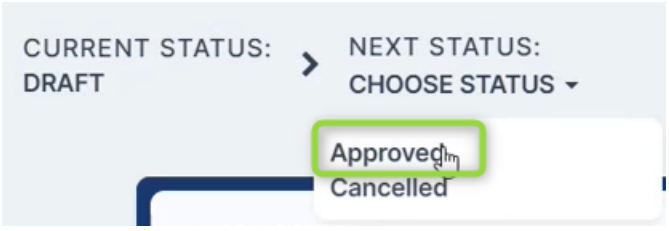
7. Enter the **Source Account** for the respective amount, the account from where the funds are taken to perform the top-up.



> **NOTE**
>
> Loan Management actually uses the financial institution's reconciliation account as a source bank account.

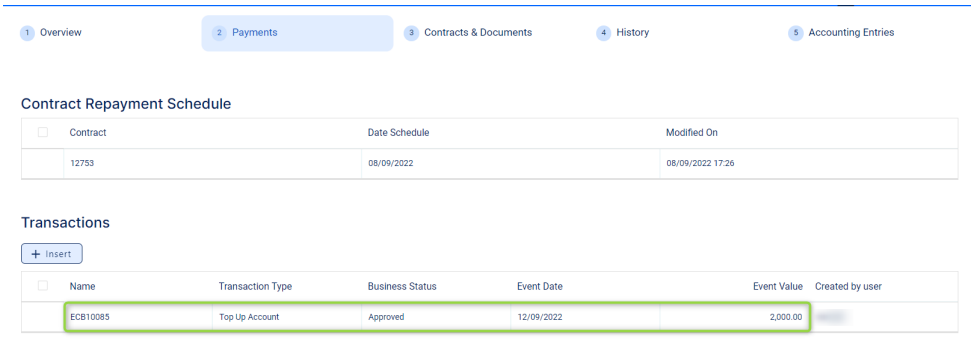
8. Click the **Save and Reload** button.
If the event value meets the business requirements defined within Loan Management, the event is saved. Otherwise, an error message appears. Change the values as instructed in the message and try saving the event again.
While the event is in **Draft** status, you can modify all the event's fields except **Transaction Type**. The event value is not applied to the contract while the event is still in this status.

9. Approve the event by changing its status to **Approved** in the upper left corner of the **Event** page.



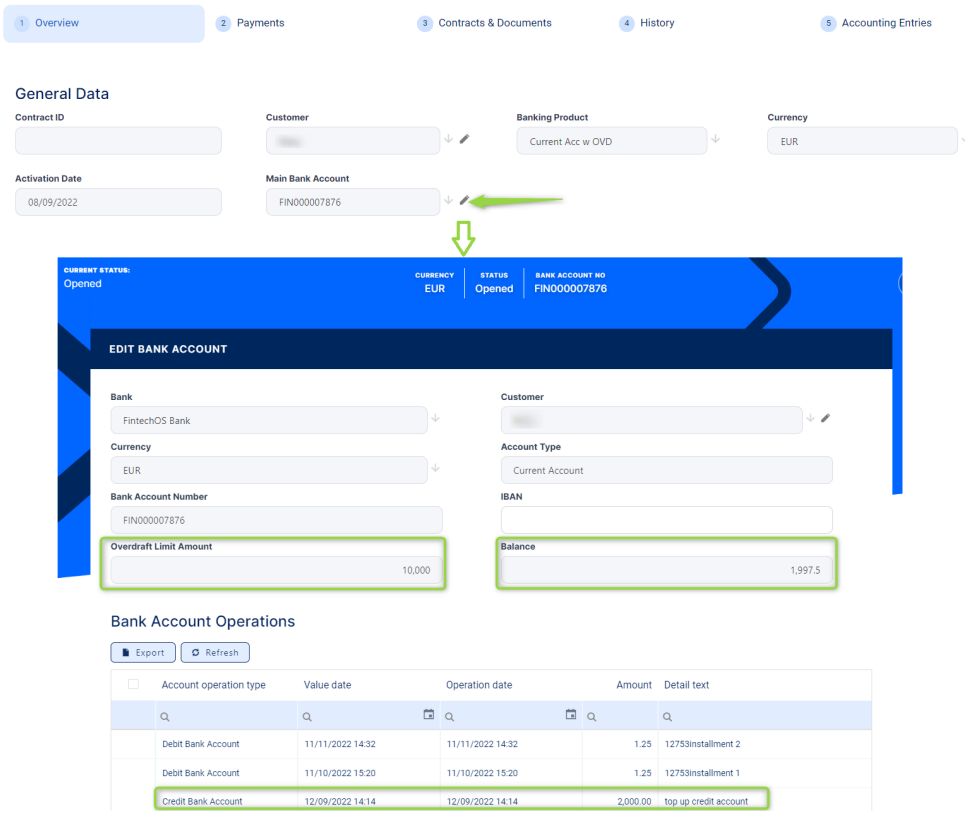
10. Confirm the change of status in the **Confirmation** window, clicking **Yes**. The event is now in **Approved** status and Loan Management applies the transaction to the contract, moving the funds specified in the event value from the source account into the current account.

    The transaction is visible in the **Transactions** section.



> **NOTE**
>
> For approved transactions on current accounts, Loan Management updates the repayment schedule when running the `End of Day job` on the last day of the month.

11. View the balance of the account after approving the transaction, clicking the pencil icon next to the **Main Bank Account** field in the contract's **Overview** tab:



> **NOTE**
> All existing versions of the contract in **Contract Version Draft** status are automatically changed to **Contract Version Closed** when a payment event is approved for that contract.

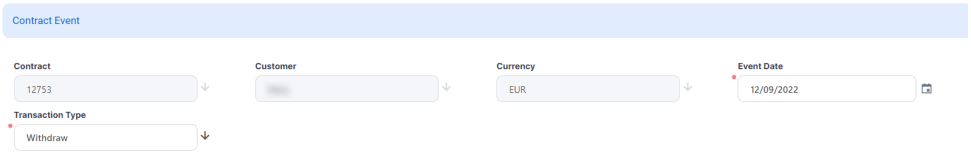
Withdrawing Funds from an Approved Contract

A withdrawal transaction represents removing funds from a bank account.

You can add withdrawal transactions to an approved contract via Loan Management's user interface or through API calls, using the Loan Management endpoints. Read more about these endpoints in the Loan Management Developer Guide.

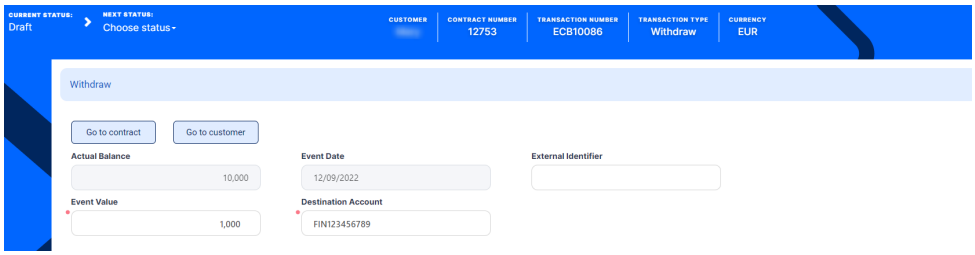
In order to add a withdrawal transaction to a contract through the menus available in Loan Management, follow these steps:

1. In FintechOS Portal, select a contract with **Approved** status and double-click to open it.
2. Navigate to the contract's **Payments** tab and click the **Insert** button above the **Transactions** section. The **Event** page is displayed.



3. Fill in the following fields:
   - **Event Date** - This is pre-filled with the current date.
   - **Transaction Type** - Select from the list the **Withdraw** transaction type. If you can't find it, then the transaction type is not associated with the banking product which is at the base of the contract.

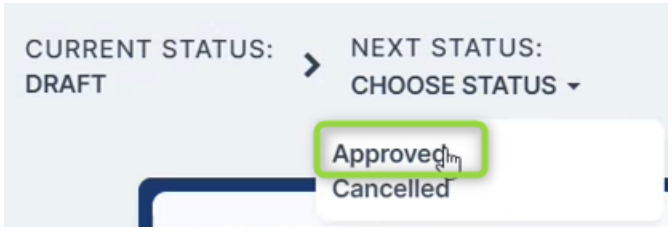
   Other values are automatically completed: contract, customer, and currency.
4. Click the **Save and Reload** button.
   The event is saved in **Draft** status and a transaction number is automatically generated for it. The **Edit Contract Event** page corresponding to the selected transaction type is displayed. The account's actual balance is automatically calculated, and you can't edit it.
5. Fill in the **external identifier** of the transaction, if available.
6. In the **Event Value** field, enter the amount that is removed from the account.
7. Enter the **Destination Account** for the respective amount, the account where the funds are being moved to.



> **NOTE**
>
> Loan Management actually uses the financial institution's reconciliation account as a destination bank account.

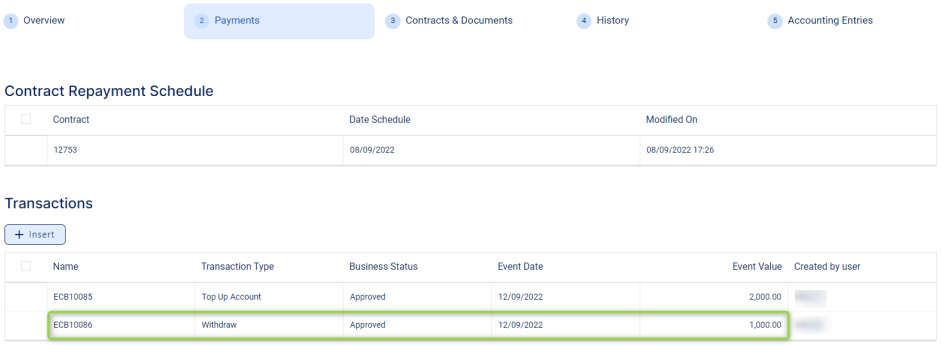
8. Click the **Save and Reload** button.
   If the event value meets the business requirements defined within Loan Management, the event is saved. Otherwise, an error message appears. Change the values as instructed in the message and try saving the event again.
   While the event is in **Draft** status, you can modify all the event's fields except **Transaction Type**. The event value is not applied to the contract while the event is still in this status.

9. Approve the event by changing its status to **Approved** in the upper left corner of the **Event** page.



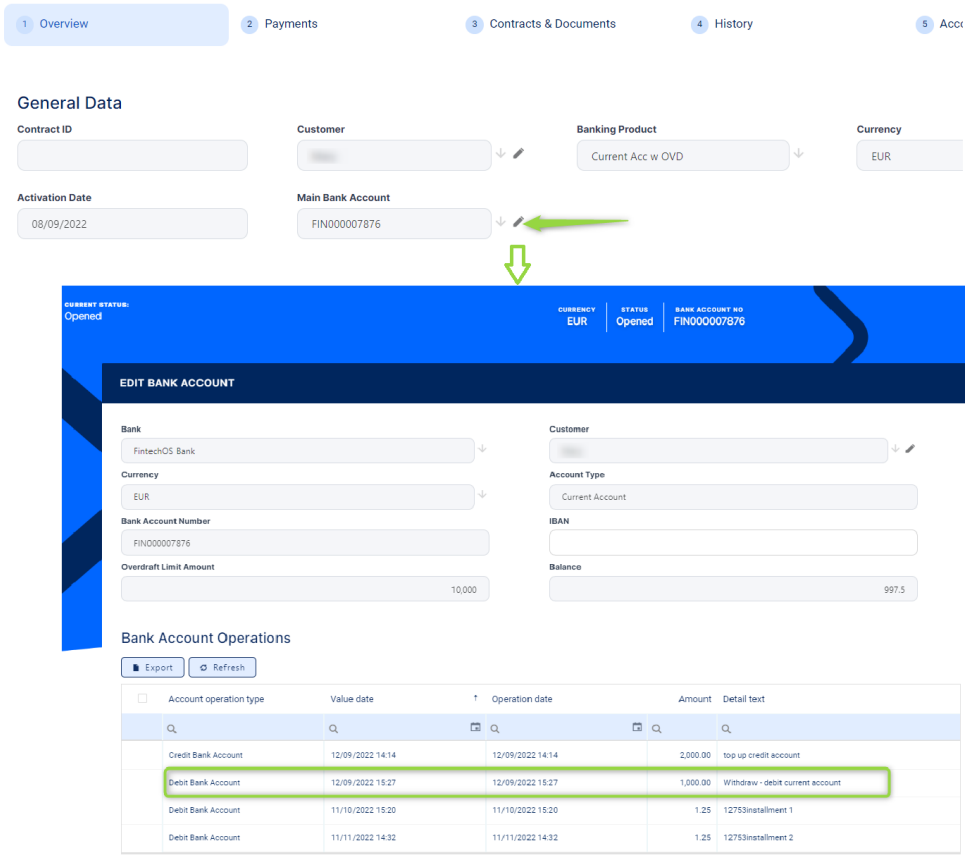
10. Confirm the change of status in the **Confirmation** window, clicking **Yes**.
    The event is now in **Approved** status and Loan Management applies the transaction to the contract, removing the funds specified in the event value from the current account.

    The transaction is visible in the **Transactions** section.



> **NOTE**
>
> For approved transactions on current accounts, Loan Management updates the repayment schedule when running the `End of Day job` on the last day of the month.

11. View the balance of the account after approving the transaction, clicking the pencil icon next to the **Main Bank Account** field in the contract's **Overview** tab:



> **NOTE**
> All existing versions of the contract in **Contract Version Draft** status are automatically changed to **Contract Version Closed** when a payment event is approved for that contract.

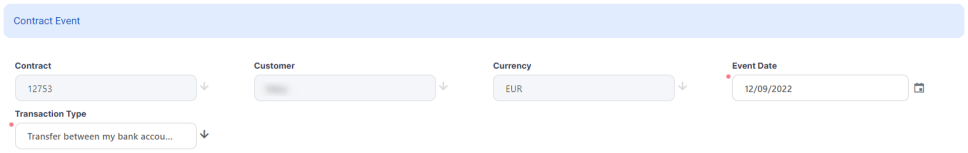
Transferring Funds between the Customer's Accounts

A `transfer between my bank accounts` transaction represents the process of moving funds between the same customer's bank accounts.

You can add transfer transactions to an approved contract via Loan Management's user interface or through API calls, using the Loan Management endpoints. Read more about these endpoints in the Loan Management Developer Guide.

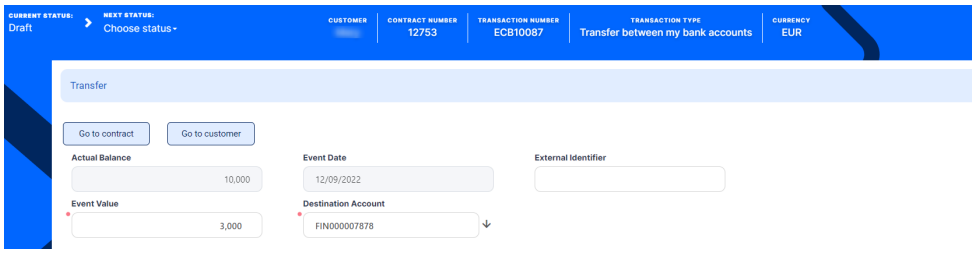
In order to add a transfer transaction to a contract through the menus available in Loan Management, follow these steps:

1. In FintechOS Portal, select a contract with **Approved** status and double-click to open it.
2. Navigate to the contract's **Payments** tab and click the **Insert** button above the **Transactions** section. The **Event** page is displayed.



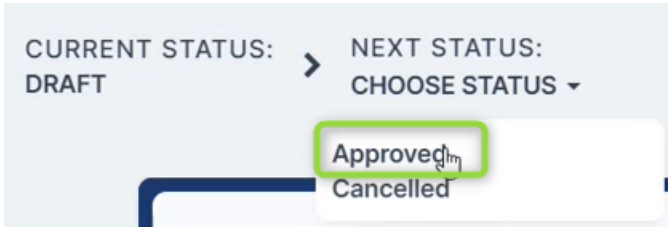
3. Fill in the following fields:
   - **Event Date** - This is pre-filled with the current date.
   - **Transaction Type** - Select from the list the **Transfer between my bank accounts** transaction type. If you can't find it, then the transaction type is not associated with the banking product which is at the base of the contract.

   Other values are automatically completed: contract, customer, and currency.
4. Click the **Save and Reload** button.
   The event is saved in **Draft** status and a transaction number is automatically generated for it. The **Edit Contract Event** page corresponding to the selected transaction type is displayed. The account's actual balance is automatically calculated, and you can't edit it.
5. Fill in the **external identifier** of the transaction, if available.
6. In the **Event Value** field, enter the amount that is added to the account.
7. Select the **Destination Account** where the respective amount should be transferred into. You can choose from the list of accounts that belong to the contract's customer and have the same currency.



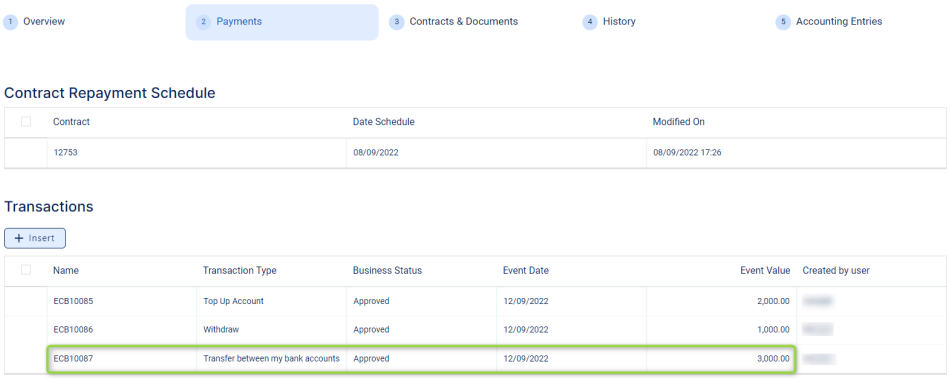
8. Click the **Save and Reload** button.
   If the event value meets the business requirements defined within Loan Management, the event is saved. Otherwise, an error message appears. Change the values as instructed in the message and try saving the event again.
   While the event is in **Draft** status, you can modify all the event's fields except **Transaction Type**. The event value is not applied to the contract while the event is still in this status.

9. Approve the event by changing its status to **Approved** in the upper left corner of the **Event** page.



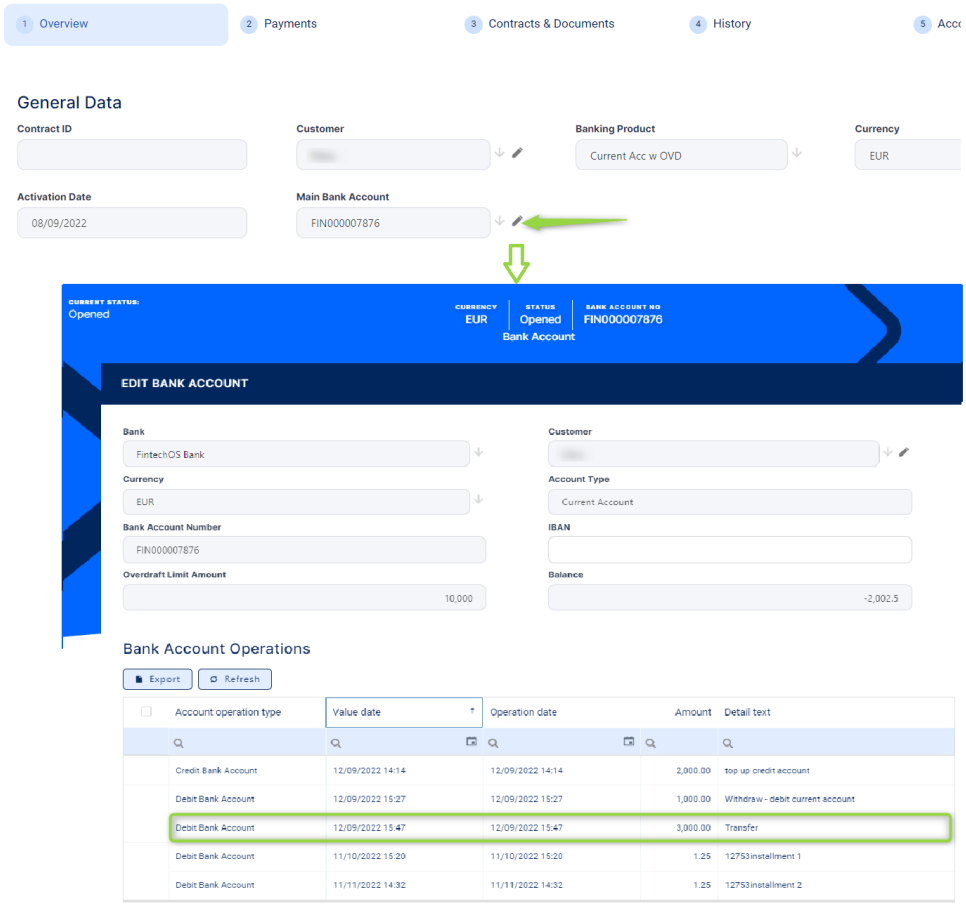
10. Confirm the change of status in the **Confirmation** window, clicking **Yes**.
    The event is now in **Approved** status and Loan Management applies the transaction to the contract, moving the funds specified in the event value from the current account into the destination account of the same customer.

    The transaction is visible in the **Transactions** section.



> **NOTE**
>
> For approved transactions on current accounts, Loan Management updates the repayment schedule when running the `End of Day job` on the last day of the month.

11. View the balance of the account after approving the transaction, clicking the pencil icon next to the **Main Bank Account** field in the contract's **Overview** tab:



NOTE

All existing versions of the contract in **Contract Version Draft** status are automatically changed to **Contract Version Closed** when a payment event is approved for that contract.

Processing Overdraft Repayments

Based on defined product pricing and established parameters, financial institutions can manage the billing and collection process on a current account with overdraft contract fully automatically. Once the withdraw from the overdraft amount is performed, Loan Management keeps track of the amounts and on the end of the

month it updates the repayment schedule and generates underlying notifications. When generating the notification, the system checks if the amounts are fully or partially available on the overdraft and uses them to settle the notification. Otherwise, the amounts are pending recovery on the notification. They will be recovered if and when the amounts are available on the account. If funds are not available, overdue amounts and days are calculated and penalties applied. Read about repayment notifications in the Managing Repayment Notifications topic.

Once a current account with overdraft approved and used, you can check the repayment schedule built based on contract details, on the **Payments** tab's **Contract Repayment Schedule** section, as described in the Viewing a Contract's Repayment Schedule topic.

When the system reaches the dates that appear on schedule projections, the amounts resulting are made due, and Loan Management automatically triggers the notifications. Depending on the availability of funds in the current account and the direct debit setup, Loan Management settles those notifications, marking them with the **Recovered** status. Any amount that is not recovered on due date stays on the notification, and when funds become available, Loan Management automatically recovers and allocates them to the pending notifications based on the **Payment Allocation Method** setup at the product level, in the Lean Core tab. When the notifications are recovered, you can see the underlying debit transactions on the bank account – there is always such traceability of the funds.

Payment Schedule Types, defined at the Banking Product Factory level, define how Loan Management handles the following:

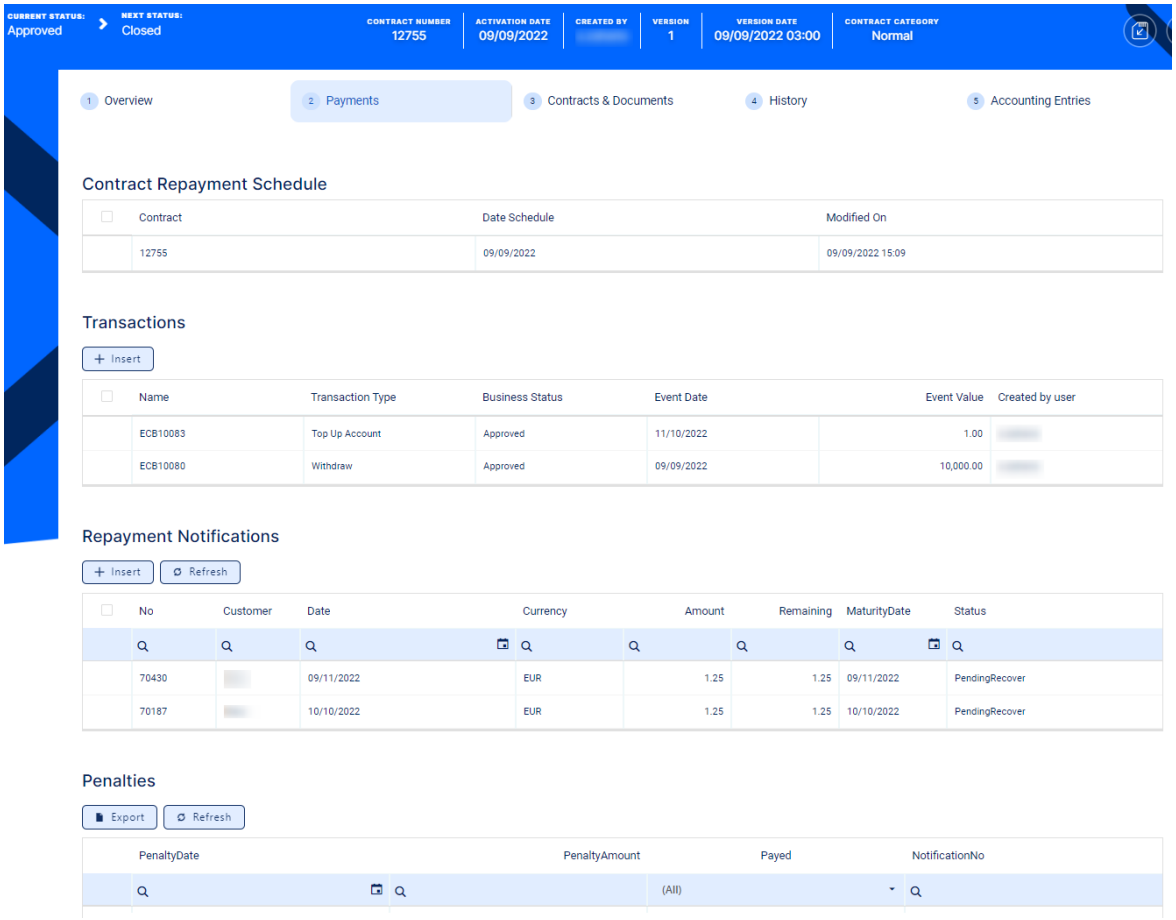
- How the overdraft interest is calculated (day basis: 30/360, actual/ 360, etc).
- Fees you want to include in the repayment schedule.
- Frequency of the installment (monthly, every 30 days, etc).

