

fintech 

Onfido 22.1.0000

User Guide

TOC

Overview	3
Benefits	3
Applications	4
Onfido Installation	5
Prerequisites	5
1 Install the SysPacks	5
2 Set up the Onfido service subscription key	6
Setting Up an Onfido Automation Processor	8
1 Create a digital flow settings group	8
2 Add the Onfido processor settings to a digital flow settings group	9
Onfido Settings	9
Example	14
Onfido Mappings	16
Example	17
Onfido UI Customization	19
Onfido Request Responses	21
Example	21
Getting Status Changes Notifications Using Webhooks	30
Triggering Identity Verification in a Digital Journey	31

Overview

Onfido helps you validate a user's identity by checking the authenticity of the submitted identity document. The solution scans the ID front and back to validate the name introduced by the user with data extracted from the document. If, for example, there's a middle name on the document that's not specified by the user, then the data doesn't match and the check can fail. In addition to the data extracted from the document scans Onfido retrieves the breakdowns and validations that depend on each configuration. This information is stored in the `FTOS_ONFIDO_CHECKREPORT` defined in JSON format.

Apart from identity documents validation, the Onfido automation processor offers face similarity checks by comparing a clear photo or a video provided by the applicant to their identity document. Using the face similarity functionality, it quickly verifies that the applicant is a real person and they are the same as the person from the identity document.

IMPORTANT!

The data that Onfido handles must be GDPR compliant. To delete data from Onfido, you must do so manually from the Onfido account.

Benefits

- **Digitization.** Validate your users' digital accounts based on their real identities.
- **Customer Acquisition.** Streamline your customer onboarding process with a simple user experience.
- **Fraud Detection.** Prevent identity fraud with document verification.
- **Security.** Manual checks for warning cases on top of automated validations.
- **Compliance.** Implement KYC and AML requirements at scale.

Applications

- Customer onboarding
- Account opening
- Loan applications
- Compliance related processes
- Claims handling
- Mortgage processing

Onfido Installation

Prerequisites

Before installation, you need:

- An active FintechOS account. For details, contact a FintechOS sales representative.
- Make sure you have the *dcs-sdk-onfido.js* installed on your environment in your custom folder. For details, contact a FintechOS sales representative.

1 Install the SysPacks

Make sure you have the **SysPacks v.22.1.0000** installed on your system. To do so:

1. Using a web browser, log in to your [FintechOS Community](#) account.
2. Select the **Release Hub**.
3. Open the **FintechOS 22.R2** release.
4. Open the **HPFI** folder.
5. Download the **SySDigitalSolutionPackages v22.1.0000.zip** archive.
6. Unzip the archive and follow the instructions in the **How to install FintechOS SysPacks v21.2.2000.pdf** file to install the SysPacks.

2 Set up the Onfido service subscription key

IMPORTANT!

Make sure that the Vault configurations are done and that the [Webhook](#) is created.

In Vault

Key Path	Key Name	Value
kv/<environment>/<Fintech OS Portal instance>/app-settings	FTOSServiceOnfidoEndpoint	DCS web app endpoint URL provided by FintechOS.
kv/<environment>/<Fintech OS Portal instance>/app-settings	FTOSServicesOnfidoAppId	ID for the Onfido service subscription.
kv/<environment>/<Fintech OS Portal instance>/app-settings	FTOSServicesOnfidoSubscriptionKey	Subscription key for the Onfido service.

DCS Onfido configurations are stored using **Vault** secrets inside a KV (key-value) Service, using the following structure path: `kv/dcs/kyc-onfido`.

(Deprecated) Adding Keys In web.config

Key	Value
FTOSServiceOnfidoEndpoint	DCS web app endpoint URL provided by FintechOS.
FTOSServicesOnfidoAppId	ID for the Onfido service subscription.
FTOSServicesOnfidoSubscriptionKey	Subscription key for the Onfido service.

```
<add key="FTOSServicesOnfidoEndpoint" value="https://"/>
```

```
<add key="FTOSServicesOnfidoAppId" value=""/>  
<add key="FTOSServicesOnfidoSubscriptionKey" value=""/>
```

Setting Up an Onfido Automation Processor

1 Create a digital flow settings group

The Onfido automation processor must be hosted inside a digital flow settings group. A digital flow settings group can include multiple automation processors and is typically used as a container for the automation processors called by a specific digital journey.

