Buy Now, Pay Later 1.0.0

User Guide

Copyright © FintechOS 2022. All rights reserved.

TOC

Overview

Buy Now, Pay Later (BNPL) is a type of short-term financing that allows consumers to make purchases and pay for them at a future date, often interest-free. The Buy Now, Pay Later solution is a digital journey that allows customers to obtain a loan for consumer needs. The form driven flow helps the customer buy something they need/ want. The application process is tied to the checkout solution for the marketplace so that the experience is seamless and very fast compared to a more classic loan offering. You can apply for it online to pay for a product you wish to purchase. All the steps of acquiring the loan are digital and in real-time, including inserting the card details to pay for the product.

> **NOTE**
> The solution does not include a payment processing mechanism. it is subject to internal implementation.

The solution is also built to handle multiple types of scoring mechanisms, from a fully integrated user score based on the marketplace behavior to a more classical approach with Credit Bureau checks and Open Banking capabilities. The solution employs an Open Banking method for markets where financial institutions must perform income checks.

However, the journey can be modified as per the business requirements using our low-code Innovation Studio. The solution determines if customers are eligible for a solution before applying. The user can be both an **un-authenticated customer**, i.e., a new customer to the bank or **an existing customer**. The customer segment is retail.

The journey is accessible through APIs, therefore the accelerator integrates with any marketplace for a seamless experience, for example, e-commerce websites for electronic goods or clothing websites.

> **NOTE**
> The solution is both mobile-friendly and desktop appropriate.

Business Pain Points

The Buy Now, Pay Later digital journey by FintechOS is aimed at resolving some of the pain points when applying for a loan:

- the most common alternative, the credit card, usually needs the customer to already have a credit card, which entails they need to first apply for the credit card, then buy the product
- alternative options for this type of financing are usually expensive
- lack of payment options
- lack of integration with the e-commerce websites
- filling in multiple fields with personal details.

Advantages of Buy Now, Pay Later

Here are some of the major benefits the solution offers through FintechOS:

- Instant loan origination. You do not need to take any action as the origination is embedded into the buying process.
- Improved Experience. You are aided financially when buying a product.
- Time-efficient. The solution is quick to determine whether you are eligible for the Buy Now, Pay Later financial offering without having to go through an extensive application review process.
- Variety of banking products to apply for. You can choose to pay for your products in 12 instalments.

- Integrations. APIs calling endpoints to trigger integration with a core banking system and e-commerce websites.
- Self-service. You do not need to come in contact with a bank employee.

Solution Walkthrough

The Buy Now, Pay Later accelerator has a few stages to ensure fast delivery of a BNPL product:

- select a product from the marketplace

  FintechOS built a dummy e-commerce website to serve as a starting point, however, you can integrate the Buy Now, Pay Later accelerator to your website for your customers to acquire products using this type of loan. For details, see "Calling the Solution within a Journey" on page 44.

- select a Buy Now, Pay Later product present in "Configuring the Banking Products" on page 60

- sign into your account or create a new account

- insert your email address and phone number for validation with two One-Time-Passwords

- complete your account with additional identity information based on the risk level the platform calculates

- use Credit Kudos to prove your income and link an account

> **NOTE**
>
> This third-party platform is a recommendation from FintechOS. Credit Kudos enable businesses to leverage Open Banking to enhance affordability and risk assessments. It stores store the following data: account details (account name, number and sort-code, account balance, card number, account transactions (details of incoming and outgoing transactions, and contact details (address, telephone numbers and email

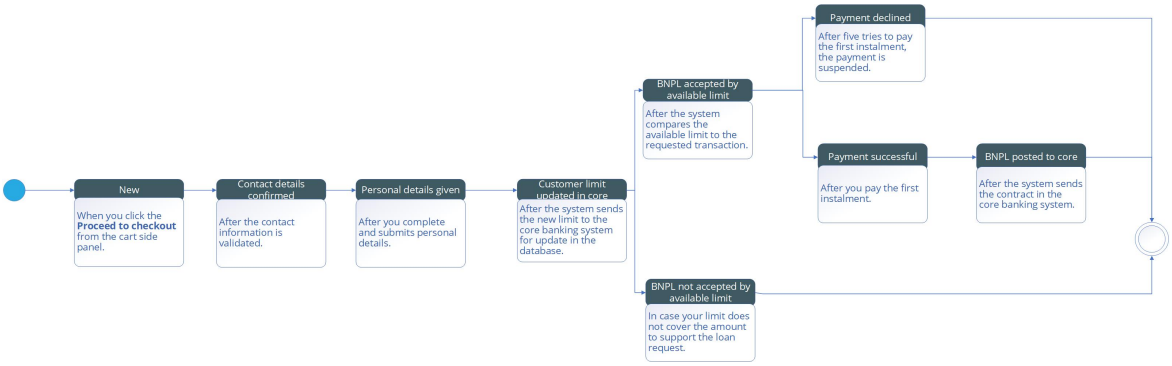
address as held by your bank/card issuer). However, any other third-party system with exposed APIs can be integrated.

- scan an ID document and take a selfie for security reasons using "OCR" on page 133 and "Processor Setting for Liveness" on page 133
- add your credit card details to pay for the first instalment or choose from a list of saved cards
- give feedback on your experience.

Below is a UML diagram showing the main actions that were taken to apply for Buy Now, Pay Later.

To download the diagram, click here.

Below is a diagram with the statuses of the solution.

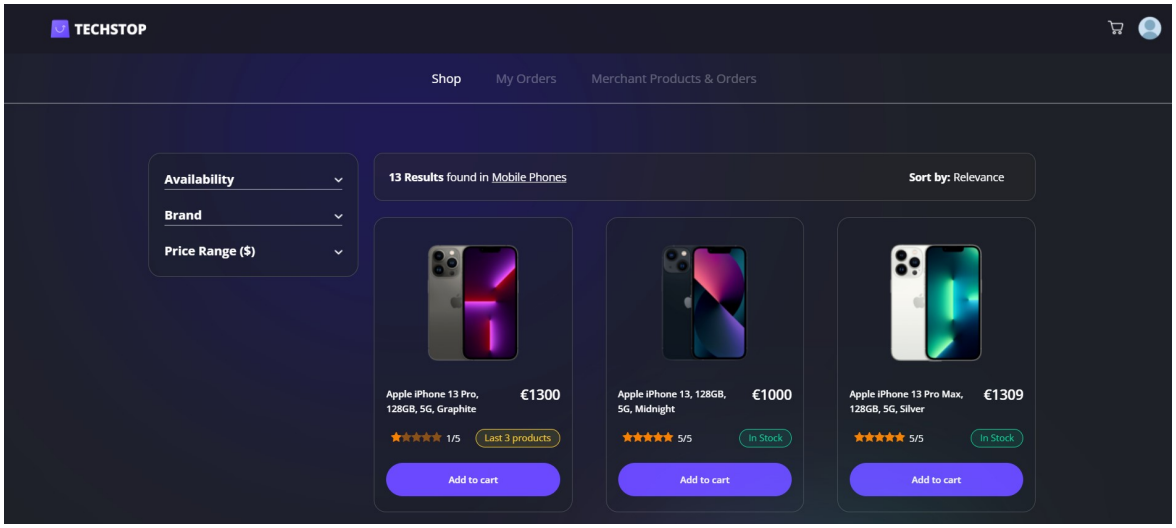


To download the diagram, click here.

Shop

This screen represents the e-commerce website named TechStop from where you can choose to buy a product, the starting point for the Buy Now, Pay Later flow. This is the website accessed by the customers looking to buy goods such as electronics or furniture, etc. For demonstration purposes, the solution sells cell phones. There is also the data model in place for the marketplace. Add to the chart one or more products for which you are going to apply for a loan.

IMPORTANT!
This dashboard is not part of the package. You must integrate your marketplace.



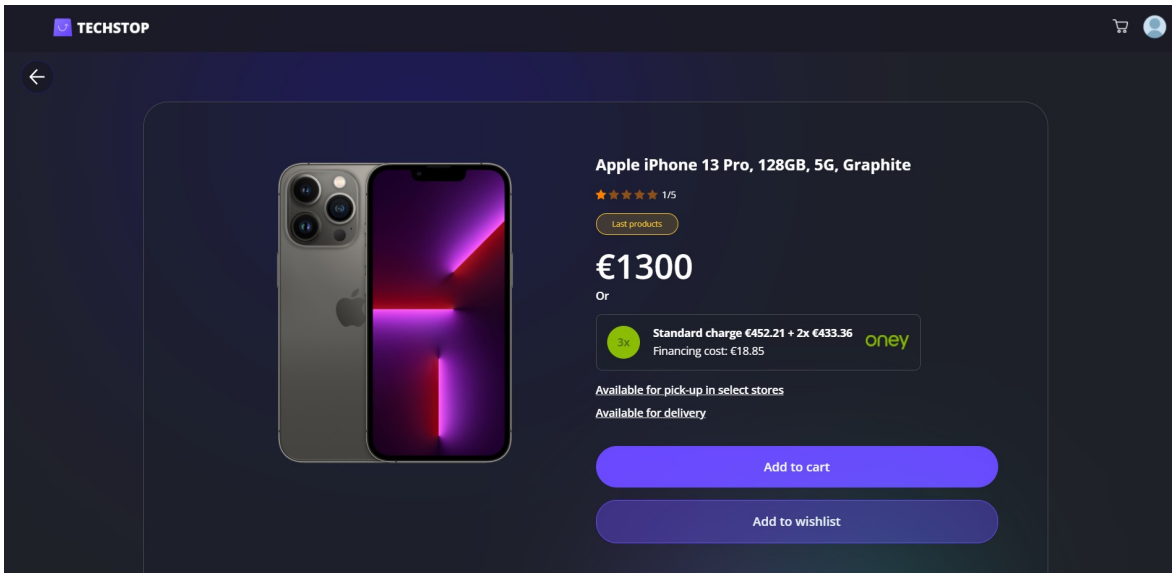
Each product has displayed:

NOTE
The products here are defined in Managing Products.

- a product picture
- a product name
- a product description
- the price and currency (to modify the currency, see "Defaults" on page 144)
- the product rating.

To add a product, click **Add to chart**.

To open the details about a product, click its image. The details screen depict:



- a product picture
- a product name
- the rating
- the stock
- a product description
- product price or the **monthly instalment**[1] per three months including:

  the number of payments: the first payment including the commission, it is paid by you from your money and two more payments that are financed by the Buy Now, Pay Later solution

  financing cost
- availability.

Underneath, there are two buttons:

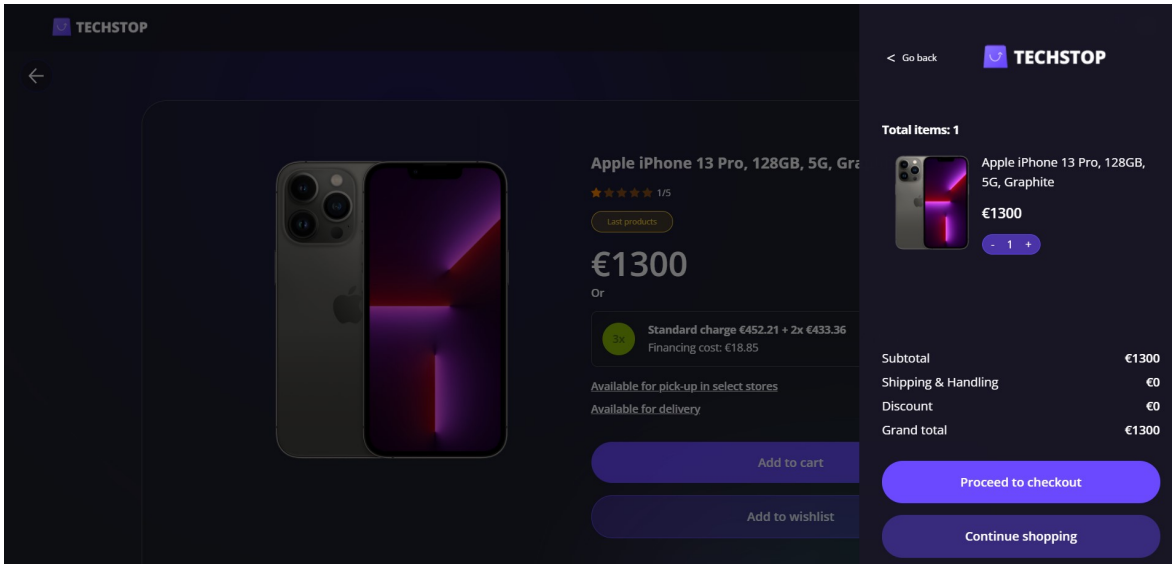
[1]The value of one of a number of successive payments in settlement of a debt.

- **Add to chart**. Click it to select the product.
- **Add to wishlist**.

> **NOTE**
>
> The button's functionality was not implemented for the purpose of this digital journey.

Checkout

The chart displays the product selected (a picture, a product description, a price, the number of products selected from that category with the options to increase/decrease the number) and the total sum to be paid. To delete an item from the chart, decrease the number to zero.



The cost is outlined:

- Subtotal: the price for the goods
- Shipping: free

- Discount: not applied
- Grand total: the sum of the subtotal, shipping and discount.

Click the **Proceed to checkout** button to continue to the next step. This button triggers the calculations of the formulas:

- "BNPL_KO" on page 95
- "BNPL_Scoring" on page 98
- "BNPL_Limit" on page 106
- "BNPL_RK_AvailableProducts" on page 107
- "BNPL_RK_RiskLevel" on page 108.

Click **Continue shopping** to return to the listings.

Payment

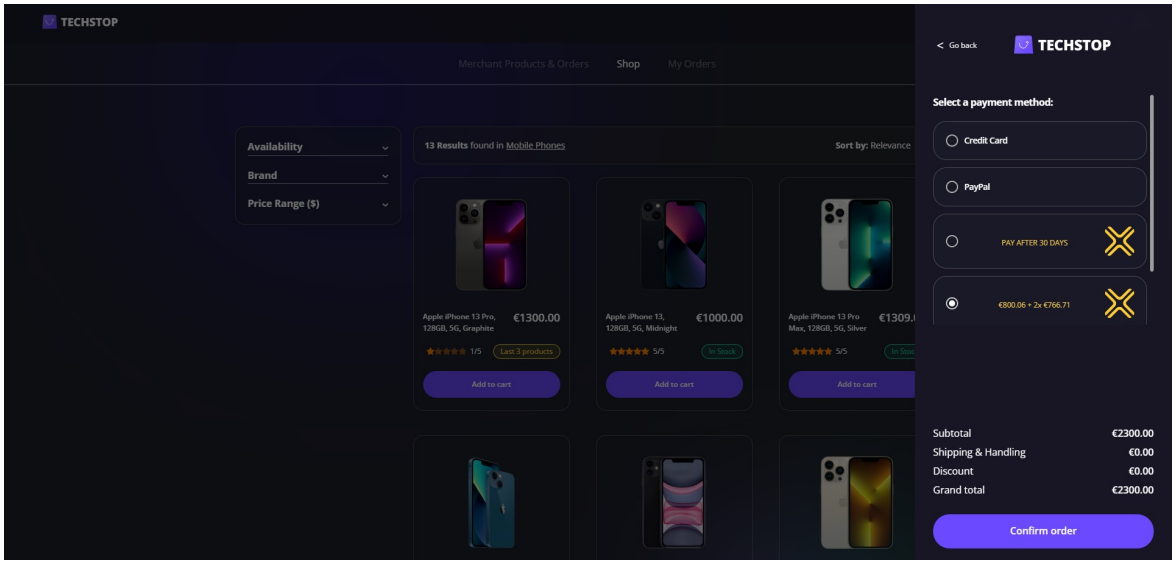
The system asks you to select a payment method from the list:

- Credit Card
- PayPal
- PAY AFTER 30 DAYS (if you choose this option, then the slider to choose the number of instalments is no longer editable since the full amount is to be payed after 30 days)
- the first instalment + the number of instalments times the value, i.e., €174.25 + 8x €145.45 (the first payment followed by eight instalments, each valued at €145.45).

  This option is selected by default.

  Use the slider to select the number of instalments from 3 to 12 instalments are available. Drag the slider from left to right to increase the number of instalment. The payment options refresh after each time you choose a number on this slider.

NOTE
The slider was created using "FTOS_EC_ShoppingCartLoader" on page 122.

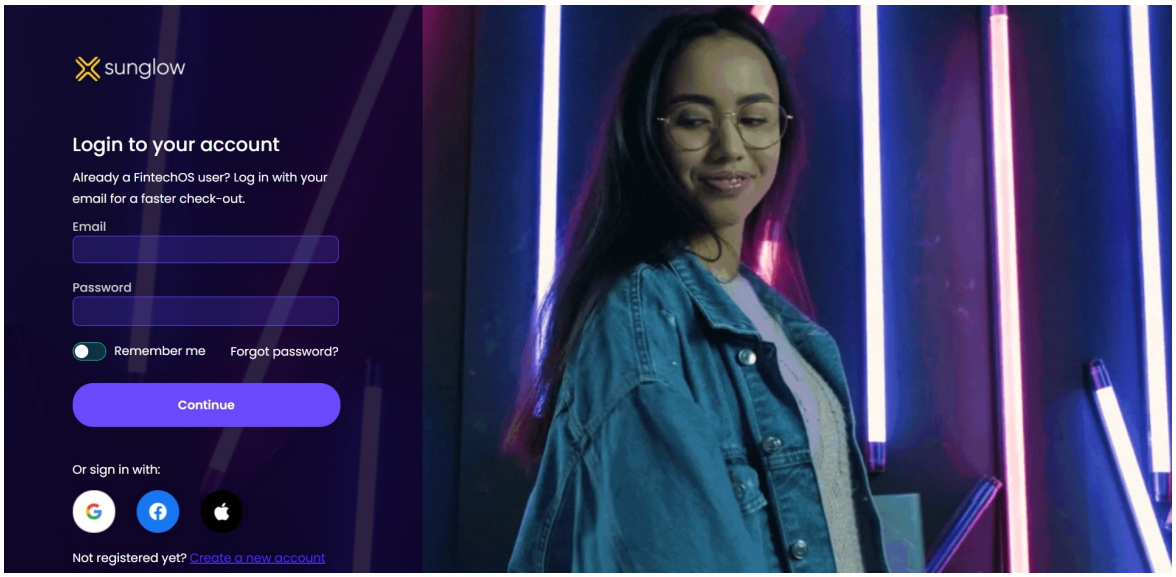


NOTE
For each of the BNPL loans above, i.e., each number of instalments, e.g., 3, 4, 10, have a product was defined. For details, see "Configuring the Banking Products" on page 60.

Click **Confirm order** to continue.

Sign In

You are directed to the login page where if you already have an account you can log in using your credentials or create a new account.



You can use to sign in:

- email and password

  The email must include @ symbol and must include a “.” after it, otherwise, it is not valid.

  > **IMPORTANT!**
  >
  > The email cannot be changed once inserted and it must be unique. The system checks the credentials. If they are not correct a toast message is displayed. The validation is done in the from driven flow FTOS_BNKAP_BNPLAutentificate > step Login > Advanced > After Events and in Before Section Save the script "FTOS_BNKAP_AuthenticateUser" on page 124 is called.

- Google account
- Facebook account

- Apple account.

> **NOTE**
>
> The integration with these three providers was not developed. It is subject to internal implementation.

For the users without an account, click the link: **Not registered yet? *Create a new account***. This directs you to the page "Contact Info" below.

To save the credentials, toggle the **Remember me** field.

If you cannot remember your password, click **Forgot password?** to reset it.

> **NOTE**
>
> These two functionalities to remember the credentials and reset the password are not functional.

After inserting your email and password, click **Continue**.

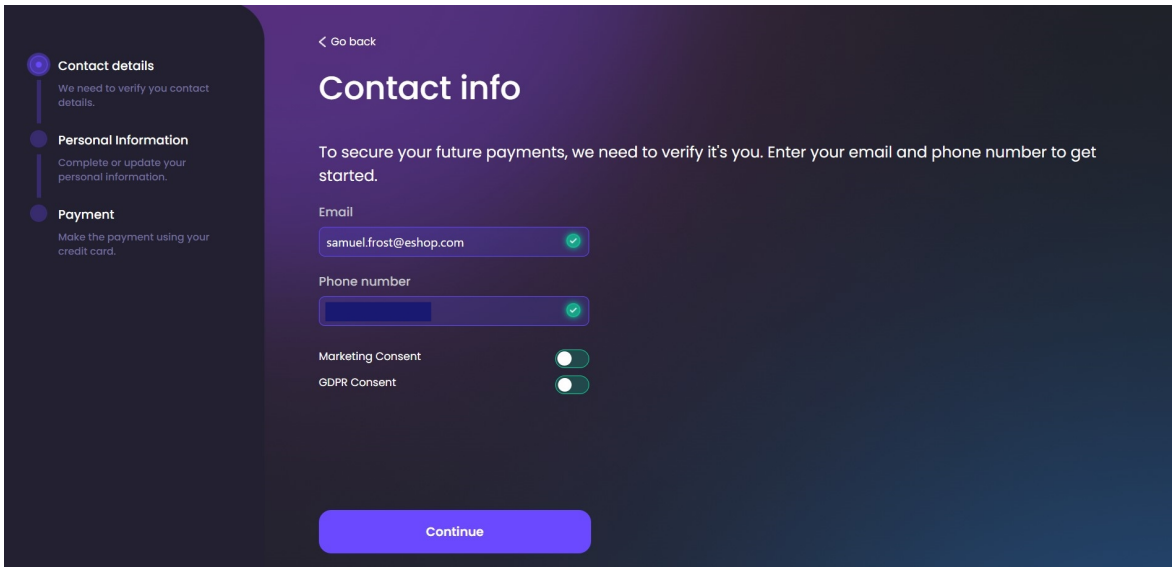
Contact Info

For security purposes, the email address and phone number must be inserted. They already exist in the system, but you can change them. The system also runs checks on the format: must include @ character and . after it. For the phone number, the field accepts only numbers. If the fields do not match the requirements, then they turn red and an exclamation point is displayed.