When amounts are not available to cover notified amounts and there is a penalty interest defined for the product, the missed amounts are subject to automatic penalty calculation. Loan Management calculates and notifies the penalty interest daily. All the penalties applied by automated processes at the contract level are visible on the **Payments** tab, in the **Penalties** section, as described in the Viewing a Contract's Penalties topic. When the penalty interest is defined, there are specific Operation Items linked to it so that the system knows what types of amounts are subject to

penalty: overdue principal, interest, commissions. Alternatively, the penalty interest can be applied to all missed payments. Penalty interest is defaulted from the product level and, if allowed, it can also be amended at contract level.



You can find all the existing transactions, payments, penalties, bank account operations, repayment schedules, schedule versions, repayment notifications for a contract on the **Payments** tab. The tab has no information to display for contracts in **Draft** status. Approve the contract to perform any transactions on the contract.

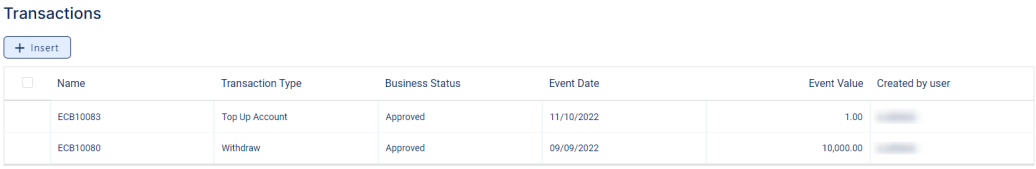


Managing a Contract's Transactions

Contract transactions are events/ changes performed at the **Approved** contract's level. Such events are top-ups, withdrawals, and transfers between the customer's accounts. Read more information about the available

transaction types in the "Transaction Types Used in Loan Management" on page 80 topic.

The **Transactions** section within the **Payments** tab holds all the transactions performed at the contract level, in any status. This section only has information if the contract is in **Approved** status and transactions were already created.



Event Statuses

An event (transaction) record has the following statuses, visible in the top left corner of any **Event** page:

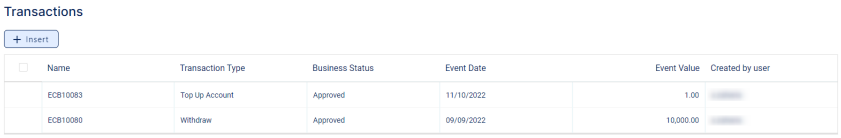
- **Draft** - the status of a newly created event record that was not yet sent for approval. The event value is not applied to the contract while the event is still in this status. While in this status, you can edit some fields. Approve after editing all the necessary details.
- **Approved** - the status of an event record after being authorized. The event value is applied to the contract and you cannot edit any of the event's details. There is no further transition from this status.
- **Canceled** - the status of an event record after being canceled. The event value is not applied to the contract and you cannot edit any of the event's details. There is no further transition from this status.

> **NOTE**
> As a best practice, new records or new versions of existing records created on a specific day should be approved on the same day.

Viewing Existing Events

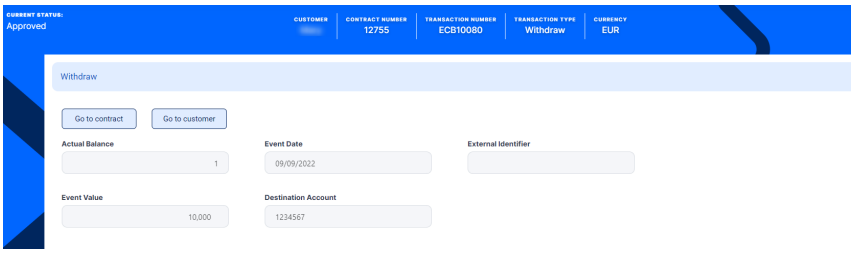
To view the events on a contract, follow these steps:

1. In FintechOS Portal, select a contract with **Approved** status and double-click to open it.
2. Navigate to the contract's **Payments** tab and view the list of events displayed in the **Transactions** section.



Here you can see only basic information about the transactions, such as event number, status, date, transaction type, value and the user who created it.

3. To view detailed information about the transaction, double-click the event record to open the **Event** page:



You can't edit the information displayed on this page.

4. View the following information displayed about each event, with some variations depending on the event type:
    - Transaction status, contract number, customer name, transaction number, type, and currency, all displayed in the header of the page.

The following details are displayed in the body of the page:

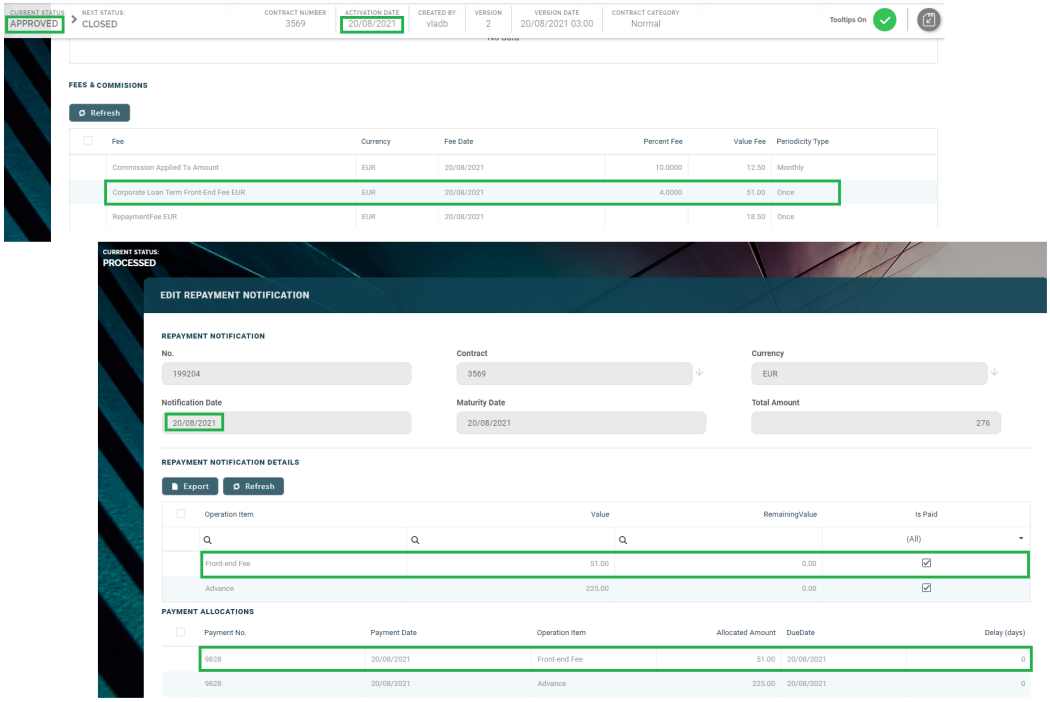
- Event date and value. Contract events added through API integration also contain an external identifier.
- Contract's actual amounts and source/ destination accounts.

Managing Repayment Notifications

Loan Management automatically generates notifications for each installment that has to be paid for existing contracts with overdrawn amounts. There can be various types of notifications generated for fees, commissions, and so on. Loan Management also assists you reconcile wrongly processed cases and other situations resulted from delayed processing/ wrong updates. Thus, depending on your user rights, you can manually add notifications for an active contract, or add notifications for amounts to be recovered even if there is no active contract. Loan Management includes all such manual notifications in the recovery processes.

All the repayment notifications that are not paid in full after their maturity dates are subject to automated penalty calculation processes performed by Loan Management. Thus, for repayment notifications linked to a contract, Loan Management uses the penalty interests defined within the interests list attached at the banking product level, and for repayment notifications not linked to a contract, it uses the penalty interest list specified in a `system parameter`.

NOTE
All the `Front-End Fee` commission types with `Once` periodicity type applied to a contract are notified and must be paid when the contract is approved. The Loan Management system parameter `FrontEndFee` defines the type of commission that is automatically notified at the contract approval.



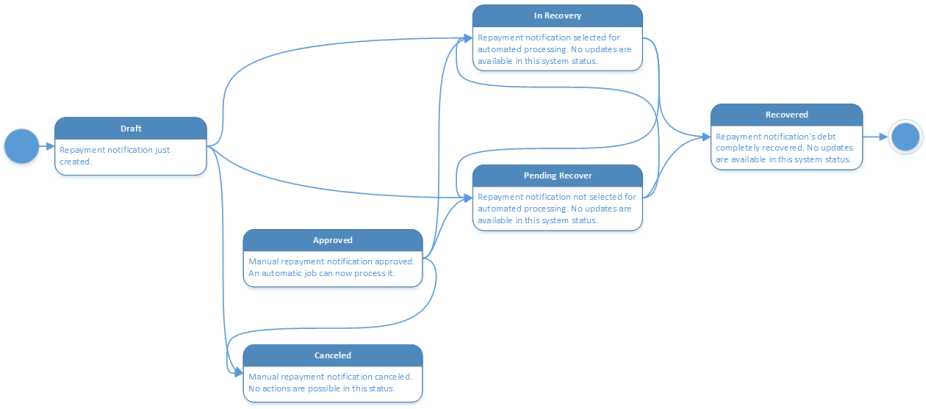
Repayment Notification Statuses

A repayment notification record has the following business workflow statuses:

- **Draft** - the status of a newly created repayment notification record, either automatic or manual.
- **Approved** - the status of a manual repayment notification record after being authorized by a user with notification approval competencies. While in this status, you cannot edit the record's details. From this status, the record is picked up by a scheduled job and its status is automatically changed, depending on the direct debit settlement settings. If the `Direct Debit Settlement Account` field at the contract level = `True`, then the manual notification's status changes to **In Recovery**, otherwise it changes to **Pending Recover**.

- **Canceled** - the status of a manual repayment notification after canceling it straight from the **Draft** status. You can only cancel a manual notification if its `Total Amount` = `Remaining Value`.
- **Pending Recover** - this is a system status applied to repayment notification when `Direct Debit Settlement Account` at the contract level is set to `False`. No updates are available in this system status.
- **In Recovery** - this is a system status applied to repayment notification when `Direct Debit Settlement Account` at the contract level is set to `True`. No updates are allowed on the record.
- **Recovered** - the last status of a repayment notification, after the complete recovery of the notification's debt. No updates are allowed on the record.

The repayment notification status transitions are illustrated below:



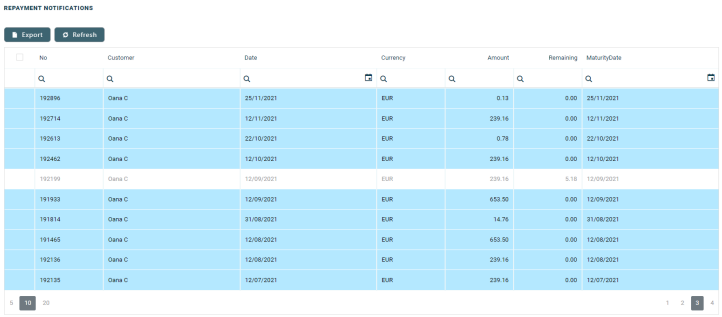
Accessing Repayment Notifications

Loan Management enables you to access notifications in several places, for your convenience.

Accessing a contract's repayment notifications

To view the notifications generated for a specific contract, follow these steps:

1. On the **Contract** page, navigate to the **Payments** tab > **Repayment Notifications** section.
2. View all the repayment notifications generated for the contract. This section only has information if the contract is in **Approved** status and disbursements were already performed.



Repayment notifications highlighted in blue are already paid, allocated or closed to payment, while the ones not highlighted remain to be paid.

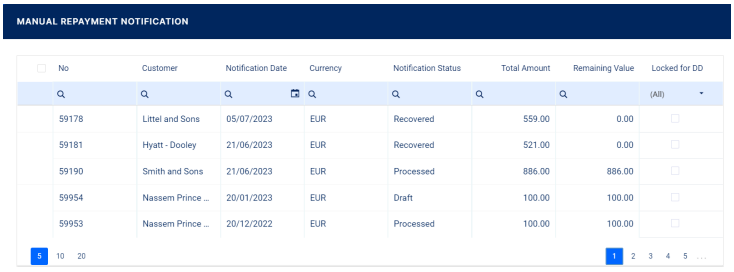
3. View the information is displayed about each notification:
    - Number, date, and status of the notification
    - Customer and currency of the contract
    - Amount of the installment for which the notification was generated
    - Remaining amount from the installment to be paid
    - Maturity date of the notification, automatically calculated adding the value of the Grace

`period for repayment` field at the banking product level to the notification date.

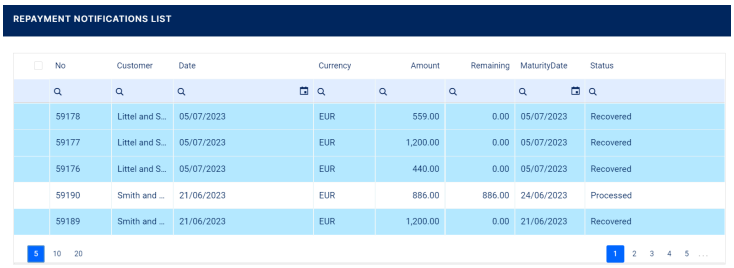
Accessing all the repayment notifications generated by Loan Management

To access all the notifications created in Loan Management, follow these steps:

1. In the FintechOS Portal, click the main menu icon and expand the **Loan Management Operational** menu.
2. To access only manually captured notifications, click **Manual Repayment Notification** menu item to open the **Manual Repayment Notifications** page.

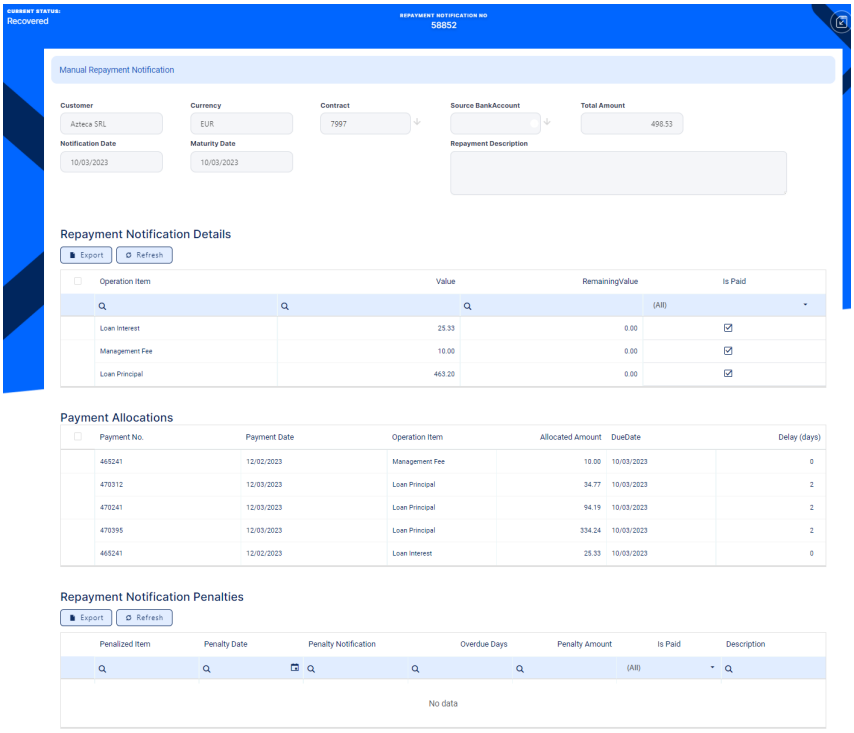


3. To access automatic and manual notifications, click **Repayment Notification** menu item to open the **Repayment Notifications List** page.



Viewing Repayment Notifications

1. To view the details of a repayment notification, double-click the desired record. The **Edit Repayment Notification** page is displayed for automatically generated notifications, or the **Edit Manual Repayment Notification** page for manual notifications, both presenting the repayment notification details.



> **NOTE**
>
> Automatically generated notifications can't be edited!
>
> You can only edit the details of manual notifications in `Draft` status.

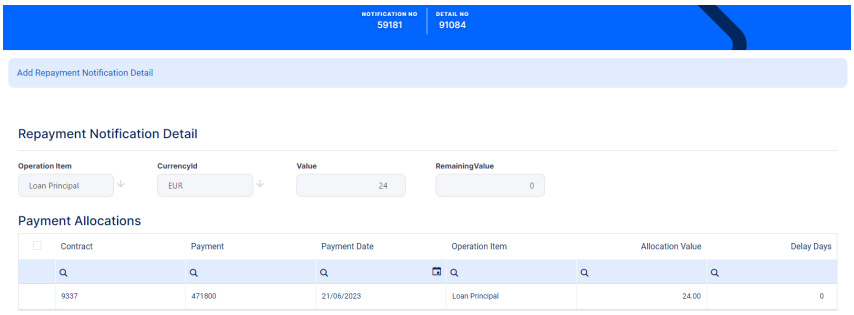
2. View notification specific data in the **Repayment Notification** section:

- **Repayment Notification No.** - The number of the repayment notification record.
- **Customer** - The customer for whom the notification was generated.
- **Currency** - The currency of the notification.
- **Contract** - The number of the contract for which the notification was generated.
- **Notification Date** - The date when the notification was generated.
- **Maturity Date** - The maturity date of the notification. This is calculated by adding the value of the `Grace period for repayment` field at the banking product level to the notification date.
- **Source Bank Account** - The bank account from where the notified amount should be allocated.
- **Total Amount** - The total amount to be paid within the notification (the sum of all the details' values).
- **Repayment Description** - A description of the manual notification.

3. View details (lines) of the notification in the **Repayment Notification Details** section:
   - **Operation Item** - The operation item for which the notification detail is generated.
   - **Value** - The value of the notification detail.
   - **Remaining Value** - The remaining value still to be paid from the notification value.
   - **Is Paid** - This checkbox is automatically marked as true when the full amount is allocated to the detail value. You cannot change this value.

> **NOTE**
>
> Notification details are automatically marked as paid when a repayment transaction performed and approved for the contract is allocated by the system to cover the value of the notification detail.

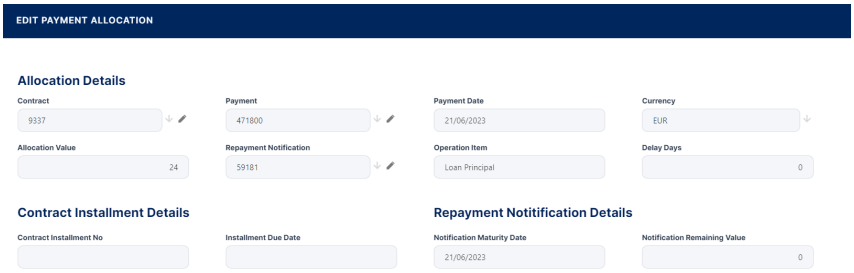
4. To view more information about a notification detail, double-click it to open the **Repayment Notification Details** page:



5. View information about the payments allocated for the notification details in the **Payment Allocation** section:
    - **Payment No.** - The number of the payment.
    - **Payment Date** - The date when the payment was performed.
    - **Operation Item** - The operation item from the notification for which the payment was allocated.
    - **Allocated Amount** - The amount allocated from the payment.
    - **Due Date** - The due date of the notification.

- **Delays (days)** - The number of days passed since the notification's due date.

6. To view more information about a payment allocation, double-click it to open the **Edit Payment Allocation** page:



You cannot edit any of the fields from this page.

> **NOTE**
>
> The operation item is used in the payment allocation process. If the repayment notification is not linked to a contract, then Loan Management takes the operation item value from the allocation method configured within the `ManualAllocationMethod` system parameter. If a repayment notification is created for a contract with `Closed` status, then Loan Management takes the operation item value from the allocation method selected at the banking product level.

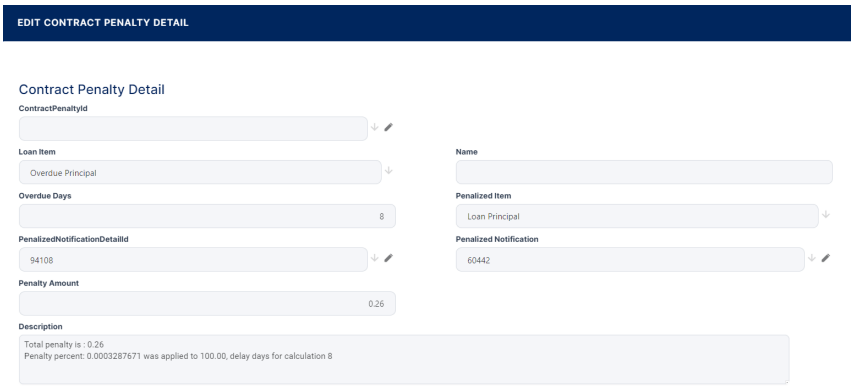
7. View information about the penalties calculated for the manual repayment notifications that were not paid in full until their maturity date in the **Viewing Notification Penalties** section:



All the repayment notifications that are not paid in full after their maturity dates are subject to automated penalty calculation processes performed by Loan Management. Thus, for repayment notifications linked to a contract, Loan Management uses the penalty interests defined within the interests list attached at the banking product level. For repayment notifications which are not linked to a contract, Loan Management uses the penalty interest list specified in the `ManualPenaltyInterestList` system parameter.

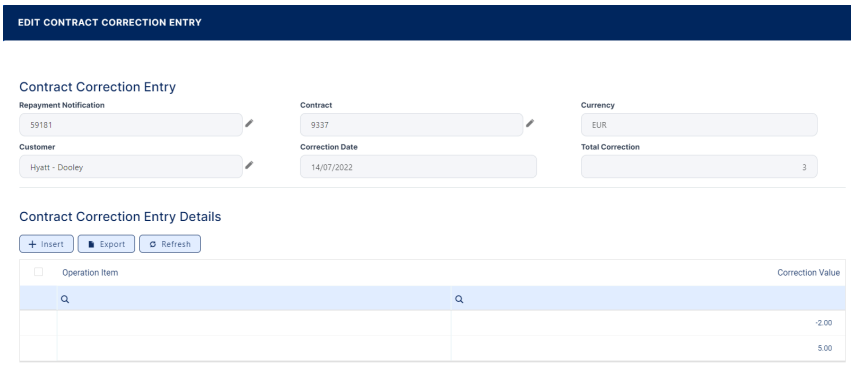
Each penalty displays information about the penalized item, the penalty date, the number of penalty notification, the number of overdue days after the repayment notification's maturity date, the penalty amount, a description, and whether the penalty was pad or not.

8. To view more information about a correction entry, double-click it to open the **Edit Contract Penalty Detail** page:



You can't edit any of the fields from this page.

9. View information about any correction entries created for the notification in the **Viewing Corrections** section. Contract correction entries are automatically generated, for notifications that are overdue, when creating a **Reschedule Overdue** transaction type contract event. Here you can see information about the customer of the contract, the date and currency of the correction entry, and the total amount of the correction (the sum of all the correction details' values).
10. To view more information about a correction entry, double-click it to open the **Edit Contract Correction Entry** page:



- **Repayment Notification** - The repayment notification number.
- **Contract** - The contract number associated with the notification.
- **Currency** - The currency of the notification.
- **Customer** - The customer associated with the notification.
- **Correction Date** - The date when the correction was created.
- **Total Correction** - The sum of all correction entry detail records associated with the current correction entry.

11. Additionally, you can view information about each detail within the correction:

- **Operation Item** - The operation item of the transaction for which the correction entry detail was inserted.
- **Correction Value** - The value of the correction entry detail, in the correction entry's currency.

Understanding Automated Settlement of Repayment Notifications (Direct Debit Settlement Account)

The automated settlement of repayment notification, or direct debit settlement account, is the functionality whereby, if funds are available on the settlement account and the contract has repayment notifications pending for recovery, Loan Management automatically uses the available balance up to full settlement of repayment notifications.

When you have restrictions of any kind on the settlement account or the allocation simply needs to be done as per a legal authority instructions, you can turn off the automated settlement of `Installment` type repayment notifications functionality (the payment allocation) at the **contract level** using the `Direct Debit Settlement Account` checkbox. Deselecting the checkbox leads to the underlying amounts on notifications pending recovery not being retrieved automatically even if there are available funds in settlement account. Thus, financial institution can manage the contracts in case of blocked accounts and control the allocation of funds to outstanding `Installment` type notifications in case of need to impose a block on the settlement account, or manage the settlement of multiple loans from the same settlement account when short on funds and exceptional rules might apply.