If you already have a digital flow settings group you wish to host your Onfido automation processor, skip to "[2 Add the Onfido processor settings to a digital flow settings group](#)" on the next page. Otherwise, follow the instructions below to create a new digital flow settings group:

1. In Innovation Studio, go to **Main Menu > Digital Experience > Digital Journeys > Processor Settings**.
2. In the Digital Flow Settings list page, click the **Insert** button at the top right corner to add a new digital flow settings group.
3. In the Add Digital Flow Settings window, enter a **Name** for your digital flow settings group.
4. If you already have a digital journey set up where you wish to call the Onfido automation processor, select it from the Digital Journey drop-down box.
5. Click the **Save and Close** button at the top right corner to save your flow settings group.

2 Add the Onfido processor settings to a digital flow settings group

1. In Innovation Studio, **Main Menu > Digital Experience > Digital Journeys > Processor Settings**.
2. In the Digital Flow Settings list, double click the desired digital flow settings group.
3. In the Processor Settings section, click the **Insert** button.
4. Fill in the following fields:
 - Name – Enter a name for your processor settings.
 - Flow Settings – Leave the default value.
 - Digital Processor Type – Select **Onfido**.
 - Settings – JSON code for the automation processor's settings. For details, see ["Onfido Settings" below](#).
 - Mapping – JSON code for the automation processor's mappings. For details, see ["Onfido Mappings" on page 16](#).
5. Click **Save and Close** at the top right corner of the screen.

Onfido Settings

The Onfido settings are defined in JSON format as key-value pairs. The following settings are available:

JSON Key	Description
SourceEntityName	The entity associated with the digital journey that calls the Onfido processor. Needed only if the Onfido processor is used on an edit form (to alter an existing record) to update the workflow entity's business status after the scan. If the Onfido processor is used on an insert form (to create a new record), this key is not needed.
UseProcessorForOCR	Boolean value. When set to <code>true</code> , an OCR request is sent to DCS.
UseProcessorForCheck	Boolean value. When set to <code>true</code> , the raw identity check reports received from the Onfido service are attached to the FTOS_ONFIDO_CHECK entity (in the FTOS_ONFIDO_CHECKREPORT related entity).
UploadFallback	Boolean value - default: <code>false</code> . When set to <code>true</code> , an optimized camera UI is used to take a live photo of the identity document. When this is not possible (because of an unsupported browser or mobile device), it falls back to the mobile device's default camera application. <div style="background-color: #f9c796; padding: 10px; border: 1px solid #ccc;"> <p>IMPORTANT! This method does not guarantee live capture, because some mobile device browsers and camera applications allow uploads from the user's gallery of photos.</p> </div>
OnfidoCountry	The identity document's issuing country specified using a 3-letter ISO 3166-1 alpha-3 country code.
OnfidoDocumentType	An array with desired document types. Allowed values are: "national_identity_card", "passport", "driving_licence", and "residence_permit".
Referrer	The URL of the web page where the Web SDK is used. The referrer sent by the browser must match the referrer URL pattern in the SDK token for the SDK to successfully authenticate.
DocumentHasTwoSides	Boolean value indicating if the user must scan both sides of the identity document.
UseDefaultValues	Boolean value. When set to <code>true</code> , the processor creates a new Onfido applicant with the information defined in "DefaultApplicantData" on the next page .
UseFaceSimilarity	Boolean value. When set to <code>true</code> , the Face Similarity Type field is displayed to validate face similarity.