Field	Required	Data Type	Description
Email	Yes	Text	The email is populated with the value from the database, but you can edit it. The data that is already displayed is hard-coded in "FTOS_EC_ ShoppingCartLoader" on page 122.

Field	Required	Data Type	Description
Phone Number	Yes	Text	The phone number is populated with the value from the database, but you can edit it. The data that is already displayed is hard-coded in "FTOS_EC_ ShoppingCartLoader" on page 122.
Marketing Consent[1]	Yes	Bool	Tick the toggle, otherwise, you cannot proceed.
GDPR [2] Consent	Yes	Bool	Tick the toggle, otherwise, you cannot proceed.

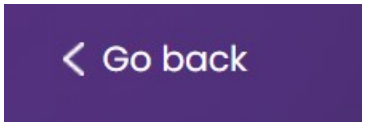
Click **Continue** to proceed.



> **NOTE**
> If the email already exists, a toast message is displayed. This was configured in the form driven flow FTOS_BNKAP_BNPLContactInfo > step ContactInfo > Advanced > Before Section Save.

[1]It is the practice of only contacting consumers that have given their prior express written consent to be contacted.
[2]The General Data Protection Regulation is a regulation in EU law on data protection and privacy in the European Union and the European Economic Area.

The journey contains a **Go back** button that allows the user to steer to the previous step, without losing any information already entered.



Contact Validation

For security purposes, the email address and phone number must be validated against the two One-Time-Passwords (OTPs) codes sent in the previous step. Two different codes are sent, one to the email address and one to the phone number. Both are needed to complete the validation process.

The validation process is timed. A timer is displayed on the screen with the default values of 60 seconds for the text message to the phone number, and 120 seconds for the OTP code sent to the email address. Both timers start after you access the page. You have five retries.

> **NOTE**
> For details on how the email and SMS validations were created, see "FTOS_DFP_BNPL_OTP" on page 144.

To change the email and phone number, click the Edit contact information or the back button to return to the previous screen.

The first code sent is to the phone number provided. You have 60 seconds to fill it in. After you write it in the field, click anywhere outside the box to validate it. The timer stops and displays a green checkmark for validated.

> **NOTE**
> The email and phone number are hidden. This was configured in the form driven flow FTOS_BNKAP_BNPLContactInfo > step ContactInfo > Advanced > After Section Save.

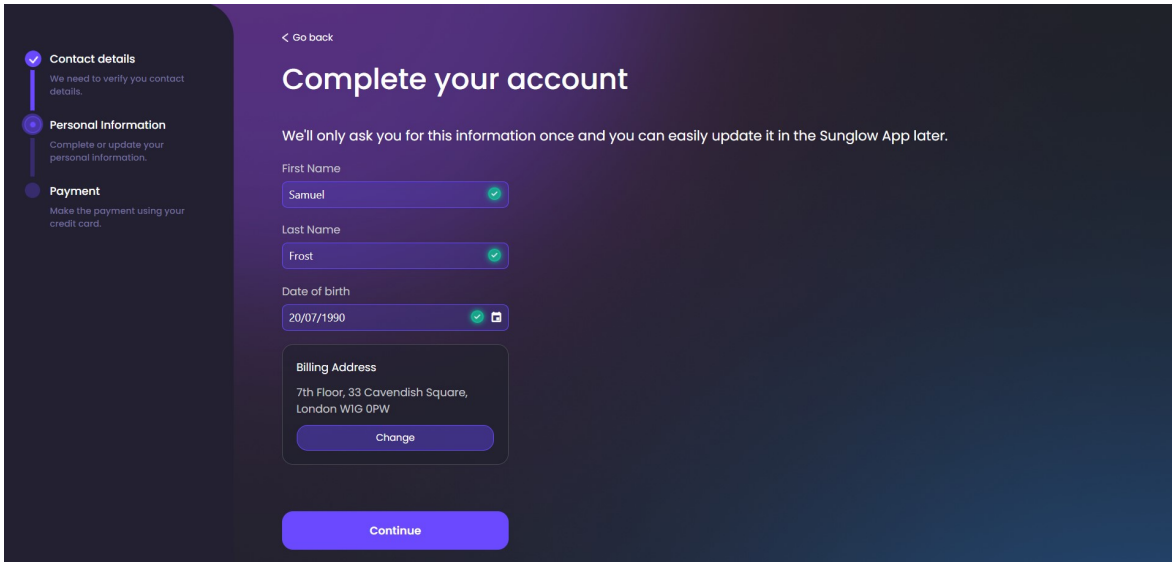
If the time expires, the button's label changes to **Resend**, and you can set the application to resend the codes on both the email address and the phone number. Afterward, the timer starts again. A toast message is displayed to inform you that a new SMS was sent.

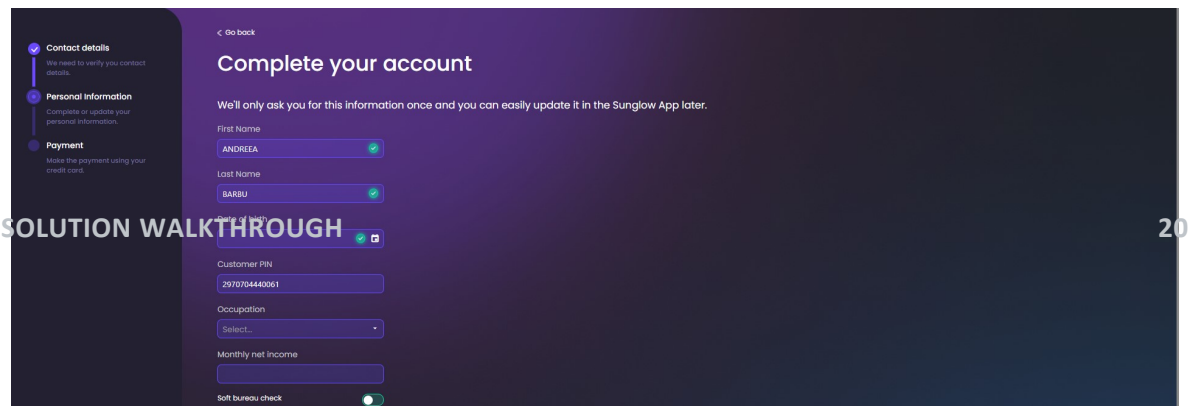
After the SMS code was certified, insert the code received on your email address. You have 120 seconds to do so. After the email code was inserted and validated, click anywhere outside the field to validate. If the wrong code has been inserted more than five times, you fail the validation and the flow ends. In this case, you must visit a branch to initiate the process.

To continue to the next step, the system automatically redirects you, if both codes were validated.

Complete Your Account

This stage displays the data from the marketplace to be used for the risk level. This data was hard-coded for this solution, however, once the Buy Now, Pay Later accelerator is integrated with the actual e-commerce website, then the accelerator gets this data from the e-commerce database.



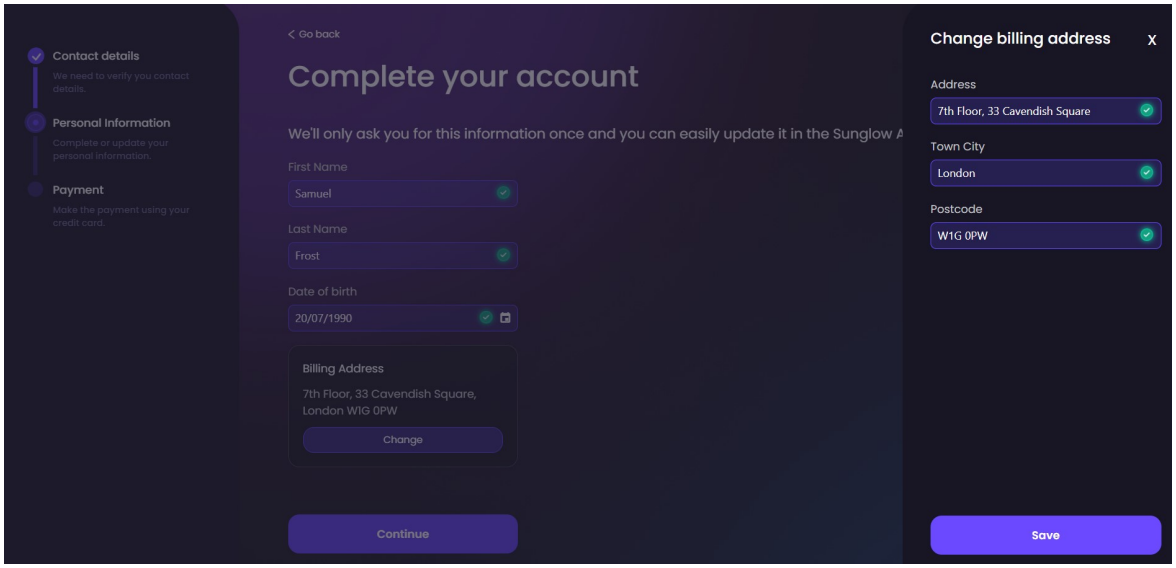


Field	Required	Data Type	Description	Displayed for risk level
First Name	Yes	Text	The data here is already displayed, but you can edit your first name. The data here comes from the marketplace.	All risk levels.
Last Name	Yes	Text	The data here is already displayed, but you can edit your surname. The data here comes from the marketplace.	All risk levels.
Date of birth	Yes	Date	The data here is already displayed, but you can edit your date of birth. The data here comes from the marketplace. **NOTE** The age is checked in this flow FTOS_ BNKAP_ BNPLConfirmDetails > ConfirmDetails step > Advanced > Before Section Save.	All risk levels.
Password	Yes	Text	Insert a password.	1

Field	Required	Data Type	Description	Displayed for risk level
Confirm Password	Yes	Text	Repeat the password. They must match for you to continue the flow. **NOTE** The passwords are matched in this flow FTOS_ BNKAP_ BNPLConfirmDetails > ConfirmDetails step > Advanced > Before Section Save.	1
Customer PIN	Yes	Text	Insert your Personal Identification Number. The system allows any types of PINs regardless of the issue country.	2
Occupation	Yes	Option set	Select from the drop-down list the job status: • employed • retired • inactive • freelancer.	2
Monthly net income	Yes	Money	Insert the amount of money you receive during one month.	2
Soft bureau check	Yes	Bool	Tick the checkbox if you agree to have the system run a soft credit bureau check on you.	3
Credit score interrogation	Yes	Bool	Tick the checkbox if you agree to have the system run a credit score check on you.	3
Billing Address	Yes	Text	The data here is already displayed, but you can edit your address.	All risk levels.

NOTE
The fields here are dynamically displayed, depending on the outcome from the calculations done within the risk level as explained in "Data set BNPL_RK_RiskLevel" on page 109. To display/hide the fields, access the Form driven flow FTOS_BNKAP_BNPLConfirmDetails > ConfirmDetails step > Advanced > After Events. Every user from risk level 2 that passes the eligibility step needs to have available **DTI**[1] for this loan.

Click **Change** to modify your address.



Edit the fields:

Field	Data Type	Description
Address	Text	Edit the full address. The data here comes from the marketplace.
Town City	Text	Edit the city. The data here comes from the marketplace.
Postcode	Text	Edit the postal code. The data here comes from the marketplace.

Click **Save** if you are done editing or click **X** to abandon the flow.

[1]The Debt-to-Income ratio (DTI) is a personal indicator of a good balance between debt and income detremined by essential expenditure/ income with bank transfers.

NOTE
The email address is checked in this flow FTOS_BNKAP_BNPLConfirmDetails > ConfirmDetails step > Advanced > Before Section Save > the script "FTOS_BNPL_VerifySistemUserEmail" on page 132.

Click **Continue** after you are done. Several actions are triggered:

- calculate the limit
- update the customer limit in the core banking system
- fetch the newly available limit from the core banking system after the update is done
- an email is sent with the account confirmation
- create a new system user as configured in FTOS_BNKAP_BNPLConfirmDetails > ConfirmDetails step > Advanced > After Section Save > "FTOS_BNPL_CreateSystemUser" on page 127.

Scoring Done by Bank

If the scoring is done by the bank and it is not calculated after the **Proceed to checkout**, then this screen displays multiple fields:

Field	Required	Data Type	Description
First Name	Yes	Text	The data here is already displayed, but you can edit your first name. The data here comes from the marketplace.
Last Name	Yes	Text	The data here is already displayed, but you can edit your surname. The data here comes from the marketplace.
Date of birth	Yes	Date	The data here is already displayed, but you can edit your date of birth. The data here comes from the marketplace.
Customer PIN	Yes	Text	Insert your Personal Identification Number. The system allows any types of PINs regardless of the issue country.

Field	Required	Data Type	Description
Occupation	Yes	Option set	Select from the drop-down list the job status: • retired • student • unemployed • full-time employed • self-employed • management • mandate contract • housewife.
Monthly net income	Yes	Numeric	Insert the amount of money you receive during one month.
Marital Status	Yes	Option set	Select one from the list: • Married • Civil Partnership • Single • Widow • Separated.
Employer name	Yes	Text	Insert the name of the company that employs you.
Date of employment	Yes	Date	The date you started working at your current job.

Field	Required	Data Type	Description
Education	Yes	Option set	Select one from the list: • High school • University • Middle school • Elementary school.
Billing Address	Yes	Text	The data here is already displayed, but you can edit your address.

NOTE
The validation on this page are configuring in the from driven flow FTOS_BNKAP_BNPLCompleteAccountScoring > step AccountScoring > Advanced > After Events.

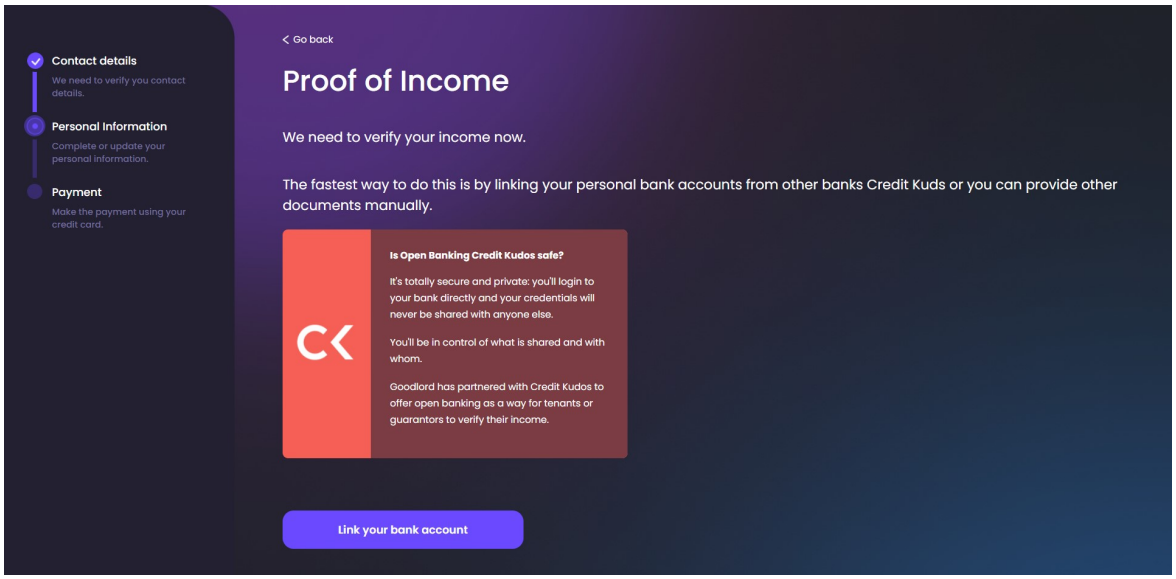
Click **Continue**. This button triggers the calculation of:

- "BNPL_RK_AvailableProducts" on page 107
- "BNPL_KO_Risk" on page 113
- "BNPL_Scoring_Risk" on page 116
- "BNPL_MaxDTI_Risk" on page 114
- "BNPL_FinancialCalculation" on page 112.

Proof of Income

This step verifies the income you earn. This validation is needed by the financial institution because any loan requires a verification of the source of income. This is done using the **Open Banking**[1] integration with **Credit Kudos**[2]. This is a suggestion FintechOS has. However, any third-party platform can be integrated.

> **IMPORTANT!**
> Every customer must have at least one account linked.



Click **Link Your Bank Account** to start the process.

FintechOS fetches the following data from Credit Kudos:

Report Details

---

[1]Open banking connects banks, third-parties and technical providers to securely exchange data.

[2]Credit Kudos' intelligent products enable businesses to leverage Open Banking to enhance affordability and risk assessments. Their products help lenders streamline underwriting, improve accuracy in decision-making, and support customers after acquisition through our engagement tools.

Predicted monthly income with transfers `incomeWithBankTransfers`

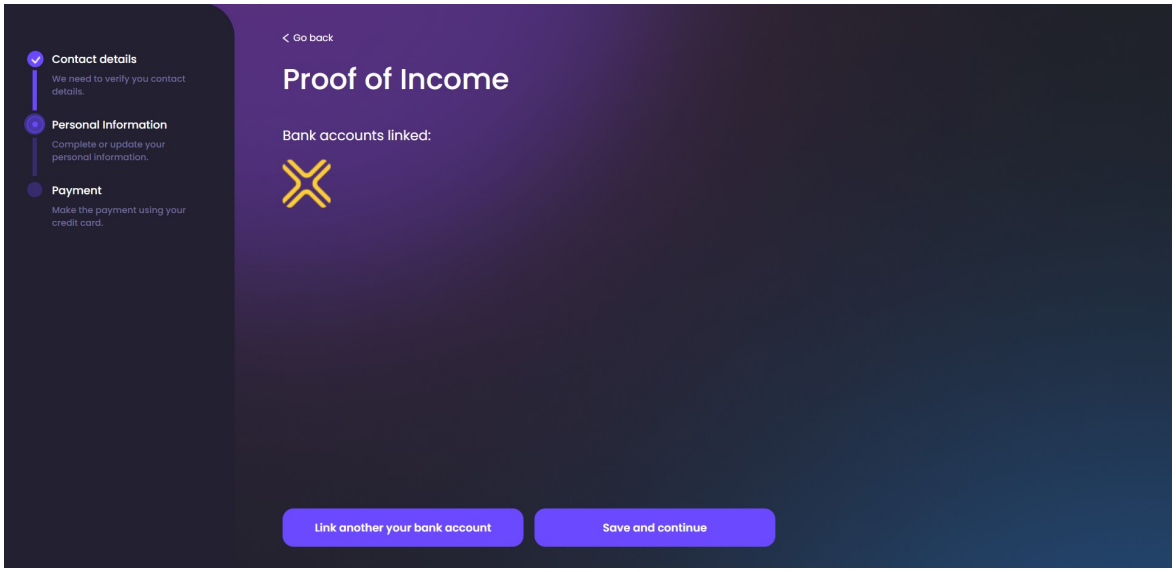
Predicted monthly essential expenditure `essentialExpenditure`:

- Value
- Currency.

They are used to determine the **DTI**[1].

Account linked

After the process within the Credit Kudos platform is over, you are directed to this step with two options:



- **Add another account**: click it to launch the Credit Kudos solution again to add a secondary account

  or

[1]The Debt-to-Income ratio (DTI) is a personal indicator of a good balance between debt and income detremined by essential expenditure/ income with bank transfers.

- **Save and continue**: click it to open the next page and trigger the formula "BNPL_ FinancialCalculation" on page 112 and the new limit is stored within the core banking system.

> **NOTE**
> The income and expenses are hard-coded in the form driven flow FTOS_BNKAP_ BNPLOpenBanking > step BankAccounts >After Events to income = 15000; expense = 100.