This parametrization is available at product level, you can it amended at the contract level, and it is also available at customer level with a system parameter to instruct Loan Management if the customer level setup should impact underlying contracts or not. Thus, you can manage the `Direct Debit Settlement Account` setting at the **customer level**. The customer level setting takes

precedence over the setting at the contract level when creating new contracts. For existing contracts, Loan Management applies the setting configured within the `CustomerToContractDirectDebitSettlementAcc` system parameter.

If the automated settlement of repayment notification functionality is turned off, the contract is pending for manual repayment. You can turn it back on as required, when required, and allow Loan Management to allocate the funds according to its automated processes, using any funds that become available in the settlement account in order to cover pending notifications. When the functionality is turned on or off, the notifications already processed remain unchanged. You can turn the automated settlement functionality on or off even after the maturity of a contract, as long as the contract is not closed.

The following validations are performed for the `Direct Debit Settlement Account` field at the contract level:

- If `Direct Debit Settlement Account` = `True` and new `Installment` type repayment notifications are generated, the system automatically tries to recover the values from Settlement Amount. When the repayment notification is fully paid, Loan Management automatically changes the `Installment` type repayment notification's status to `Recovered`.
- If `Direct Debit Settlement Account` = `True` and old unpaid `Installment` type repayment notifications already exist, the system tries to create recover debt records for the remaining amount for all unpaid `Installment` type repayment notifications, and changes their status to `In recovery`.

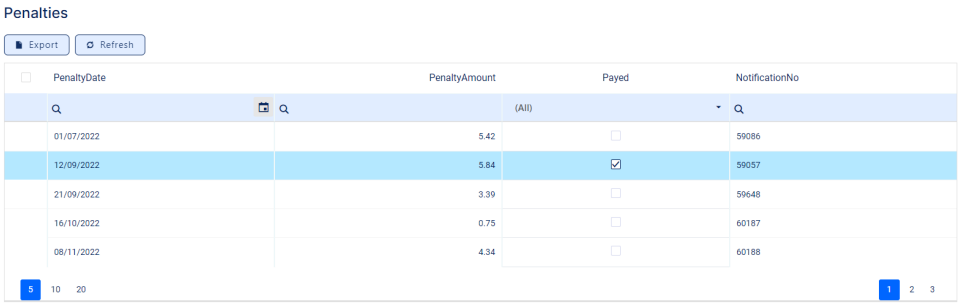
- If `Direct Debit Settlement Account` = `False` and new `Installment` type repayment notifications are generated, the system doesn't register any debt to recover, and changes the notification's status to `Pending Recover`.
- if `Direct Debit Settlement Account` = `False` and old `Installment` type repayment notifications are generated, the system removes debts to recover from the Settlement Account, and changes the status to `Pending Recover`.

Viewing a Contract's Penalties

You can view the penalty interest already notified for the contract in the **Penalties** section of the **Payments** tab. These penalties are automatically calculated by Loan Management for an approved contract based on all the interests with selected **Is Penalty** checkbox that are applied to this contract.

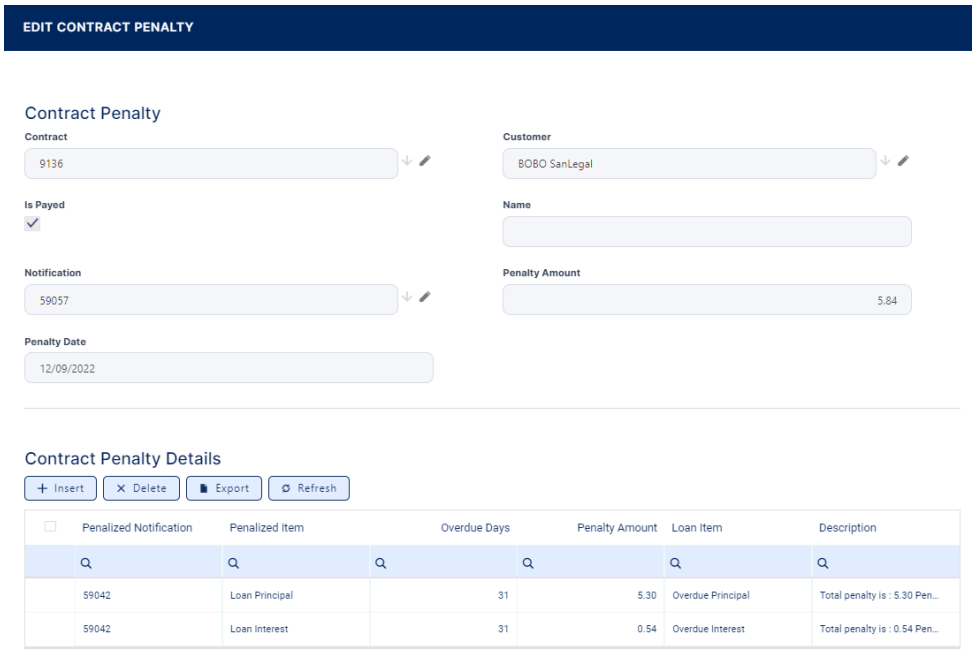
To view the penalties applied to a contract, follow these steps:

1. On an approved contract's **Payments** tab, navigate to the **Penalties** section. If any penalty interest was calculated for the contract, they are displayed here:



2. View basic information about the penalties in the list, such as penalty date, amount, notification number and whether it was paid or not. Payed penalties are also highlighted in blue, for your convenience.

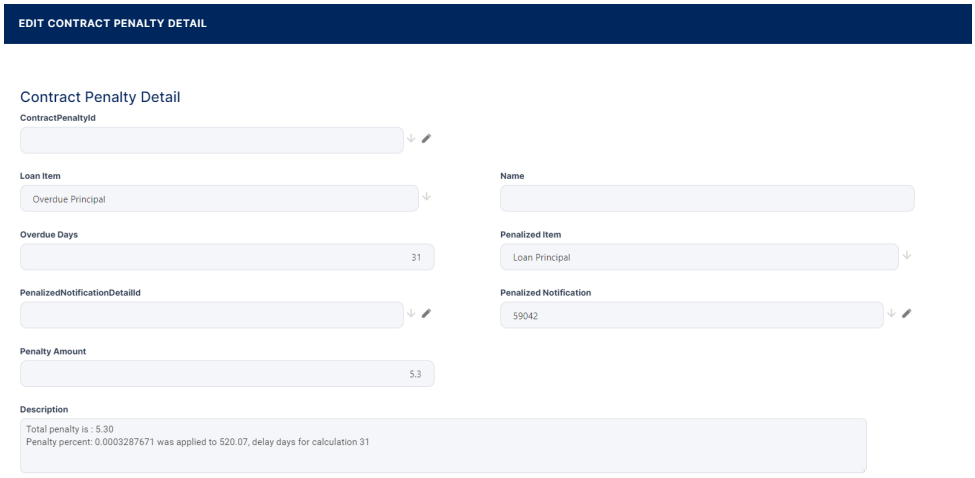
3. To see detailed information about one of the applied penalties, double-click on the desired penalty record. The **Contract Penalty** page is displayed with the selected penalty's details:



You can't edit the information displayed on this page.

4. View the information in the **Contract Penalty** section, as displayed:
   - **Contract** - The number of the contract for which the penalty is applied.
   - **Customer** - The customer for whom the contract was created.
   - **Is Paid** - A checkbox indicating whether the penalty was already paid through a payment allocation or not.
   - **Name** - The name of the penalty.
   - **Notification** - The number of the notification where the penalty is included.
   - **Penalty Amount** - The amount of the penalty expressed in the contract's currency.
   - **Penalty Date** - The date when the penalty was calculated.

5. View the information in the **Contract Penalty Details** section, as displayed:
    - **Penalized Notification** - The notification which was not paid in time and for which the penalty is calculated.
    - **Penalized Item** - The item to which the penalty interest was applied.
    - **Overdue Days** - The number of days since the notification was overdue for payment.
    - **Penalty Amount** - The calculated amount of the penalty.
    - **Loan Item** - The loan item which is used to calculate the penalty interest.
    - **Description** - The description of the contract penalty detail. It contains the total penalty value, the penalty percent or value applied to the number of overdue, and the delay days for calculation.
6. Double-click a detail record to view the details of the penalty on a separate page, **Edit Contract Penalty Detail**:



You can't edit the information displayed on this page.

Manually Capture Notifications

Apart from the notifications automatically generated by Loan Management for each installment that has to be paid for existing contracts that disbursed various amounts to customers, Loan Management also assists you reconcile wrongly processed cases and other situations resulted from delayed processing/ wrong updates. Thus, depending on your user rights, you can manually add notifications for an active contract based on lending product types, term loans, and mortgages, or add notifications for amounts to be recovered even if there is no active contract. Loan Management includes all such manual notifications in the recovery processes.

> **NOTE**
> Loan Management enables you to manage manual repayment notification via the user interface or via integration through APIs. For information about the available endpoints, please visit the Loan Management Developer Guide.
> For information about managing manual repayment notification via the user interface, continue reading this page.

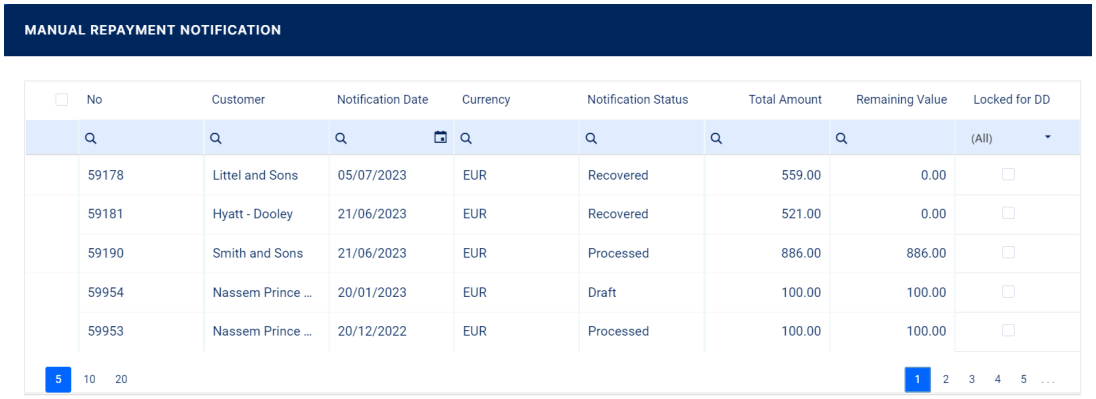
> **NOTE**
> You need one of the **Corporate Credit Officer**, **Retail Credit Officer**, or **Loan Admin Officer** security roles to view, create, delete, and update manual repayment notifications.
> You need the **Loan Admin Officer** security role to update their status to Approved.

Adding Repayment Notifications

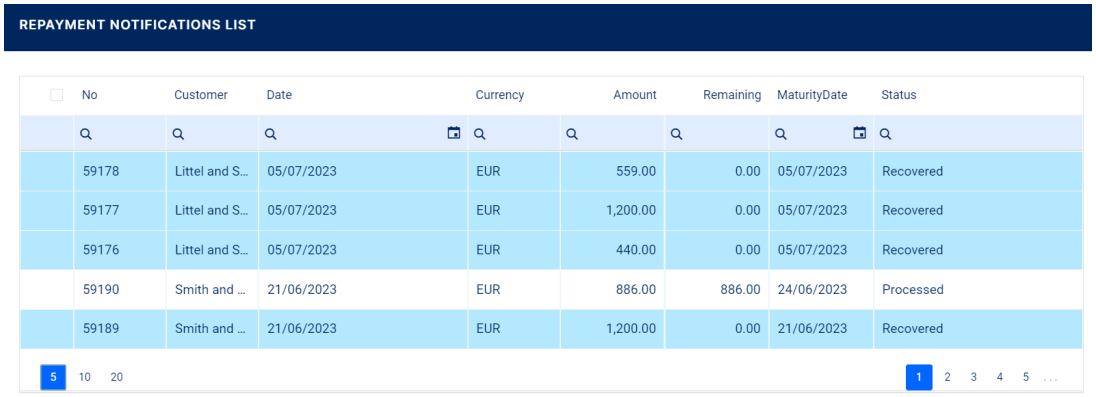
Follow these steps to manually add a repayment notification:

1. In the FintechOS Portal, click the main menu icon and expand the **Loan Management Operational** menu.

2. Click **Manual Repayment Notification** menu item to open the **Manual Repayment Notifications** page.



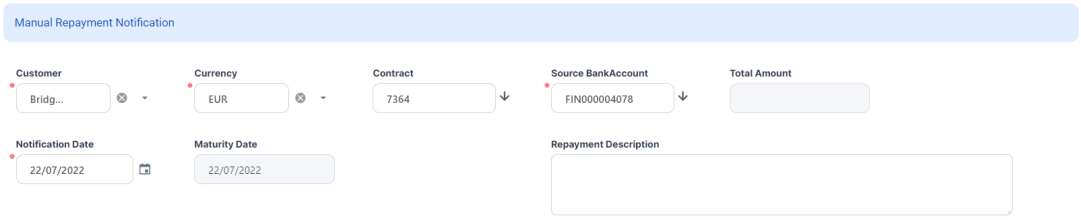
Or, click **Repayment Notification** menu item to open the **Repayment Notifications List** page.



Within the list, the notifications highlighted in blue are already paid, allocated, or closed to payment, while notifications not highlighted (displayed on a white background) remain to be paid.

3. On the **Manual Repayment Notifications** page, click **Insert** to open the **Add Manual Repayment Notification** page.

4. Fill in the following details regarding the notification:

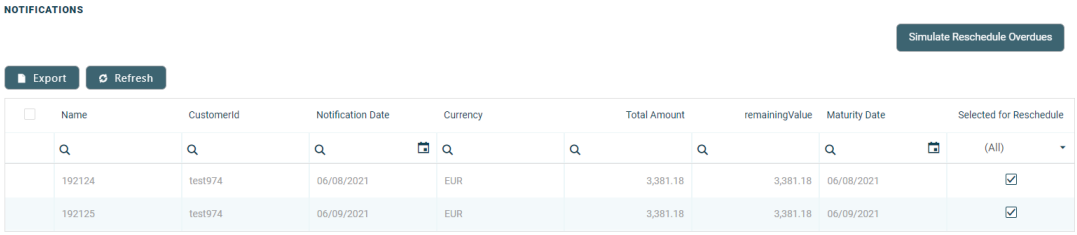


- **Customer** - Select the customer for whom the notification is created.
- **Contract** - Select the number of the contract for which the notification is generated. You can choose from the approved and closed contracts of the selected customer. The currency and the source bank account are automatically filled in using the values from the selected contract. If the notification is not linked to an active contract, you must select a source bank account.
- **Source Bank Account** - Automatically filled in if the contract was selected. Select the bank account from where the notified amount should be allocated. After selecting a source bank account, the currency is changed with the bank account's currency.
- **Notification Date** - Select the date when the notification is created.

> **NOTE**
>
> You can also add manual repayment notification from the contract level's **Payments** tab, clicking **Insert** within the **Repayment Notification** section. In the displayed **Add Manual Repayment Notification** page, some of the fields are automatically filled in based on the contract's information and can't be modified.

5. Optionally, view or edit the following details:
   - **Currency** - Automatically filled in with the currency of the notification, if the contract or the source bank account was selected.
   - **Total Amount** - This read-only field holds the total amount to be paid within the notification, calculated as the sum of all the details' values.
   - **Maturity Date** - This field is automatically filled in with the maturity date of the notification, calculated by adding the value of the `Grace period for repayment` field at the banking product level to the notification date. If no contract is selected, hence there is grace period to consider from the banking product level, then the `ManualGraceRepayment` Loan Management system parameter is used for maturity date calculation.
   - **Repayment Description** - Enter a description for the manual notification.

6. Click the **Save and Reload** button. The manual notification is saved by Loan Management in `Draft` status. You can now continue by adding repayment notification details to it.
   You can view the notifications generated for a specific contract on the **Contract** page, in the **Payments** tab > **Repayment Notifications** section:

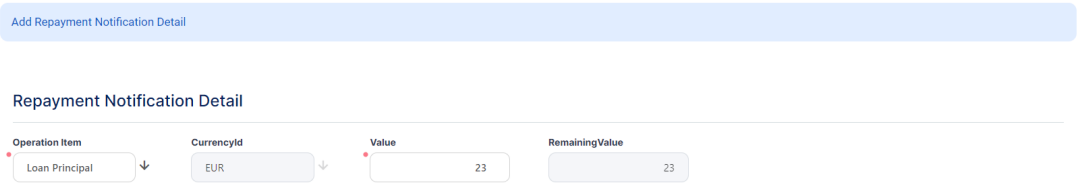


> **NOTE**
> Once the repayment notification is in `Draft` status, you can edit the currency and the source bank account only if there are no notification details created for it.

Adding Repayment Notification Details

Follow these steps to manually add a repayment notification detail:

1. On the **Edit Manual Repayment Notifications** page, click **Insert** to open the **Add Manual Repayment Notification Detail** page. The page already has the currency of the notification and the remaining value still to be paid from the notification value completed. When you create the notification detail, `Remaining Value = Value`.
2. Fill in the following details regarding the notification detail:



- **Operation Item** - Select the operation item for which the notification detail is created.
  The operation item is used in the payment allocation process. If you select an operation item that is not included in the allocation method used for manual notifications (stored in the `ManualAllocationMethod` system parameter), then Loan

Management displays a warning message.

- **Value** - Enter the value of the notification detail. It must be must be greater than 0.

3. Click the **Save and Close** button. The notification detail is saved by Loan Management. You can add as many details as needed to a manual repayment notification in `Draft` status.

> **NOTE**
> For the payment allocation job to process the details, you must first approve the manual repayment notification record.

Approving Manual Repayment Notifications

After adding all the details you need to a manual repayment notification, make sure you approve it by changing its status to **Approved**. Otherwise, the payment allocation automated jobs don't process it.

Loan Management performs the following validations before approving a manual repayment notification:

- The `Total Amount` of the repayment notification must be greater than 0;
- The `Value` fields at the details level must be greater than 0;
- The operation items selected at details level must be included in the allocation method used for manual notifications (stored in the `ManualAllocationMethod` system parameter).

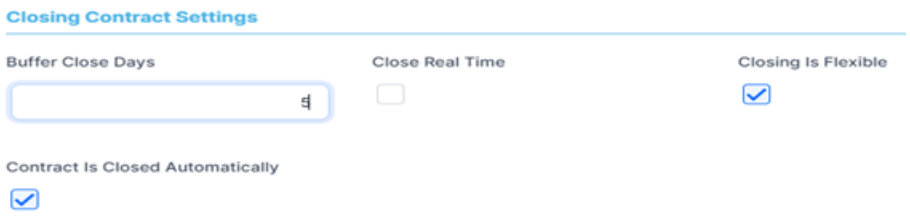
After approval, Loan Management automatically transitions manual repayment notifications from the **Approved** status into **Pending Recover** or **In Recovery** statuses, using the `Auto Process Manual Repayment Notifications` scheduled job. Further, the automated settlement of repayment notification takes the notification and processes it, allocating funds from the source bank account to settle the debt.

Closing a Current Account

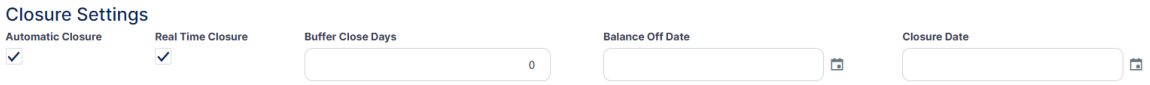
Current account contracts with all their financial obligations met can be closed. Loan Management enables you to close these contracts automatically through scheduled jobs or manually, according to a series of settings defined at the banking product and at the contract level. The automatic closure of contract is triggered whenever the maturity date of the contract is reached and there are no overdue amounts on the contract, or according to the closure settings.

There are cases when you might expect the account to get closed once all amounts recovered, maturity date reached, or you might want such contracts to be closed after a certain number of days, allowing for possible reconciliations, or even leave them to be manually closed or with a localized job. All this is enabled from product level and, if set as negotiable, you can also change the default at contract level. You might need such settings if you work with direct debit and need to allow for the number of days the direct debit can bounce to pass before you really close the deal.

You can configure the closure settings during product definition, in the **Lean Contract Settings**' tab -> **Closing Contract Settings** section, as described in the Banking Product Factory user guide:



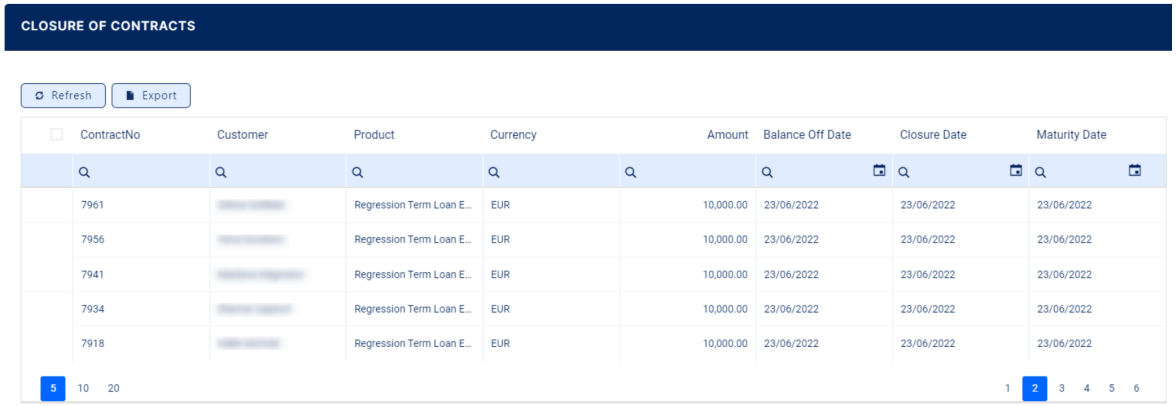
If allowed from the product definition, you can amend the closure settings of the contract, the way Loan Management should behave once there are no more due amounts and the contract can be closed. Perform these configurations in the **Closure Settings** section of the **Overview** tab, during contract creation, for contracts based on banking products having the `Closing Is Flexible = True` setting:



Depending on the real time closure setting, Loan Management uses the one following scheduled jobs to close the contracts automatically:

- `Close Contracts (CB) Job` - this job closes automatically all contracts with `Automatic Closure = True` and `Real Time Closure = False`, with zero available amount and with no further amounts to be recovered, that have `Balance Off Date` filled in and `Closure Date = Current Date`.
- `Close Contracts RealTime(CB) Job` - this job closes automatically all contracts with `Automatic Closure = True` and `Real Time Closure = True`, with zero available amount and with no further amounts to be recovered.

You can see the list of contracts that are ready to be closed in the **Closure of Contracts** report:



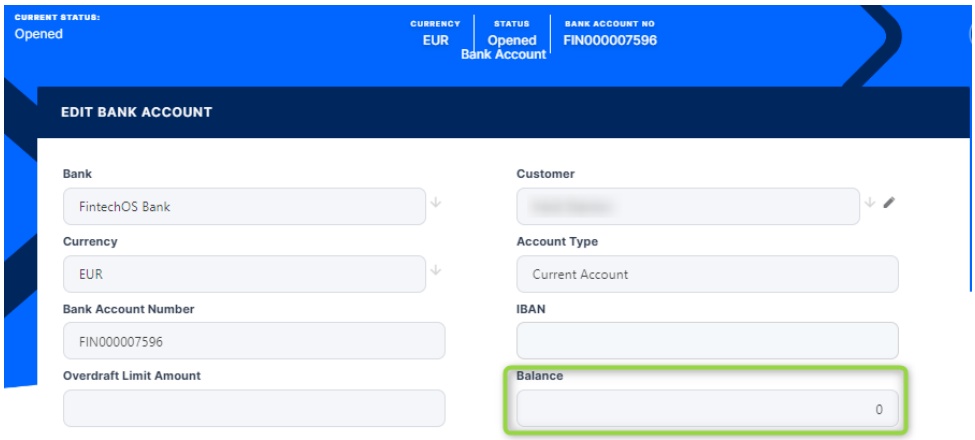
You can also use the `GetClosureOfContracts` endpoint to fetch the same information within your own API integration.

Manually Closing Current Account without Overdraft Contracts

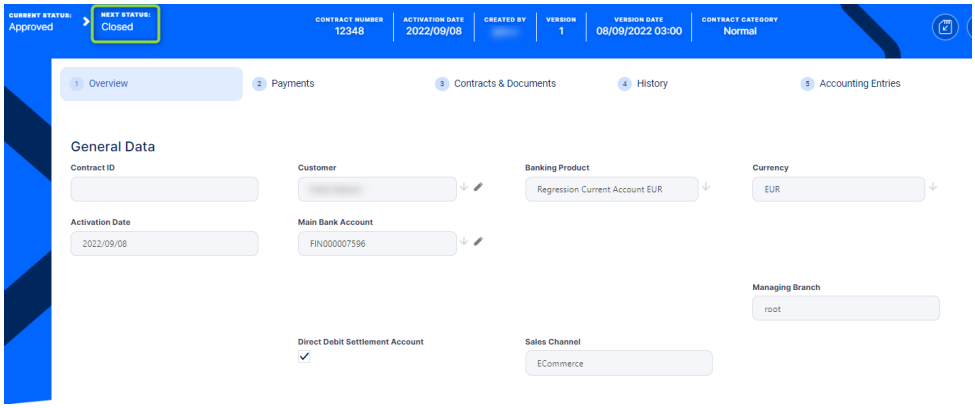
If you opted to close a contract with all the obligations met manually, and not automatically, before its maturity date, then follow these steps:

1. Double-click an approved contract with zero amounts to be recovered and zero available balance, opening it for editing.