JSON Key	Description
FaceSimilarityType	<p>Displayed if the Use Face Similarity field is set to <i>true</i>. The following options are available</p> <ul style="list-style-type: none"> • Standard, in which you take a simple selfie. • Video, in which you need to make the requested moves for validation.
DefaultApplicantData	<p>JSON object containing key-value pairs for the default Onfido applicant's data.</p>
ApplicantData	<p>The attributes from which the Onfido applicant's data is retrieved.</p>
Entities	<p>Indicates the entities that populates with the scanned values.</p> <ul style="list-style-type: none"> • DestinationEntityName – Name of the entity that is populated with the scanned data. • SourceEntityName – Entity associated with the digital journey that uses the automation processor. If this is the same entity that is populated, this value is identical to DestinationEntityName. • SourceLookupDestinationName – Name of the SourceEntityName lookup key that points to DestinationEntityName. If they are the same entity, enter the primary key.

JSON Key	Description
OptionSets	<p>Replaces values returned by the Onfido processor with entries from an option set.</p> <ul style="list-style-type: none"> • OptionSetName – Name of the option set that stores the valid replacement values. • MappingName – Name of the field as returned by the Onfido processor. • OptionSetItems – Key-value pairs that map the value returned by the Onfido processor (the key) to the replacement value from the option set (the value). <p>If the value returned by the Onfido processor is not found in the OptionSetItems keys, the entry is not be populated. The user is able to manually select only values from OptionSetName.</p>

JSON Key	Description
LookupEntities	<p>Validates values returned by the Onfido processor based on records in an entity.</p> <ul style="list-style-type: none"> • MappingName – Name of the field as returned by the Onfido processor. • EntityName – Name of the entity that stores the valid values. • AttributeKey – Name of the attribute in EntityName that stores the valid values. <p>Parent – Defines hierarchical relationships between lookup entities. For instance, you can check if a city name belongs to a valid county name.</p> <ul style="list-style-type: none"> • MappingParentName – Name of the parent field as returned by the Onfido processor. • AttributeParentKey – Name of the EntityName lookup key linked to the parent entity. <p>If the value returned by the Onfido processor is not found in the lookup entity, the entry does not populate. The user is able to manually select only values from the lookup entity.</p>
MaskNextStepUrlSuccess	<p>Location in the user interface where the user is redirected after a successful check.</p> <ul style="list-style-type: none"> • entity – Entity name. • form – Form name of the above entity. • section – Optional parameter for the section name of the above form.

JSON Key	Description
MaskNextStepUrlFail	<p>Location in the user interface where the user is redirected after a failed check.</p> <ul style="list-style-type: none"> entity – Entity name. form – Form name of the above entity. section – Optional parameter for the section name of the above form.
UseLocalization	<p>Boolean value. Setting it to <i>true</i> enables user interface localization. Localization uses the FTOS_DFP_OnfidoLocalization client script library to populate the interface text fields. By default, only the en-US locale is defined. Edit the library to add additional languages.</p>
DocumentType	<p>The document type that identifies the set of attribute mappings used by the processor (see "Onfido Mappings" on page 16). Only one document type can be enabled per automation processor.</p>

Example

```
{
  "SourceEntityName": "FTOS_BARET_AccountApplication",
  "UseProcessorForOCR": true,
  "UseProcessorForCheck": true,
  "UploadFallback": true,
  "OnfidoCountry": "ROU",
  "OnfidoDocumentType": ["national_identity_card"],
  "Referrer": "https://myOnfidoPortal.net/",
  "DocumentHasTwoSides": false,
  "UseDefaultValues": true,
  "UseFaceSimilarity": true,
  "FaceSimilarityType": "standard" / "video",
  "DefaultApplicantData": {
    "LastName": "DefaultLastName",
    "FirstName": "DefaultFirstName",
    "Email": "DefaultEmail@email.com"
  },
  "ApplicantData": {
    "LastName": {
      "AttributeName": "retailApplicantId_lastName",