Proof of ID

The **Proof of ID** screen allows you to provide personal identification documents to prove your identity. This step followed by the Liveness check are displayed only to risk level 3 customers. A card or any bank product cannot be issued without verifying the identity of the applicant. The following ways of providing identification proof are available. Select one by clicking on the corresponding field:

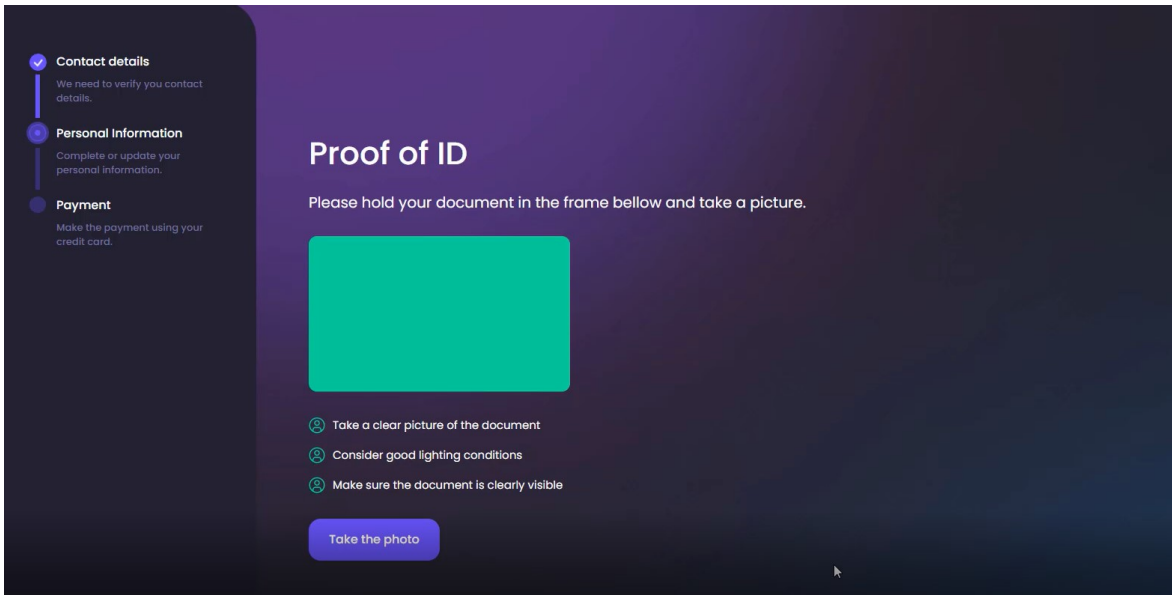
- **Driving License**: requires you to take a photo of your driver's license
- **Passport**: requires to scan your passport
- **Personal ID**: requires you to take a photo of your ID.

It is mandatory to choose at least one way of providing identification information, otherwise, the button is not available. You have five retries to take the photo, otherwise, the step is failed.

> **NOTE**
> The three methods are defined using "Configuring the Flow Settings" on page 132 and they are called in the form driven flow FTOS_BNKAP_BNPLScanOCR > step selectDocumentType> After Events.

Take a photo of your driver's license

This option allows you to open the camera and take a photo of the driver's license. Select the option and click **Continue** to proceed.



Place the license inside the frame and click **Take the photo**. Click **Process file**. If the photo is not successful, click **Retake photo** to reinitiate the process.

Take a photo of my ID

This option allows you to open the camera and take a photo of the front of the ID card. The photo is taken by our automation processor Computer Vision, which gathers the data from the document and saves the photo of the customer's face. This photo is later compared with the selfie to match the appearance of the customer. Allow your browser to take photos when the pop-up appears.

If the first attempt to take the photo is unsuccessful, then you have five more tries. Once the final attempt has failed, the flow ends. This number of retries can be modified within "Configuring the Flow Settings" on page 132.

Click **Continue** to proceed.

Passport

If you choose the passport option, click **Continue** to proceed. This alternative is a proposal to scan the ID card, however, it was not developed within this solution.

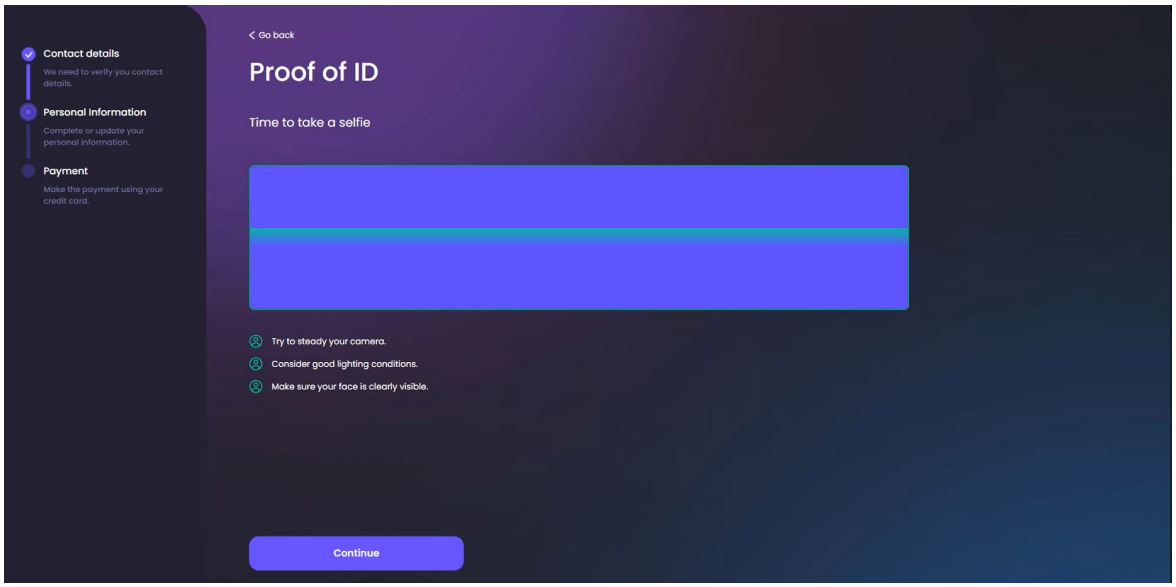
If any of those methods fail, the system informs you that it could not retrieve the information.

Click the back button to return to the **Proof of ID** page and try again.

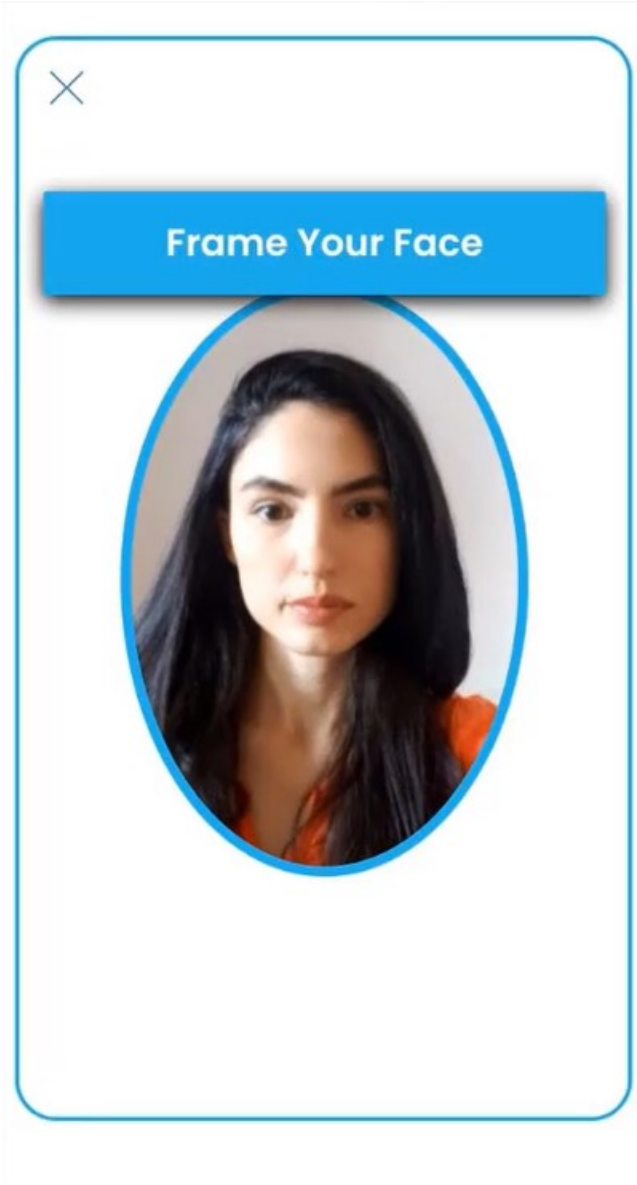
Liveness Check

The **Proof of ID** screen allows you to confirm your identity by taking a selfie. However, you need to follow a few conditions:

- Try to steady your camera.
- Consider good lighting conditions.
- Make sure your face is clearly visible.



Click **Continue** to proceed.



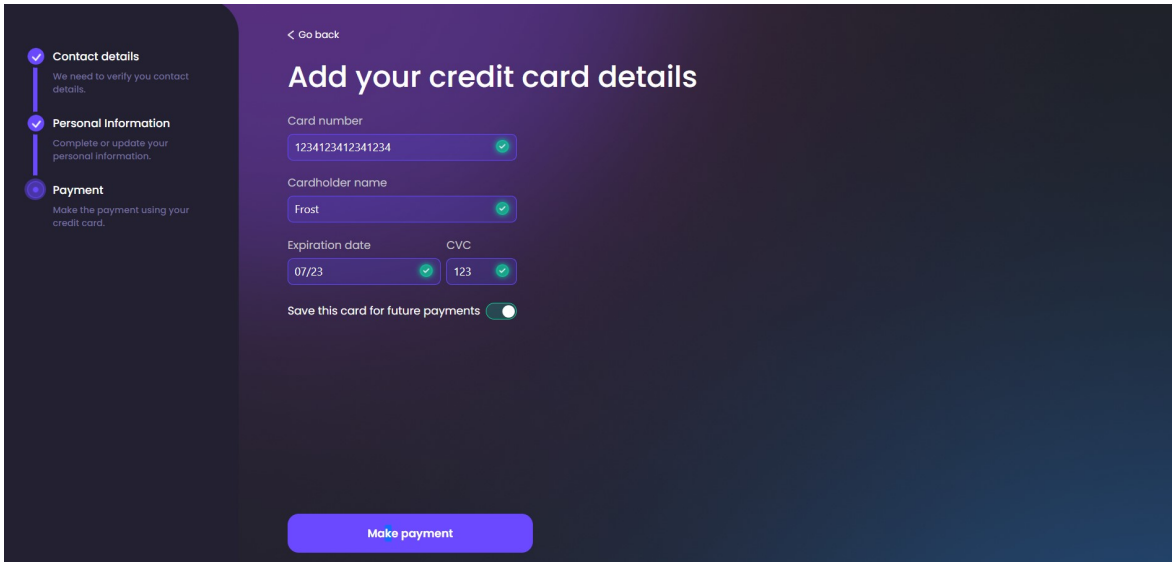
This option allows you to open the camera and take a photo. The photo is taken by our automation processor Face Recognition with Liveness. This photo of the ID is later compared with this selfie, to match the appearance of the customer. If the first attempt to take the selfie is unsuccessful, then you have five more tries. Once the final attempt has failed, the flow ends.

NOTE
This number of retries can be modified within "Processor Setting for Liveness" on page 146.

Add Your Credit Card Details

This payment screen is available only for customers eligible for the loan. You have to pay for the first installment using a card. There are two possible screens here depending whether you already have a card saved on our platform. If you do not have a card the **Add Your Credit Card Details** screen is displayed. If you do have a card saved in the system, the "Credit Card" on the next page page is displayed.

On this screen, you have to insert the card. The maximum number of days to pay for the goods is 30 days. Insert the details:



Field	Mandatory	Data Type	Description
Cardholder name	Yes	Text	Insert the name on the card. It must contain 16 digits.
Card number	Yes	Number	Insert the 16 digits.
Expiration date	Yes	Text	Insert the month and year, e.g., 07/23.
CVC[1]	Yes	Number	Insert the three digits.

[1]Card Verification Code. It is located on the back of your credit/debit card.

Field	Mandatory	Data Type	Description
Save this card for future payments	No	Bool	Toggle this field if you wish to save the card details. **NOTE** This is achieved in the form driven flow "FTOS_BARET_BNPLPayment" on page 41 > Advanced > Before Events and in > step Payment > Advanced > After Events > the script "FTOS_BNPL_SaveCard" on page 129.

Click **Make payment**. To pay in the next 30 days, a transaction with the value of 0 is done to validate the card. After the first payment is received, the loan amount is approved and sent to the core banking system for disbursement. If the payment is not successful, you have four more tries to make the payment. After all the attempts are used, an error is displayed.

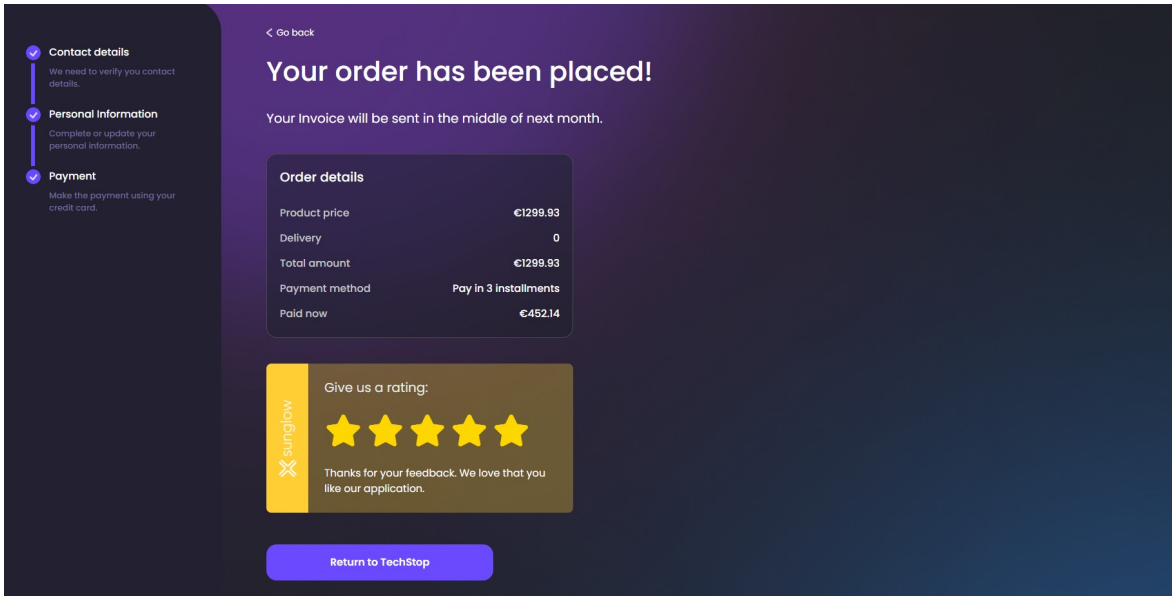
Credit Card

This page is displayed to the users who already have card(s) saved into the platform. The page loads with the existing cards, click the card you wish to use and then click **Make payment**. If you wish to pay using a different card, click **Use a new card** and the page described above is displayed.

NOTE
The saved cards are brought by the script "FTOS_BNPL_GetSavedCards" on page 128 from the form driven flow "FTOS_BARET_BNPLPayment" on page 41 > step Payment > Advanced > After Events.

Your Order Has Been Placed

The confirmation screen is displayed to re-inform you of the order and to ask for feedback on the experience.



The **Order details** are depicted to you:

- Product price
- Delivery
- Total amount (including the commissions)
- Payment method
- Pay Now (the value of the next instalment). Its value is 0 if the product is *Pay in 30 days*. For split payment, the total amount paid in the previous step as first payment includes commission.

For the customers who opted for **Pay in 30 days**, the option is available.

For the customers who opted for a split payment, *Pay in X installments* is displayed to them.

Feedback

The solution prompted you to state your opinion about the journey.

> **NOTE**
> The script "FTOS_BARET_AddFeedback" on page 132 is called in the form driven flow FTOS_BARET_BNPLOrderConfirm > step ConfirmOrder > Advanced > After Events.

Click on the number of stars to give a rating from 0.5 to 5 stars. The default value is five.

Click **Return to TechStop** to return to the marketplace. This button triggers the script "FTOS_BNPL_GoToShop" on page 128.

Installation

This version is compatible with High Productivity Fintech Infrastructure v22.1. The supported operating system is Microsoft Windows 10. The supported browser is Google Chrome.

Prerequisites

Download the package from the Marketplace.

Install and configure:

- Standard FTOS infrastructure (with B2CPortal, B2CProxy, JobServer, MessageBus, MessageComposer for the "Contact Validation" on page 19) v22.1 as explained here
- Banking Product Factory v3.0.2
- OTP project (found in the folder Buy Now, Pay Later-v1.0.0.zip > prereq)
- B2C Setup project (found in the folder Buy Now, Pay Later-v1.0.0.zip > prereq) OR a manually configured B2C security role assigned to the Guest user and a front-end domain named B2C.
- Optionally, consider the Credit Kudos and Core Banking integrations, which are subject to internal implementation. The Core Banking integration is needed because the Buy Now, Pay Later solution searches for a customer number with an associated bank account to it. Consider creating a customer with a filled in customer number and a bank account.

After the FintechOS environment is installed, proceed with the next steps.

Package

Once you downloaded the package Buy Now, Pay Later-1.0.0.zip, unzip the package. It contains the following set-up:

- prereq
- solution (the solution with the templates for the B2C Portal, containing the digital assets)
- data-model
- additional-files (the library with custom controls; the custom files and the report document templates)
- default-products (the banking products)
- reset (it contains SQL procedures to delete records from the database)
- an installation guide.

1 Install the Packages

1. Import the packages using Digital Solutions Packages found within the `solution` folder.
2. Add the latest syspackdeployer in the reset folder, edit the installer and run it in admin mode.
3. Add the latest syspackdeployer in the data-model folder, edit the installer and run it in admin mode.

4. Add the latest syspackdeployer in the solution folder, edit the installer and run it in admin mode.

2 Set up the Additional Files

Copy or upload the custom folder in the B2CPortal folder.

Copy or upload the content of the UploadEBS in your UploadEBS folder.

Core Banking Integration

For the Core Banking integration, set the value to 1 (true). If you don't want to integrate the core banking system, set the key CB-use-integration to 0 (false) and you can leave the other keys with the default values.

```
<add key="B2CProxy" value="LinkToProxy" />
<add key="CB-use-integration" value="1" />
<add key="CB-URL" value="core-banking-api-baseurl" />
<add key="CB-user" value="core-banking-api-username" />
<add key="CB-pass" value="core-banking-api-password" />
<add key="CB-bank-code" value="core-banking-api-defaultBank" />
```

Add the keys needed for the Cognitive Processors and SMTP in the web.config of the B2CPortal.

Add the keysneeded for the Cognitive Processors in the serviceSettings of the Jobserver.

Add the proxy routes in proxy config file.

Restart the application pools/services for B2CPortal and B2CProxy.

3 Install the Default Banking Products

Add the latest syspackdeployer in the default-products folder, edit the installer and run it in admin mode.

Configurations

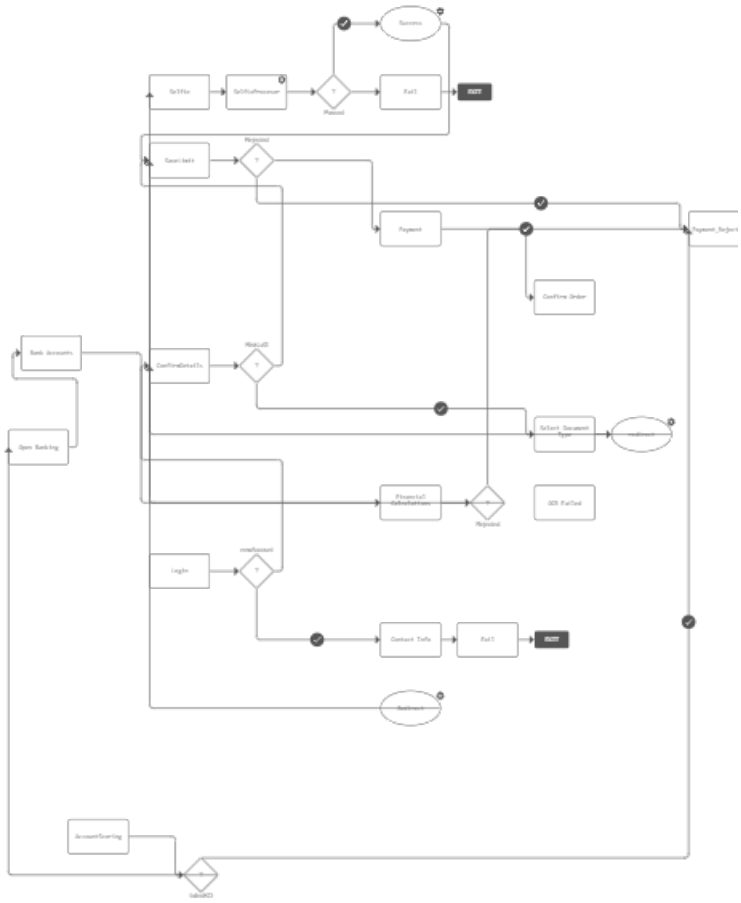
The following chapter of this guide explains the automation processors used to build the products, as well as technical details of how automation scripts aid the validation process or the process of comparing, returning, and storing data. Automation is a powerful way to make business processes more time-efficient. Automation processors can help achieve exactly that. They can be configured to be consumed by any digital journey using the Innovation Studio.