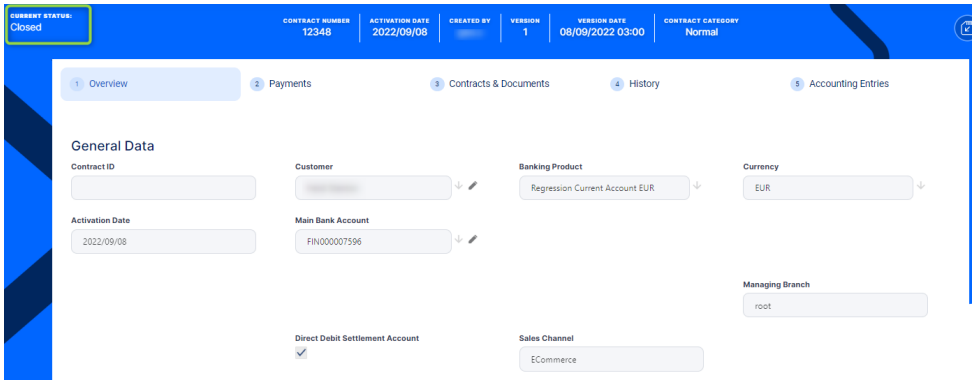
   This example shows you a current account with its balance = 0 and no overdraft:



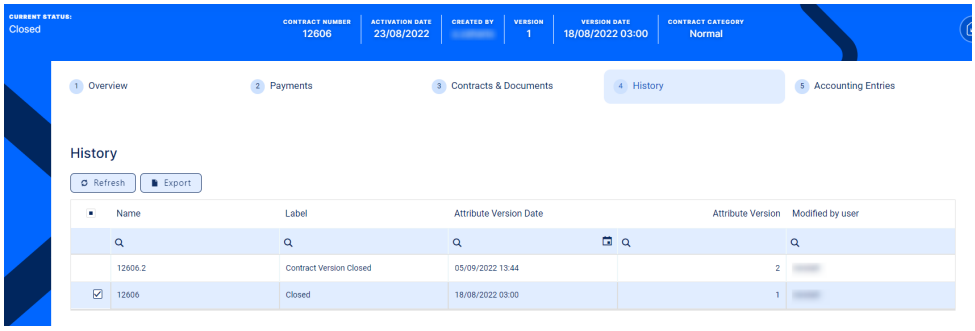
2. Change the contract's **Next Status** into **Closed**.



If Loan Management performs all the validations and finds that the financial obligations are met and there are no more amounts to be recovered or to be transferred out, then the contract's status becomes **Closed**. You can't perform any other operations on this contract.



Any existing versions of the contract are also automatically closed, as you can see in the **History** tab.

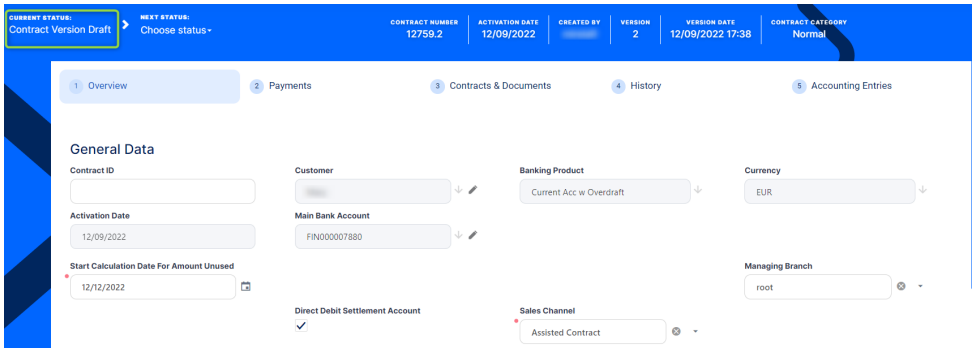


Manually Closing Current Account with Overdraft Contracts

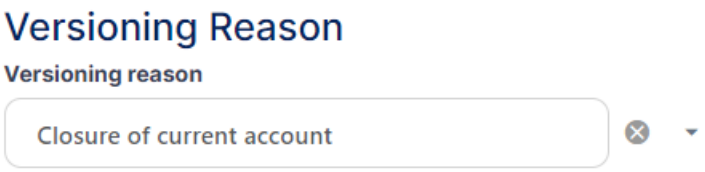
IMPORTANT!
When you plan to close a current account with an attached overdraft functionality, first you have to settle the costs linked to the overdraft, transfer any remaining balance to another account, and only then you can proceed to close the account.

If you opted to close a current account with overdraft contract with all the obligations met manually, and not automatically, before its maturity date, then follow these steps:

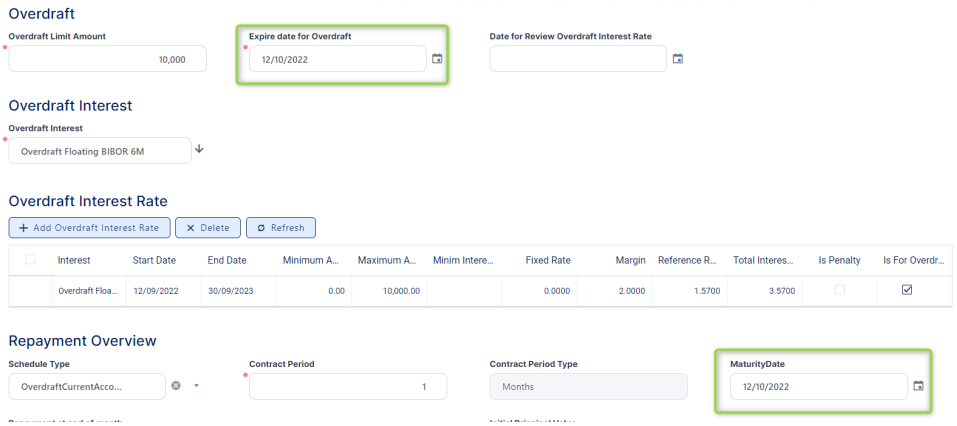
1. Create a new version of the **Approved** contract.



2. Scroll down in the new version's **Overview** tab and fill in a `Versioning Reason`.

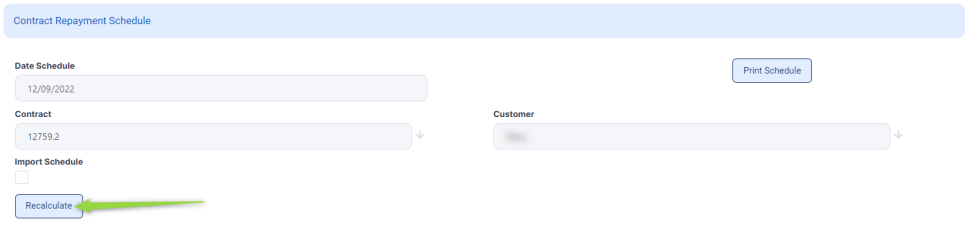


3. Change the `Maturity Date` of the contract to an agreed upon date.



4. If the `Expire Date for Overdraft` was not reached yet, then you must also change it to the same date as you did for `Maturity Date`.

5. Go to the contract's **Payments** tab and double-click the repayment schedule to open it. Click **Recalculate** for Loan Management to perform a recalculation of the repayment plan, to have the last installment due on the new maturity date, with all the accrued amount added into the designated columns.



6. Approve the contract version.

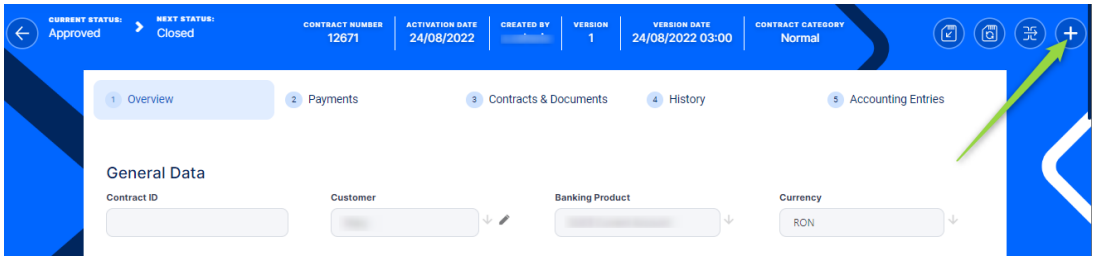
7. At the end of that day, Loan Management notifies the installment. Whenever a due amount is notified before the expiration date of the overdraft, the balance of the current account is credited with the notification amount when running the Loan Management End Of Day scheduled job. When the expiration date of the overdraft is reached, any due amounts are moved to the last, expiration installment of the repayment schedule, on the principal, and the balance, the overdraft limit amount and the overdraft available amount are zeroed. The expiration installment amount can be covered with a top-up or a new overdraft on the same current account.
8. The next day, settle the last installment, performing a top-up.
9. Close the current account manually, performing the same steps described in the "Manually Closing Current Account without Overdraft Contracts" on page 458 section.

Creating New Versions of Existing Current Account Contracts

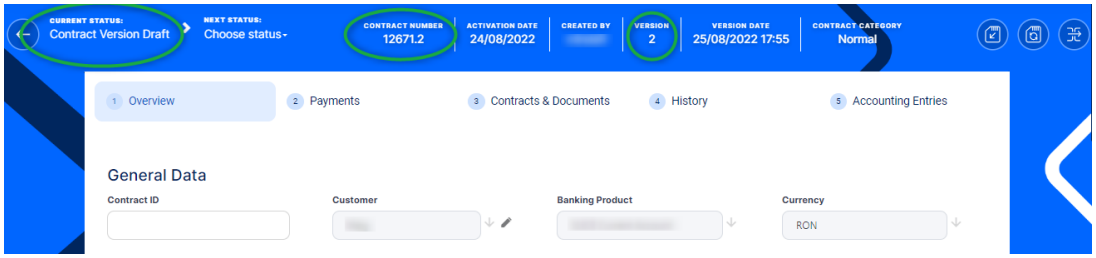
In Loan Management, the contracts are set up for versioning. Thus, if you want to update the details of an approved contract, then you must create a new version of the record.

To create a new version for a record with the **Approved** status, follow these steps:

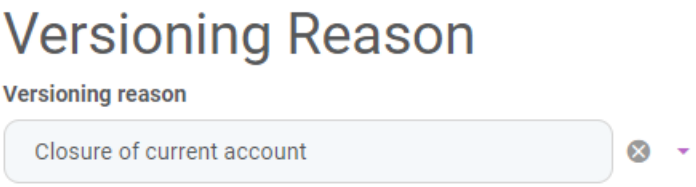
1. While in the **Contract** page of the record selected for updates, click the **New Version** button.



2. View the new version of the contract created by Loan Management, with **Contract Version Draft** status.



3. Edit the desired fields in the **Overview** tab. You can only edit a set of fields for contracts based on specific banking products.
4. Select a versioning reason in the newly displayed editable **Versioning Reson** section.



> **NOTE** Select the `Closure of current account` reason when closing the current account with overdraft contract, as it signals Loan Management to perform the procedures needed in order to settle all the costs of the overdraft and of the current account.

5. Click the **Save and Reload** button.

If you approve the contract in **Contract Version Draft** status, then the original record transitions into the **Contract Version Closed** status and the secondary version becomes the **Approved** currently active contract record.

Read more details about versioning a record on the How to Version an Entity Record page.

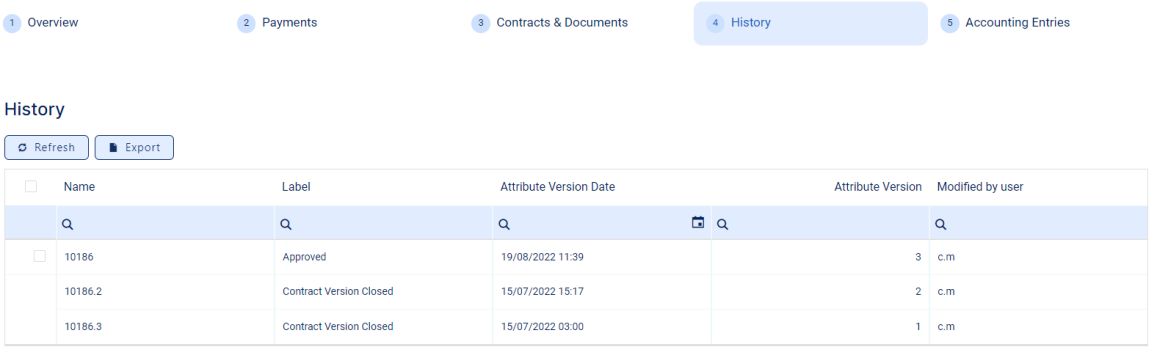
Possible Changes on New Current Account with Overdraft Contract Versions

- The **Financed Amount** value can either be increased or decreased. The amount can be decreased with a number smaller than or equal to the **Available amount**. Financed amount can be increased up to the maximum value specified at banking product level.
- The **Current Account** attached to the contract can be changed to any other active account belonging to the customer.
- **Product Interest** can be changed to any other type set at banking product level.
- **Schedule Type** can be changed with any other type set at banking product level.
- **Contract Period** cannot exceed the maximum set at banking product level.
- **Interest Grace Period** can be changed up to the maximum number of months set at banking product level.

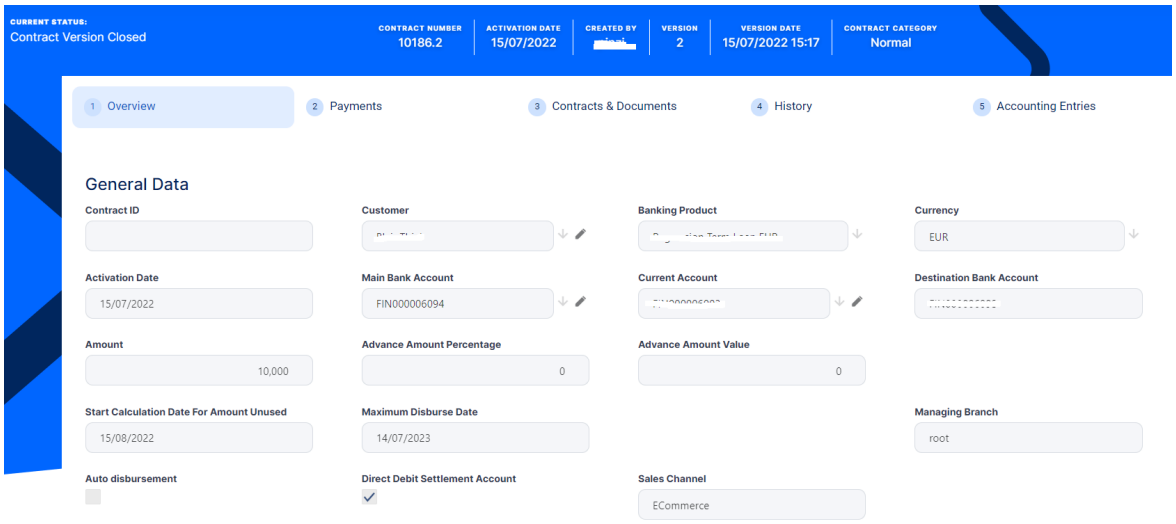
After any of the above changes, in order to approve the new version of contract, the **Contract Repayment Schedule** must be recalculated.

Viewing a Contract's History

You can view the versions of the contract, along with workflow status and the user who modified the record, in the contract's **History** tab.



A contract can have only one **Draft** version, one **Current** version, but it may have multiple **History** versions, which are displayed in this section. Here you can track the contract's life cycle and view older versions that are no longer active. Double-click a version in the list to view its details.



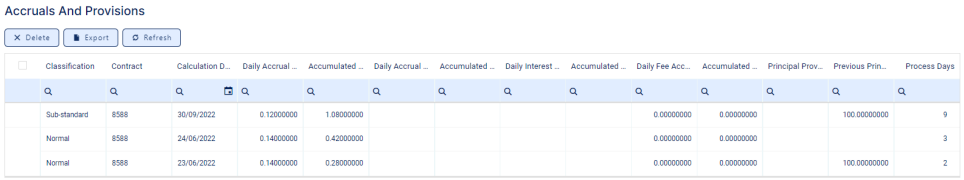
Viewing a Contract's Accounting Entries

You can view all the accounting entries, accounting totals, and accruals and provisions recorded for a contract within the **Accounting Entries** tab of the contract. These records are automatically generated by the system, after performing transactions for an approved contract.

View Accruals and Provisions

To view the records containing daily accrual and provisions, generated automatically by the system respecting the definition of the contract, product dimensions, system parameters and jobs, follow these steps:

1. Navigate to the contract's **Accounting Entries** tab > **Accruals and Provisions** section.

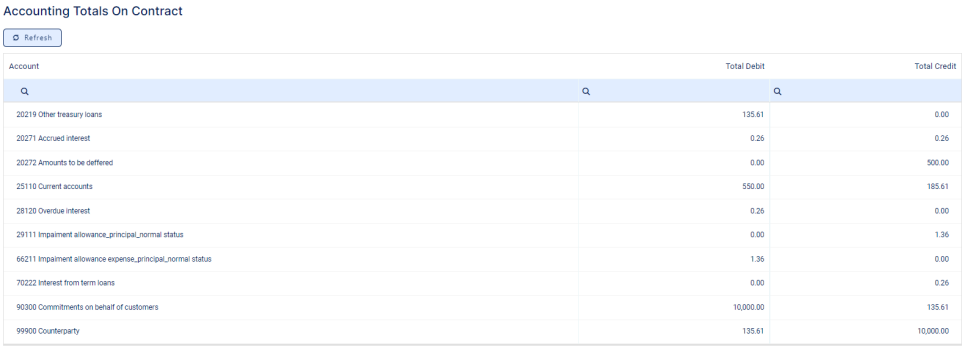


2. View the information displayed for each accrual and provision entry:
    - **Classification** - The classification of the accrual and provision entry. The classification is determined based on the records created in the **Loan Classification** menu. These records classify transactions based on the number of days since a repayment notification is overdue.
    - **Contract** - The number of the current contract.
    - **Calculation Date** - The date when the accrual and provision calculation was performed.
    - **Daily Accrual Interest** - The amount of interest accrued on that day.
    - **Accumulated Interest Accrual** - The total amount of interest accrued until that day.
    - **Daily Interest Provision** - The amount of interest provisioned on that day.
    - **Accumulated Interest Provision** - The total amount of interest provisioned until that day.
    - **Daily Fee Accrual** - The amount of fees and commissions accrued on that day.
    - **Accumulated Fee Accrual** - The total amount of fees and commissions accrued until that day.
    - **Principal Provision** - The amount of principal provisioned.
    - **Previous Principal Provision** - The previous amount of principal provisioned.
    - **Process Days** - The number of days processed.

View Accounting Totals on Contract

To view an overview of the total amounts specified in accounting records generated by the **Generate Accounting Entries** service in the Core Banking END OF DAY (CB) daily job, follow these steps:

1. Navigate to the contract's **Accounting Entries** tab > **Accounting Totals on Contract** section.


Accounting Totals On Contract

Refresh

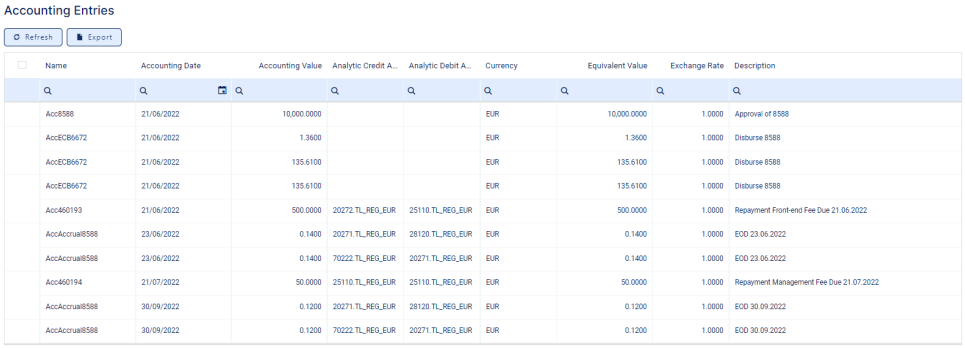
Account	Total Debit	Total Credit
20219 Other treasury loans	135.61	0.00
20271 Accrued interest	0.26	0.26
20272 Amounts to be deffered	0.00	500.00
25110 Current accounts	550.00	185.61
28120 Overdue interest	0.26	0.00
29111 Impairment allowance_principal_normal status	0.00	1.36
66211 Impairment allowance expense_principal_normal status	1.36	0.00
70222 Interest from term loans	0.00	0.26
90300 Commitments on behalf of customers	10,000.00	135.61
99900 Counterparty	135.61	10,000.00

2. View the information displayed for each total amount:
   - **Account** - The account where the operation was performed.
   - **Total Debit** - The amount which was debited from the account.
   - **Total Credit** - The amount which was credited to the account.

View Accounting Entries

To view the accounting for the transactions related to the loan contract generated by the **Generate Accounting Entries** service in the Core Banking END OF DAY (CB) daily job, follow these steps:

1. Navigate to the contract's **Accounting Entries** tab > **Accounting Entries** section.


Accounting Entries

Refresh Export

Name	Accounting Date	Accounting Value	Analytic Credit A...	Analytic Debit A...	Currency	Equivalent Value	Exchange Rate	Description
Acc8588	21/06/2022	10,000.0000			EUR	10,000.0000	1.0000	Approval of 8588
AccECB6672	21/06/2022	1.3600			EUR	1.3600	1.0000	Disburse 8588
AccECB6672	21/06/2022	135.6100			EUR	135.6100	1.0000	Disburse 8588
AccECB6672	21/06/2022	135.6100			EUR	135.6100	1.0000	Disburse 8588
Acc460193	21/06/2022	500.0000	20272.TL_REG_EUR	25110.TL_REG_EUR	EUR	500.0000	1.0000	Repayment Front-end Fee Due 21.06.2022
AccAccrual8588	23/06/2022	0.1400	20271.TL_REG_EUR	28120.TL_REG_EUR	EUR	0.1400	1.0000	EOD 23.06.2022
AccAccrual8588	23/06/2022	0.1400	70222.TL_REG_EUR	20271.TL_REG_EUR	EUR	0.1400	1.0000	EOD 23.06.2022
Acc460194	21/07/2022	50.0000	25110.TL_REG_EUR	25110.TL_REG_EUR	EUR	50.0000	1.0000	Repayment Management Fee Due 21.07.2022
AccAccrual8588	30/09/2022	0.1200	20271.TL_REG_EUR	28120.TL_REG_EUR	EUR	0.1200	1.0000	EOD 30.09.2022
AccAccrual8588	30/09/2022	0.1200	70222.TL_REG_EUR	20271.TL_REG_EUR	EUR	0.1200	1.0000	EOD 30.09.2022

2. View the information displayed for each accounting entry:
    - **Name** - The id of the accounting entry.
    - **Accounting Date** - The date when the entry was generated.
    - **Accounting Value** - The value of the accounting entry.
    - **Analytic Credit Account Code** - The code of the analytic credit account.
    - **Analytic Debit Account Code** - The code of the analytic debit account.
    - **Currency** - The currency of the accounting entry.
    - **Equivalent Value** - The equivalent value of the accounting entry expressed in the contract's currency.
    - **Exchange Rate** - The exchange rate between the accounting entry currency and the contract currency.
    - **Description** - The description of the accounting operation.

Credit Facilities

A credit facility is a grouping of multiple credit products that a customer has arranged with a financial institution under a single credit limit. Financial institutions can offer companies a credit limit for the company as a whole, and the company can then take on different loan products without the need for separate risk assessments. This simplifies access to funds for companies and greatly reduces time-to-cash. Credit facilities also create operational efficiencies for the financial institution, because individual loans no longer need separate risk assessments.

FintechOS Loan Management allows financial institutions to create credit facility agreements for their customers based on approvals.

Credit Facility Implementation Notes

- The credit facility approval is made according to the specifications of the financial institution set during the implementation process.

Business Logic

Let's say a financial institution approves a credit facility for a group or a customer up to EUR 100.000, to be used by the credit facility participants among various currencies:

- guarantees allowed in EUR and USD;
- term loans allowed in EUR and GBP;
- overdrafts allowed in EUR.

First, an agreement is made between a financial institution and a customer (total exposure limit). This limit is used while creating the credit facility, making sure that the credit facility limit amount does not exceed the total exposure limit of the customer. The approval can be revolving or non-revolving, thus both limit and facility have the same nature.

