



**PLATFORM DEEP DIVE / Plugins / Custom Plugins / ocr-plugin**

# Contents

- PLATFORM DEEP DIVE / Plugins / Custom Plugins / OCR plugin
  - Using the OCR plugin
    - Use case
    - Scenario for FLOWX.AI generated documents
    - Setup guide

## PLATFORM DEEP DIVE / Plugins / Custom Plugins / OCR plugin

The OCR (Optical Character Recognition) plugin is a powerful tool that enables you to read barcodes and extract handwritten signatures from .pdf documents with ease.

Before using the OCR service for reading barcodes and extracting signatures, please note the following requirements:

### CAUTION

- All \*.pdf documents that are sent to the OCR service for reading barcodes and extracting handwritten signatures should be scanned at a minimum resolution of 200DPI (approximately 1654x2339 px for A4 pages)
- Barcode is searched on the top 15% of each image (scanned page)
- Signatures are detected on boxes with a border: 4px black solid
- Only two signatures per image (scanned page) are detected.

- All \*.pdf documents should be scanned at a minimum resolution of 200DPI (approximately 1654x2339 px for A4 pages).
- The barcode is searched in the top 15% of each scanned page.
- Signatures are detected within boxes with a 4px black solid border.
- The plugin detects up to two signatures per scanned page.
- Only two signatures per image (scanned page) are detected.

#### ! INFO

The plugin supports **1D Code 128** barcodes. For more information about this barcode type, please refer to the [documentation here](#).

## Using the OCR plugin

You can utilize the OCR plugin to process generic document templates by either using a specific flow on FLOWX.AI (HTML template) or any other document editor.

#### ! INFO

Using a specific flow on FLOWX.AI offers several advantages:

- Centralized management of templates and flows within a single application.
- Access to template history and version control.

## Use case

1. Prepare and print generic document templates.

2. End-users complete, sign, and scan the documents.
3. Upload the scanned documents to the flow.
4. FLOWX validates the template (barcode) and the signatures.

## Scenario for FLOWX.AI generated documents

1. Utilize the **Documents plugin** to create a **document template**.

» Generating documents based on templates

FLOWX.AI

Processes

Definitions

Active process

Process Instances

Failed process start

Content Management

Enumerations

Substitution tags

Content models

Languages

Source systems

Plugins

Task Manager

All tasks

Hooks

Stages

Allocation rules

Out of office

Notification templates

John Doe

Process Definitions

Search by process defin

Drafts / In progress

Name	Version	Edited at	Edited by	
S...	1	30 Sep 2022, 5:41 PM	Silviu Grigore	▶ ✎ ⋮
d...	2	30 Sep 2022, 2:32 PM	andrei antal	▶ ✎ ⋮
t...	1	30 Sep 2022, 11:15 AM	QA FlowX	▶ ✎ ⋮
T...	4	30 Sep 2022, 10:20 AM	QA FlowX	▶ ✎ ⋮

Published

Name	Version	Published at	Published by	
A...	1	03 Oct 2022, 8:12 AM	QA FlowX	▶ ✎ ⋮
C...	19	30 Sep 2022, 3:08 PM	Silviu Grigore	▶ ✎ ⋮
d...	1	30 Sep 2022, 10:34 AM	Bogdan Ionescu	▶ ✎ ⋮
T...	1	30 Sep 2022, 10:18 AM	QA FlowX	▶ ✎ ⋮

2. Create a process and add a **Kafka Send Action** to a **Message event send** node. Here you specify the **kafka topic** (address) where the template will be generated.

Node Config Actions

Actions +  
send\_main\_... +

### Action Edit

ID: 584701

Name  
send\_main\_applicant\_consent

Order  
1

Timer Expression

Kafka Send Action

☒ Automatic ☐ Manual

☒ Mandatory ☐ Optional

☐ Repeatable

Autorun Children? ☐

### Parameters

Custom From integration

### Topics

ai.flowx.in.sandbox.document.html.generate.v1

#### ! INFO

The Kafka topic for generating the template must match the topic defined in the **KAFKA\_TOPIC\_DOCUMENT\_GENERATE\_HTML\_IN** variable. Refer to the **Kafka configuration guide** for more details. For additional information, please see the **Documents plugin setup guide**.

3. Fill in the **Message**. The request body should include the following values:

- **documentList** - a list of documents to be generated, including properties such as name and values to be replaced in the document templates

- **customId** - client ID
- **templateName** - the name of the template to be used
- **language**
- **includeBarcode** - true/false
- **data** - a map containing the values that should replace the placeholders in the document template, the keys used in the map should match those defined in the HTML template

## Parameters

Custom

From integration

## Topics

```
ai.flowx.in.sandbox.document.html.generate.v1
```

## Message

```
1 {
2   "documentList": [
3     {
4       "customId": "1234",
5       "templateName": "Agreement",
6       "language": "ro",
7       "data": {
8         "offerName": "The greatest offer - deluxe edition",
9         "companyName": "Test Company SRL",
10        "cui": "R01234567",
11        "firstName": "Bess",
12        "lastName": "Twishes",
13        "offerExpiryDate": "31.12.2099"
14      },
15      "includeBarcode": true
16    }
17  ]
18 }
```

## Advanced configuration

Show Headers ☐























INFO

The **data parameters** must be defined in the document template beforehand. For more information, check the **WYSIWYG Editor** section.

Documents Templates - Test\_docs

Body

Data model

Name	Type	Mandatory	
customId	STRING	false	 
templateName	STRING	false	 
language	STRING	false	 
version	NUMBER	false	 
draft	BOOLEAN	false	 
offerName	STRING	false	 
companyName	STRING	false	 
cui	STRING	false	 
lastName	STRING	false	 
firstName	STRING	false	 

#### 4. Add a **barcode**.





- to include a **default barcode**, add the following parameter to the message body: `includeBarCode: true`.
- to include a **custom barcode**, set `includeBarCode: false` and provide the desired data in the `data` field

5. Add a **message received event** node and specify the topic where you want to receive the response.

### CAUTION

Ensure that the topic matches the one defined in the `KAFKA_TOPIC_DOCUMENT_GENERATE_HTML_OUT` variable.

Node: **receive\_consent** (ID: 584502)

#### Node Config

##### General Config

Node name

Can go back? ☒

Swimlane



##### Response Timeout

Response Timeout (PT30S)

### Data stream topics

Custom

From integration

Topic Name

Key Name

ai.flowx.updates.sandbox.document.html.gener

ocrBarcodeResponse



Add stream

6. Add a **user task node** and configure an **Upload file action** to send the file (defined by the **KAFKA\_TOPIC\_DOCUMENT\_PERSIST\_IN** variable) to the storage solution (for example, S3).

## Parameters

---

### Topics

☐ Replace Values

### Document Type

☐ Replace Values

### Folder

☒ Replace Values

### Advanced configuration

Show Headers ☐

## Data to send

---

[Add Key](#)

- Next, the response will be sent back to the kafka topic defined by **KAFKA\_TOPIC\_DOCUMENT\_PERSIST\_OUT** environment variable through a callback action/subprocess.
- Next, send the response to the OCR Kafka topic defined at **KAFKA\_TOPIC\_OCR\_IN** variable (representing the path to the S3 file)

9. Display the result of the OCR validation on the kafka topic defined at

**KAFKA\_TOPIC\_OCR\_OUT**.

## Setup guide

Refer to the OCR plugin setup guide for detailed instructions on setting up the OCR plugin:

» [OCR plugin setup](#)

**Was this page helpful?**