The Buy Now, Pay Later digital journey is comprised of several automation processors that ensure a smooth process of applying for a loan. Several banking products were configured to offer the customer a real product. This section of the user guide unveils information about which code snippet was used where, and what it enables in the journey. To create a personalized Buy Now, Pay Later digital journey to fit your requirements and offerings, more automation processors and other features are available in the Innovation Studio to build a customer-centric digital flow.

To access the digital journey, access Innovation Studio > Digital Experience > Digital Journeys > Digital Journeys > **BNPL**:

Type	Name	Description
Form Driven Flow	FTOS_BARET_BNPLOrderConfirm	"Your Order Has Been Placed" on page 35
Form Driven Flow	FTOS_BARET_BNPLPayment	"Add Your Credit Card Details" on page 33
Form Driven Flow	FTOS_BARET_BNPLPayment_ Rejected	Loan Rejected
Form Driven Flow	FTOS_BARET_BNPLSelfie	"Liveness Check" on page 31
Form Driven Flow	FTOS_BNKAP_BNPLAutentificate	"Sign In" on page 15

Type	Name	Description
Form Driven Flow	FTOS_BNKAP_ BNPLCompleteAccountScoring	"Scoring Done by Bank" on page 24 The solution has two options: • get the scoring of the customer from the marketplace • the bank calculates the scoring within this solution by inserting this screen.
Form Driven Flow	FTOS_BNKAP_ BNPLConfirmDetails	"Complete Your Account" on page 20 In this flow > Advanced > Before Events, the "Defaults" on page 144 flow settings are called.
Form Driven Flow	FTOS_BNKAP_BNPLContactInfo	"Contact Info" on page 17
Form Driven Flow	FTOS_BNKAP_BNPLOpenBanking	"Proof of Income" on page 27
Form Driven Flow	FTOS_BNKAP_BNPLSaveLimit	After the "Proof of ID" on page 29 is a success, it calls the script "FTOS_BNPL_ SaveCustomerLimit" on page 130 in After Events and gets the `decision`.
Form Driven Flow	FTOS_BNKAP_BNPLScanOCR	"Proof of ID" on page 29
Form Driven Flow	FTOS_BNPL_FinancialCalculations	After FTOS_BNKAP_BNPLOpenBanking flow, it calls the script "FTOS_BNPL_ RunFinancialCalculation" on page 128.



Consult the following pages to read more on the internal set-up of the solution:

Calling the Solution within a Journey

This journey is meant to be integrated with an existing e-commerce website, therefore, you need to call it. APIs are used to achieve this integration. We recommend using Postman to call all the endpoints from this solution in the following order:

1. FTOS_EC_GetUserShoppingCart
2. FTOS_EC_UpdateCartItemQuantity
3. FTOS_BNPL_CalcScoringAndLimit
4. FTOS_BNPL_GetAvailableProducts
5. FTOS_BNPL_SimulateOfferSummary
6. FTOS_BNPL_GetRiskLevel
7. FTOS_EC_CreateOrder
8. FTOS_BNPL_GetMerchantBankAccount
9. FTOS_BNPL_AddBNPLApplication
10. FTOS_BNPL_UpdateBNPLApplication
11. FTOS_BNPL_GetBNPLApplicationLink.

FTOS_BNPL_SimulateOfferSummary

request

```
"request": {
                "method": "GET",
                "header": [],
                "url": {
                    "raw": "{{host.schema}}://
{{host.name}}/api/Authorize/GetToken?client_
id=postman&username={{auth.username}}&password=
{{auth.password}}&response_type=token",
                    "protocol": "{{host.schema}}",
                    "host": [
                        "{{host.name}}"
                    ],
                    "path": [
                        "api",
                        "Authorize",
                        "GetToken"
                    ],
                    "query": [
                        {
                            "key": "client_id",
                            "value": "postman"
                        },
                        {
                            "key": "username",
                            "value": "{{auth.username}}"
                        },
                        {
                            "key": "password",
                            "value": "{{auth.password}}"
                        },
                        {
                            "key": "response_type",
                            "value": "token"
                        }
                    ]
                }
            },
```

response

```
    "response": []
        },
```

```
        {
            "name": "SimulateOfferSummary",
            "request": {
                "method": "POST",
                "header": [],
                "body": {
                    "mode": "raw",
                    "raw": "{\r\n\t\"ApiInfo\":
{\r\n        \"UserName\":\"
{{auth.username}}\",\r\n        \"Token\":\"
{{ftos.auth.accessToken}}\"\r\n    },\r\n    \"Request\":
{\r\n\t\t\"ActionName\":\"FTOS_BNPL_
SimulateOfferSummary\",\r\n        \"Data\": \"
{\\\"productAmount\\\":1000,\\\"bankingProductCode\\\":
\\\"BNPL3X\\\" }\"\r\n    }\r\n    \r\n}",
                    "options": {
                        "raw": {
                            "language": "json"
                        }
                    }
                },
                "url": {
                    "raw": "{{host.schema}}://
{{host.name}}/api/OpenApi/CallAction",
                    "protocol": "{{host.schema}}",
                    "host": [
                        "{{host.name}}"
                    ],
                    "path": [
                        "api",
                        "OpenApi",
                        "CallAction"
                    ]
                }
            },
```

FTOS_BNPL_CalcScoringAndLimit

```
"response": []
        },
        {
            "name": "CalcScoringAndLimit",
            "request": {
                "method": "POST",
```

```
                "header": [],
                "body": {
                    "mode": "raw",
```

```
                    "raw": "{\r\n    \"Request\":
{\r\n        \"ActionName\": \"FTOS_BNPL_
CalcScoringAndLimit\",\r\n        \"Data\": \"
{\\\"KOInput\\\":
{\\\
"hasModifiedCredentialsPast24Hours\\\"
:
false
,\\\
"ordersReturnedPercentage\\\"
:
0.1
,\\\
"dpdForBnplActiveProducts\\\"
:
0
,\\\
"bnplWithDpdPast12Months\\\"
:
10
,\\\
"bnplRefusedPaymentsNoLast30Days\\\"
:4},\\\"scoringInput\\\":
{\\\
"BNPLUsageRate\\\"
:\\\
"20\\\"
,\\\
"maxDPD\\\"
:\\\
"N/A\\\"
,\\\
"payedInAdvance\\\"
:\\\
"2\\\"
,\\\
"averageCheckoutTicketSize\\\"
:\\\"31\\\",\\\"paymentInstrument\\\":\\\"Bank
Transfer\\\"
,\\\
"uniqueCardsNo\\\"
:\\\
"3\\\"
,\\\
"customerLoyalty\\\"
:\\\
"4\\\"
,\\\

:\\\

,\\\

:\\\

,\\\"userScore\\\":141}}\"\r\n    },\r\n    \"ApiInfo\":
```

```
:141}}\"\r\n    },\r\n    \"ApiInfo\":
{\r\n        \"UserName\": \"
{{auth.username}}\",\r\n        \"Token\": \"
{{ftos.auth.accessToken}}\"\r\n    }\r\n}",
                    "options": {
                        "raw": {
                            "language": "json"
                        }
                    }
                },
                "url": {
                    "raw": "{{host.schema}}://
{{host.name}}/api/OpenApi/CallAction",
                    "protocol": "{{host.schema}}",
                    "host": [
                        "{{host.name}}"
                    ],
                    "path": [
                        "api",
                        "OpenApi",
                        "CallAction"
                    ]
                }
            },
```

FTOS_BNPL_GetAvailableProducts

```
"response": []
        },
        {
            "name": "GetAvailableProducts",
            "request": {
                "method": "POST",
                "header": [],
                "body": {
                    "mode": "raw",
```

```
                    "raw": "{\r\n\t\"Request\":
{\r\n\t\t\"ActionName\":\"FTOS_BNPL_
GetAvailableProducts\",\r\n        \"Data\": \"
{\\\
"score\\\"
:200,\\\"countryCode\\\":\\\"FR\\\",\\\"productCodes\\\":
[\\\
"BNPL30D\\\"
,\\\
"BNPL3X\\\"
,\\\
"BNPL4X\\\"
,\\\
"BNPL5X\\\"
,\\\
"BNPL6X\\\"
,\\\
"BNPL7X\\\"
,\\\
"BNPL8X\\\"
,\\\
"BNPL9X\\\"
,\\\
"BNPL10X\\\"
,\\\
"BNPL11X\\\"
,\\\"BNPL12X\\\"]}\"\r\n    },\r\n    \"ApiInfo\":
{\r\n        \"UserName\":\"
{{auth.username}}\",\r\n        \"Token\":\"
{{ftos.auth.accessToken}}\"\r\n    }\r\n}",
                    "options": {
                        "raw": {
                            "language": "json"
                        }
                    }
                },
                "url": {
                    "raw": "{{host.schema}}://
{{host.name}}/api/OpenApi/CallAction",
                    "protocol": "{{host.schema}}",
                    "host": [
                        "{{host.name}}"
                    ],
                    "path": [
                        "api",
                        "OpenApi",
                        "CallAction"
```

```
                    ]
                }
            },
```

FTOS_BNPL_GetRiskLevel

```
"response": []
        },
        {
            "name": "GetRiskLevel",
            "request": {
                "method": "POST",
                "header": [],
                "body": {
                    "mode": "raw",
                    "raw": "{\r\n\t\"Request\":
{\r\n\t\t\"ActionName\":\"FTOS_BNPL_
GetRiskLevel\",\r\n        \"Data\": \"
{\\\
"amount\\\"
:
300
,\\\
"productCode\\\"
:\\\"BNPL3X\\\"}\"\r\n    },\r\n    \"ApiInfo\":
{\r\n        \"UserName\":\"
{{auth.username}}\",\r\n        \"Token\":\"
{{ftos.auth.accessToken}}\"\r\n    }\r\n}",
                    "options": {
                        "raw": {
                            "language": "json"
                        }
                    }
                },
                "url": {
                    "raw": "{{host.schema}}://
{{host.name}}/api/OpenApi/CallAction",
                    "protocol": "{{host.schema}}",
                    "host": [
                        "{{host.name}}"
                    ],
                    "path": [
                        "api",
                        "OpenApi",
```

```
                        "CallAction"
                    ]
                }
            },
```

FTOS_BNPL_AddBNPLApplication

```
"response": []
        },
        {
            "name": "AddBNPLApplication",
            "request": {
                "method": "POST",
                "header": [],
                "body": {
                    "mode": "raw",
                    "raw": "{\r\n\t\"Request\":
{\r\n\t\t\"ActionName\":\"FTOS_BNPL_
AddBNPLApplication\",\r\n        \"Data\":
\"\"\r\n    },\r\n    \"ApiInfo\":
{\r\n        \"UserName\":\"
{{auth.username}}\",\r\n        \"Token\":\"
{{ftos.auth.accessToken}}\"\r\n    }\r\n}",
                    "options": {
                        "raw": {
                            "language": "json"
                        }
                    }
                },
                "url": {
                    "raw": "{{host.schema}}://
{{host.name}}/api/OpenApi/CallAction",
                    "protocol": "{{host.schema}}",
                    "host": [
                        "{{host.name}}"
                    ],
                    "path": [
                        "api",
                        "OpenApi",
                        "CallAction"
                    ]
                }
            },
```

FTOS_BNPL_UpdateBNPLApplication

```
"response": []
        },
        {
            "name": "UpdateBNPLApplication",
            "request": {
                "method": "POST",
                "header": [],
                "body": {
                    "mode": "raw",
                    "raw": "{\r\n\t\"Request\":
{\r\n\t\t\"ActionName\":\"FTOS_BNPL_
UpdateBNPLApplication\",\r\n        \"Data\": \"
{\\\"data\\\":{\\\"loan\\\":{\\\"id\\\":\\\"1e17e93e-eb34-
4d15-8baa-ab8ff33c796b\\\",\\\"data\\\":
{\\\
"bankingProductCode\\\"
:\\\"BNPL3X\\\"}}}}\"\r\n    },\r\n    \"ApiInfo\":
{\r\n        \"UserName\":\"
{{auth.username}}\",\r\n        \"Token\":\"
{{ftos.auth.accessToken}}\"\r\n    }\r\n}",
                    "options": {
                        "raw": {
                            "language": "json"
                        }
                    }
                },
                "url": {
                    "raw": "{{host.schema}}://
{{host.name}}/api/OpenApi/CallAction",
                    "protocol": "{{host.schema}}",
                    "host": [
                        "{{host.name}}"
                    ],
                    "path": [
                        "api",
                        "OpenApi",
                        "CallAction"
                    ]
                }
            },
```

FTOS_BNPL_GetBNPLApplicationLink

```
"response": []
        },
        {
            "name": "GetBNPLApplicationLink",
            "request": {
                "method": "POST",
                "header": [],
                "body": {
                    "mode": "raw",
                    "raw": "{\r\n\t\"Request\":
{\r\n\t\t\"ActionName\":\"FTOS_BNPL_
GetBNPLApplicationLink\",\r\n        \"Data\": \"
{\\\"loanId\\\":\\\"1e17e93e-eb34-4d15-8baa-
ab8ff33c796b\\\"}\"\r\n    },\r\n    \"ApiInfo\":
{\r\n        \"UserName\":\"
{{auth.username}}\",\r\n        \"Token\":\"
{{ftos.auth.accessToken}}\"\r\n    }\r\n}",
                    "options": {
                        "raw": {
                            "language": "json"
                        }
                    }
                },
                "url": {
                    "raw": "{{host.schema}}://
{{host.name}}/api/OpenApi/CallAction",
                    "protocol": "{{host.schema}}",
                    "host": [
                        "{{host.name}}"
                    ],
                    "path": [
                        "api",
                        "OpenApi",
                        "CallAction"
                    ]
                }
            },
```


FTOS_BNPL_GetMerchantBankAccount


```
    "response": []
        },
        {
            "name": "GetMerchantBankAccount",
```

```
            "request": {
                "method": "POST",
                "header": [],
                "body": {
                    "mode": "raw",
                    "raw": "{\r\n\t\"Request\":
{\r\n\t\t\"ActionName\":\"FTOS_BNPL_
GetMerchantBankAccount\",\r\n        \"Data\": \"
{\\\
"merchantName\\\"
:\\\
"TechShop\\\"
,\\\
"currencyCode\\\"
:\\\"EUR\\\"}\"\r\n    },\r\n    \"ApiInfo\":
{\r\n        \"UserName\":\"
{{auth.username}}\",\r\n        \"Token\":\"
{{ftos.auth.accessToken}}\"\r\n    }\r\n}",
                    "options": {
                        "raw": {
                            "language": "json"
                        }
                    }
                },
                "url": {
                    "raw": "{{host.schema}}://
{{host.name}}/api/OpenApi/CallAction",
                    "protocol": "{{host.schema}}",
                    "host": [
                        "{{host.name}}"
                    ],
                    "path": [
                        "api",
                        "OpenApi",
                        "CallAction"
                    ]
                }
            },
```

FTOS_BNPL_CreateMerchantBankAccount

```
"response": []
        },
        {
```

```
            "name": "CreateMerchantBankAccount",
            "request": {
                "method": "POST",
                "header": [],
                "body": {
                    "mode": "raw",
                    "raw": "{\r\n\t\"Request\":
{\r\n\t\t\"ActionName\":\"FTOS_BNPL_
CreateMerchantBankAccount\",\r\n        \"Data\": \"
{\\\
"merchantName\\\"
:\\\
"TechShop\\\"
,\\\
"fiscalRegistrationNo\\\"
:\\\
"100243\\\"
,\\\
"commercialRegistration\\\"
:\\\"100243\\\"}\"\r\n    },\r\n    \"ApiInfo\":
{\r\n        \"UserName\":\"
{{auth.username}}\",\r\n        \"Token\":\"
{{ftos.auth.accessToken}}\"\r\n    }\r\n}",
                    "options": {
                        "raw": {
                            "language": "json"
                        }
                    }
                },
                "url": {
                    "raw": "{{host.schema}}://
{{host.name}}/api/OpenApi/CallAction",
                    "protocol": "{{host.schema}}",
                    "host": [
                        "{{host.name}}"
                    ],
                    "path": [
                        "api",
                        "OpenApi",
                        "CallAction"
                    ]
                }
            },
```

FTOS_BNPL_ActivateContract

```
    "response": []
        },
        {
            "name": "ActivateContract",
            "request": {
                "method": "POST",
                "header": [],
                "body": {
                    "mode": "raw",
                    "raw": "{\r\n\t\"Request\":
{\r\n\t\t\"ActionName\":\"FTOS_BNPL_
ActivateContract\",\r\n        \"Data\": \"
{\\\
"contractNo\\\"
:\\\
"455\\\"
,\\\
"merchantCustomerNo\\\"
:\\\"251\\\"}\"\r\n    },\r\n    \"ApiInfo\":
{\r\n        \"UserName\":\"
{{auth.username}}\",\r\n        \"Token\":\"
{{ftos.auth.accessToken}}\"\r\n    }\r\n}",
                    "options": {
                        "raw": {
                            "language": "json"
                        }
                    }
                },
                "url": {
                    "raw": "{{host.schema}}://
{{host.name}}/api/OpenApi/CallAction",
                    "protocol": "{{host.schema}}",
                    "host": [
                        "{{host.name}}"
                    ],
                    "path": [
                        "api",
                        "OpenApi",
                        "CallAction"
                    ]
                }
            },
```


FTOS_BNPL_UpdateContract

```
"response": []
        },
        {
            "name": "UpdateContract",
            "request": {
                "method": "POST",
                "header": [],
                "body": {
                    "mode": "raw",
                    "raw": "{\r\n\t\"Request\":
{\r\n\t\t\"ActionName\":\"FTOS_BNPL_
UpdateContract\",\r\n        \"Data\": \"
{\\\"contractNo\\\":\\\"455\\\",\\\"updateData\\\":
{\\\"amountDue\\\":1200}}\"\r\n    },\r\n    \"ApiInfo\":
{\r\n        \"UserName\":\"
{{auth.username}}\",\r\n        \"Token\":\"
{{ftos.auth.accessToken}}\"\r\n    }\r\n}",
                    "options": {
                        "raw": {
                            "language": "json"
                        }
                    }
                },
                "url": {
                    "raw": "{{host.schema}}://
{{host.name}}/api/OpenApi/CallAction",
                    "protocol": "{{host.schema}}",
                    "host": [
                        "{{host.name}}"
                    ],
                    "path": [
                        "api",
                        "OpenApi",
                        "CallAction"
                    ]
                }
            },
```


FTOS_BNPL_SaveEarlyRepayment


```
"response": []
        },
        {
            "name": "SaveEarlyRepayment",
```

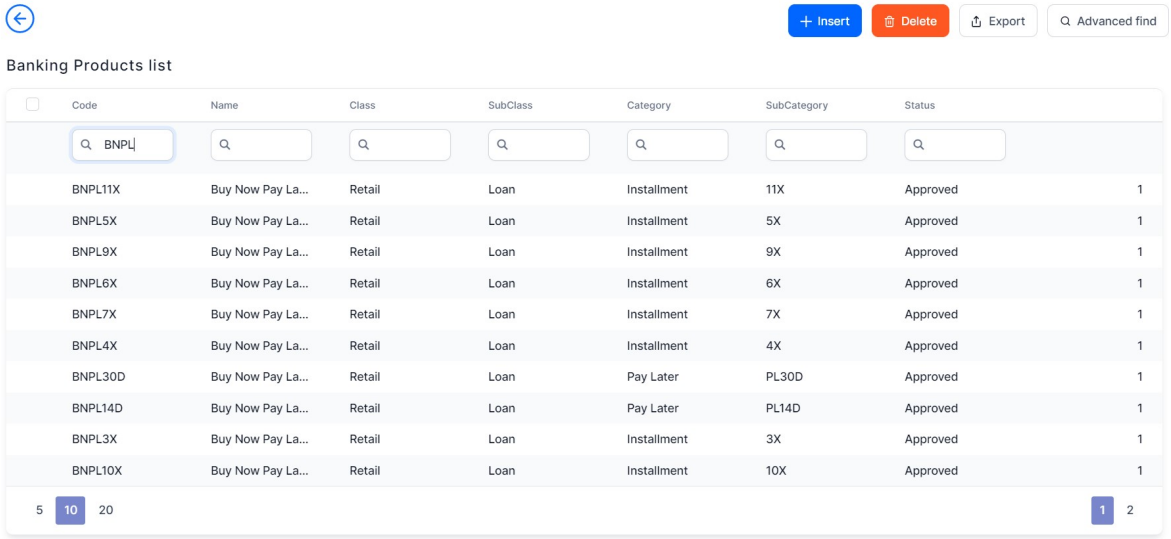
```
            "request": {
                "method": "POST",
                "header": [],
                "body": {
                    "mode": "raw",
                    "raw": "{\r\n\t\"Request\":
{\r\n\t\t\"ActionName\":\"FTOS_BNPL_
SaveEarlyRepayment\",\r\n        \"Data\": \"
{\\\
"contractNo\\\"
:\\\
"459\\\"
,\\\"eventValue\\\":100}\"\r\n    },\r\n    \"ApiInfo\":
{\r\n        \"UserName\":\"
{{auth.username}}\",\r\n        \"Token\":\"
{{ftos.auth.accessToken}}\"\r\n    }\r\n}",
                    "options": {
                        "raw": {
                            "language": "json"
                        }
                    }
                },
                "url": {
                    "raw": "{{host.schema}}://
{{host.name}}/api/OpenApi/CallAction",
                    "protocol": "{{host.schema}}",
                    "host": [
                        "{{host.name}}"
                    ],
                    "path": [
                        "api",
                        "OpenApi",
                        "CallAction"
                    ]
                }
            },
```

Configuring the Banking Products

This processor creates and maintains products that are used in FintechOS form driven flows and later in digital journeys. To visualize and to administer them, access the dedicated menu for **Product Factory** within the Innovation Studio. The custom products are included in the solution with pre-loaded filters and ranking systems. They are a great place to start, but if for some reason you need further configurations, you have limitless possibilities to modify, extend or define new custom offerings for your customers. It offers configurations such as product dimensions for commissions, interests, and product filters to navigate through the list of products created based on a filtering scope. The Buy Now, Pay Later journey has the products:

Product	Name	Class	Subclass	Category
BNPL	Buy Now Pay Later	N/A	N/A	N/A
BNPL3X	Buy Now Pay Later 3X	Retail	Loan	Installment
BNPL4X	Buy Now Pay Later 4X	Retail	Loan	Installment
BNPL5X	Buy Now Pay Later 5X	Retail	Loan	Installment
BNPL6X	Buy Now Pay Later 6X	Retail	Loan	Installment
BNPL7X	Buy Now Pay Later 7X	Retail	Loan	Installment
BNPL8X	Buy Now Pay Later 8X	Retail	Loan	Installment
BNPL9X	Buy Now Pay Later 9X	Retail	Loan	Installment
BNPL10X	Buy Now Pay Later 10X	Retail	Loan	Installment
BNPL11X	Buy Now Pay Later 11X	Retail	Loan	Installment
BNPL12X	Buy Now Pay Later 12X	Retail	Loan	Installment
BNPL30D	Buy Now Pay Later - 30 Days	Retail	Loan	Installment

1. Log into Innovation Studio in developer mode.
2. Click the main menu icon at the top left-hand corner of the screen.
3. In the main menu, click **Product Factory**.
4. Click **Banking Products** to open the **Banking Products List** page.