The credit facility holds details about:

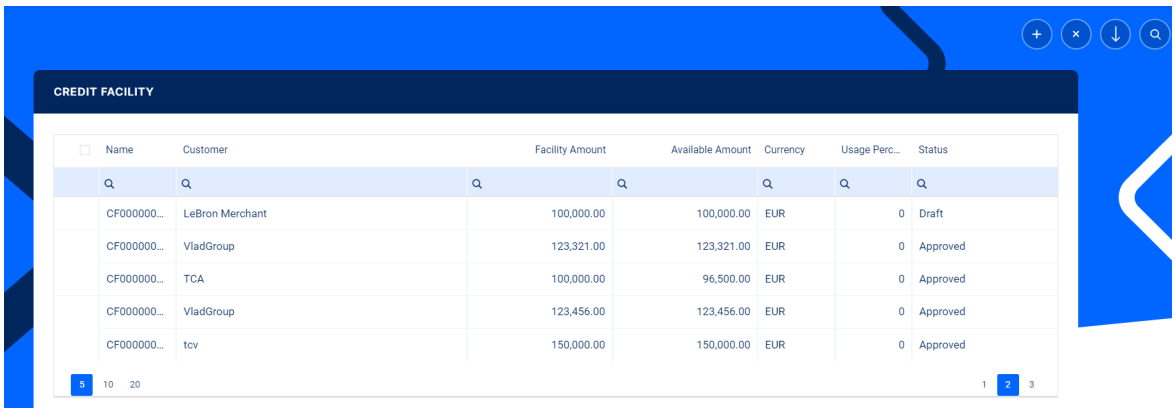
- Allowed banking products, with their preset currency;
- Allowed customers, if the facility is granted for a specific group of customers;
- Covenants, if needed;
- Prices:
  - Unused amount fee, as a percentage to be applied to daily unused amount. The fee is collected from current/ servicing account with a given frequency/ periodicity;
  - Interest and commission related elements, for negotiable product costs.

Loan contracts are entered whenever the customer asks for disbursements, according to the credit facility setup.

Managing Credit Facilities

To manage credit facilities:

1. In FintechOS Portal, click the main menu icon and expand the **Loan Management Operational** menu.
2. Click **Credit Facility** menu item to open the **Credit Facility** page.



On the **Credit Facility** page, you can create a new credit facility, search, edit, or delete existing ones in Draft status.

Credit Facility Life Cycle and States

Credit facilities are complex agreements between a bank and its customers. Therefore the four-eyes principle is applicable here, meaning that a record should be approved by a second bank employee, with higher authorization rights.

A credit facility record has the following business workflow statuses:

- **Draft** - the status of a newly created credit facility record that was not yet sent for approval. While in this status, you can edit the fields from the record's **Credit Facility** tab, but you can't add utilizations to it. Send the record to approval after editing all the necessary details.
- **Pending** - a system status applied to credit facilities sent for approval, but not yet approved. You can't perform any updates in this system status.
- **Approved** - the status of a credit facility record after being authorized by a user with credit facility approval competencies. While in this status, you can't edit the record's details, but you can add utilizations to it within the **Credit Facility Utilizations** tab. If you need to alter the credit facility's details, create a new version based on the current credit facility.

  > **NOTE**
  >
  > Each facility utilization must also be approved by a user with credit facility utilization approval competencies, otherwise, the disbursement of the utilization is performed by Loan Management.

- **Unapproved** - the last status of a credit facility, after manually canceling it directly from Draft status. You can't perform any updates on the record.
- **Closed** - the last status of a credit facility, after manually closing it or after creating a new version based on the current version. You can't perform any updates on the record.

IMPORTANT!
In order to use the credit facility, it must be in the **Approved** status.

Credit Facility Versioning

Loan Management allows you to create new versions for an existing credit facility if you need to modify an existing approved one.

A credit facility version can have the following statuses:

- **Version Draft** - the status of a newly created credit facility version record that was not yet sent for approval. While in this status, you can edit some fields, but you can't add utilizations to it. Send the record to approval after editing all the necessary details.
- **Approved** - the status of a credit facility version record after being authorized by a user with credit facility approval competencies. While in this status, you cannot edit the record's details, but you can add utilizations to it.
- **Version Closed** - the last status of a credit facility version, after manually closing it or after creating another new version based on the current version. You can't perform any updates on the record.

Credit Facility Life Cycle

First, an agreement is made between a financial institution and a customer - usually, a legal entity, for the customer to have easy access to funds whenever in whichever banking product they need it. The amount cannot exceed the customer's approved **Total Exposure** type limit.

This agreement is recorded in the financial institution's system by a clerk, in the form of a credit facility. All details of the agreement are captured while creating the credit facility record: who are the participants with access to funding, what's the usable amount in the chosen currency, what products can be used within this agreement, when is the agreement applicable, under which conditions, whether the facility's

amount increases or decreases over time, and so on. The clerk fills in all the mandatory details, saves the record still in **Draft** status, and then sends it for approval.

Another employee of the financial institution, with higher authorization rights and with credit facility competencies, consults the record and approves or rejects the credit facility, depending on the details entered before by the creator of the record. If rejected, the credit facility's status becomes **Closed**.

If approved, the credit facility, now in **Approved** status, can be used by the customer to access funds. Its details cannot be altered anymore, but the clerk can add utilizations to it up until the credit facility's maturity date, in the form of contracts for banking products listed in the credit facility.

These utilizations, being in fact banking contracts, after creation are still in **Draft** status, and thus have to be further approved by a second employee of the bank, with corresponding contract approval rights. After being approved, a utilization disburses its amount in the customer's account. This amount is taken from the credit facility, thus the available amount is lowered with the sum of the approved utilization.

> **NOTE**
> The total amount of approved utilizations, in any of the banking products' currencies, cannot exceed the amount approved in the credit facility, calculated in the facility's currency based on the exchange rate valid on each day.

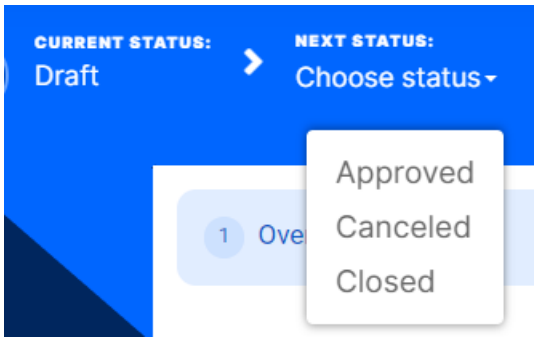
Fee values and accruals are calculated for the approved utilizations and displayed in the **Credit Facility Utilizations** tab, along with any repayment notifications.

You can manually close credit facilities if needed. Records in **Closed** status cannot be altered in any way.

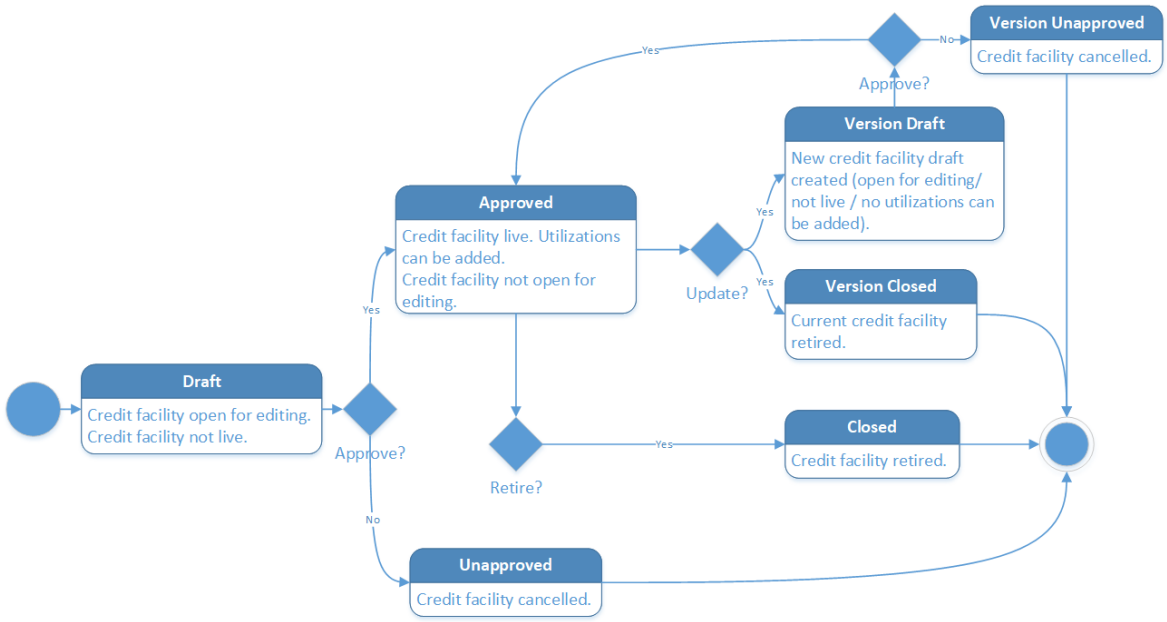
If you have to update the details of an approved credit facility, then you must create a new version of the record. The new version of the record is created in Draft status, thus restarting the life cycle.

Changing Credit Facility Statuses

You can manage a credit facility's life cycle by changing its status from the top right corner of the screen.



The credit facility status transitions are illustrated below:



Note that:

- Once a record is live, its settings can no longer be modified.
- If you want to update the details of a live credit facility, you must create a new credit facility version.
- When you create a new credit facility version, the current version is retired; no updates are allowed on the retired version.
- Every credit facility version starts in a draft state and must go through an approval

process before going live.

- Only one version of a credit facility can be live at one time.

> **NOTE**
> As a best practice, new records or new versions of existing records created on a specific day should be approved on the same day.

Creating Credit Facilities

A credit facility is a grouping of multiple credit products that a customer has arranged with a financial institution under a single credit limit.

Before creating a credit facility, make sure that:

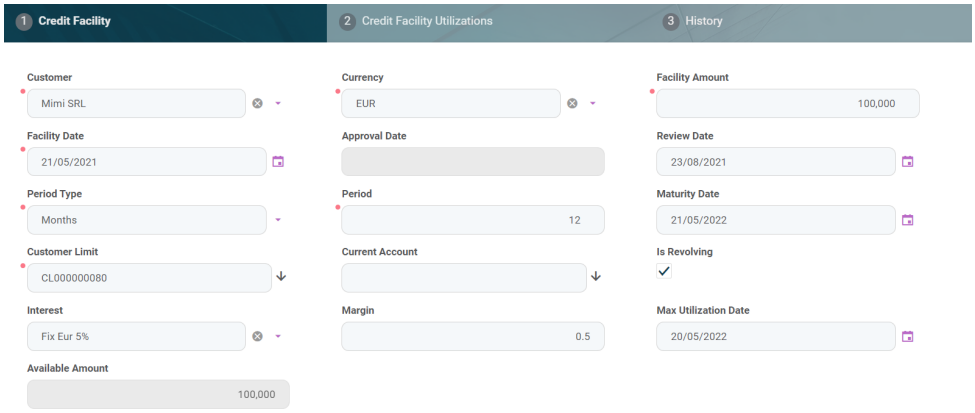
- the customer is recorded in Loan Management,
- a settlement account (a current account contract for the same customer) is set up for the desired currency,
- and the limits are configured according to Loan Management's setup.

To create a new credit facility:

1. Add Details

1. Open the **Credit Facility** page as described in the Managing Credit Facilities section.
2. Click the **Insert** button to display the **Add Credit Facility** page.
   The **Credit Facility** tab requires the basic elements for the creation of a credit facility such as customer, facility amount and currency, period, attached customer limit. Other important details such as participants, products, plans, fees, and contract covenants are captured in specialized sections of the same tab.

3. Fill in the following fields, also available for completion when updating a record in **Draft** status:



- **Customer** - Select from the list the name of the customer with whom the financial institution agreed upon the credit facility. Changing the selected customer at a later point of the record creation process leads to emptying the `Current Account` and `Customer Limit` fields, if these were already selected.
- **Currency** - Select from the list the currency of the credit facility. If the banking products attached to the credit facility are defined in different currencies, then their values are converted in this currency when calculating the facility's available amount. Changing the selected currency at a later point of the record creation process leads to emptying the `Current Account` field, if this was already selected.
- **Facility Amount** - Enter the amount agreed upon to grant within the credit facility, expressed in the currency selected above.

  The facility amount cannot exceed the selected customer limit's value.
- **Facility Date** - Select the date when the facility becomes active. The maturity date is automatically calculated following the formula: `Facility Date` + (`Period` * `Period Type`).
- **Period Type** - Select from the list the period type for the facility's validity.

- **Period** - Enter the number of periods during which the facility is valid.
- **Customer Limit** - Select a customer limit from the list of limits approved for the chosen customer. The list is already filtered to display only the selected customer's already approved **Total Exposure** type limits that have `Is Revolving = True` at the limit level.

> **NOTE**
>
> The previously entered facility amount cannot exceed the selected customer limit's value.

- **Current Account** - Select the customer's bank account where the credit facility amount can be disbursed. The list is already filtered to display only the selected customer's bank accounts in the currency selected before for the credit facility.

4. Optionally, view or fill in the following fields:
   - **Approval Date** - This is the date when the credit facility record is approved by a user with credit facility approval competencies. This date is automatically displayed when the record's status changes to **Approved**.
   - **Review Date** - Enter the date when the credit facility's amount should be reviewed for possible adjustments.
   - **Maturity Date** - This is the automatically calculated maturity date of the credit facility. You can modify this date from the attached calendar, if needed.
   - **Is Revolving** - Select this checkbox to mark the credit facility as revolving. This means that the customer can borrow money repeatedly up to the entered facility amount while repaying a portion of the current balance due in regular installments. Each payment, minus the interest and fees charged, replenishes the available amount.

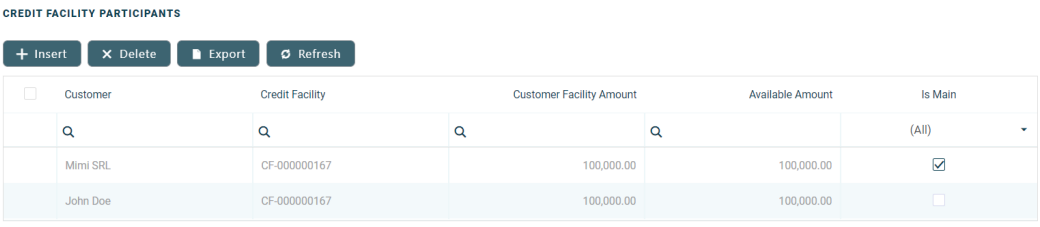
- **Interest** - Select from the list the interest applicable for the credit facility amount. The list is already filtered to display only the interests defined in the selected currency.
- **Margin** - Enter a margin for the credit facility amount.
- **Max Utilization Date** - Select from the calendar the maximum date when the credit facility's available amount can be disbursed through utilizations.
- **Available Amount** - This is the amount still available in the credit facility after disbursing the amounts specified in the approved utilizations, expressed in the facility's currency.
  At creation time, `Available Amount` = `Facility Amount`.

5. Click the **Save and Reload** button.

> **NOTE**
> When creating a credit facility, fill in all the mandatory fields. After saving the credit facility, all the other sections of the **Credit Facility** page become visible and can be completed.

2. Insert Participants

You can insert, delete, or export customers who can participate in this credit facility within the **Credit Facility Participants** section. After the first save operation, Loan Management adds the customer as the main facility participant. If the customer is a group, then all the group members are also added.



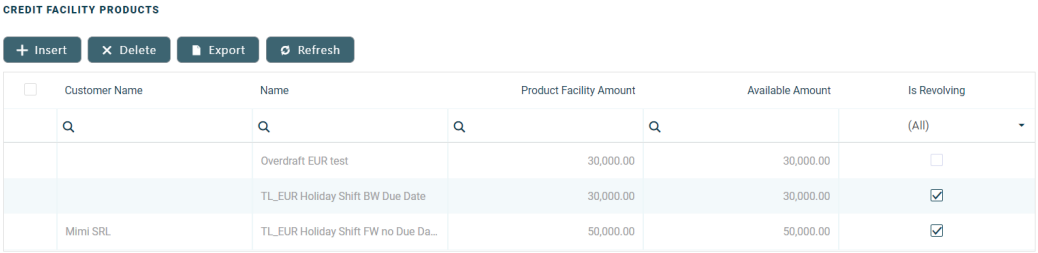
To add a participant, follow these steps:

1. Click the **Insert** button to display the **Credit Facility Participant** page.
2. Fill in, modify or view the following fields:
   - **Customer** - Select from the list the name of the customer who can participate to the selected credit facility.
   - **Customer Facility Amount** - This is automatically filled with the facility amount. You can modify the amount that this specific participant can use within the credit facility. The entered amount cannot exceed the available amount of the facility.
   - **Available Amount** - This read-only field displays the available amount of the facility.
3. Click the **Save and Close** button.

> **NOTE**
> For information purposes, the **Credit Facility Participant** page also displays the **Facility Utilizations** section, containing a list with all the credit facility utilizations corresponding to the selected customer. You cannot perform any action on the records within this list.

3. Associate Banking Products

You can insert, delete or export banking products which can be utilized through this credit facility in the **Credit Facility Products** section.



To add a banking product, follow these steps:

1. Click the **Insert** button to display the **Credit Facility Products** page.
2. Fill in, modify or view the following fields:

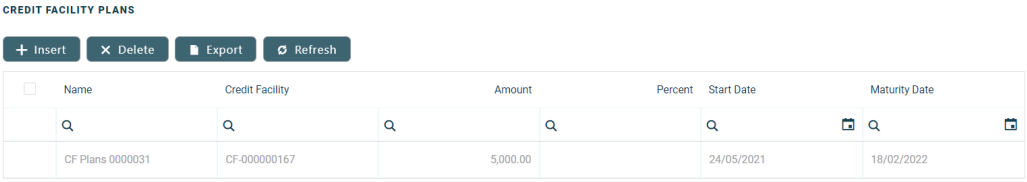
- **Product Facility Amount** - This is automatically filled with the facility amount. You can modify the amount that can be disbursed through the use of this product within the credit facility's utilizations. The entered amount cannot exceed the facility amount.
- **Available Amount** - This read-only field displays the available amount of the facility.
- **Allowed Customer** - Select from the list the customers who is allowed to use this banking product through credit facility utilizations, if the use of this banking product has to be restricted to certain customers. The list is already filtered to display only the customers defined as participants in this credit facility record.
- **Banking Product** - Select from the list the banking product that can be used through credit facility utilizations.
- Is revolving - Select this checkbox to mark the banking product used through the credit facility utilizations as revolving. This means that the customer can borrow money repeatedly up to the entered product facility amount while repaying a portion of the current balance due in regular installments. Each payment, minus the interest and fees charged, replenishes the available amount.

3. Click the **Save and Close** button.

> **NOTE**
> For information purposes, the **Credit Facility Products** page also displays the **Facility Utilizations** section, containing a list with all the credit facility utilizations already created for the selected banking product. You cannot perform any action on the records within this list.

4. Create Plans

You can insert, delete or export plans for the increase or decrease of the facility amount during the credit facility's duration in the **Credit Facility Plans** section.



To add a plan, follow these steps:

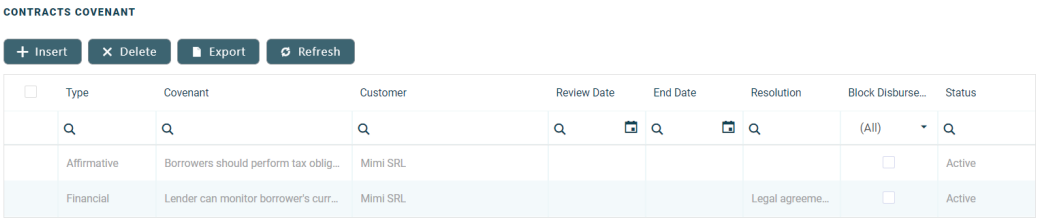
1. Click the **Insert** button to display the **Credit Facility Plans** page.
2. Fill in, modify or view the following fields:
    - **Amount** - Enter the amount which affects the credit facility plan. Use negative values if you wish to decrease the facility amount. Positive values increase the facility amount. This field is mandatory only if `Percent` is not filled in, otherwise, it can't be completed.
    - **Periodicity Type** - Select from the list the periodicity type applicable for the facility plan. The possible values are `Semestrial, Weekly, Monthly, Annual, Bimonthly, Trimestrial, Once, 4 Weeks`.
    - **No Times** - Enter the number of times the plan should increase or decrease of the facility amount, until the credit facility's maturity date.
    - **Percent** - Enter the percent of facility amount which affects the credit facility plan. Use negative percent values if you wish to decrease the facility amount. Positive percent values increase the facility amount. Mandatory only if `Amount` is not filled in, otherwise, it can't be completed.
    - **Start Date** - Select from the calendar the first date when the plan should be executed. Depending on the periodicity type and number of times already completed for plan execution, the maturity date of the plan is calculated. The plan's maturity date cannot exceed the credit facility's maturity date.
    - **Maturity Date** - This read-only field displays the plan's maturity date based on the start date, periodicity type and number of

times already completed for plan execution.

3. Click the **Save and Close** button.

5. Enter Contract Covenants

You can insert, delete or export covenants, certain conventions that customers must abide by after getting the facility in the **Contract Covenants** section.



To add a covenant, follow these steps:

1. Click the **Insert** button to display the **Contracts Covenant** page.
2. Fill in, modify or view the following fields:
    - **Covenant** - Select from 3 possible covenants:
        - *Borrowers should perform tax obligations* - the lenders expect the borrowers to perform their tax obligations to both the business and towards their employees. This covenant is of affirmative type.
        - *Lender can monitor borrower's current ratio* - the lender may continuously monitor the borrower's current ratio to ensure it stays relatively attractive and promising. This covenant is of financial type.
        - *Lender posses the right to prevent merges or acquisitions* - a clear stipulation that the lender possesses the right to prevent

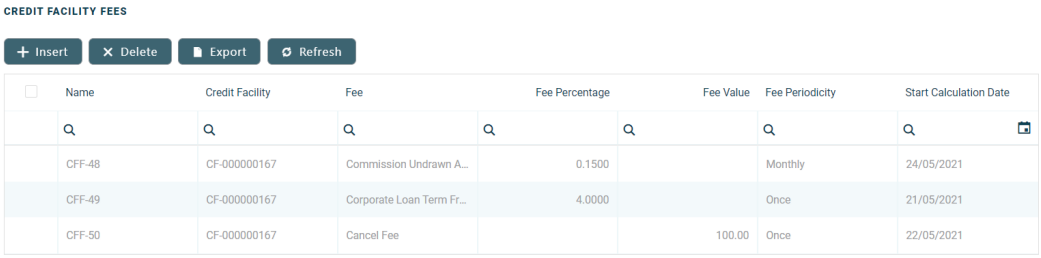
merges of acquisitions without proper notification or full knowledge of the process. This covenant is of negative type.

- **Value** - Enter the value for the covenant.
- **Covenant Type** - This field displays the type of the selected covenant. You can edit it, selecting one the possible values: `Financial, Affirmative`, or `Negative`.
- **Review Frequency (Months)** - Enter the number of months applicable for the covenant review frequency.
- **Review Date** - Enter the date when the covenant should be reviewed.
- **Customer** - Select the customer who must abide by the covenant's terms. The list is already filtered to display only the customers defined as participants in this credit facility record.

3. Click the **Save and Reload** button.
   Loan Management displays a series of fields after the save operation.
4. Fill in the following fields:
   - **Grace Period (Months)** - Enter the number of months acting as grace period for this covenant's resolution, if applicable.
   - **Resolution** - Select from the list the resolution of this covenant, if applicable.
   - **Resolve Date** - Enter the date when the covenant is achieved, if applicable.
   - **End Date** - Enter the last day when this covenant is applicable.
   - **Start Early Termination** - If you select this checkbox, then the credit facility agreement is terminated before its maturity date.
   - **Block Disbursement** - If you select this checkbox, then Loan Management blocks any further disbursements if the covenant is not achieved after end date.
5. Activate the covenant by changing its status to **Active**.
6. Click the **Save and Close** button.

6. Add Fees

You can insert, delete or export fees or commissions that are added to this credit facility in the **Credit Facility Fees** section.



To add a fee, follow these steps:

1. Click the **Insert** button to display the **Credit Facility Fees** page.
2. Fill in, modify or view the following fields:
   - **Credit Facility** - This read-only field displays the id of the selected credit facility record.
   - **Start Calculation Date** - Enter the start date for fee calculation.
   - **Fee** - Select a fee to apply to the credit facility from the list of defined fees & commissions.
   - **Fee Value, Fee Percentage**, and **Fee Periodicity** - These read-only fields display the value, the percentage, and the periodicity of the selected fee, as defined in Loan Management.
   - **Use specific day for aggregation** - Enter a day of the month when the fee accrual should be aggregated.
   - **Use End Of Month for Aggregation** - Select the checkbox to mark the last day of the month as aggregation day for the fee accrual. Mandatory only if the `Use specific day for aggregation` field is not completed.
3. Click the **Save and Reload** button.
   A new list, **Credit Facility Fee Values**, is displayed for viewing after the save operation, containing the calculated fee values for the saved fee. The list displays the fee name, date, value and currency.

After filling in all the mandatory details in the **Credit Facility** tab, the record is still in **Draft** status. Change its status to **Send to Approved** to send it for approval.

> **NOTE**
> You can add utilizations only for credit facility records with **Approved** status.

Sending Credit Facilities for Approval

After creating a new credit facility and filling in all the mandatory details within the **Credit Facility** tab, the record is still in **Draft** status. In this status, the customer cannot access any funding through utilizations (the term for contracts opened for banking products attached to the facility).

You must first send the record for approval to an employee of the financial institution with corresponding competencies, following the 4-eyes principle.

As a clerk, you should change the credit facility's status to **Approved**. Loan Management automatically sends the record for approval to users with credit facility approval competencies.