```

```

        "Extension": "retailApplicantData"
    },
    "FirstName": {
        "AttributeName": "retailApplicantId_firstName",
        "Extension": "retailApplicantData"
    },
    "Email": {
        "AttributeName": "retailApplicantId_email",
        "Extension": "retailApplicantData"
    }
},
"Entities":[
    {
        "DestinationEntityName":"FTOS_BNKAP_RetailApplicantData",
        "SourceEntityName":"FTOS_BARET_AccountApplication",
        "SourceLookupDestinationName":"retailApplicantId"
    },
    {
        "DestinationEntityName":"FTOS_BNKAP_
RetailApplicantAddress",
        "SourceEntityName":"FTOS_BARET_AccountApplication",
        "SourceLookupDestinationName":"FTOS_BNKAP_
RetailApplicantAddressId"
    }
],
"OptionSets":[
    {
        "OptionSetName":"Gender Type",
        "MappingName":"Sex",
        "OptionSetItems":{
            "Male":"Male",
            "Female":"Female"
        }
    }
],
"LookupEntities":[
    {
        "MappingName":"DistrictCode",
        "EntityName":"District",
        "AttributeKey":"Code"
    },
    {
        "MappingName":"BirthDistrictBody",
        "EntityName":"District",
        "AttributeKey":"Code"
    }
],

```

```

    {
      "MappingName": "IssuingCountry",
      "EntityName": "FTOS_CMB_Country",
      "AttributeKey": "code"
    },
    {
      "MappingName": "City",
      "EntityName": "City",
      "AttributeKey": "Name",
      "Parent": {
        "AttributeParentKey": "DistrictId",
        "MappingParentName": "DistrictCode"
      }
    }
  ]
,
  "maskNextStepURLSuccess": {
    "entity": "FTOS_BARET_AccountApplication",
    "form": "TestOnfido",
    "section": "step2"
  },
  "maskNextStepURLFail": {
    "entity": "FTOS_BARET_AccountApplication",
    "form": "TestOnfido",
    "section": "Step1"
  },
  "UseLocalization": true,
  "DocumentType": "IdRom"
}

```

Onfido Mappings

The Onfido mappings match the field names as returned by the Onfido processor (keys) with the populated entities' attributes (values).

Setting Name	Description
DocumentsMapping	Holds attribute mappings for various types of scanned documents.
Type	Scanned document type.

Setting Name	Description
Map	Key-value pairs that match the field name as returned by the Onfido processor (the key) to the attribute name in the destination entity (the value).

Example

The *side* key indicates which side of the identity document the field belongs to. For one-sided documents, set it always to *front*.

```
{
  "DocumentsMapping":
  [
    {
      "type": "IdRom",
      "Map":
      {
        "PictureAttribute": { "attribute": "imgFaceOCR",
"side": "front" },
        "LastName": { "attribute": "lastName", "side":
"front" },
        "GivenName": { "attribute": "firstName", "side":
"front" },
        "FullName": { "attribute": "fullName", "side":
"back" },
        "SpouseName": { "attribute": "spouseName", "side":
"back" },
        "WidowName": { "attribute": "widowName", "side":
"back" },
        "AliasName": { "attribute": "aliasName", "side":
"back" },
        "TypeOfDocument": { "attribute": "typeOfDocument",
"side": "back" },
        "DocumentNumber": { "attribute": "IdCardSeries",
"side": "front" },
        "StreetType": { "attribute": "streetType", "side":
"front" },
        "PersonalNumber": { "attribute": "PIN", "side":
"front" },
        "BirthDate": { "attribute": "dateOfBirth", "side":
"front" },
        "PlaceOfBirthBody": { "attribute": "placeOfBirth",
"side": "front" },