Buy Now Pay Later

Main Info

The first tab requires the basic elements for the creation of a product such as product type, name, code, hierarchy and features.

Field	Required	Description	Example
Product Type	Yes	The type of product.	Term Loan
External Code	No	The code of the product imported from an external system, if applicable. It can have 10 characters and it is not used in the contract.	BNPL
Banking Product Code	Yes	The code of the product. It can have 10 characters and it is used in the contract. It uses a sequencer and the code of the product type.	BNPL
Name	Yes	The name of the product.	Buy Now Pay Later
Class	No	This field is used to place the product in a hierarchy.	
Subclass	No	This field is used to place the product in a hierarchy.	

Field	Required	Description	Example
Category	No	This field is used to place the product in a hierarchy.	
Start Date	Yes	The date when the product becomes available.	27/09/2021
End Date	Yes	The last date from when the product is available. From that date forward, the product is no longer available.	27/09/2099
Description	No	Write any description or additional text here.	

Details

The Details tab requires further elements such as interest, payment type, top-ups and withdrawals.

Field	Required	Description	Example
Bank Account Type	Yes	Select the type.	Loan Term Account
Is Revolving	No	Allows a business to borrow money as needed for funding working capital needs and continuing operations such as meeting payroll and playable.	null
Auto Disbursement	Yes	Specifies if the disbursement is automatically performed when the contract is approved.	true
Max No Disbursements	No	The maximum number of disbursements that can be configured for this product.	
Is Guaranteed	Yes	This checkbox marks the product as secured or unsecured.	null

Field	Required	Description	Example
Allow Collateral Partial Release	No	A partial release is a mortgage provision that allows some of the collateral to be released from a mortgage after the borrower pays a certain amount of the loan.	null
Collateral Cover Percent	No	The percent that the collateral person on the contract pays. It usually is over 100%.	null
Allow CoDebtor	No	Select if another debtor exists for this product.	null
Allow Refinancing	No	Select if the account can be refinanced for this product.	null
Number of CoDebtors	No	Set the maximum number of debtors possible for this product.	

Payment Schedule Types grid

In the Payment Schedule Types section, the following fields are available:

Field	Required	Description	Example
Periodicity Type	Yes	Select the regularity of payments.	15Days

For Associated Payment Schedule Types, the Payment Schedule Types contain:

BNPL

Field	Required	Description	Example
Name	No	Insert a suggestive name for the type.	BNPL
Payment schedule code	No	Insert a code for the type to keep track of them.	BNPL

Field	Required	Description	Example
Product Type	No	Select a product type to associate with the payment schedule type. Depending on the Product Type, different calculation rules are triggered. For example, the product type Overdraft has only the payment at maturity.	Term Loan

Field	Required	Description	Example
Schedule Interest Calculation Type	No	Select from the list a type of calculation for the interest. When an annual interest rate is specified, in order to calculate the Installment for an interval of days, first the annual interest rate should be transformed in to a daily base. To make this transformation there are some accepted conventions. Innovation Studio implemented the following conventions: 30/360, 30/365, Actual/Actual, where Actual for years can be either 365 or 366. Other schedule interest calculation types can also be defined, as needed.	30/360

Field	Required	Description	Example
		In practice may be also encountered the Actual/360 or Actual/365.	
Is With Equal Installments	No	Select the checkbox if the installments are equal. If there are Commissions that appear on the Payment Schedule, these Commissions are added to the equal Installments, not included within.	null

Field	Required	Description	Example
Installment Value Custom	No	If you select the checkbox, with multiple disbursements, the Principal component of the Installments is the one calculated for the entire Financed Amount, even if it was not entirely disbursed. For example, if Financed Amount is 10.000 EURO and the value calculated for Principal component of the Installments is 800 EURO, and the customer disburses only 5.000 EURO, the Principal component remains 800, but the Interest is calculated for 5.000 EURO that were disbursed.	null

Field	Required	Description	Example
Use Fix Maturity Date (from Activation Date)	Yes	If you select the checkbox, then the Maturity Date equals to Activation Date plus the Contractual Period in Months, i.e. the number of installments depends on the Activation Date. If the checkbox remains unselected, the number of installments are fixed, the Maturity Date is equal to the First Installment plus the Contractual Period in Months, e.g. Installment date is on the first day of the month, this results in the Maturity day to be the first day of the month.	null

Field	Required	Description	Example
Measurement Unit	Yes	Select from the list the type of measurement unit applicable for the payment schedule type.	Days

For the PAYMENT SCHEDULE TYPE DETAILS:

Column Repayment Schedule	Title	Calculation Method
RemainingValue	REMAINING	RemainingFormula
Interest	INTEREST	Effective Rate
Principal	PRINCIPAL	Once
TotalInstallment	TOTALINST	ColumnFormula

Availability

The Availability tab determines the monetary range and the time frame when the product is available for customers.

Field	Required	Description	Example
Currency	Yes	Choose the currency for this banking product.	EUR
Period Type	No	Choose one: Days/Weeks/Months/Years/Once.	Days
Minimum Period	No	The minimum duration of the product mentioned in the contract.	15
Maximum Period	No	The maximum duration of the product mentioned in the contract.	15

Dimensions

The Dimensions tab displays the interests, commissions, insurances, discount and questions valid for a product.

Interests & Commissions: for Interest BMPL

Field	Value
Banking Product	Buy Now Pay Later

Field	Value
Code	BNPL
Item Name	BNPL
Start Date	27/09/2021
End Date	27/09/2099
Interest List	
Commission List	BNPL Commision List
Is Negotiable	true

Buy Now Pay Later 3X

Main Info

The first tab requires the basic elements for the creation of a product such as product type, name, code, hierarchy and features.

Field	Required	Description	Example
Product Type	Yes	The type of product.	Term Loan
External Code	No	The code of the product imported from an external system, if applicable. It can have 10 characters and it is not used in the contract.	BNPL3X
Banking Product Code	Yes	The code of the product. It can have 10 characters and it is used in the contract. It uses a sequencer and the code of the product type.	BNPL3X
Name	Yes	The name of the product.	Buy Now Pay Later 3X

Field	Required	Description	Example
Class	No	This field is used to place the product in a hierarchy.	Retail
Subclass	No	This field is used to place the product in a hierarchy.	Loan
Category	No	This field is used to place the product in a hierarchy.	Installment
Subcategory		This field is used to place the product in a hierarchy.	3x
Start Date	Yes	The date when the product becomes available.	19/10/2020
End Date	Yes	The last date from when the product is available. From that date forward, the product is no longer available.	31/12/2030

Field	Required	Description	Example
Benefits	No	Write any description or additional text here.	Order the products that you want and slice the payment in 3 parts: 1st slice + additional charge of 1.45% of the total amount of the order for a 3 installment on order date 2ndslice = order date +30 days 3rdslice = order date + 60 days (2nd slice + 30 days) The overdue amount of non paid instalment is a base for Interests calculation Each subsequent missed payment slice instalment shall increase overdue principal balance as a basis for penalty interest calculation Penalty interests is a nominal annual interest rate that calculates on daily basis on the overdue principal (24,00%) Due to the application of penalty interests the system shall allow to parametrize repayment

Field	Required	Description	Example
			order as per collection strategy that can change over time. Order of collection: overdue principal / balance due principal overdue interests

Details

The Details tab requires further elements such as interest, payment type, top-ups and withdrawals.

Field	Required	Description	Example
Bank Account Type	Yes	Select the type.	Loan Term Account
Is Revolving	No	Allows a business to borrow money as needed for funding working capital needs and continuing operations such as meeting payroll and playable.	null
Auto Disbursement	Yes	Specifies if the disbursement is automatically performed when the contract is approved.	true
Max No Disbursements	No	The maximum number of disbursements that can be configured for this product.	undefined
Is Guaranteed	Yes	This checkbox marks the product as secured or unsecured.	true

Field	Required	Description	Example
Allow Collateral Partial Release	No	A partial release is a mortgage provision that allows some of the collateral to be released from a mortgage after the borrower pays a certain amount of the loan.	null
Collateral Cover Percent	No	The percent that the collateral person on the contract pays. It usually is over 100%.	undefined
Allow CoDebtor	No	Select if another debtor exists for this product.	true
Allow Refinancing	No	Select if the account can be refinanced for this product.	false
Number of CoDebtors	No	Set the maximum number of debtors possible for this product.	2

Payment Schedule Types grid

In the Payment Schedule Types section, the following fields are available:

Field	Required	Description	Example
Periodicity Type	Yes	Select the regularity of payments.	Monthly
Holiday Shift For Repayment Installments	Yes	This checkbox marks if the holidays are considered for the calculation of the maturity schedule.	undefined

Field	Required	Description	Example
Holiday Shift Method	No	Select from the list the method to be used when calculating the due date if that date falls to a holiday. The due date can be shifted before or after the holiday. Possible values: • None - the due date is not shifted. • Forward - the due date is shifted to the last working day before the initially calculated due date. • Backward - the due date is shifted to the next working day after the initially calculated due date.	Forward
Defer Due Date	Yes	If you select the checkbox, the payment schedule calculates the next payment amount as if the due date has not changed even when the due date falls on a holiday. This checkbox is selected by default.	true
Grace Type	No	Select one from the list.	none
Product Grace	No	Select one from the list.	
Grace Days for Repayment	No	The number of days for which the grace applies.	undefined

Field	Required	Description	Example
Penalty for grace period	No	If you select the checkbox, the penalty interest is applied on the loan contract without taking into consideration the grace period defined at contract level, being calculated for the difference between system date - due date, if the grace period passed and the customer didn't pay the due amounts. If you leave this checkbox unselected, the penalty interest is applied on the loan contract taking into consideration the grace period defined at contract level, being calculated for system date - due date + grace days for repayment.	undefined

For Associated Payment Schedule Types, the Payment Schedule Types contain:

Equal installments

Field	Required	Description	Example
Name	No	The name for the type.	Equal installments
Payment schedule code	No	The code for the type to keep track of them.	MEIM360

Field	Required	Description	Example
Product Type	No	The product type to associate with the payment schedule type. Depending on the Product Type, different calculation rules are triggered. For example, the product type Overdraft has only the payment at maturity.	Mortgage

Field	Required	Description	Example
Schedule Interest Calculation Type	No	Select from the list a type of calculation for the interest. When an annual interest rate is specified, in order to calculate the Installment for an interval of days, first the annual interest rate should be transformed in to a daily base. To make this transformation there are some accepted conventions. Innovation Studio implemented the following conventions: 30/360, 30/365, Actual/Actual, where Actual for years can be either 365 or 366. Other schedule interest calculation types	30/360

Field	Required	Description	Example
		can also be defined, as needed. In practice may be also encountered the Actual/360 or Actual/365.	
Is With Equal Installments	No	Select the checkbox if the installments are equal. If there are Commissions that appear on the Payment Schedule, these Commissions are added to the equal Installments, not included within.	true

Field	Required	Description	Example
Installment Value Custom	No	If you select the checkbox, with multiple disbursements, the Principal component of the Installments is the one calculated for the entire Financed Amount, even if it was not entirely disbursed. For example, if Financed Amount is 10.000 EURO and the value calculated for Principal component of the Installments is 800 EURO, and the customer disburses only 5.000 EURO, the Principal component remains 800, but the Interest is calculated for 5.000 EURO that were disbursed.	true

Field	Required	Description	Example
Use Fix Maturity Date (from Activation Date)	Yes	If you select the checkbox, then the Maturity Date equals to Activation Date plus the Contractual Period in Months, i.e. the number of installments depends on the Activation Date. If the checkbox remains unselected, the number of installments are fixed, the Maturity Date is equal to the First Installment plus the Contractual Period in Months, e.g. Installment date is on the first day of the month, this results in the Maturity day to be the first day of the month.	null

Field	Required	Description	Example
Measurement Unit	Yes	Select from the list the type of measurement unit applicable for the payment schedule type.	Months

For the PAYMENT SCHEDULE TYPE DETAILS:

Column Repayment Schedule	Calculation Method
AnalysisFee	FeeOnce
RemainingValue	RemainingFormula
PMT	FixedValue
Interest	Effective Rate
Principal	ColumnFormula
TotalInstallment	ColumnFormula

Availability

The Availability tab determines the monetary range and the time frame when the product is available for customers.

Field	Required	Description	Example
Currency	Yes	Choose the currency for this banking product.	GDP
Period Type	No	Choose one: Days/Weeks/Months/Years/Once.	Months
Minimum Period	No	The minimum duration of the product mentioned in the contract.	1
Maximum Period	No	The maximum duration of the product mentioned in the contract.	480
Minimum Amount	No	The minimum amount of the product for which the bank opens a contract.	1000
Maximum Amount	No	The maximum amount of the product for which the bank opens a contract.	9500,000
Minimum Advance (%)	No	The minimum down payment that must be paid for the leasing contract to be signed.	15%

Field	Required	Description	Example
Maximum Advance (%)	No	The maximum advance that can be paid for the leasing contract to be signed.	85%

Dimensions

The Dimensions tab displays the interests, commissions, insurances, discount and questions valid for a product.

Interests & Commissions: for Interest BNPL

Field	Value
Banking Product	2Y Fixed
Code	IRWM_02
Item Name	Interest RWM_01
Start Date	19/01/2022
End Date	29/11/2029
Interest List	IRWM_02
Commission List	Mortgage fee 999
Minimum Interest Rate (%)	undefined
Is Negotiable	null

Buy Now Pay Later - 30X

Main Info

The first tab requires the basic elements for the creation of a product such as product type, name, code, hierarchy and features.

Field	Required	Description	Example
Product Type	Yes	The type of product.	Term Loan
External Code	No	The code of the product imported from an external system, if applicable. It can have 10 characters and it is not used in the contract.	BNPL30X
Banking Product Code	Yes	The code of the product. It can have 10 characters and it is used in the contract. It uses a sequencer and the code of the product type.	BNPL30X

Field	Required	Description	Example
Name	Yes	The name of the product.	Buy Now Pay Later - 30X
Class	No	This field is used to place the product in a hierarchy.	Retail
Subclass	No	This field is used to place the product in a hierarchy.	Loan
Category	No	This field is used to place the product in a hierarchy.	Installment
Subcategory	No	This field is used to place the product in a hierarchy.	30x
Start Date	Yes	The date when the product becomes available.	19/10/2020
End Date	Yes	The last date from when the product is available. From that date forward, the product is no longer available.	31/12/2030
Benefits	No	Write any description or additional text here.	
Product image	No	Insert an image here.	

Details

The Details tab requires further elements such as interest, payment type, top-ups and withdrawals.

Field	Required	Description	Example
Bank Account Type	Yes	The type of account.	Loan Term Account
Is Revolving	No	Allows a business to borrow money as needed for funding working capital needs and continuing operations such as meeting payroll and playable.	null
Auto Disbursement	Yes	Specifies if the disbursement is automatically performed when the contract is approved.	true

Field	Required	Description	Example
Max No Disbursements	No	The maximum number of disbursements that can be configured for this product.	1
Is Guaranteed	Yes	This checkbox marks the product as secured or unsecured.	null
Allow Collateral Partial Release	No	A partial release is a mortgage provision that allows some of the collateral to be released from a mortgage after the borrower pays a certain amount of the loan.	null
Collateral Cover Percent	No	The percent that the collateral person on the contract pays. It usually is over 100%.	null
Allow CoDebtor	No	Select if another debtor exists for this product.	null
Allow Refinancing	No	Select if the account can be refinanced for this product.	null

Payment Schedule Types grid

In the Payment Schedule Types section, the following fields are available:

Field	Required	Description	Example
Periodicity Type	Yes	The regularity of payments.	30Days
Repayment Allocation Method	No	The repayment allocation method represents the recovery order of amounts from repayment notifications. You can specify if charges should be recovered with priority or maybe interest or principal. You can also split order by number of days overdue. Select a repayment allocation method from the list.	CostOrder

Field	Required	Description	Example
Defer due date	No	If you select the checkbox, the payment schedule calculates the next payment amount as if the due date has not changed even when the due date falls on a holiday.	true
Grace Type	No	Select one from the list.	Both
Product Grace	No	Select one from the list.	FiveDaysGrace

For Associated Payment Schedule Types, the Payment Schedule Types contain:

EqualPrincipal_30DaysForDueDate_BNPL

Field	Required	Description	Example
Name	No	The name for the type.	EqualPrincipal_14DaysForDueDate_BNPL
Payment schedule code	No	The code for the type to keep track of them.	30D
Product Type	No	The product type to associate with the payment schedule type. Depending on the Product Type, different calculation rules are triggered. For example, the product type Overdraft has only the payment at maturity.	Term Loan

Field	Required	Description	Example
Schedule Interest Calculation Type	No	Select from the list a type of calculation for the interest. When an annual interest rate is specified, in order to calculate the Installment for an interval of days, first the annual interest rate should be transformed in to a daily base. To make this transformation there are some accepted conventions. Innovation Studio implemented the following conventions: 30/360, 30/365, Actual/Actual, where Actual for years can be either 365	30Days

Field	Required	Description	Example
		or 366. Other schedule interest calculation types can also be defined, as needed. In practice may be also encountered the Actual/360 or Actual/365.	
Is With Equal Installments	No	Select the checkbox if the installments are equal. If there are Commissions that appear on the Payment Schedule, these Commissions are added to the equal Installments, not included within.	true

Field	Required	Description	Example
Installment Value Custom	No	If you select the checkbox, with multiple disbursements, the Principal component of the Installments is the one calculated for the entire Financed Amount, even if it was not entirely disbursed. For example, if Financed Amount is 10.000 EURO and the value calculated for Principal component of the Installments is 800 EURO, and the customer disburses only 5.000 EURO, the Principal component remains 800, but the Interest is	null

Field	Required	Description	Example
		calculated for 5.000 EURO that were disbursed.	