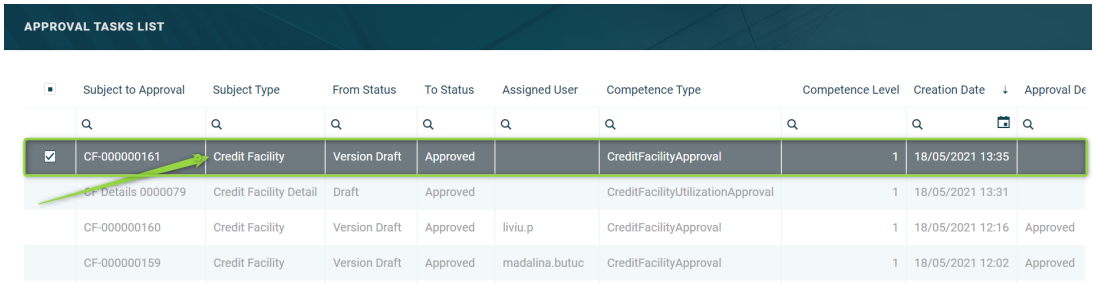
For more details on how to perform a change of status, read the Changing Credit Facility Statuses section.

Approving Credit Facilities

Credit facility records in **Draft** status must be approved in order to add utilizations to it. The record's status can be changed to **Approved** by users with credit facility approval competencies.

To approve a credit facility:

1. Log into FintechOS Portal with a user with credit facility approval competencies.
2. Access **Main** menu > **Approval Tasks** > **My Approval Tasks** to view your list of approval tasks.
3. Find the desired Credit Facility record in the **Approval Tasks List** page and double-click it to open.



4. Click the **Approve** button.

Follow the steps described on the Approve Workflow Transitions page for more detailed instructions.

After approving a credit facility, you can't edit the record's details, but you can add utilizations to it within the **Credit Facility Utilizations tab**. If you need to alter the credit facility's details, create a new version based on the current credit facility.

For more details, read about user competencies and workflow transition approvals.

Adding Utilizations to Credit Facilities

A credit facility utilization is a contract opened for banking products attached to the facility. You can add utilizations to a credit facility record when the record has the **Approved** status.

Follow these steps to add utilizations to a credit facility record:

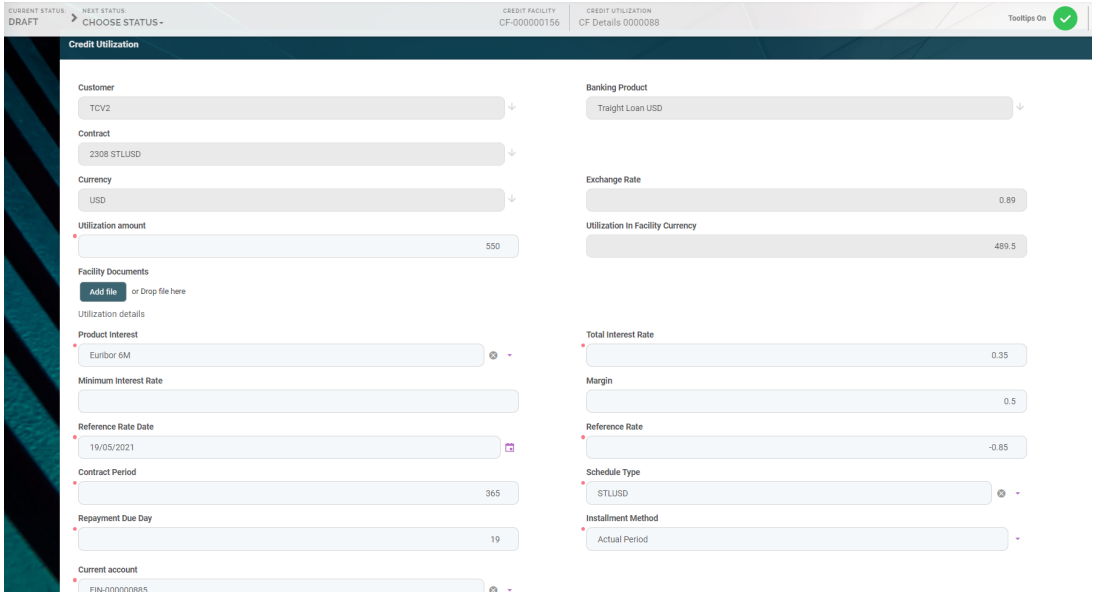
1. On the **Credit Facility** page, double-click the desired credit facility with **Approved** status to edit it.
2. Select the **Credit Facility Utilizations** tab.

3. Click the **Insert** button in the **Facility Utilizations** section to open the **Insert credit Utilization** page.
4. Fill in the following fields:
    - **Customer** - Select the customer who becomes the owner of this utilization. The list is already filtered to display only the customers entered as participants in the credit facility.
    - **Banking Product** - Choose the banking product which is the object of this credit facility utilization. The list is already filtered to display only the banking products attached to the credit facility.
    - **Contract** - You can select an existing contract of the same customer containing the selected banking product where you can attach this credit facility utilization. The list is already filtered to display only the selected customer's contracts that contain that same banking product.
      If no contract is selected, Loan Management automatically creates a new contract for this credit facility utilization.

      > **NOTE**
      >
      > The contract's start date cannot precede the utilization's start date.

    - **Currency**, **Exchange Rate** - These read-only fields display the selected banking product's defined currency, respectively exchange rate for that currency.
    - **Utilization amount** - Enter the amount to be disbursed through this utilization, expressed in the selected banking product's currency.
    - **Utilization In Facility Currency** - This read-only field displays the utilization amount expressed in the credit facility's currency, calculated using the exchange rate displayed above.
    - **Facility Documents** - Add any credit facility utilization documents needed in this field, either by dragging and dropping the file, or by clicking the **Add file** button and selecting the desired file.
5. Click the **Save and Reload** button.
   After saving the credit facility, other fields of the **Insert credit Utilization** page become visible and can be completed.

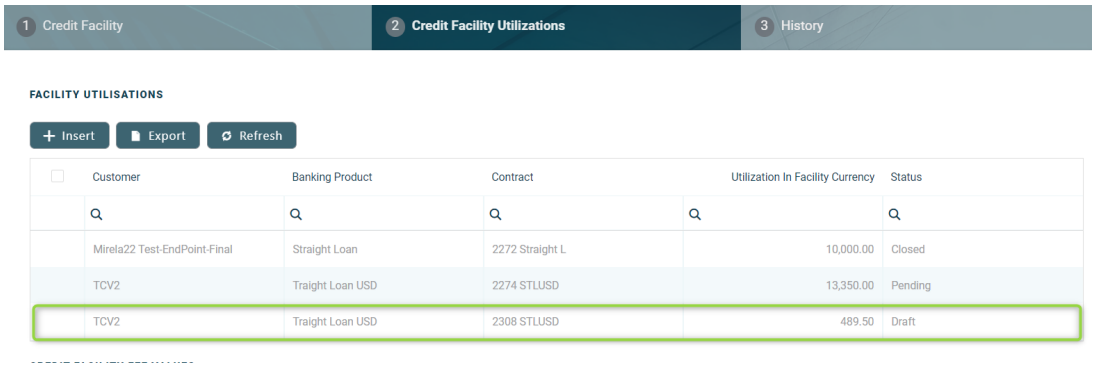
6. Fill in the following fields in the **Utilization details** section:



- **Product Interest** - Select the product interest applicable for this utilization. The list is already filtered to display only the interests defined for the selected banking product.
- **Total Interest Rate** - This field displays the total interest rate of the utilization, as it was calculated for the selected banking product. You can edit the value.
- **Minimum Interest Rate** - Enter a minimum interest rate applicable for this utilization, if needed.
- **Margin** - Enter a margin applicable for this utilization, if needed.
- **Reference Rate Date** - Select from the calendar the date for the exchange reference rate to be used for utilization amount calculation, when the selected banking product was defined with a currency different from the credit facility's currency.
- **Reference Rate** - This field displays the value of the selected exchange reference rate. You can edit the value.
- **Contract Period** - This field displays the contract period in days, as it was defined for the selected banking product. You can edit the value.

- **Schedule Type** - Select the schedule type applicable for this utilization. The list is already filtered to display only the schedule types defined for the selected banking product.
- **Repayment Due Day** - This field displays the day in the month when the repayment is due, as it was defined for the selected banking product. You can edit the value.
- **Installment Method** - This field displays the installment calculation method, as it was defined for the selected banking product. You can edit the value, selecting one of the possible values:
  - `Actual Period` - The first installment should be paid on this month's repayment due date.
  - `Next Period` - The first installment should be paid on the next month's repayment due date.
- **Current account** - Select the current account for disbursing the amount for this utilization. The list is already filtered to display only the bank accounts opened in the facility's currency for the selected customer.

7. Click the **Save and Close** button.
The utilization record is saved in **Draft** status.



8. Double-click the utilization and send it to approval by changing its status to **Send to Approved**.

For more details on how to perform a change of status, read the Changing Credit Facility Statuses section.

IMPORTANT!
Each facility utilization must be approved by a user with credit facility utilization approval competencies, otherwise, Loan Management doesn't perform the disbursement of the utilization.

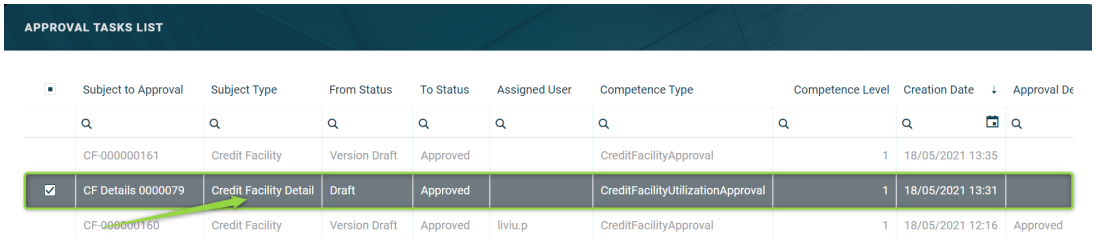
At any given time, the available amount of the credit facility = the facility amount - (the sum of all approved utilizations expressed in the facility's currency).

Approving Utilization Requests

Credit facility utilization records (contracts opened based on the selected credit facility) in **Draft** status must be approved before the utilization's amount can be disbursed in the customer's account. The utilization record's status can be changed to **Approved** by users with credit facility utilization approval competencies.

To approve a credit facility utilization:

1. Log into FintechOS Portal with a user with credit facility utilization approval competencies.
2. Access **Main** menu > **Approval Tasks** > **My Approval Tasks** to view your list of approval tasks.
3. Find the desired Credit Facility Detail record on the **Approval Tasks List** page and double-click it to open.



4. Click the **Approve** button.

Follow the steps described on the Approve Workflow Transitions page for more detailed instructions. For more information, read about user competencies and workflow transition approvals.

After approving a credit facility utilization, you can't edit the utilization's details.

An approved utilization disburses its amount in the customer's account. Loan Management takes this amount from the credit facility, thus the facility's available amount is lowered with the sum of the approved utilization.

> **NOTE**
> The total amount of approved utilizations, in any of the banking products' currencies, can't exceed the amount approved in the credit facility, calculated in the facility's currency based on the exchange rate valid on each day.

Fee values and accruals are automatically calculated by Loan Management for the approved utilizations, and displayed in the **Credit Facility Utilizations** tab, along with any repayment notifications. Read the "Managing Credit Facility Utilization Details" below for more details.

Managing Credit Facility Utilization Details

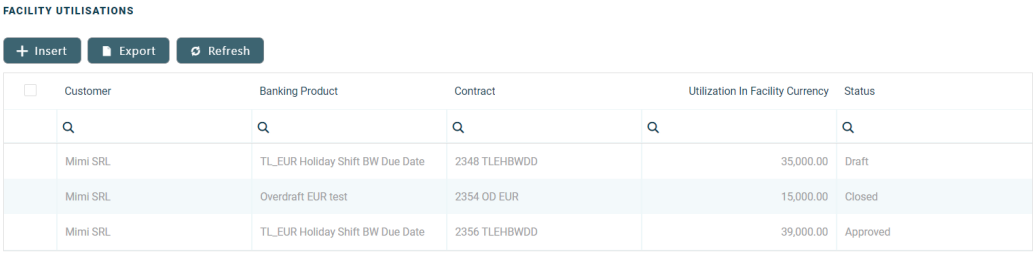
A credit facility utilization is a contract opened for banking products attached to the facility.

You can find the details of the credit facility such as facility utilizations, fee values, accruals and repayment notifications in a credit facility record's **Credit Facility Utilizations** tab. There is no information here to display for records in **Draft** status. You can add utilizations only after the record reaches **Approved** status.

Here are the actions that you can perform on utilizations already added to a credit facility record:

View & Update Utilizations

You can view, insert or export contracts based on the banking products added to this credit facility in the **Facility Utilizations** section within a credit facility record's **Credit Facility Utilizations** tab. These contracts are known as utilizations.



To add a utilization, perform the steps described on the Adding Utilizations to Credit Facilities page.

The already added utilizations are displayed in a list with the following fields:

- **Customer** - The name of the customer who is the owner of this utilization.
- **Banking Product** - The banking product which is the object of this credit facility utilization.
- **Contract** - The number of the contract holding this credit facility utilization, either selected when adding the utilization, or automatically created by Loan Management.
- **Utilization in Facility Currency** - The amount specified in the credit facility utilization, expressed in the credit facility's currency.
- **Status** - The status of the credit facility utilization record. The possible values are:
    - **Draft** - The utilization was created, but it needs further approval.
    - **Approved** - The utilization was approved and its amount was disbursed in the customer's designated current account.
    - **Closed** - The utilization reached its final unalterable status either by being rejected during the utilization approval process, or manually by changing the record's status.

To update a utilization in **Draft** status:

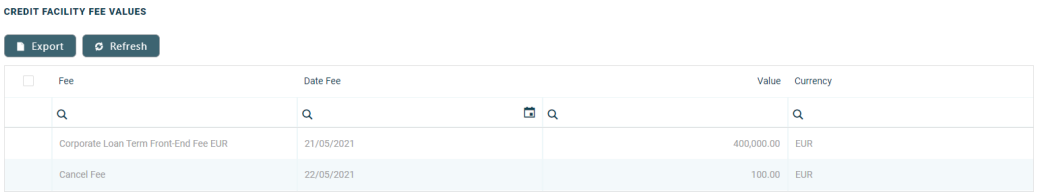
1. Double-click to open the **Credit Utilization** page.
2. Update the editable fields of the utilization according to your needs.

3. Click the **Save and Close** button.

> **IMPORTANT!**
> Approve credit facility utilizations to instruct Loan Management to disburse the amount of the contract in the customer's account.

View Fee Values

You can see or export the fee values already applied to this credit facility in the **Credit Facility Fee Values** section within the **Credit Facility Utilizations** tab.



To view the details of a fee:

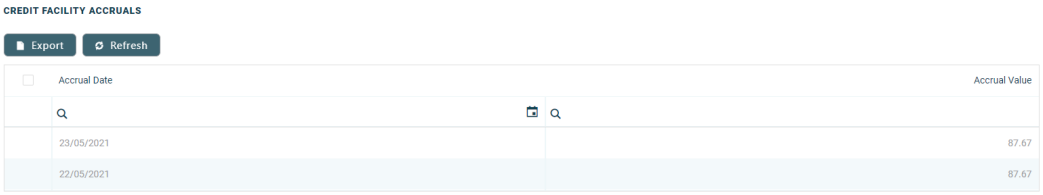
1. Double-click the fee to display the **Credit Facility Fee Value** page.
2. View the following information regarding the fee:
    - **Credit Facility** - The id of the selected credit facility record.
    - **Fee** - The fee applied to the credit facility.
    - **Date Fee** - The date when the fee was applied to the credit facility.
    - **Customer** - The customer who must pay the fee value.
    - **Currency** - The currency of the fee.
    - **Repayment Notification** - The number of the repayment notification automatically generated by Loan Management.

- **Loan Item** - The type of the fee.
- **Value** - The value of the fee, expressed in the fee's currency.

3. Click the **Save and Close** button.

View Accruals

You can see or export the values of the accrual automatically calculated by Loan Management for this credit facility in the **Credit Facility Accruals** section of the **Credit Facility Utilizations** tab.

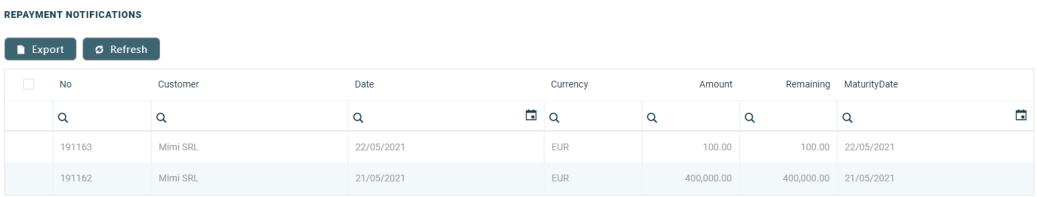


Each accrual lists the following information:

- **Accrual Date** and **Accrual Value** - The date and the value of the accrual calculation, expressed in the credit facility's currency.
- **Fee** - The fee or commission based on which the accrual was calculated. This is only displayed on the **Credit Facility Accrual** page, opened if you double-click an accrual record for viewing purposes.

View & Correct Repayment Notifications

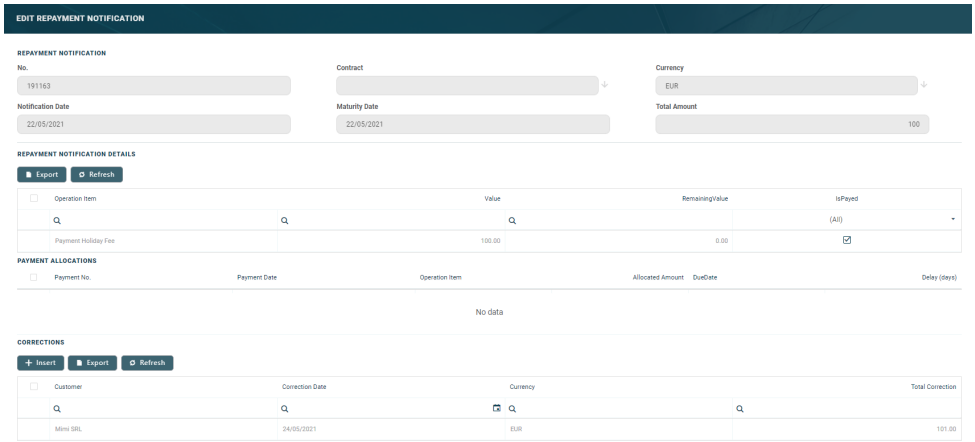
You can see, update or export the repayment notifications automatically issued by Loan Management for this credit facility in the **Repayment Notifications** section of the **Credit Facility Utilizations** tab.



Each repayment notification lists the following information: number, issuance date, the customer participant to the credit facility for whom the repayment notification was issued, the amount, the currency, and the due date of the notification, as well as the amount that remains to be paid by the customer for this repayment notification.

To update a repayment notification:

1. Double-click a record to open the **Edit Repayment Notification** page.
2. View the details of the repayment notification.



3. If needed, insert corrections by clicking the **Insert** button next to the **Corrections** section.
4. On the newly displayed **Add Contract Correction Entry** page, view in the following details: repayment notification number, contract, currency, the name of the customer, the date when the correction entry is saved, and the sum of values entered for in the correction entry details section.
5. In the **Contract Correction Entry Details** section, fill in the following details:
    - **Repayment Notification Detail** - Select the repayment notification detail to be corrected.
    - **Operation Item** - Select the operation item.
    - **Correction Value** - Enter the desired value.

6. The correction entries must be approved in order to be processed. Change the record's status to **Approved**.
7. Click the **Save and Reload** button after performing the desired updates.

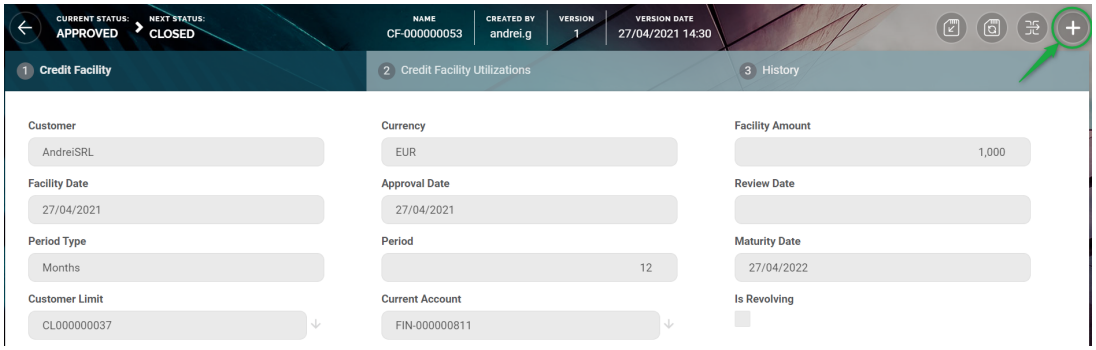
The repayment notification is automatically marked as paid (the **IsPaid** checkbox in the **Repayment Notification details** section is selected) after the payment is processed, either by a Loan Management process or by adding a manual correction.

Creating New Versions of Existing Credit Facilities

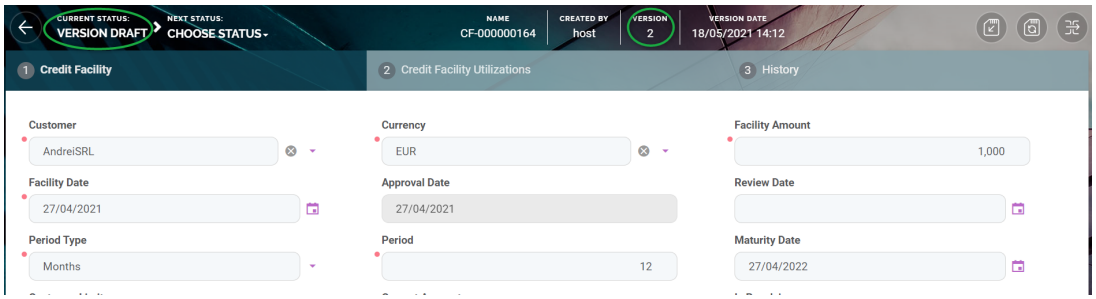
In Loan Management, the credit facilities are set up for versioning. Thus, if you want to update the details of an approved credit facility, then you must create a new version of the record.

To create a new version for a record with the **Approved** status, follow these steps:

1. While in the **Credit Facility List** page, double-click the credit facility record selected for updates.
2. Click the **New Version** button in the top right corner of the page.



A new version of the credit facility is created, with **Version Draft** status, thus restarting the life cycle.



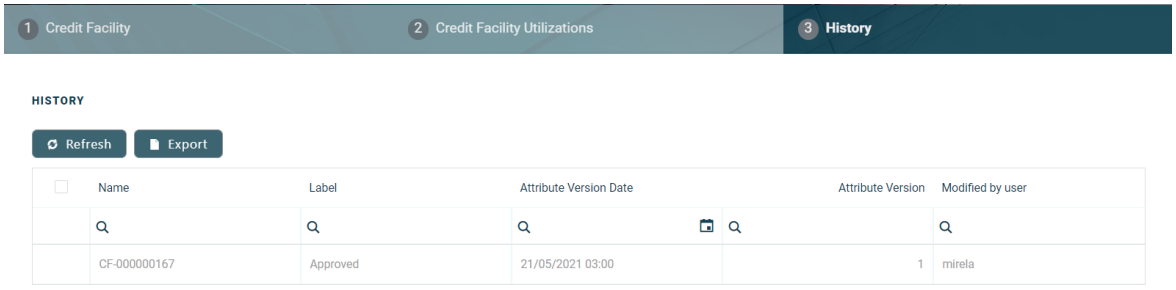
3. Edit the desired fields in the **Credit Facility** tab.
4. Click the **Save and Reload** button.

If you approve the draft version, then the original record transitions into the **Version Closed** status and the secondary version becomes the **Approved** currently active credit facility record.

Read more details about versioning a record on the How to Version an Entity Record page.

Viewing a Credit Facility's History

You can view the versions of the credit facility, their workflow status and the user who modified the record, in the credit facility's **History** tab.



Here you can track the record's life cycle and review older versions that are no longer active (for details, see Credit Facility Statuses).

There are no edits allowed in this tab. Double-click a version in the list to view its details.

Dashboards and Reports

Loan Management facilitates user interaction with a series of in-built dashboards and reports. According to their specific destination, they aid the bank employees in their daily tasks, displaying important, up-to-date information on the statuses of different contracts, events, limits, needed approval tasks, generating reports or offering easy navigation through a button to record creation pages.

These dashboards can be accessed from the FintechOS Portal's **Home** page in accordance with each user's specific access rights.

The following dashboards and reports come along with your Loan Management package:

- **Contracts** - displays a list of the contracts along with a pie-chart specifying the number of contracts in each business status, a list of contract approval requests and a button to access the **Add Contract** page.
- **Customer Limits** - displays a list of the existing customer limit records, a list of the customer limit approval requests and a button for adding new customer limits.
- **Soon to Expire Overdrafts** - displays a list of contracts based on `current accounts with overdraft` banking products whose overdraft functionality is about to expire.
- **Credit Facility Dashboard** - displays a list of the credit facility records along with a pie-chart specifying the number of credit facilities in each business status, separate lists of credit facility approval requests, utilizations and utilizations approval requests, and a button to access the **Create Credit Facility** page.
- **Reports** - contains links to a series of reports such as repayment notifications past due, collaterals in default, limits, closure of contracts, future installments or past due installments.
- **Records To Be Purged** - displays the list of records in `Draft` status that are scheduled to be deleted at the current day's end, grouped on tabs specific for each transaction type: disbursement, early repayments, top-ups, early termination for deposits, loan contracts, payment holidays, reschedule overdue, withdraws, transfers, returned amount or goods, or agreements. Also displays

tabs with agreement records in `Draft` status that are scheduled to be purged.