```

```

    "BirthCountryBody": { "attribute": "birthCountry",
"side": "front" },
    "Address": { "attribute": "fullAddress", "side":
"front" },
    "Sex": { "attribute": "gender", "side": "front" },
    "DistrictCode": { "attribute": "DistrictId",
"side": "front" },
    "Nationality": { "attribute": "nationality",
"side": "front" },
    "IssuedBy": { "attribute": "IssueInstitution",
"side": "front" },
    "IssuedAt": { "attribute": "IdIssueDate", "side":
"front" },
    "IssuedUntil": { "attribute": "IdExpirationDate",
"side": "front" },
    "IssuingCountry": { "attribute": "isuingCountry",
"side": "front" },
    "Confidence": { "attribute": "confidence", "side":
"front" },
    "mrz_line1": { "attribute": "mrz_line1", "side":
"front" },
    "mrz_line2": { "attribute": "mrz_line2", "side":
"front" },
    "mrz_line3": { "attribute": "mrz_line3", "side":
"front" },
    "AddressLine1": { "attribute": "AddressLine1",
"side": "front" },
    "AddressLine2": { "attribute": "AddressLine2",
"side": "front" },
    "AddressLine3": { "attribute": "AddressLine3",
"side": "front" },
    "AddressLine4": { "attribute": "AddressLine4",
"side": "front" },
    "AddressLine5": { "attribute": "AddressLine5",
"side": "front" }
    }
  }
]
}

```

Onfido UI Customization

The Onfido component supports language and UI customization through the `language` and `customUI` parameters.

- `language`: Customize the language displayed on the SDK by passing a string for the supported languages or object. The following languages are supported: `en_US`, `de_DE`, `es_ES`, `fr_FR`, `it_IT`, `pt_PT`, `nl_NL`.

For unsupported languages, send an object containing local string, a local tag and phrases, and an object containing the keys you want to override. For example:

```
{
  locale: 'en_US',
  phrases: {
    "welcome": {
      "next_button": "Select document",
      "subtitle": "This will take a few minutes.",
      "title": "Validate identity."
    }
  }
}
```

- `customUI`: Customize text, the SDK main container, buttons, links, icon background color, and pop-ups by using the following properties:

```
// Typography
fontFamilyTitle: string;
fontFamilySubtitle: string;
fontFamilyBody: string;
fontSizeTitle: string;
fontSizeSubtitle: string;
fontSizeBody: string;
fontWeightTitle: string;
fontWeightSubtitle: string;
```

```
fontWeightBody: string;
colorContentTitle: string;
colorContentSubtitle: string;
colorContentBody: string;
// Modal
colorBackgroundSurfaceModal: string;
colorBorderSurfaceModal: string;
borderWidthSurfaceModal: string;
borderStyleSurfaceModal: string;
borderRadiusSurfaceModal: string;
// Buttons
colorContentButtonPrimaryText: string;
colorBackgroundButtonPrimary: string;
colorBackgroundButtonPrimaryHover: string;
colorBackgroundButtonPrimaryActive: string;
colorBorderButtonPrimary: string;
colorContentButtonSecondaryText: string;
colorBackgroundButtonSecondary: string;
colorBackgroundButtonSecondaryHover: string;
colorBackgroundButtonSecondaryActive: string;
colorBorderButtonSecondary: string;
colorContentDocTypeButton: string;
colorBackgroundDocTypeButton: string;
colorBorderDocTypeButton: string;
colorBorderDocTypeButtonHover: string;
colorBorderDocTypeButtonActive: string;
colorBackgroundIcon: string;
// Shared Buttons
borderRadiusButton: string;
buttonGroupStacked: boolean;
// Links
colorContentLinkTextHover: string;
colorBorderLinkUnderline: string;
colorBackgroundLinkHover: string;
colorBackgroundLinkActive: string;
// Warning Popups
colorContentAlertInfo: string;
colorBackgroundAlertInfo: string;
colorContentAlertInfoLinkHover: string;
colorContentAlertInfoLinkActive: string;
// Error Popups
```

```

colorContentAlertError: string;
colorBackgroundAlertError: string;
colorContentAlertErrorLinkHover: string;
colorContentAlertErrorLinkActive: string;
// Info Header/Highlight Pills
colorBackgroundInfoPill: string;
colorContentInfoPill: string;
// Icon Buttons
colorBackgroundButtonIconHover: string;
colorBackgroundButtonIconActive: string;
// Camera Shutter Button
colorBackgroundButtonCameraHover: string;
colorBackgroundButtonCameraActive: string;

```

Onfido Request Responses

The Onfido OCR processor is composed of a client side script, automation scripts, automation script libraries, and a client script library. When the applicant's documents are validated, the call is made using the below method. The OCR data is then saved in the entity based on the [Settings](#) configurations and the response is returned in JSON format.