Field	Required	Description	Example
Use Fix Maturity Date (from Activation Date)	Yes	If you select the checkbox, then the Maturity Date equals to Activation Date plus the Contractual Period in Months, i.e. the number of installments depends on the Activation Date. If the checkbox remains unselected, the number of installments are fixed, the Maturity Date is equal to the First Installment plus the Contractual Period in Months, e.g. Installment date is on the first day of the month, this results in the Maturity day	null

Field	Required	Description	Example
		to be the first day of the month.	
Measurement Unit	Yes	Select from the list the type of measurement unit applicable for the payment schedule type.	Days

For the PAYMENT SCHEDULE TYPE DETAILS:

Column Repayment Schedule	Title	Calculation Method
Increment	Increment	ColumnFormula
RemainingValue	RemainingValue	RemainingFormula
Interest	Interest	Effective Rate
Principal	Principal	Liniar
TotalInstallment	Total Installment	ColumnFormula

Availability

The Availability tab determines the monetary range and the time frame when the product is available for customers.

Field	Required	Description	Example
Currency	Yes	The currency for this banking product.	EUR
Period Type	No	The period type: Days/Weeks/Months/Years/Once.	Days
Minimum Period	No	The minimum duration of the product mentioned in the contract.	30
Maximum Period	No	The maximum duration of the product mentioned in the contract.	30

Field	Required	Description	Example
Maximum Period For Disbursement After Activation (Months)	No	The maximum period of disbursement after the this option is set. The number of months during which the disbursement must be made.	1
Minimum Principal For Early Repayment	No	The minimum principal for when early repayments are made.	0.01
Minimum Amount	No	The minimum amount of the product for which the bank opens a contract.	1
Maximum Amount	No	The maximum amount of the product for which the bank opens a contract.	2000

Dimensions

The Dimensions tab displays the interests, commissions, insurances, discount and questions valid for a product.

Interests & Commissions: for Interest Buy Now Pay Later 14X Interest & Commision List

Field	Value
Banking Product	Buy Now Pay Later 30X
Code	BNPL30X_IC
Item Name	Buy Now Pay Later 14X Interest & Commision List
Start Date	11/02/2022
End Date	11/02/2062
Interest List	BNPL 1X Interest List
Is Negotiable	null

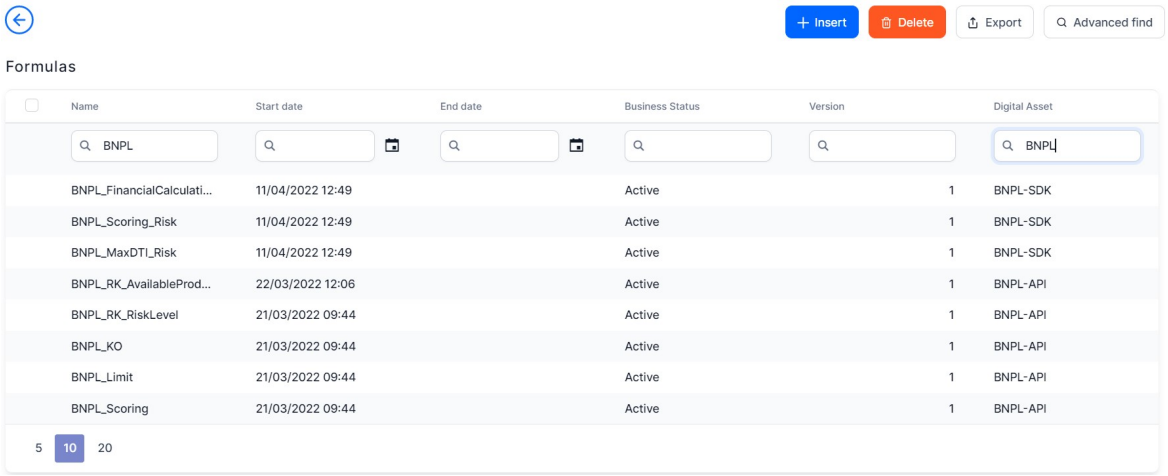
Configuring the Business Formulas

The Business Formulas processor handles different inputs from your digital journey in order to generate desired outputs. By inserting arguments and using them in steps, the system creates complex calculations to be triggered in a flow. The order of the formula execution for the scoring done by the marketplace:

- "BNPL_KO" on the next page
- "BNPL_Scoring" on page 98
- "BNPL_Limit" on page 106
- "BNPL_RK_AvailableProducts" on page 107
- "BNPL_RK_RiskLevel" on page 108

The order of the formula execution for the scoring done by the bank is:

- "BNPL_RK_AvailableProducts" on page 107
- "BNPL_KO_Risk" on page 113
- "BNPL_Scoring_Risk" on page 116
- "BNPL_MaxDTI_Risk" on page 114
- "BNPL_FinancialCalculation" on page 112.



IMPORTANT!
The formulas and matrices are for example purposes only. They are subject to internal modifications for each implementation.

BNPL_KO

This formula calculates the knock-out criteria for the customer.

Field	Example
Name	BNPL_KO
Formula Input	BNPL_KO
Start date	07/03/2022 16:42

Formula steps:

Name	Description	Master Type	SubType	Calculation Type	Formula
KO_ bnplWithDpdPast12Months	It checks if the number of days is over 15.	Simple Type	Boolean	Normal	`result = DataSet("BNPL_KO_BnplWithDpdPast12Months", ("bnplWithDpdPast12Months", bnplWithDpdPast12Months)) == 1;`
KO_ dpdForBnplActiveProducts	It checks if the number of active products is zero.	Simple Type	Boolean	Normal	`result = DataSet("BNPL_KO_DpdForBnplActiveProducts", ("dpdForBnplActiveProducts", dpdForBnplActiveProducts)) == 1;`
KO_ ordersReturnedPercentage	It checks if the orders returned percetage is 0.5.	Simple Type	Boolean	Normal	`result = DataSet("BNPL_KO_OrdersReturnedPercentage", ("ordersReturnedPercentage", ordersReturnedPercentage)) == 1;`
KO_ bnplRefusedPaymentsNoLast30Days	It checks if the refused paymments in the last 30 days is over 5.	Simple Type	Boolean	Normal	`result = DataSet("BNPL_KO_BnplRefusedPaymentsNoLast30Days", ("bnplRefusedPaymentsNoLast30Days", bnplRefusedPaymentsNoLast30Days)) == 1;`

Name	Description	Master Type	SubType	Calculation Type	Formula
KO_hasModifiedCredentialsPast24Hours	It checks if the credentials were modified in the last 24h.	Simple Type	Boolean	Normal	`result = DataSet("BNPL_KO_HasModifiedCredentialsPast24Hours", ("hasModifiedCredentialsPast24Hours", hasModifiedCredentialsPast24Hours)) == 1;`
KO_Final	It determines the knock-out of the customer.	Simple Type	Boolean	Normal	`var criterias = KO_bnplWithDpdPast12Months && KO_dpdForBnplActiveProducts && KO_ordersReturnedPercentage && KO_bnplRefusedPaymentsNoLast30Days && KO_hasModifiedCredentialsPast24Hours; result = !criterias;`

Data Set **BNPL_KO_BnplWithDpdPast12Months**

Matrix		
bnplWithDpdPast12Months	**Values**	**Description**
(15;]	0	The customer fails the check.
[;15]	1	The customer passes the check.

Data Set **BNPL_KO_DpdForBnplActiveProducts**

Matrix		
dpdForBnplActiveProducts	**Values**	**Description**
(0;]	0	The customer fails the check.
[;0]	1	The customer passes the check.

Data set **BNPL_KO_ordersReturnedPercentage**

Matrix		
ordersReturnedPercentage	**Values**	**Description**
(0.5;]	0	The customer fails the check.
[;0.5]	1	The customer passes the check.

Data set **BNPL_KO_bnplRefusedPaymentsNoLast30Days**

Matrix		
bnplRefusedPaymentsNoLast30Days	**Values**	**Description**
(5;]	0	The customer fails the check.
[;5]	1	The customer passes the check.

Data set **BNPL_KO_hasModifiedCredentialsPast24Hours**

Matrix		
hasModifiedCredentialsPast24Hours	**Values**	**Description**
(0;]	0	The customer fails the check.
[;0]	1	The customer passes the check.

BNPL_Scoring

After the customer continued to the check out page, the system decided if the customer is eligible for the limit. The data is hard-coded, but it can be fetched via API. For eligibility rules, the customer fails if:

- Buy Now, Pay Later with DPD >15 days in the past 12 months
- DPD more then 0 for Buy Now, Pay Later active products
- orders returned in the past 30 days >50%
- payments for Buy Now, Pay Later initiated at checkout and refused >5 in the past 30 days
- customer has modified the credentials in the past 24 hours.

Field	Example
Name	BNPL_Scoring
Formula Input	BNPL_Scoring
Start date	07/03/2022 17:40

Formula steps:

Name	Description	Master Type	SubType	Calculation Type	Formula
SCORING_ customerLoyalty	It determines the customer loyalty.	Simple Type	Whole Number	Normal	`result = DataSet ("BNPL_Scoring_ CustomerLoyalty", ( "CustomerLoyalty" , customerLoyalt y));`
SCORING_ averageCheckout TicketSIze	It determines the average checkout ticket size in the past 12 months.	Simple Type	Whole Number	Normal	`result = DataSet ("BNPL_Scoring_ AverageCheckoutT icketSize", ( "AverageCheckout TicketSize", averageCheckoutT icketSize));`
SCORING_ mostCommonIns trument	It determines the type of payment used by the customer used in the last 12 months.	Simple Type	Whole Number	Normal	`result = DataSet ("BNPL_Scoring_ MostCommonPaymen tInstrument", ( "MostCommonPayme nt", paymentInstrumen t));`

Name	Description	Master Type	SubType	Calculation Type	Formula
SCORING_maxDPD	It determines the maximum DPD in the past 12 months for Buy Now, Pay Later payments.	Simple Type	Whole Number	Normal	`result = DataSet ("BNPL_Scoring_MaxDPD", ("maxDPD", maxDPD));`
SCORING_advancePay	It determines how many times the customer paid in advanced in the past 12 months.	Simple Type	Whole Number	Normal	`result = DataSet ("BNPL_Scoring_AdvancePayments", ( "AdvancePayments" , payedInAdvance));`

Name	Description	Master Type	SubType	Calculation Type	Formula
SCORING_usageRate	It determines the usage rate for Buy Now, Pay Later in past 12 months.	Simple Type	Whole Number	Normal	`result = DataSet("BNPL_Scoring_UsageRate", ("BNPLUsageRate", BNPLUsageRate));`
SCORING_returnedProductsPercent	It determines the % of returned products in the past 12 months, min 10 orders placed.	Simple Type	Whole Number	Normal	`result = DataSet("BNPL_Scoring_ReturnedProductsPercent", ("returnedProductsPercent", returnedProductsPercent));`

Name	Description	Master Type	SubType	Calculation Type	Formula
SCORING_ uniqueCardsUsed	It determines the number of unique cards used for transactions in the past 12 months, it must be more than 10 orders.	Simple Type	Whole Number	Normal	`result = DataSet ("BNPL_Scoring_ UniqueCardsUsed", ( "UniqueCardsUsed" , uniqueCardsNo));`
SCORING_ CustomerAge	It determines the age of the customer.	Simple Type	Whole Number	Normal	`result = DataSet ("BNPL_Scoring_ CustomerAge", ("CustomerAge", customerAge));`
Userscore	It calculates the scoring per customer.	Simple Type	Whole Number	Normal	`result = SCORING_ customerLoyalty + SCORING_ averageCheckoutT icketSIze + SCORING_ mostCommonInstru ment + SCORING_ maxDPD + SCORING_ advancePay + SCORING_usageRate + SCORING_ returnedProducts Percent + SCORING_ uniqueCardsUsed + SCORING_ CustomerAge;`

Data set **BNPL_Scoring_CustomerLoyalty**

The hard-coded values from the following data sets are placed in "FTOS_EC_ShoppingCartLoader" on page 122.

It is measured in years.

Matrix		
Customer Loyalty	**Values**	**Description**
[-1;-1]	5	N/A
[0;1)	5	customer for <1 year
[1;2]	10	customer for 1-2 years
(2;3]	15	customer for 2-3 years
(3;4]	20	customer for 3-4 years The hard-coded value is 4.
(4;]	25	customer for >4 years

Data set **BNPL_Scoring_AverageCheckoutTIcketSize**

It is measured in euros.

Matrix		
AverageCheckoutTicketSize	**Values**	**Description**
[-1;-1]	5	N/A
[0;10)	5	<50 EURO
[10;30]	10	<20 EURO
[31;50]	15	20 - 50 EURO The hard-coded value is 31.
[51;100]	20	51 -100 EURO
[101;300]	25	101-300 EURO
[301;500]	30	301- 500 EURO
[501;]	35	>500 EURO

Data set **BNPL_Scoring_MostCommonPaymentInstrument**

Matrix		
MostCommonPayment	**Value**	**Description**
N/A	5	
Cash at delivery	5	
Debit Card	10	
Credit Card	15	

Matrix		
MostCommonPayment	**Value**	**Description**
Bank Transfer	10	The hard-coded value is Bank Transfer.
BNPL	20	

Data set **BNPL_SCORING_maxDPD**

Matrix		
maxDPD	**Values**	**Description**
[-1;-1]	5	N/A
[0;0]	25	0 days past due
[1;5]	15	between 1 and 5 DPD
[6;10]	10	between 6 and 10 DPD
(10;15]	5	between 10 and 15 DPD

Data Set **BNPL_Scoring_AdvancePayments**

Matrix		
AdvancePayments	**Values**	**Description**
[-1;-1]	5	N/A
[0;0]	5	Never payed in advance
[1;1]	10	Payed in advance at least once
(1;]	20	Payed in advance more than once The hard-coded value is 2.

Data Set **BNPL_Scoring_UsageRate**

Matrix		
BNPLUsageRate	**Values**	**Description**
[-1;-1]	5	The customer has never paid using Buy Now, Pay Later.
[0;0]	5	
[1;5]	10	The customer has paid using Buy Now, Pay Later 1-5 times.
[6;10]	15	The customer has paid using Buy Now, Pay Later 6-10 times.
[11;20]	20	The hard-coded value is 20. The customer has paid using Buy Now, Pay Later 11-20 times.
[21;40]	25	The customer has paid using Buy Now, Pay Later 21-40 times.

Matrix		
BNPLUsageRate	**Values**	**Description**
(40;]	30	The customer has paid using Buy Now, Pay Later >40 times.

Data set **BNPL_SCORING_returnedProductsPercent**

Matrix		
returnedProductsPercent	**Values**	**Description**
[-1;-1]	5	N/A
[0;5]	20	<5% of returned products
(5;10]	15	5-10% of returned products
(10;25]	10	>10% - 25% of returned products The hard-coded value is 20.
(25;50]	5	>25%- 50% of returned products

Data set **BNPL_SCORING_uniqueCardsUsed**

Matrix		
UniqueCardsUsed	**Values**	**Description**
[-1;-1]	5	N/A
[0;2)	30	<2
[2;3]	20	between 2 - 3 cards The hard-coded value is 3.
[4;5]	10	between 4 -5 cards
(5;]	5	>5 cards

Data set **BNPL_SCORING_CustomerAge**

Matrix		
CustomerAge	**Values**	**Description**
[-1;-1]	5	N/A
[18;25]	5	between 18 and 25 years
(25;35]	10	between 25 and 35 years
(35;50]	15	between 35 and 50 years The hard-coded value is 40.
(50;]	20	over 50 years

BNPL_Limit

It calculates the limit of the customer. If the user score is over 100, the system assigns a limit to that user and progresses to the next calculation Product availability rules. If the user score is below 100, the system does not show any Buy Now, Pay Later products in the Payment screen. If the total check out value is over or equal to the limit, then the system does not trigger the risk calculations and the customer cannot pay with Buy Now, Pay Later.

Field	Example
Name	BNPL_Scoring
Formula Input	BNPL_Scoring
Start date	07/03/2022 17:40

Formula steps:

Name	Master Type	SubType	Calculation Type	Formula
LIMIT_ userscore	Simple Type	Whole Number	Normal	`result = DataSet ("BNPL_Limit", ("UserScore", userScore), ("currencyCode", currencyCode));`

The `currencyCode` is hard-coded to EUR.

Data set **BNPL_Limit**

Matrix		
UserScore	**Values**	**Description**
[45;100]	0	
[101;150]	250	
[151;180]	500	
[181;225]	1000	The hard-coded value is 200.

BNPL_RK_AvailableProducts

This formula calculates the available products.

Field	Example
Name	BNPL_RK_AvailableProducts
Formula Input	BNPL_RK_AvailableProducts
Start date	07/03/2022 12:54

Formula steps:

Name	Description	Master Type	SubType	Calculation Type	Number of iterations	Formula

availableProducts	This formula calculates based on the data set BNPL_Risk_AvailableProducts containing the product code and the score the available products to be offered.	Collection	Text	Iteration	productCodes	`var available = DataSet("BNPL_Risk_AvailableProducts", ("Score", score), ("ProductCode", productCodes[i]), ("CountryCode", countryCode));`
						`if (available == 1) {`
						`result[i] = productCodes[i];`
						`}`
						`else result[i] = null;`

Data set **BNPL_Risk_AvailableProducts** has two data set values depending on the country Germany or France.

BNPL_Risk_AvailableProducts - DE

Score/ Product	BNPL30D	BNPL3X	BNPL4X	BNPL5X	BNPL6X	BNPL7X	BNPL8X	BNPL9X	BNPL10X	BNPL11X	BNPL12X
[101;150]	0	1	0	0	0	0	0	0	0	0	0
[151;180]	0	1	1	1	1	0	0	0	0	0	0
[181;]	0	1	1	1	1	0	0	0	0	0	0
[;100]	0	0	0	0	0	0	0	0	0	0	0

BNPL_Risk_AvailableProducts - FR

Score/ Product	BNPL30D	BNPL3X	BNPL4X	BNPL5X	BNPL6X	BNPL7X	BNPL8X	BNPL9X	BNPL10X	BNPL11X	BNPL12X
[;100]	0	0	0	0	0	0	0	0	0	0	0
[101;150]	1	1	0	0	0	0	0	0	0	0	0
[151;180]	1	1	1	1	1	1	1	0	0	0	0
[181;]	1	1	1	1	1	1	1	1	1	1	1

BNPL_RK_RiskLevel

After the customer selected the product type, the system calculates the risk level based on the total cart value and the product used. Based on the risk level, the system can decide to ask for additional information from the customer so that they can access the Buy Now, Pay Later product.

Field	Example
Name	BNPL_RK_RiskLevel
Formula Input	BNPL_RK_RiskLevel
Start date	07/03/2022 14:56

Formula steps:

Name	Description	Master Type	SubType	Calculation Type	Formula
riskLevel	This formula determines the risk level based on the amount and product code. Each amount range has a risk level.	Simple Type	Text	Normal	`result = DataSet ("BNPL_RK_ RiskLevel", ("amount", amount), ( "productCode" , productCod e));`

Data set **BNPL_RK_RiskLevel**

Name	Matrix	
BNPL_RK_ RiskLevel - BNPL10X	Amount	Risk
	(;300)	1
	[300;800]	2
	(800;)	3
BNPL_RK_ RiskLevel - BNPL11X	Amount	Risk
	(;300)	1
	[300;800]	2
	(800;)	3

Name	Matrix	
BNPL_RK_ RiskLevel - BNPL12X	Amount	Risk
	(;300)	1
	[300;800]	2
	(800;)	3
BNPL_RK_ RiskLevel - BNPL30D	Amount	Risk
	(;100)	1
	[100;300]	2
	(300;)	3
BNPL_RK_ RiskLevel - BNPL3X	Amount	Risk
	(;200)	1
	[200;600]	2
	(600;)	3
BNPL_RK_ RiskLevel - BNPL4X	Amount	Risk
	(;200)	1
	[200;600]	2
	(600;)	3

Name	Matrix	
BNPL_RK_ RiskLevel - BNPL5X	Amount	Risk
	(;200)	1
	[200;600]	2
	(600;)	3
BNPL_RK_ RiskLevel - BNPL6X	Amount	Risk
	(;200)	1
	[200;600]	2
	(600;)	3
BNPL_RK_ RiskLevel - BNPL7X	Amount	Risk
	(;250)	1
	[250;700]	2
	(700;)	3
BNPL_RK_ RiskLevel - BNPL8X	Amount	Risk
	(;250)	1
	[250;700]	2
	(700;)	3

Name	Matrix	
BNPL_RK_ RiskLevel - BNPL9X	Amount	Risk
	(;250)	1
	[250;700]	2
	(700;)	3

BNPL_FinancialCalculation

This formula calculates whether the loan is approved or not.

Field	Example
Name	BNPL_FinancialCalculation
Formula Input	BNPL_FinancialCalculation
Start date	04/04/2022 17:54

Formula steps:

Name	Master Type	Sub Type	Calculation Type	Formula
DTI	Simple Type	Decimal	Normal	`result = expense / income;`
availableDTI	Simple Type	Decimal	Normal	`result = maxDTI - DTI;`
maxInstallment	Simple Type	Decimal	Normal	`if (availableDTI > 0) { result = (decimal) (availableDTI * income); } else result = (decimal)(0);`

Name	Master Type	Sub Type	Calculation Type	Formula
decision	Simple Type	Text	Normal	`if (availableDTI > 0 && maxInstallment >= monthlyInstallment) result = "Approved"; else result = "Rejected";`

BNPL_KO_Risk

This formula determines the risk level of a customer based on age and employment status.