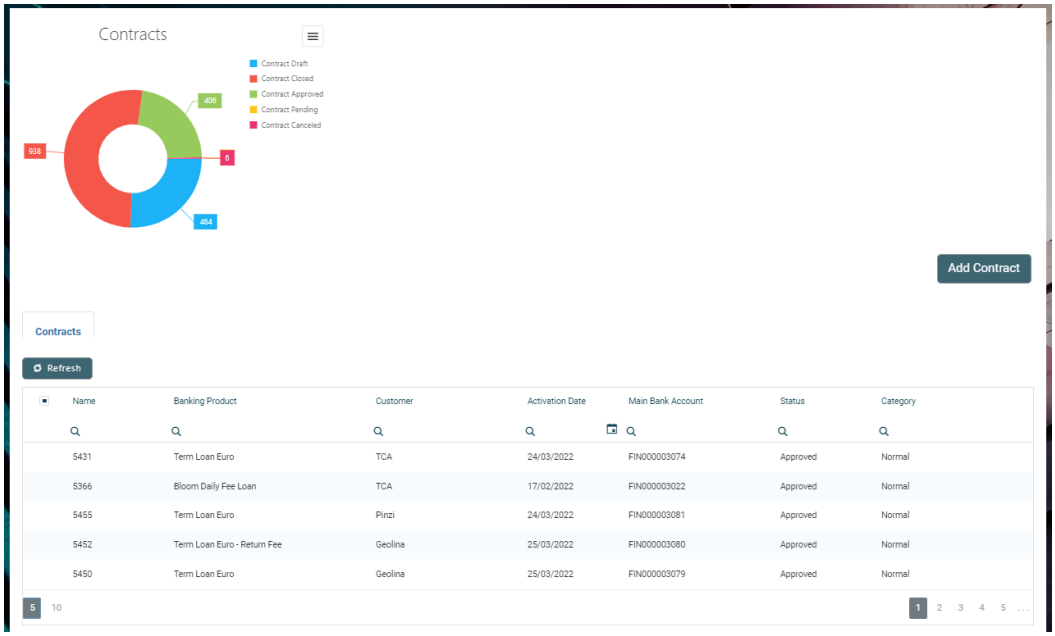
- **Loan Admin Officer Dashboard** - focuses on improving users' experience and productivity for users with the associated role of **Loan Admin Officer** in day-to-day tasks, displaying important notices, overview information about contracts, customers, collateral evaluations, credit facilities, third-party agreements, the user's upcoming tasks, calendar, as well as an integrated chat with colleagues.
- **DLQ Messages** - the dead letter queue dashboard stores messages about the requests that the Async Engine could not process for any reason and allows you to send them for reprocessing.

Contracts

The **Contracts** dashboard displays a list of the contracts created in the system and a list of contract approval requests. The lists can be filtered on every column. Access records from the lists by double-clicking them.

The **Add Contract** button facilitates your access to the **Create Contract** page, where you can create new contracts.

The dashboard also shows a visual of the contracts within the system, displaying a pie-chart that specifies the number of contracts in each status: `Contract Draft`, `Contract Closed`, `Contract Approved`, `Contract Pending` and `Contract Canceled`.



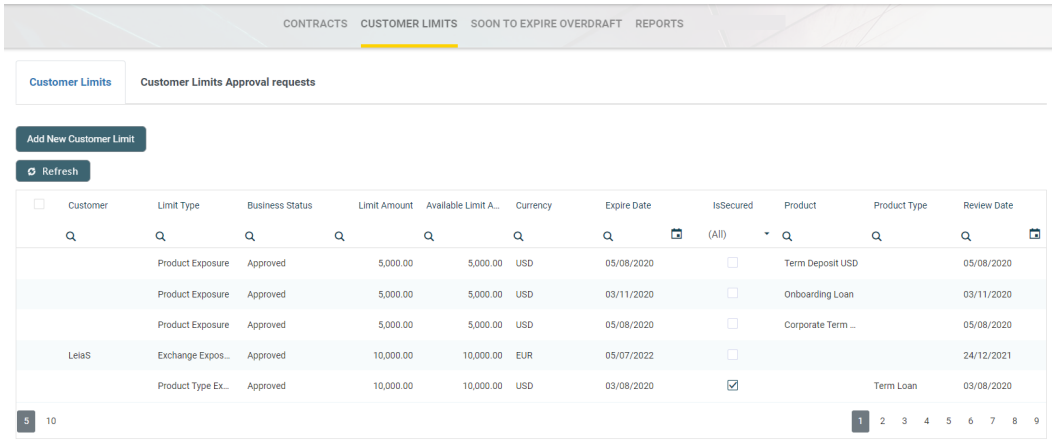
This dashboard can be accessed by users with the following predefined security roles, but note that some actions may be limited according to the role setup:

- Loan Admin Officer
- Supervisor Corporate Officer
- Supervisor Retail Loans Officer
- Corporate Credit Officer
- Retail Credit Officer
- Supervisor Risk Officer
- Risk Officer.

Customer Limits

The **Customer Limits** dashboard displays a list of the customer limit records created in the system and a list of customer limit approval requests. The lists can be filtered on every column. Access records from the lists by double-clicking them.

The **Add New Customer Limit** button helps you add new customer limits.



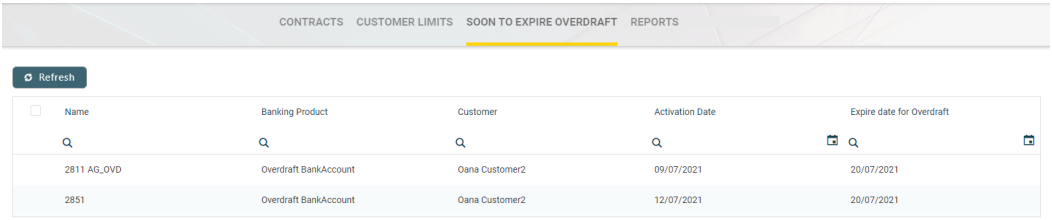
This dashboard can be accessed by users with the following predefined security roles, but note that some actions may be limited according to the role setup:

- Loan Admin Officer
- Supervisor Corporate Officer
- Corporate Credit Officer
- Supervisor Risk Officer
- Risk Officer.

Soon to Expire Overdrafts

The **Soon to Expire Overdrafts** dashboard displays a list of the contracts created in the system based on `current account with overdraft` banking products whose overdraft functionality is about to expire. The Loan Management system parameter `CurrentAccount_WithOverdraft_DaysBeforeExpire` determines the number of days before overdraft expiration when the contract can be displayed in this dashboard.

The lists can be filtered on every column. Access records from the lists by double-clicking them.



This dashboard can be accessed by users with the following predefined security roles, but note that some actions may be limited according to the role setup:

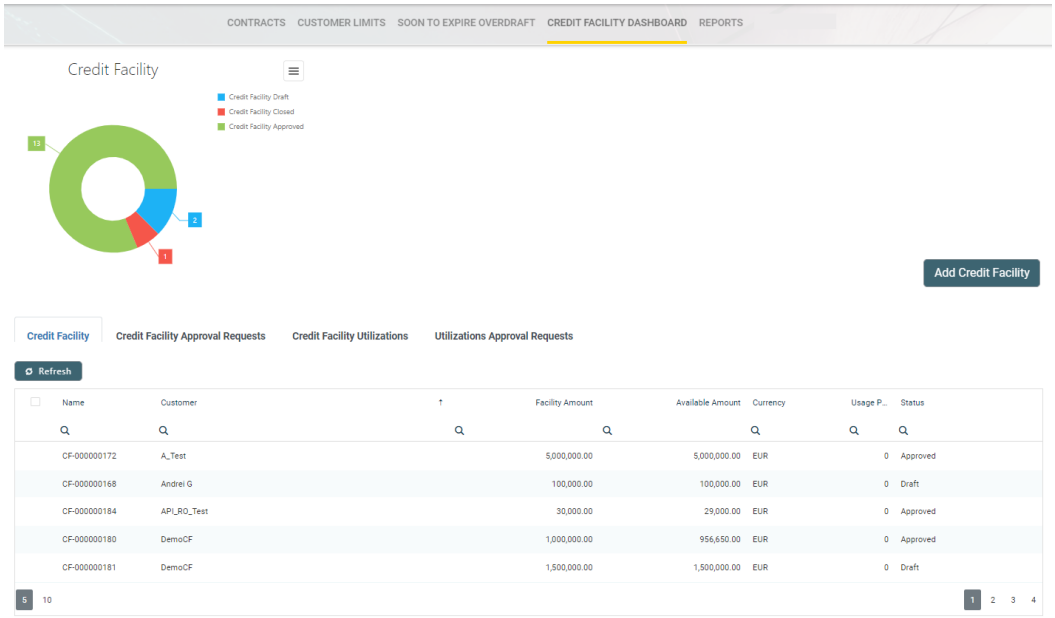
- Loan Admin Officer
- Supervisor Corporate Officer
- Supervisor Retail Loans Officer
- Corporate Credit Officer
- Retail Credit Officer
- Supervisor Risk Officer
- Risk Officer.

Credit Facility Dashboard

The **Credit Facility** dashboard displays a list of the credit facility records created in the system. It also displays separate lists of credit facility approval requests, utilizations and utilizations approval requests. The lists can be filtered on every column. Access records from the lists by double-clicking them.

The **Add Credit Facility** button facilitates your access to the **Create Credit Facility** page, where you can create new credit facilities.

The dashboard also shows a visual of the credit facilities within the system, displaying a pie-chart that specifies the number of records in each status: `Credit Facility Draft`, `Credit Facility Closed`, and `Credit Facility Approved`.



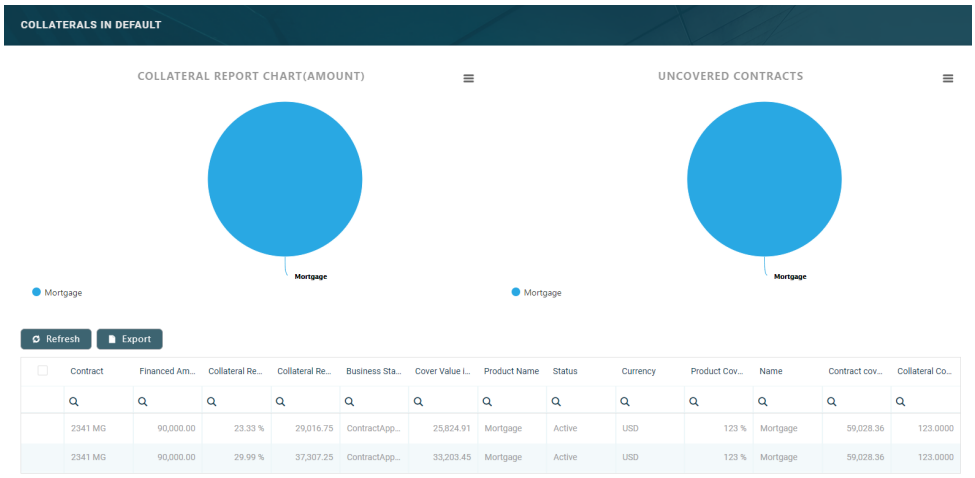
This dashboard can be accessed by users with the following predefined security roles, but note that some actions may be limited according to the role setup:

- Loan Admin Officer
- Supervisor Corporate Officer
- Corporate Credit Officer
- Supervisor Risk Officer
- Risk Officer.

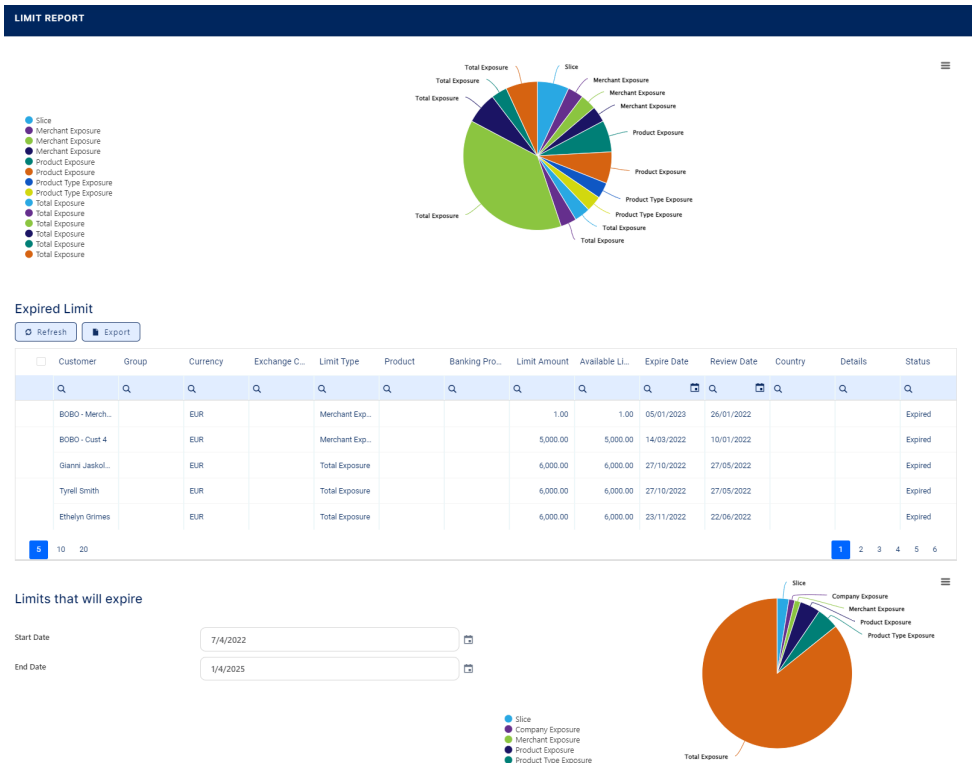
Reports

The **Reports** dashboard contains links to a series of reports:

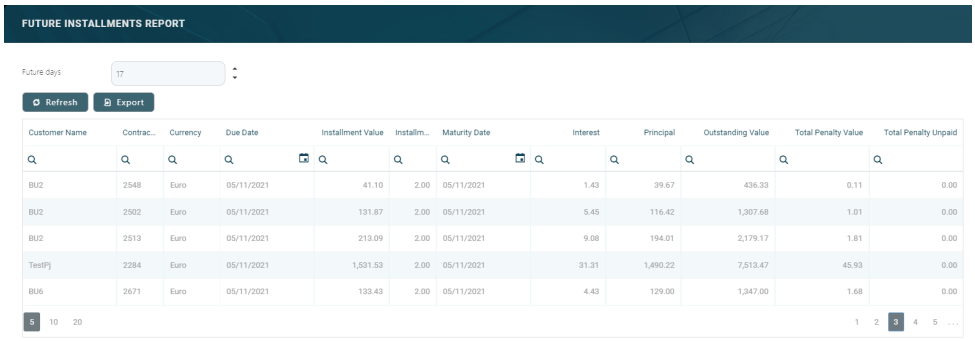
- **Report Days Past Due** - Click this link to display the report of repayment notifications past due date.
- **Collaterals in Default** - Click this link to display the report of collateral records in default.



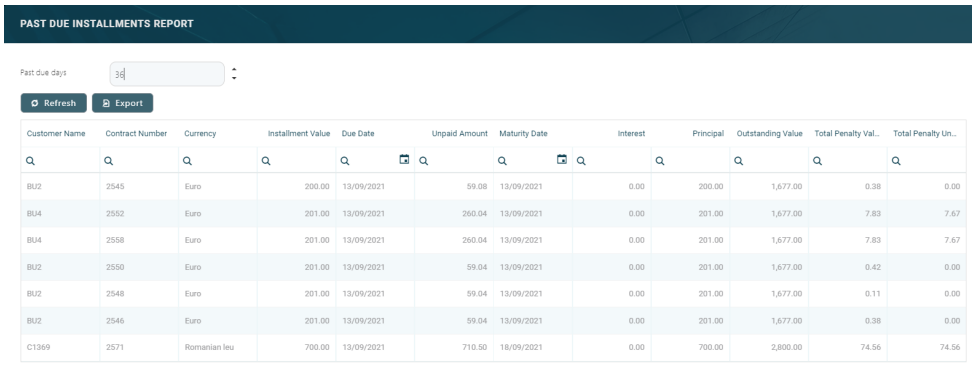
- **Limit Report** - Click this link to display the report of limits records in Loan Management. The report displays different sections for expired limits, limits with available amount lower than 0, limits about to expire and limits to be reviewed, the latest two with the option to select the desired interval of dates. The reports are run automatically with a default value defined in the `DefaultIntervalLimitsReport` Loan Management system parameter, but you can change the intervals according to your needs directly from the report.



- **Future Installments** - Click this link to display the list of installments that are due in the following X number of days from the current date. X represents a default value taken from the `DaysFutureInstallmentsReport` Loan Management system parameter. You can generate the report for a different number of days simply by changing the value of the **Future days** field within the **Future Installments Report** page. The report displays the following information about the future installments: customer name, contract number, currency, due date, installment value, installment number, maturity date, interest, principal, outstanding value, total penalty value and total penalty unpaid.

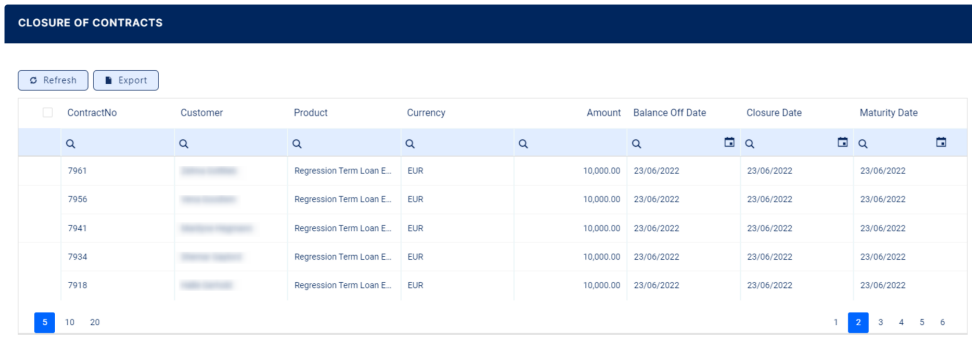


- **Past Due Installments** - Click this link to display the list of installments that were due but not have been fully paid, no matter their origin - normal installments, penalties, transaction fees, etc, - in the last Y number of days from the current date. Y represents a default value taken from the `DaysPastDueInstallmentsReport` Loan Management system parameter. You can generate the report for a different number of days simply by changing the value of the **Past due days** field within the **Past Due Installments Report** page. The report displays the following information about the past due installments: customer name, contract number, currency, installment total value, due date, unpaid amount, maturity date, interest, principal, outstanding value, total penalty value and total penalty unpaid.



- **Closure of Contracts** - Click this link to display the list of contracts that are ready to be closed because they meet the following conditions: the contracts are in Approved status, their maturity date < the current system date, their loan balance = 0, and the remaining amount for all their notifications = 0. The report displays the following information about the contracts that can be closed: contract number, customer

name, product, currency, amount, balance off date, closure date, and maturity date.



You can also use the `GetClosureOfContracts` endpoint to fetch the same information within your own API integration.

The lists can be filtered on every column. Access records from the lists by double-clicking them.

The charts can be downloaded by clicking the Chart context menu in the top right corner of each chart and selecting the desired format: PNG or JPEG image, PDF document or SVG vector image.

The reports can be accessed by users with the following predefined security roles, but note that some actions may be limited according to the role setup:

- Loan Admin Officer
- Supervisor Corporate Officer
- Corporate Credit Officer
- Supervisor Risk Officer
- Risk Officer.

The **Future Installments** and the **Past Due Installments** reports can also be accessed by users with Supervisor Retail Officer and Retail Credit Officer roles.

Records To Be Purged

The **Records To Be Purged** dashboard displays the records in `Draft` status that are due be purged on the current day and have their transaction type's **To Be Purged** field marked as `True`.

> **NOTE**
> In order to be purged on the current day, the record's **Created On** date + the value of the **Purge Number of Days** parameter at transaction type level must be equal with the current date. If the **Purge Number of Days** parameter at transaction type is `0`, then the value of the `DaysBeforePurge` system parameter is considered instead.

The job performing the deletion is `Delete Purged Entries` and it should be scheduled at the bank's level. The lists can be filtered on every column. You can select to display only the records created on a specific day from the calendar button next to the **Created On** column.

The following tabs are available to display the records to be purged, based on their transaction type:

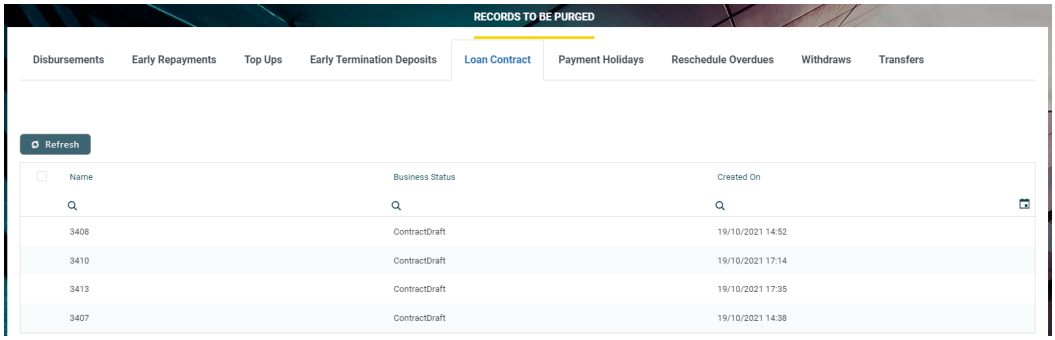
- Disbursements - displays all the disbursement type transactions in `Draft` status which are due to be purged on the current system date;
- Early Repayment - displays all the early repayments type transactions in `Draft` status which are due to be purged on the current system date;
- Top-Ups - displays all the top-up account type transactions in `Draft` status which are due to be purged on the current system date;
- Early Termination Deposit - displays all the early termination deposits type transactions in `Draft` status which are due to be purged on the current system date;
- Loan Contract - displays all the contracts in `Draft` status created based on Term Loan banking products which are due to be purged on the current system date;
- Payment Holidays - displays all the payment holidays type transactions in `Draft` status which are due to be purged on the current system date;

- Reschedule Overdues - displays all the reschedule overdues type transactions in `Draft` status which are due to be purged on the current system date;
- Withdraws - displays all the withdraw type transactions in `Draft` status which are due to be purged on the current system date;
- Transfers - displays all the transfer type transactions in `Draft` status which are due to be purged on the current system date.
- Return Fees - displays all the Returned Amount of Goods type transactions in `Draft` status which are due to be purged on the current system date.
- Agreements - displays all the Agreement type transactions in `Draft` status which are due to be purged on the current system date.

> **NOTE**
> For each transaction type that can be purged (marked with **Yes** in the Predefined Transaction Types table's **Can Be Purged** column), Loan Management displays a tab in the **Records To Be Purged** dashboard only if their **To Be Purged** field is marked as `True`.

For each record, the following information is displayed: name, business status, creation date and transaction type.

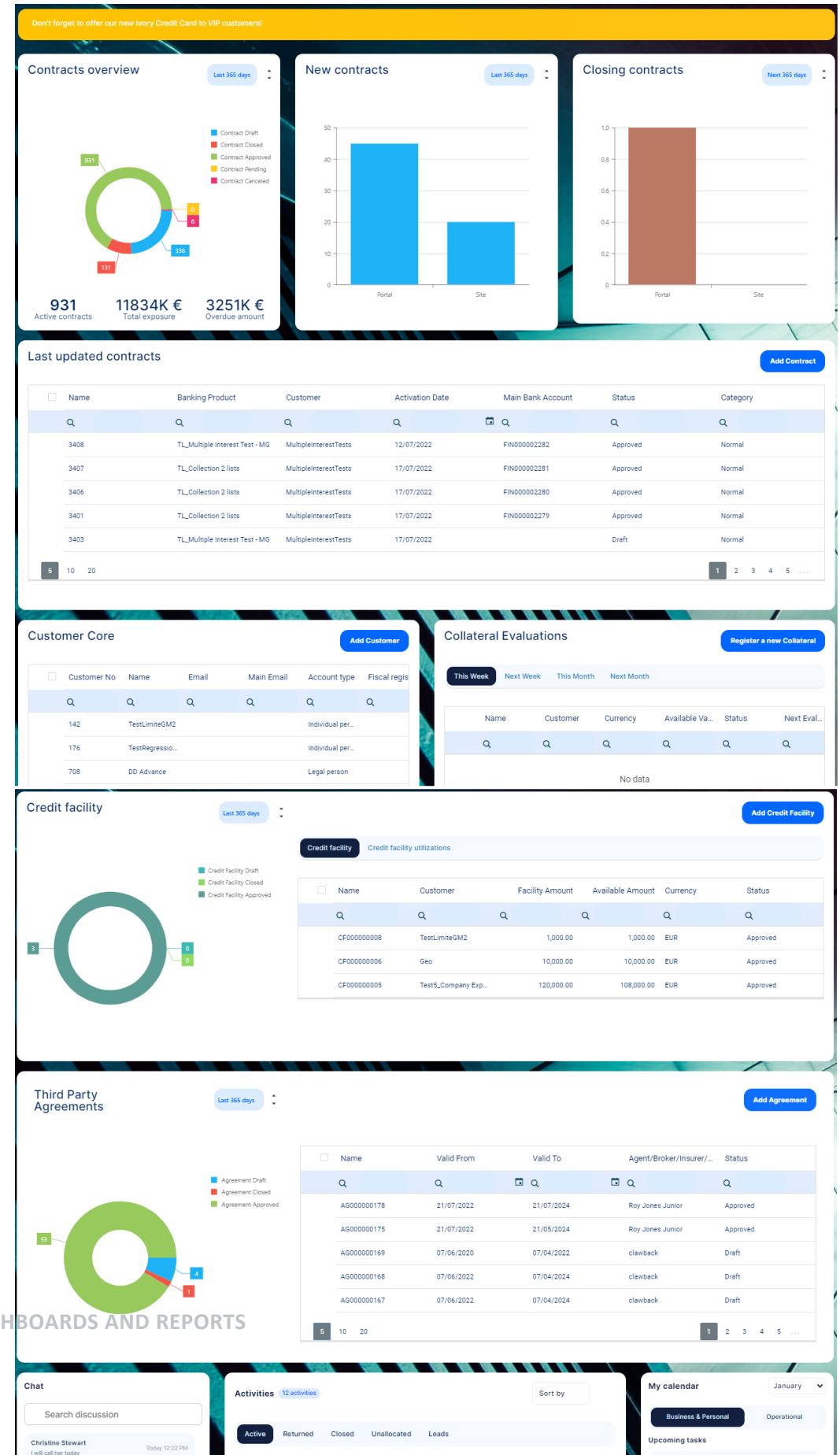
The example below shows the **Loan Contract** tab, which displays all the contracts in `Draft` status created based on Term Loan banking products and which are due to be purged on the current system date.