NOTE

When the Onfido automation processor is called, the response returned is saved in JSON format in the customer's FintechOS Digital Journey. The parsing of the returned JSON object depends on the implementation team.

- `getReport`: the result contains the report for a related check.

Example

Sample JSON code for Onfido `getReport` response

```

{
  "id": "571ddd12-5e9a-43c9-ad87-3c2f434ece9b",
  "created_at": "2022-04-27T14:14:34Z",
  "name": "document",
  "href": "/v3.2/reports/571ddd12-5e9a-43c9-ad87-3c2f434ece9b",
  "status": "complete",
  "result": "consider",
  "sub_result": "suspected",
  "breakdown": {
    "age_validation": {
      "breakdown": {
        "minimum_accepted_age": {
          "properties": {

          },
          "result": "clear"
        }
      },
      "result": "clear"
    },
    "compromised_document": {
      "result": "clear"
    },
    "data_comparison": {
      "breakdown": {
        "date_of_birth": {
          "properties": {

          },
          "result": null
        },
        "date_of_expiry": {
          "properties": {

          },
          "result": null
        },
        "document_numbers": {
          "properties": {

          },
          "result": null
        },
        "document_type": {

```

```

        "properties":{
        },
        "result":null
    },
    "first_name":{
        "properties":{
        },
        "result":null
    },
    "gender":{
        "properties":{
        },
        "result":null
    },
    "issuing_country":{
        "properties":{
        },
        "result":null
    },
    "last_name":{
        "properties":{
        },
        "result":null
    }
    },
    "result":null
},
"data_consistency":{
    "breakdown":{
        "date_of_birth":{
            "properties":{
            },
            "result":"clear"
        },
        "date_of_expiry":{
            "properties":{
            },
            "result":"clear"
        }
    }
}

```

```
    },
    "document_numbers":{
      "properties":{

      },
      "result":"clear"
    },
    "document_type":{
      "properties":{

      },
      "result":"clear"
    },
    "first_name":{
      "properties":{

      },
      "result":"clear"
    },
    "gender":{
      "properties":{

      },
      "result":"clear"
    },
    "issuing_country":{
      "properties":{

      },
      "result":"clear"
    },
    "last_name":{
      "properties":{

      },
      "result":"clear"
    },
    "multiple_data_sources_present":{
      "properties":{

      },
      "result":null
    },
    "nationality":{
      "properties":{
```



```

        },
        "result":null
    }
},
"result":"clear"
},
"data_validation":{
  "breakdown":{
    "barcode":{
      "properties":{

        },
        "result":null
      },
      "date_of_birth":{
        "properties":{

          },
          "result":"clear"
        },
        "document_expiration":{
          "properties":{

            },
            "result":"consider"
          },
          "document_numbers":{
            "properties":{
              "document_number":"clear",
              "personal_number":"clear"
            },
            "result":"clear"
          },
          "expiry_date":{
            "properties":{

              },
              "result":"clear"
            },
            "gender":{
              "properties":{

                },
                "result":"clear"
              }

```

```

    },
    "mrz":{
      "properties":{
        },
      "result":"consider"
    }
  },
  "result":"consider"
},
"image_integrity":{
  "breakdown":{
    "colour_picture":{
      "properties":{
        },
      "result":"clear"
    },
    "conclusive_document_quality":{
      "properties":{
        "abnormal_document_features":"clear",
        "corner_removed":"clear",
        "digital_document":"clear",
        "missing_back":"clear",
        "obscured_data_points":"clear",
        "obscured_security_features":"clear",
        "punctured_document":"clear",
        "watermarks_digital_text_overlay":"clear"
      },
      "result":"clear"
    },
    "image_quality":{
      "properties":{
        },
      "result":"clear"
    },
    "supported_document":{
      "properties":{
        },
      "result":"clear"
    }
  },
  "result":"clear"
}