Field	Example
Name	BNPL_KO_Risk
Formula Input	BNPL_KO_Risk
Start date	21/03/2022 09:44

Formula steps:

Name	Master Type	SubType	CalculationType	Exclude From Mapping	Formula
KO_Risk_Age	Simple Type	Boolean	Normal	FALSE	`result = DataSet ("BNPL_KO_Risk_Age", ("age", age)) == 1;`

Name	Master Type	SubType	CalculationType	Exclude From Mapping	Formula
KO_Risk_EmploymentStatus	Simple Type	Boolean	Normal	FALSE	`result = DataSet("BNPL_KO_Risk_EmploymentStatus", ("EmploymentStatus", employmentStatus)) == 1;`
KO_Risk_Decision	Simple Type	Boolean	Normal	FALSE	`result = KO_Risk_Age || KO_Risk_EmploymentStatus;`

Data set **BNPL_KO_Risk_Age**

Age	Values
[;18)	1
[18;]	0

Data set **BNPL_KO_Risk_EmploymentStatus**

EmploymentStatus	Values
Unemployed	1
Student	1
Housewife	1

BNPL_MaxDTI_Risk

This formula determines the value for the risk category based on the user score.

Field	Example
Name	BNPL_MaxDTI_Risk

Field	Example
Formula Input	BNPL_MaxDTI_Risk
Start date	01/04/2022 03:00

Formula steps:

Name	Master Type	Sub Type	Calculation Type	Formula	Exclude From Mapping
MAXDTI_riskCategory	Simple Type	Text	Normal	`result = DataSet("BNPL_Scoring_Risk_Category", ("UserScore", userScore));`	FALSE
MAXDTI	Simple Type	Decimal	Normal	`result = DataSet("BNPL_Scoring_Risk_MaxDTI", ("RiskCategory", MAXDTI_riskCategory));`	FALSE

Data set **BNPL_Scoring_Risk_Category**

User Score	Values
[45;100]	D
[101;150]	C
[151;180]	B
[181;225]	A

Data set **BNPL_Scoring_Risk_MaxDTI**

Risk Category	Values
A	0.4
B	0.3
C	0.2
D	0

BNPL_Scoring_Risk

This formula determines the scoring of a customer based on the age, marital status, employment status, time spent at current employer, DPD, active number of loans, education and user score.

Field	Example
Name	BNPL_Scoring_Risk
Formula Input	BNPL_Scoring_Risk
Start date	01/04/2022 03:00

Formula steps:

Name	Master Type	Sub Type	Calculation Type	Formula	Exclude From Mapping
SCORING_age	Simple Type	Whole Number	Normal	`result = DataSet("BNPL_Scoring_Risk_Age", ("Age", age));`	FALSE
SCORING_ maritalStatus	Simple Type	Whole Number	Normal	`result = DataSet("BNPL_Scoring_Risk_MaritalStatus", ( "MaritalStatus", maritalStatus));`	FALSE

Name	Master Type	Sub Type	Calculation Type	Formula	Exclude From Mapping
SCORING_employmentStatus	Simple Type	Whole Number	Normal	`result = DataSet("BNPL_Scoring_Risk_EmploymentStatus", ("EmploymentStatus", employmentStatus));`	FALSE
SCORING_timeAtCurrEmployer	Simple Type	Whole Number	Normal	`result = DataSet("BNPL_Scoring_Risk_EmploymentStatus", ("EmploymentStatus", employmentStatus));`	FALSE
SCORING_DPD	Simple Type	Whole Number	Normal	`result = DataSet("BNPL_Scoring_Risk_DPD", ("DPD", DPD));`	FALSE
SCORING_activeLoansNo	Simple Type	Whole Number	Normal	`result = DataSet("BNPL_Scoring_Risk_ActiveLoansNo", ("ActiveLoansNo", activeLoansNo));`	FALSE

Name	Master Type	Sub Type	Calculation Type	Formula	Exclude From Mapping
SCORING_education	Simple Type	Whole Number	Normal	`result = DataSet("BNPL_Scoring_Risk_Education", ("Education", education));`	FALSE
userScore	Simple Type	Whole Number	Normal	`result = SCORING_age + SCORING_maritalStatus + SCORING_employmentStatus + SCORING_timeAtCurrEmployer + SCORING_DPD + SCORING_activeLoansNo + SCORING_education;`	FALSE

Data set **BNPL_Scoring_Risk_Age**

Age	Values
[;26)	5
[26;31]	10
[32;41]	15
[42;51]	35
[52;]	30

Data set **BNPL_Scoring_Risk_MaritalStatus**

Marital Status	Values
Married	35
Civil partnership	10
Single	10
Widow	20
Separated	15

Data set **BNPL_Scoring_Risk_EmploymentStatus**

Employment Status	Values
Full-time employed	30
Retired	25
Self employed	15
Management/ Mandate contract	10
Unemployed/ Student/ Housewife	5

Data set **BNPL_Scoring_Risk_ActiveLoansNo**

Active Loans No	Values
[0;0]	30
[1;1]	15
[2;2]	10
[3;]	5
[-1;-1]	5
[-2;-2]	10

Data set **BNPL_Scoring_Risk_TimeSpentAtCurrentEmployer**

Time Spent Current Employer	Values
(10;)	30
(5;10]	10
(2;5]	15
[1;2]	10
(;1)	5

Data set **BNPL_Scoring_Risk_DPD**

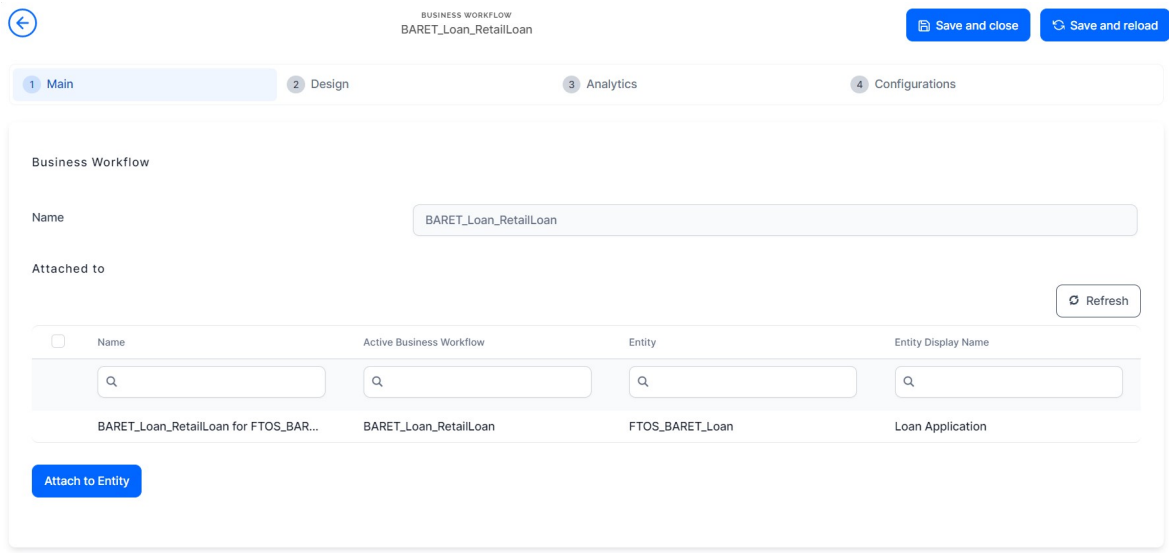
DPD	Values
[0;29]	35
[30;59]	15
[60;89]	10
[90;]	5
[-2;-2]	20
[-1;-1]	5

Data set **BNPL_Scoring_Risk_Education**

Education	Values
University	30
High school	20
Middle school	15
Elementary school	10

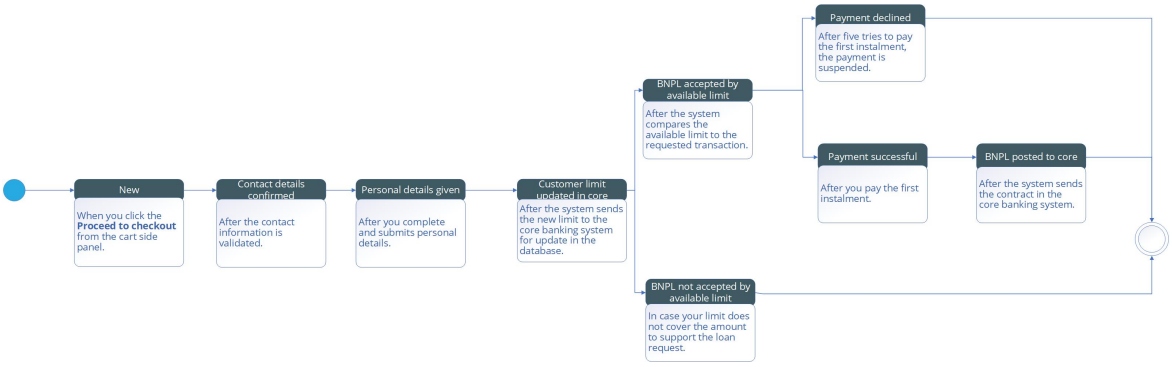
Configuring the Business Workflow

The Business Workflows Processor is used in the Buy Now, Pay Later digital journey for ensuring an easy-to-follow process by using rules-driven business workflows. The workflow is named `BARET_Loan_RetailLoan` attached to the entity FTOS_BARET_Loan. It contains all the states including for other digital journeys such as Mobile Retail Loan Origination and BuyNowPayLater Loan Application.



NOTE
All the statuses are changed using the script "FTOS_BNPL_ChangeLoanApplicationStatus" on page 126.

Status	Description	Flow
New	When you click the **Proceed to checkout** from the cart side panel.	"Click the Proceed to checkout button to continue to the next step. This button triggers the calculations of the formulas: " on page 14
Contact details confirmed	After the contact information is validated.	FTOS_BNKAP_BNPLContactInfo > step Redirect
Personal details given	After you complete and submits personal details.	FTOS_BNKAP_BNPLConfirmDetails > step ConfirmDetails
Customer limit updated in core	After the system sends the new limit to the core banking system for update in the database.	FTOS_BNKAP_BNPLSaveLimit > step SaveLimit
BNPL accepted by available limit	After the system compares the available limit to the requested transaction.	FTOS_BARET_BNPLPayment > step Payment > Advanced > After Events
BNPL not accepted by available limit	In case your limit does not cover the amount to support the loan request.	FTOS_BARET_BNPLPayment_Rejected > step Payment_Rejected
Payment successful	After you pay the first instalment.	FTOS_BARET_BNPLPayment > step Payment > Advanced > After Section Save
Payment declined	After five tries to pay the first instalment, the payment is suspended.	N/A
BNPL posted to core	After the system sends the contract in the core banking system.	FTOS_BARET_BNPLOrderConfirm



To download the diagram, click here.

The transitions are:

- New_Contact details confirmed
- Contact details confirmed_Personal details given
- Personal details given_Customer limit updated in core
- Customer limit updated in core_BNPL accepted by available limit
- Customer limit updated in core_BNPL not accepted by available limit
- BNPL accepted by available limit_Payment declined
- BNPL accepted by available limit_Payment successful
- Payment successful_BNPL posted to core.

Server Automation Scripts and Libraries

Within Innovation Studio, there are bits of code that execute several actions, which embody a business need such as: filtering products, saving session storage, returning products and setting amounts or periods. On-demand automation scripts are available for being called from any object or context.

FTOS_EC_ShoppingCartLoader

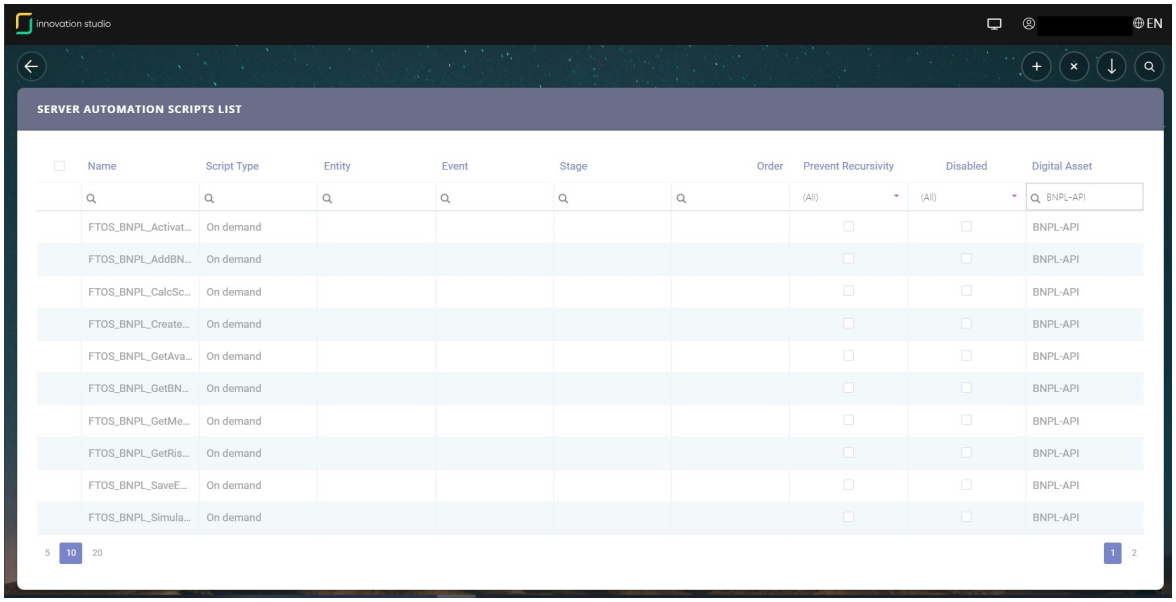
It is the client library where the scoring values are hard-coded, the customer data and the configurations for the **Select a payment** method side panel are. It also calls the script "FTOS_BNPL_SaveCustomerLimit" on page 130 and the script FTOS_EC_CreateOrder, "FTOS_BNPL_AddBNPLApplication" on page 125, FTOS_EC_GetUserShoppingCart and FTOS_EC_UpdateCartItemQuantity.

FTOS_BNPLHelper

The server library named **FTOS_BNPLHelper** gets the records from FTOS_BP_BankingProduct with the status VWApproved and it also gets the payments scheduled type for each banking product. It contains the scripts:

- FTOS_BNPL_GetAvailableProducts
- FTOS_BNPL_SimulateOfferSummary
- FTOS_BNPL_UpdateBNPLApplication.

Server Automation Scripts are:



- FTOS_BNKAP_AuthenticateUser
- FTOS_BNKAP_BNPLUpdatePaymentDetails
- FTOS_BNPL_ActivateContract
- FTOS_BNPL_AddBNPLApplication
- FTOS_BNPL_BCInsert
- FTOS_BNPL_CalcScoringAndLimit
- FTOS_BNPL_ChangeLoanApplicationStatus
- FTOS_BNPL_CreateMerchantBankAccount
- FTOS_BNPL_CreateSystemUser

- FTOS_BNPL_GetAvailableProducts
- FTOS_BNPL_GetBNPLApplicationLink
- FTOS_BNPL_GetMerchantBankAccount
- FTOS_BNPL_GetRiskLevel
- FTOS_BNPL_GetSavedCards
- FTOS_BNPL_GoToShop
- FTOS_BNPL_Risk_CalcScoringAndLimit
- FTOS_BNPL_RunFinancialCalculation
- FTOS_BNPL_SaveCard
- FTOS_BNPL_SaveContract
- FTOS_BNPL_SaveCustomerLimit
- FTOS_BNPL_SaveEarlyRepayment
- FTOS_BNPL_SimulateOfferSummary
- FTOS_BNPL_UpdateBNPLApplication
- FTOS_BNPL_UpdateContract
- FTOS_BNPL_UpdateFinancialData
- FTOS_BNPL_VerifySistemUserEmail.

FTOS_BNKAP_AuthenticateUser

This script sets the URL and the username and password.

Input: let `applicantId`; let `username`; let `password`

Output: token

FTOS_BNKAP_BNPLUpdatePaymentDetails

This script updates the contract number and card number.

Input: const `orderRef`; const `cardNo`; const `contractNo`

Output: `cardNumber`; `contractNo`.

FTOS_BNPL_ActivateContract

This script approves the contract.

Input: let `contractNo`; let `merchantCustomerNo`

Output: let `approvalData`

FTOS_BNPL_AddBNPLApplication

This script inserts a new record in the entities: FTOS_BARET_Loan, FTOS_BNKAP_RetailApplicantAddress, FTOS_BNKAP_RetailApplicantData, FTOS_BNKAP_Application, FTOS_BNKAP_RetailFinancialData, then it generates the `loanNo`. It also saves the session in the entities FTOS_BNKAP_RetailApplicantData and FTOS_BNKAP_RetailApplicantData.

Input: let `loanId`; let `addressId`; let `applicantId`; let `applicationId`; let `financialDataId`

Output: `response`

FTOS_BNPL_BCInsert

This script updates the `activeLoansNo: 1`, `worstDeliquency: 0`.

Input: var `rfd`

Output: `activeLoansNo`, `worstDeliquency`

FTOS_BNPL_CalcScoringAndLimit

This script runs the formulas "BNPL_KO" on page 95 and "BNPL_Limit" on page 106 and "BNPL_Scoring" on page 98. From the result of the BNPL_KO formula, if the:

- customerLoyalty is 'N/A', set it to -1, if not turn the value from customerLoyalty to an integer
- averageCheckoutTicketSize is 'N/A', set it to -1, if not turn the value from averageCheckoutTicketSize to an integer
- maxDPD is "N/A", set it to -1, if not turn the value from maxDPD to an integer
- payedInAdvance is 'N/A', set it to -1, if not turn the value from maxDPD to an integer
- BNPLUsageRate is 'N/A', set it to -1, if not turn the value from maxDPD to an integer
- returnedProductsPercent is 'N/A' , set it to -1, if not turn the value from maxDPD to an integer
- uniqueCardsNo is 'N/A' , set it to -1, if not turn the value from maxDPD to an integer
- customerAge =is 'N/A' , set it to -1, if not turn the value from maxDPD to an integer.

Input: var `KOInput`; var `KOFormula`

Output: var `res`

FTOS_BNPL_ChangeLoanApplicationStatus

This script changes the status of the loan within the entity FTOS_BARET_Loan.

Input: var `loanId`; var `statusName`

Output: boolean

FTOS_BNPL_CreateMerchantBankAccount

This script saves the EUR bank account of the merchant. It sets the `customerNo: customerNo`, `bankAccountNo: bankAccountNo`.

Input: let `merchantName`; let `fiscalRegistrationNo`; let `commercialRegistration`

Output: `customerNo`; `bankAccountNo`

FTOS_BNPL_CreateSystemUser

This script creates a user within the FintechOS environment for each customer.

Input: let `password`; let `applicantDataId`

Output: updates the Account with the values for: `cbURL`, `tokenCB`, `usernameCB`, `customerNo`, `applicantData`.

FTOS_BNPL_GetAvailableProducts

This script runs the formula "BNPL_RK_AvailableProducts" on page 107, it then turns into a string the result of the formula and parses it. Row by row from the formula result, for each code it adds `resultItem` to `result`.

Input: let `score`; let `productCodes`; let `countryCode`

Output: `result`

FTOS_BNPL_GetBNPLApplicationLink

This script generates the B2C link for each loan based on the `sessionID` and `loanID` to be accessed after the marketplace visit.

Input: let `loanId`; let `sessionId`

Output: the link to the B2C

FTOS_BNPL_GetMerchantBankAccount

This script gets the bank account of the e-commerce website.

Input: let `merchantName`; let `currencyCode`; let `token`

Output: let `bankAccountData`

FTOS_BNPL_GetRiskLevel

This script get the amount of the loan and the product code and runs the formula "BNPL_RK_RiskLevel" on page 108.

Input: let `amount`; let `productCode`

Output: `formulaResult`

FTOS_BNPL_GetSavedCards

This script is use to receive the credit cards of the customer from the entity FTOS_MOCK_CreditCard.

Input: let `ref`

Output: `result`

FTOS_BNPL_GoToShop

This script gets the link to the e-commerce.

Input: N/A

Output: let `link`

FTOS_BNPL_Risk_CalcScoringAndLimit

This script gets the result from the formula "BNPL_KO_Risk" on page 113. If the result is `Passed`, then the script calculates the time of employment, runs the "BNPL_Scoring_Risk" on page 116 formula and the "BNPL_MaxDTI_Risk" on page 114. It is used in the "FTOS_BNKAP_BNPLCompleteAccountScoring" on page 42.

Input: var `retailApplicantId`

Output: `res` (it contains the ko, riskCategory, maxDTI, userScore.