This dashboard can be accessed by users with the **Loan Admin Officer** predefined security role.

Loan Admin Officer Dashboard

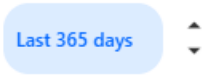
The **Loan Admin Officer Dashboard** aims to be the main overview page for users with the associated role of **Loan Admin Officer** in their day-to-day tasks. It offers in a glimpse the important notices, overview information about contracts, customers, collateral evaluations, credit facilities, third-party agreements, the user's upcoming tasks, calendar, as well as an integrated chat with colleagues, while also enabling users to access record creation or viewing pages without having to navigate to the corresponding menu items.


Don't forget to offer our new Ivory Credit Card to VIP customers!
Contracts overview
Last 365 days
Contract Draft
Contract Closed
Contract Approved
Contract Pending
Contract Canceled
931
0
6
338
131
931
Active contracts
11834K €
Total exposure
3251K €
Overdue amount
New contracts
Last 365 days
Portal
Site
Closing contracts
Next 365 days
Portal
Site
Last updated contracts
Add Contract
Name
Banking Product
Customer
Activation Date
Main Bank Account
Status
Category
3408 TL_Multiple Interest Test - MG MultipleInterestTests 12/07/2022 FIN000002282 Approved Normal
3407 TL_Collection 2 lists MultipleInterestTests 17/07/2022 FIN000002281 Approved Normal
3406 TL_Collection 2 lists MultipleInterestTests 17/07/2022 FIN000002280 Approved Normal
3401 TL_Collection 2 lists MultipleInterestTests 17/07/2022 FIN000002279 Approved Normal
3403 TL_Multiple Interest Test - MG MultipleInterestTests 17/07/2022 Draft Normal
Customer Core
Add Customer
Customer No
Name
Email
Main Email
Account type
Fiscal regis
142 TestLimiteGM2 Individual per...
176 TestRegressio... Individual per...
708 DD Advance Legal person
Collateral Evaluations
Register a new Collateral
This Week
Next Week
This Month
Next Month
Name
Customer
Currency
Available Va...
Status
Next Eval...
No data
Credit facility
Last 365 days
Add Credit Facility
Credit facility
Credit facility utilizations
Credit Facility Draft
Credit Facility Closed
Credit Facility Approved
3
0
0
Name
Customer
Facility Amount
Available Amount
Currency
Status
CF000000008 TestLimiteGM2 1,000.00 1,000.00 EUR Approved
CF000000006 Geo 10,000.00 10,000.00 EUR Approved
CF000000005 Test5_Company Exp... 120,000.00 108,000.00 EUR Approved
Third Party Agreements
Last 365 days
Add Agreement
Agreement Draft
Agreement Closed
Agreement Approved
53
4
1
Name
Valid From
Valid To
Agent/Broker/Insurer/...
Status
AG000000178 21/07/2022 21/07/2024 Roy Jones Junior Approved
AG000000175 21/07/2022 21/05/2024 Roy Jones Junior Approved
AG000000169 07/06/2020 07/04/2022 clawback Draft
AG000000168 07/06/2022 07/04/2024 clawback Draft
AG000000167 07/06/2022 07/04/2024 clawback Draft
Chat
Search discussion
Christine Stewart
I will call her today
Today 12:22 PM
Activities
12 activities
Sort by
Active
Returned
Closed
Unallocated
Leads
My calendar
January
Business & Personal
Operational
Upcoming tasks

The operations available for each user depend on their role and permissions, but users associated with the out-of-the box Loan Admin Officer security role can perform all the actions available on the dashboard.

The lists can be filtered on every column. Access records from the lists by double-clicking them. The records are then displayed in a new tab, so you don't have to leave the dashboard.

The default number of days used to generate the reports can be configured through the `DashboardDefaultLastXDays` Loan Management system parameter. You can modify the number within each section of the dashboard, using the up and down arrows next to `Last x days`, or entering the desired number.



The dashboard contains the following sections dedicated to different processes:

- **Important Notice** - The information displayed here is dependent on the Loan Management implementation. If the functionality was implemented, the section displays important notifications. This is where the financial institutions can set reminders to their employees.



- **Contracts Overview** - It displays a pie-chart that specifies the number of contracts in each status: `Contract Draft`, `Contract Closed`, `Contract Approved` , `Contract Pending` and `Contract Canceled`. It also shows the number of active contracts, the amount of total exposure of these contracts, and the total overdue amount, both amounts converted into the currency specified in the `DashboardCurrency` Loan Management system parameter.

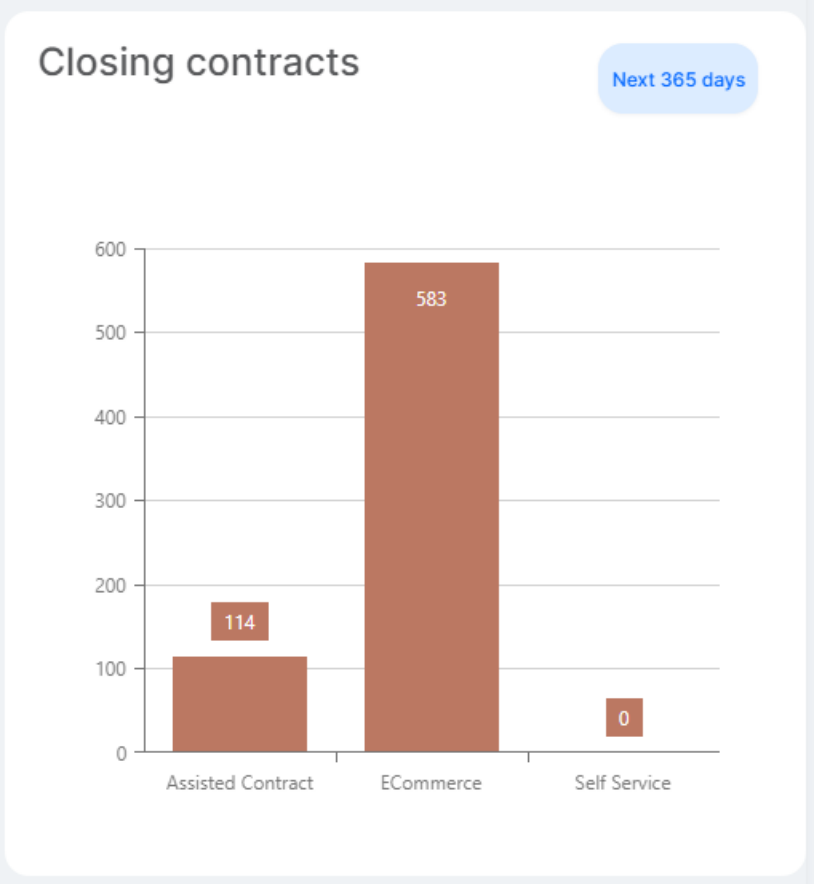


- **New Contracts** - It displays a bar chart with the exact number of approved contracts created through different sales channels, during the last x days based on activation date, where x represents the configured default number of days.



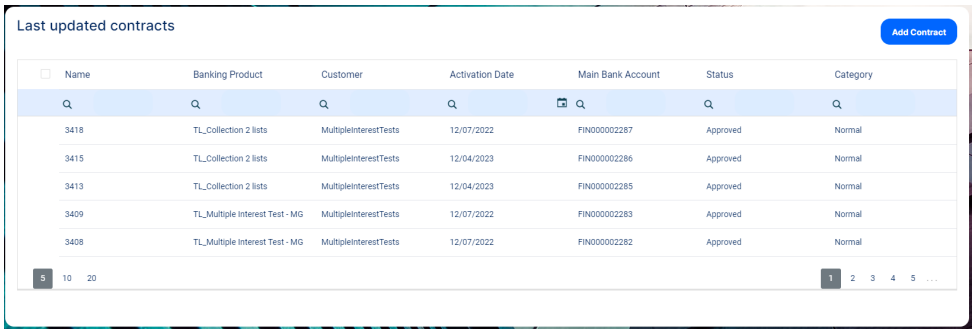
Double-click any of the bars to open a new browser tab displaying the list of contracts that fit the selected criteria.

- **Closing Contracts** - It displays a bar chart with the exact number of the contracts closed through different sales channels, during the last x days based on closing date, where x represents the configured default number of days.

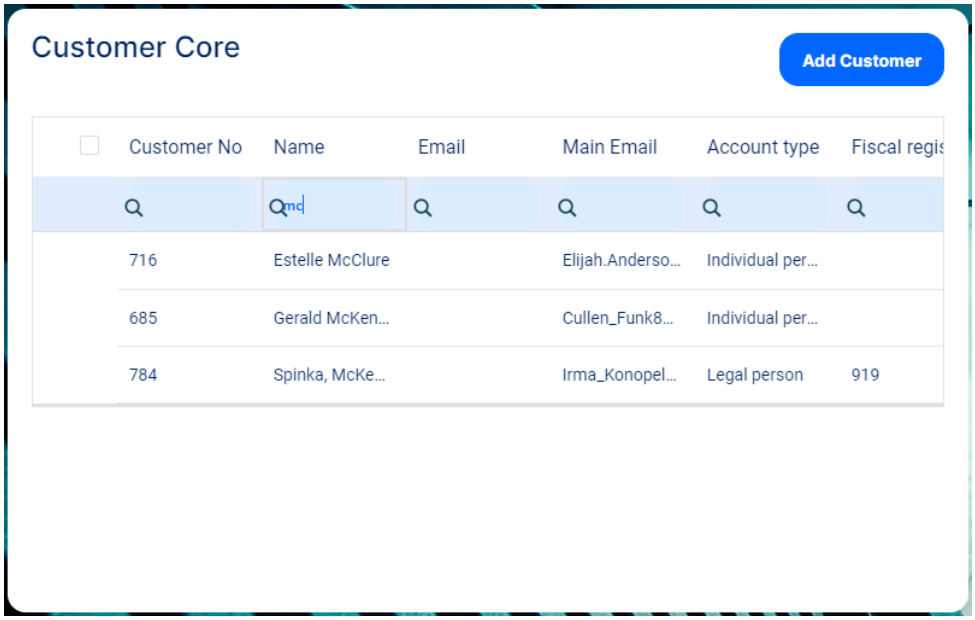


Double-click any of the bars to open a new browser tab displaying the list of contracts that fit the selected criteria.

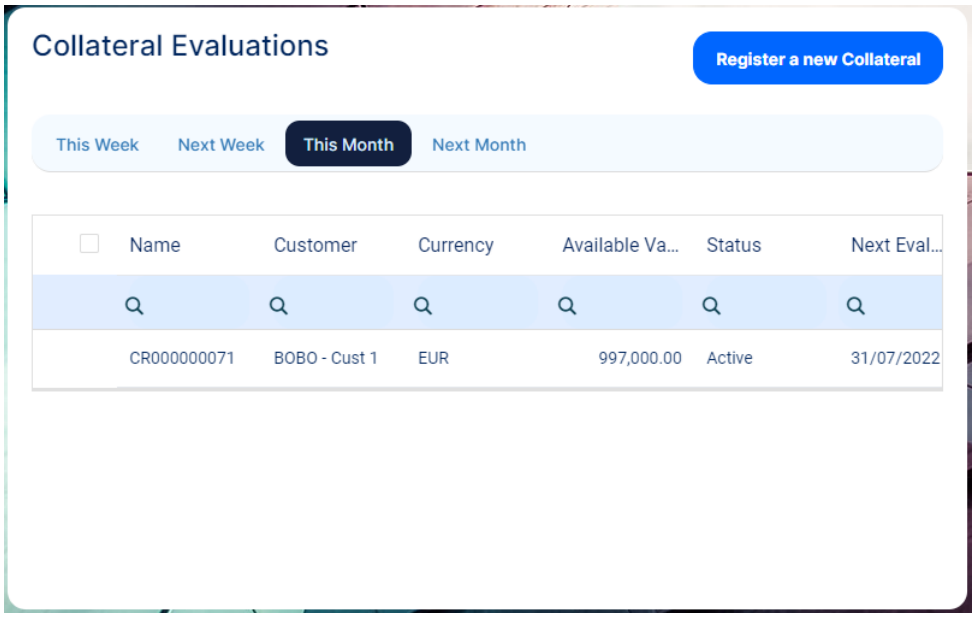
- **Last Updated Contracts** - It displays a list of the contracts updated in the system during the last x days, where x represents the configured default number of days. Click the **Add Contract** button to open the **Add Contract** page, where you can create new contracts.



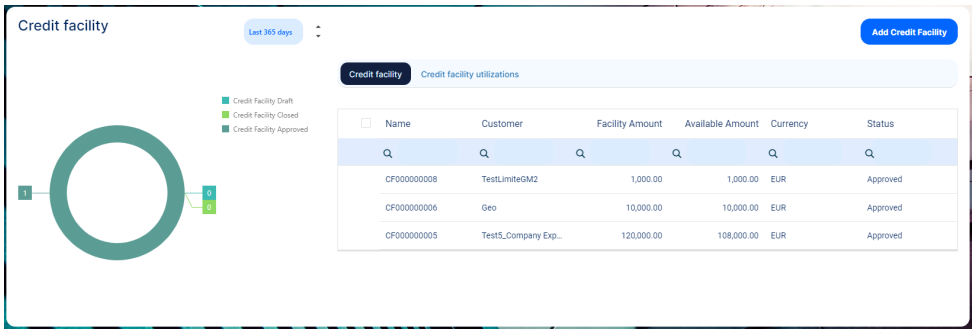
- **Customer Core** - It displays a list of the customers created in the system. Click the **Add Customer** button to open the **Add Customer** page, where you can create new customers.



- **Collateral Evaluations** - It displays a list of the collaterals with the next evaluation date within x days, where x represents the configured default number of days. You can also select to view the collaterals with the evaluations due this week, next week, this month, or next month. Click the **Register a new Collateral** button to open the **Add Collateral Register** page, where you can register new collaterals.

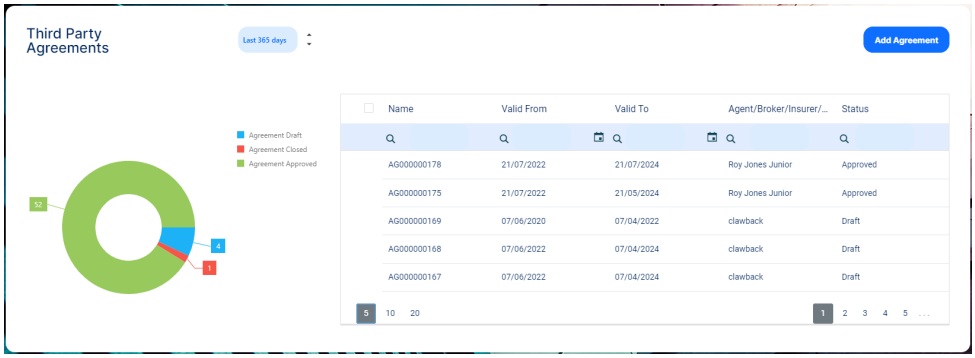


- **Credit Facility** - This section is displayed if the `UseCF` Loan Management system parameter is set to `True`. It displays lists of the credit facility and utilizations records created in the system in the last x days based on the credit facility date, where x represents the configured default number of days. It also displays a pie-chart that specifies the number of records in each status: `Credit Facility Draft`, `Credit Facility Closed`, and `Credit Facility Approved`. Click the **Add Credit Facility** button to open the **Create Credit Facility** page, where you can create new credit facilities.

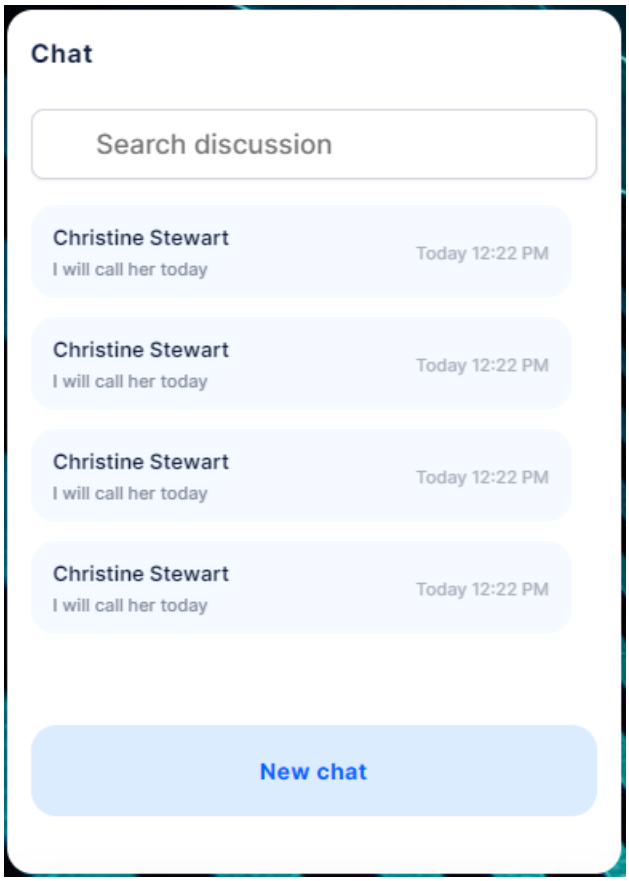


- **Third Party Agreements** - This section is displayed if the `UseTPM` Loan Management system parameter is set to `True`. It shows a list of the agreements defined within the system in last x days based on the

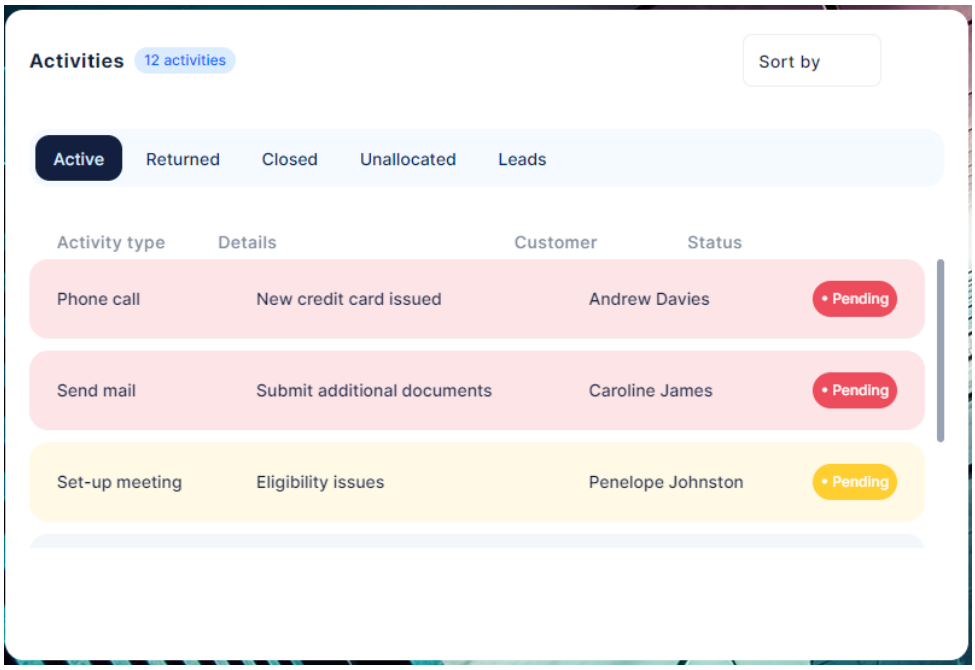
agreement date, where x represents the configured default number of days. It also displays a pie-chart that specifies the number of agreements in each status: `Draft`, `Closed` , and `Approved`. Click the **Add Agreement** button to open the **Creating Agreements for Third-Parties** page, where you can create new agreements.



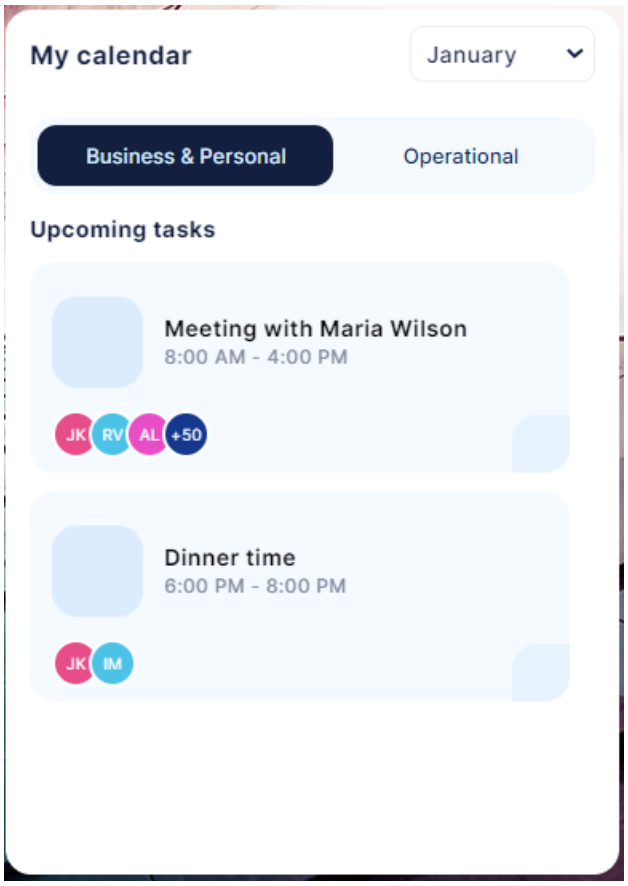
- **Chat** - The information displayed here is dependent on the Loan Management implementation. If the functionality was implemented, this section contains a chat communication channel with your colleagues. You can start a new discussion by clicking **New chat** or search in the chat for a person. You can search for chats with Customers or Team by clicking one of the buttons.



- **Activities** - The information displayed here is dependent on the Loan Management implementation. If the functionality was implemented, this section displays the activities assigned to you, filtered by active, returned, closed, unallocated, and leads.



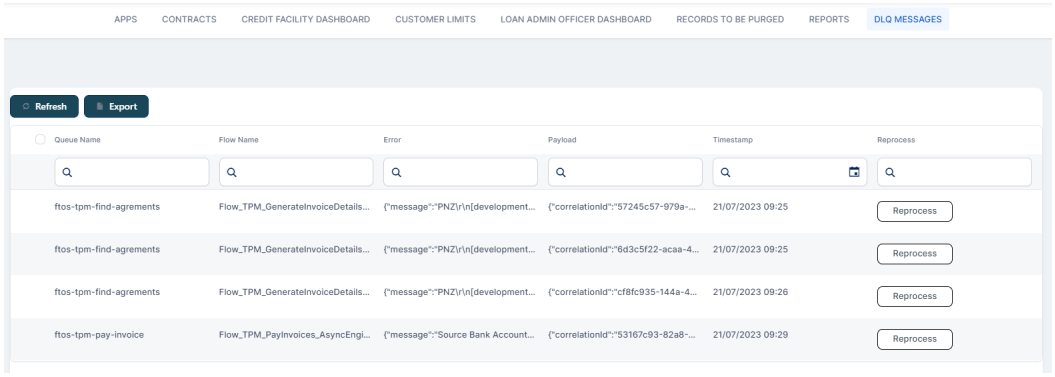
- **My calendar** - The information displayed here is dependent on the Loan Management implementation. If the functionality was implemented, the section displays the upcoming tasks filtered by month from the top right-hand corner of the screen. You can filter them by Operational or Business & Personal tasks, and you can add tasks.



DLQ Messages

The **DLQ Messages** (the dead letter queue) dashboard lists messages about the requests that the Async Engine could not process for any reason. Each message contains the queue name, the flow name, the error message, the payload and the time stamp of the unprocessed request. The lists can be filtered on every column.

The **Reprocess** button next to each message allows you to send them for reprocessing.



This dashboard can be accessed by users with the Async Engine Admin security role.