```

```

    },
    "issuing_authority":{
      "breakdown":{
        "nfc_active_authentication":{
          "properties":{

            },
          "result":null
        },
        "nfc_passive_authentication":{
          "properties":{

            },
          "result":null
        }
      },
      "result":null
    },
    "police_record":{
      "result":null
    },
    "visual_authenticity":{
      "breakdown":{
        "digital_tampering":{
          "properties":{

            },
          "result":"clear"
        },
        "face_detection":{
          "properties":{

            },
          "result":"clear"
        }
      },
      "fonts":{
        "properties":{

          },
        "result":"consider"
      },
      "original_document_present":{
        "properties":{
          "document_on_printed_paper":"clear",
          "photo_of_screen":"consider",

```

```

        "scan": "clear",
        "screenshot": "clear"
    },
    "result": "consider"
},
"other": {
    "properties": {

    },
    "result": "consider"
},
"picture_face_integrity": {
    "properties": {

    },
    "result": "consider"
},
"security_features": {
    "properties": {

    },
    "result": "consider"
},
"template": {
    "properties": {

    },
    "result": "consider"
}
},
"result": "consider"
}
},
"properties": {
    "date_of_birth": "1987-07-19",
    "date_of_expiry": "2011-07-19",
    "document_numbers": [
        {
            "type": "document_number",
            "value": "BV999999"
        },
        {
            "type": "personal_number",
            "value": "1000000000000"
        }
    ]
}

```


Getting Status Changes Notifications Using Webhooks

Webhooks are automated message sent from an application when an event is triggered. In the Onfido dashboard, webhooks need to be defined allowing you to get notifications when the applicant document check and report are completed.

NOTE

Only one webhook can be created for each client account in the Onfido dashboard.

In the Onfido [settings](#), if the setting key `UseProcessorForCheck` is set to true, a new record is created in the `FTOS_OnfidoIntegration` entity. In this entity, the following entities are attached:

- `FTOS_ONFIDO_CHECK`: for document check that contains the **Pending**, **Webhook Received**, and **Completed** statuses.
- `FTOS_ONFIDO_CHECKREPORT`: for report check that stores the **Pending**, **Webhook Received**, and **Completed** statuses. It contains the raw response for each report that is received from Onfido.

When a new check is created, the default status is **submitted**. Then, the `FTOS_Onfido_WebHook` is called from DCS, the report is saved, and the status is changed to **completed**.

Triggering Identity Verification in a Digital Journey

The Onfido SDK is responsible for capturing and uploading photos or videos. The Onfido automation processor can be added to a Digital Journey and further enhanced via Innovation Studio based on the customer's business needs.

Follow the below steps when adding this automation processor to your Digital Journey:

1. In Innovation Studio, open the customer's form driven flow in the editor.
2. Navigate to the form or form step where you wish to initiate the identity check.

3. In the After Events tab, add code based on the model below:

```
var dfpHelper = ebs.importClientScript('FTOS.DFP');
var componentName = 'FTOS_DFP_Onfido'; //name of the
componentvar
var recordId = formData.id;
var flowSettingsName = formData.FlowSettings;
var p = {};
var accountApplicationId = recordId;
p.accountApplicationId = accountApplicationId;
p.toStatus = "OCR in Progress";
ebs.callActionByName("myAccountApplication_BusinessWorkflow",
p, function(e) {
    var params = {};
    params.flowSettingsName = flowSettingsName;
    params.processorSettingsType = 'Onfido';
    params.processorSettingsName = 'Onfido_Settings'; //name
of the proccessor setting that you created
    ebs.callActionByName("FTOS_DFP_
FlowProcessorSettingsByType", params, function(f) {
        var processorSettingsId =
f.UIResult.Data.ProcessorSettingsId;
        dfpHelper.loadComponent(componentName,
processorSettingsId, recordId, false);
    });
});
```

4. Click the **Save and Close** button at the top right corner to save your digital journey.