FTOS_BNPL_RunFinancialCalculation

This script runs the formula "BNPL_FinancialCalculation" on page 112.

Input: let `loanId`

Output: let `result`

FTOS_BNPL_SaveCard

This script saved the credit card information in the entity FTOS_MOCK_CreditCard.

Input: let `cardNo`; let `cardHolder`; let `expDate`; let `cvc`; let `ref`

Output:

```
CardNumber: cardNo,
CardHolderName: cardHolder,
ExpirationDate: expDate,
CVC: cvc,
AccountRef: ref
```

FTOS_BNPL_SaveContract

This script sets the data from the contract:

- contractId
- customerNo
- productCode
- startDate
- totalInterestRate
- contractPeriod
- firstDueDate
- maturityDate
- bankAccountNo

- destinationBankAccount
- amountDue
- productInterest
- paymentScheduleType
- periodicityType
- installmentDay
- installmentMethod
- reviewInterestDate
- maxDisburseDate
- managingBranch.

Input: let `loanId`

Output: let `contractNo`

FTOS_BNPL_SaveCustomerLimit

This script saves the customer data along with the loan data and calculates the limit. Depending on the limit, the customer is approved or denied.

Input: let `retailApplicantId`

Output: N/A

FTOS_BNPL_SaveEarlyRepayment

This script gets the contract data (updates the amount within the contract) and the bank account number, as well as the top up event value for the contract (the first payment and the commission) and the event value of the early repayment reference.

Input: let `contractNo`; let `eventValue`

Output: N/A

FTOS_BNPL_SimulateOfferSummary

This script returns the banking product based on the amount and code. It also calculates the advance

Input: let `productAmount`; let `bankingProductCode`

Output: Boolean

FTOS_BNKAP_BNPLUpdatePaymentDetails

This script updates the payment details in the entity FTOS_EC_Payment.

Input: const `orderRef`; const `cardNo`; const `contractNo`

Output: object IFtosScriptableEbsExecutionContext

FTOS_BNPL_UpdateBNPLApplication

This script updates the data in the entities: FTOS_BNKAP_ RetailApplicantData, FTOS_BNKAP_RetailFinancialData and FTOS_BNKAP_ RetailApplicantAddress based on the entity name and id.

Input: let `data`

Output: `data`

FTOS_BNPL_UpdateContract

This script updates the data (advance value) within the loan contract.

Input: let `contractNo`; let `updateData`;

Output: `contractData`

FTOS_BNPL_UpdateFinancialData

This script gets the income and expenses and updates the values in the entity FTOS_BNKAP_RetailFinancialData.

Input: let `applicantId`; let `income`; let `expense`

Output: `mergedIncome: income`, `totalMonthlyPayments: expense`

FTOS_BNPL_VerifySistemUserEmail

This script checks if the user existing in the system by checking the email address.

Input: let `email`

Output: let `result`

FTOS_BARET_AddFeedback

This script inserts the feedback in the entity FTOS_BNKAP_Feedback.

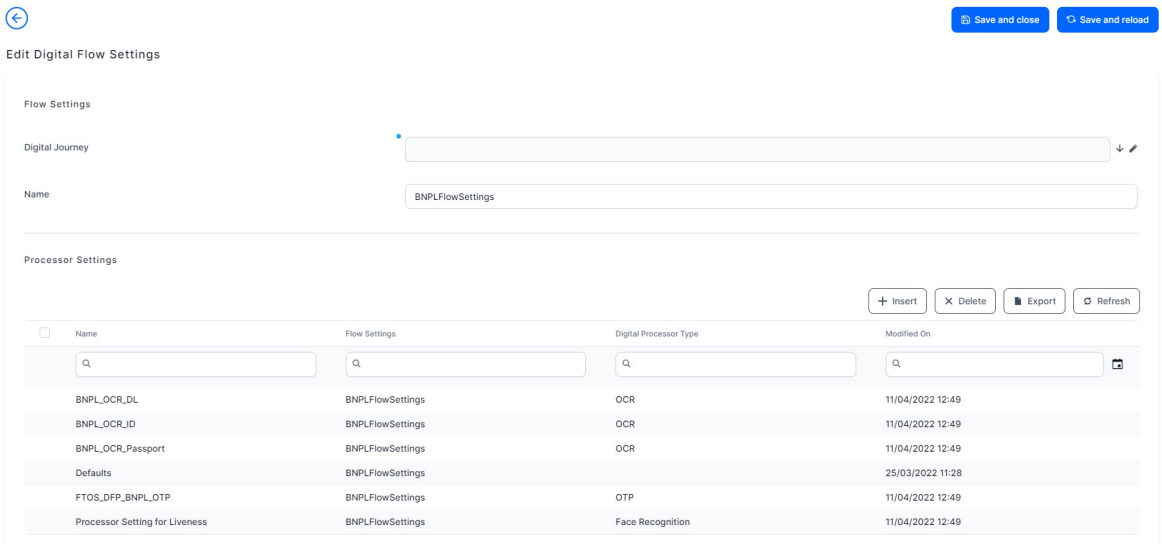
Input: var `rating`; var `sourceId`; var `applicationId`

Output: string

Configuring the Flow Settings

This menu part of Innovation Studio is used to build processors used in the journey for specific actions that are triggered such as Face Recognition and Computer Vision and contact validation. It is easy to modify any parameter of the processor by accessing the dedicated processor settings and changing the value from the key-value pair.

Access Innovation Studio > Digital Experience > Digital Journeys > Processor Settings > BNPLFlowSettings.



Flow Setting Name	Records	Type of Processor	Description
BNPLFlowSettings	BNPL_OCR_DL	OCR	These are the settings for the driver's license within the step "Proof of ID" on page 29.
	BNPL_OCR_ID	OCR	These are the settings for the personal ID card scan.
	BNPL_OCR_Passport	OCR	These are the settings for the passport scan.
	Defaults		This code contains the standard currency for the solution and the minimum age to apply.
	FTOS_DFP_BNPL_OTP	OTP	These are the settings for the sending of the one-time-password within the step "Contact Info" on page 17.
	Processor Setting for Liveness	Face Recognition	These are the settings for the selfie step "Liveness Check" on page 31.

BNPL_OCR_DL

The following are settings for the driving license scan:

- the source entity
- the file attribute name
- the upload button
- session expiration
- whether to crop the photo
- maximum number of retry the scanning
- the country of the license
- the provider
- the document type.

```
 "SourceEntityName": "RetailApplicantData",
        "Entities": [
             {
                "DestinationEntityName":
"RetailApplicantData",
                "SourceEntityName": "RetailApplicantData",
                "SourceLookupDestinationName":
"RetailApplicantDataid"
             }
        ],
             "FileAttributeName": "pictureOcr",
                "WaitUserConfirmation" : true,
                    "ShowUploadPhotoButton" : true,
                        "ShowTakePictureButton" : true,
                            "RegisterFaceFromOCR" : true,
                                "SessionExpiredMins" : 15,
                                    "RotateImage" : false,
                                        "CropImage" : false,
                                            "MaxRetry": 5,

"OptionSets": [{

"OptionSetName": "Gender Type",

"MappingName": "Sex",
```

```
"OptionSetItems": {
                                                    "M":
"Male",
                                                    "F":
"Female"
                                                }
                                            }, {
"OptionSetName": "Citizenship",
"MappingName": "Nationality",
"OptionSetItems": {
"ROU": "Romanian",
"GBR": "UK"
                                                }
                                            }],
"LookupEntities": [{
"MappingName": "DistrictCode",
"EntityName": "District",
"AttributeKey": "Code"
                                                }, {
"MappingName": "Country",
"EntityName": "Country",
"AttributeKey": "code"
                                                }],
"Validations": [],
"AvailableDocumentTypes": [{
"type": "IdRom",
"DocumentType": "IdentityCard",
"Country": "RO",
```

```
"Provider": "Azure"

 }, {

"type": "IdBG",

"DocumentType": "IdentityCard",

"Country": "BG",

"Provider": "Abbyy"

 }, {

"type": "Passport",

"DocumentType": "MRZ",

"Provider": "Azure"

 }, {

"type": "DrivingLicence",

"DocumentType": "DrivingLicence",

"Provider": "Azure"

 }],

"maskNextStepURLSuccess": {
        "entity": "RetailApplicantData",
            "form": "FTOS_BNPLScanOCR",
                "section": "redirect"
    },
    "maskNextStepURLFail": {
        "entity": "RetailApplicantData",
            "form": "FTOS_BNKAP_BNPLScanOCR",
                "section": "ocrFailed"
    },
    "DocumentType": "DrivingLicence"
}
```


BNPL_OCR_ID

-> The following are settings for the identity document:

- the source entity
- the file attribute name
- the upload button
- session expiration
- whether to crop the photo
- maximum number of retry the scanning
- the country of the license
- the provider
- the document type.

Settings

```
{
    "SourceEntityName": "RetailApplicantData",
        "Entities": [
            {
                "DestinationEntityName":
"RetailApplicantData",
                "SourceEntityName": "RetailApplicantData",
                "SourceLookupDestinationName":
"RetailApplicantDataid"
            }
        ],
            "FileAttributeName": "pictureOcr",
                "WaitUserConfirmation" : true,
                    "ShowUploadPhotoButton" : true,
                        "ShowTakePictureButton" : true,
                            "RegisterFaceFromOCR" : true,
                                "SessionExpiredMins" : 15,
                                    "RotateImage" : false,
                                        "CropImage" : false,
                                            "MaxRetry": 5,

"OptionSets": [{
```

```
"OptionSetName": "Gender Type",
"MappingName": "Sex",
"OptionSetItems": {
                                                    "M":
"Male",
                                                    "F":
"Female"
                                                }
                                            }, {
"OptionSetName": "Citizenship",
"MappingName": "Nationality",
"OptionSetItems": {
"ROU": "Romanian",
"GBR": "UK"
                                                }
                                            }],
"LookupEntities": [],
"Validations": [{
"type": "IdROM",
"Validations": "",
"CheckScriptName": "ValidateIdROM"
                                                    }],
"AvailableDocumentTypes": [{
"type": "IdRom",
"DocumentType": "IdentityCard",
"Country": "RO",
"Provider": "Azure"
```

```
 }, {
"type": "IdBG",
"DocumentType": "IdentityCard",
"Country": "BG",
"Provider": "Abbyy"
 }, {
"type": "Passport",
"DocumentType": "MRZ",
"Provider": "Azure"
 }, {
"type": "DrivingLicence",
"DocumentType": "DrivingLicence",
"Provider": "Azure"
 }],
"maskNextStepURLSuccess": {
        "entity": "RetailApplicantData",
            "form": "FTOS_BNKAP_BNPLScanOCR",
                "section": "redirect"
    },
    "maskNextStepURLFail": {
        "entity": "RetailApplicantData",
            "form": "FTOS_BNKAP_BNPLScanOCR",
                "section": "ocrFailed"
    },
    "DocumentType": "IdRom"
}
```

Mapping

```
{
    "DocumentsMapping": [{
        "type": "IdRom",
        "Map": {
            "PictureAttribute": "pictureOcr",
            "LastName": "lastName",
            "GivenName": "firstName",
            "DocumentNumber": "IdCardSeries",
            "StreetType": "streetType",
            "PersonalNumber": "PIN",
            "BirthDate": "dateOfBirth",
            "PlaceOfBirthBody": "placeOfBirth",
            "Address": "fullAddress",
            "Sex": "gender",
            "Nationality": "citizenshipId",
            "Country" : "country",
            "DistrictCode": "districtId",
            "City": "city",
            "Street": "street",
            "StreetNo": "streetNo",
            "Storey": "floor",
            "Stairway": "stairway",
            "ApartmentNo": "apartment",
            "ApHouse": "buildingNo",
            "IssuedBy": "IdIssueInstitution",
            "IssuedAt": "IdIssueDate",
            "IssuedUntil": "IdExpirationDate",
            "IssuingCountry": "cityName"
        }
    }]
}
```

BNPL_OCR_Passport

These are the setting for scanning the passport:

- the source entity
- the file attribute name
- the upload button

- session expiration
- whether to crop the photo
- maximum number of retry the scanning
- the country of the license
- the provider
- the document type.

Settings

```
{
    "SourceEntityName": "RetailApplicantData",
        "Entities": [
            {
                "DestinationEntityName":
"RetailApplicantData",
                "SourceEntityName": "RetailApplicantData",
                "SourceLookupDestinationName":
"RetailApplicantDataid"
            }
        ],
            "FileAttributeName": "pictureOcr",
                "WaitUserConfirmation" : true,
                    "ShowUploadPhotoButton" : true,
                        "ShowTakePictureButton" : true,
                            "RegisterFaceFromOCR" : true,
                                "SessionExpiredMins" : 15,
                                    "RotateImage" : false,
                                        "CropImage" : false,
                                            "MaxRetry": 5,

"OptionSets": [{

"OptionSetName": "Gender Type",

"MappingName": "Sex",

"OptionSetItems": {
                                                                "M":
"Male",
```

```
                                                                "F":
"Female"
                                                                    }
                                                            }, {
"OptionSetName": "Citizenship",
"MappingName": "Nationality",
"OptionSetItems": {
"ROU": "Romanian",
"GBR": "UK"
                                                                    }
                                                            }],
"LookupEntities": [{
"MappingName": "DistrictCode",
"EntityName": "District",
"AttributeKey": "Code"
                                                                }, {
"MappingName": "Country",
"EntityName": "FTOS_CMB_Country",
"AttributeKey": "code"
                                                                }],
"Validations": [],
"AvailableDocumentTypes": [{
"type": "IdRom",
"DocumentType": "IdentityCard",
"Country": "RO",
"Provider": "Azure"
 }, {
```

```
"type": "IdBG",

"DocumentType": "IdentityCard",

"Country": "BG",

"Provider": "Abbyy"

 }, {

"type": "Passport",

"DocumentType": "MRZ",

"Provider": "Azure"

 }, {

"type": "DrivingLicence",

"DocumentType": "DrivingLicence",

"Provider": "Azure"

 }],

"maskNextStepURLSuccess": {
        "entity": "RetailApplicantData",
            "form": "FTOS_BNKAP_BNPLScanOCR",
                "section": "redirect"
    },
    "maskNextStepURLFail": {
        "entity": "RetailApplicantData",
            "form": "FTOS_BNKAP_BNPLScanOCR",
                "section": "ocrFailed"
    },
    "DocumentType": "Passport"
}
```

```
{
    "DocumentsMapping": [{
        "type": "Passport",
        "Map": {
```

```
            "PictureAttribute": "pictureOcr",
            "LastName": "lastName",
            "GivenName": "firstName",
            "DocumentNumber": "IdCardSeries",
            "StreetType": "streetType",
            "PersonalNumber": "PIN",
            "BirthDate": "dateOfBirth",
            "PlaceOfBirthBody": "placeOfBirth",
            "BirthCountryBody": "birthCountry",
            "Address": "fullAddress",
            "Sex": "gender",
            "DistrictCode": "DistrictId",
            "Nationality": "citizenshipId",
            "City": "CityId",
            "Street": "StreetName",
            "StreetNo": "StreetNo",
            "Storey": "FloorNo",
            "Stairway": "Stairway",
            "ApartmentNo": "ApartmentNo",
            "ApHouse": "BuildingNo",
            "IssuedBy": "IdIssueInstitution",
            "IssuedAt": "IdIssueDate",
            "IssuedUntil": "IdExpirationDate",
            "IssuingCountry": "issuingCountry"
        }
    }]
}
```

Defaults

This code contains the standard currency for the solution and the minimum age to apply.

Settings

```
{
    "defaultCurrencySymbol": "€",
    "minAge": 18
}
```

FTOS_DFP_BNPL_OTP

These settings contain the key value pairs for the One-Time-Password. You can modify value for:

- source entity
- the name of the channel `sms` or `email`
- the number of digits of the code
- the maximum number of retries
- the maximum number the system sends you the code
- the maximum interval between sending you the code
- in case the OTP is inserted correctly/wrong where to navigate next: the entity, the entity form and the step as well as to set the status from the business workflow for both cases.

Settings

```
{
    "SourceEntityName" : "RetailApplicantData",
        "Channels": [
            {
                "Name": "Sms",
                "RelatedEntityName": "RetailApplicantData",
                "RelatedAttribute": "mobilePhone",
                "LookupAttribute": "RetailApplicantDataid",
                "OTPDigitsNumber": 4,
                "MaxRetry": 5,
                "MaxResendRetry": 5,
                "MaxResendRetryIntervalSeconds": 60
            },
            {
                "Name": "Email",
                "RelatedEntityName": "RetailApplicantData",
                "RelatedAttribute": "email",
                "LookupAttribute": "RetailApplicantDataid",
                "OTPDigitsNumber": 4,
                "MaxRetry": 5,
                "MaxResendRetry": 5,
                "MaxResendRetryIntervalSeconds": 120
            }
```

```
        ],
            "ValidTimeIntervalMinutes": 3,
                "maskNextStepURLChanged": {
        "entity": "FTOS_BNKAP_RetailApplicantData",
            "form": "FTOS_BNKAP_BNPLContactInfo",
                "section": "ContactInfo"
    },
    "maskNextStepURLSuccess": {
        "entity": "FTOS_BNKAP_RetailApplicantData",
            "form": "FTOS_BNKAP_BNPLContactInfo",
                "section": "Redirect"
    },
    "maskNextStepURLFail": {
        "entity": "FTOS_BNKAP_RetailApplicantData",
            "form": "FTOS_BNKAP_BNPLContactInfo",
                "section": "Fail"
    },
    "businessWorkflow": {
        "entity": "FTOS_BASME_Loan",
            "bwAttribute": "corporateLoanId",
                "successStatus": "",
                    "failedStatus": ""
    }
}
```

Processor Setting for Liveness

`isLiveness` marks the usage of the Liveness feature to detect the face of a human being with the use of a camera either smart-phone, webcam or any incorporated camera. The maximum number or retries is five. In case of success, the user is directed to the form driven flow named FTOS_BARET_BNPLSelfie, the step Success, in case of failure to the form driven flow FTOS_BARET_BNPLSelfie the step Fail. The minimum acceptance score can be 0.1 and the maximum is 1.

```
{
    "isLiveness": true,
    "DestinationEntityName": "RetailApplicantData",
    "SourceEntityName": "RetailApplicantData",
    "SourceLookupDestinationName": "RetailApplicantDataid",
    "FileAttributeName": "pictureOcr",
    "MaxRetry": 5,
```

```
    "MinimumAcceptedConfidence": 0.2,
    "maskNextStepURLSuccess":
{"entity":"RetailApplicantData", "form": "FTOS_BARET_
BNPLSelfie","section":"Success" },
    "maskNextStepURLFail": {"entity":"RetailApplicantData",
"form": "FTOS_BARET_BNPLSelfie","section":"Fail" },
    "businessStatusSuccess": "",
    "businessStatusFail": ""
}
```

Mapping

```
{
    "Confidence": "confidence"
}
```

Glossary

A

ANAF/NAFA

Agenția Națională de Administrare Fiscală (ANAF) or National Agency of Fiscal Administration (NAFA).
It is the specialized body of the central public administration in Romania. It has its own legal persona, by detaching the directions with attributions in the administration of the state revenues within the Ministry of Public Finance. It it ensures the resources for the public expenses of the state by collecting and administering effectively and efficiently the taxes, taxes, contributions and other amounts due to the general consolidated budget.

C

Credit Kudos

Credit Kudos' intelligent products enable businesses to leverage Open Banking to enhance affordability and risk assessments. Their products help lenders streamline underwriting, improve accuracy in decision-making, and support customers after acquisition through our engagement tools.

CVC

Card Verification Code. It is located on the back of your credit/debit card.

D

Debt-to-Income (DTI)

The Debt-to-Income ratio (DTI) is a personal indicator of a good balance between debt and income detremined by essential expenditure/ income with bank transfers.

F

Financial institution (FI)

A financial institution (FI) is a company that manages financial and monetary transactions such as deposits, loans, investments, and currency exchange, i.e business operations within the financial services sector including banks, trust companies, insurance companies, brokerage firms, and investment dealers.

G

GDPR

The General Data Protection Regulation is a regulation in EU law on data protection and privacy in the European Union and the European Economic Area.

I

IBAN

International Bank Account Number

Interest rate

The interest rate is the amount a lender charges a borrower and is a percentage of the principal, i.e., the amount loaned.

M

Marketing Consent

It is the practice of only contacting consumers that have given their prior express written consent to be contacted.

Monthly instalment

The value of one of a number of successive payments in settlement of a debt.

O

One-Time-Password (OTP)

It is a password generated by the system to validate a piece of information or sign a document.

Open Banking

Open banking connects banks, third-parties and technical providers to securely exchange data.

T

Total commissions

The value of the total amount of commissions paid for the entire loan duration.

Total interest

It is the total amount paid for the entire loan duration as interest.

Total principal

It is the total amount paid for the entire loan duration.

Total repayable amount

It represents the sum of all scheduled or projected payments of funds that the recipient agrees to pay to the provider.