



**PLATFORM DEEP DIVE / Plugins / Custom Plugins / Documents plugin / Using the plugin / getting-urls-to-documents**

# Contents

- PLATFORM DEEP DIVE / Plugins / Custom Plugins / Documents plugin /  
Using the plugin / Getting URLs for documents
  - Sending the request
  - Receiving the reply

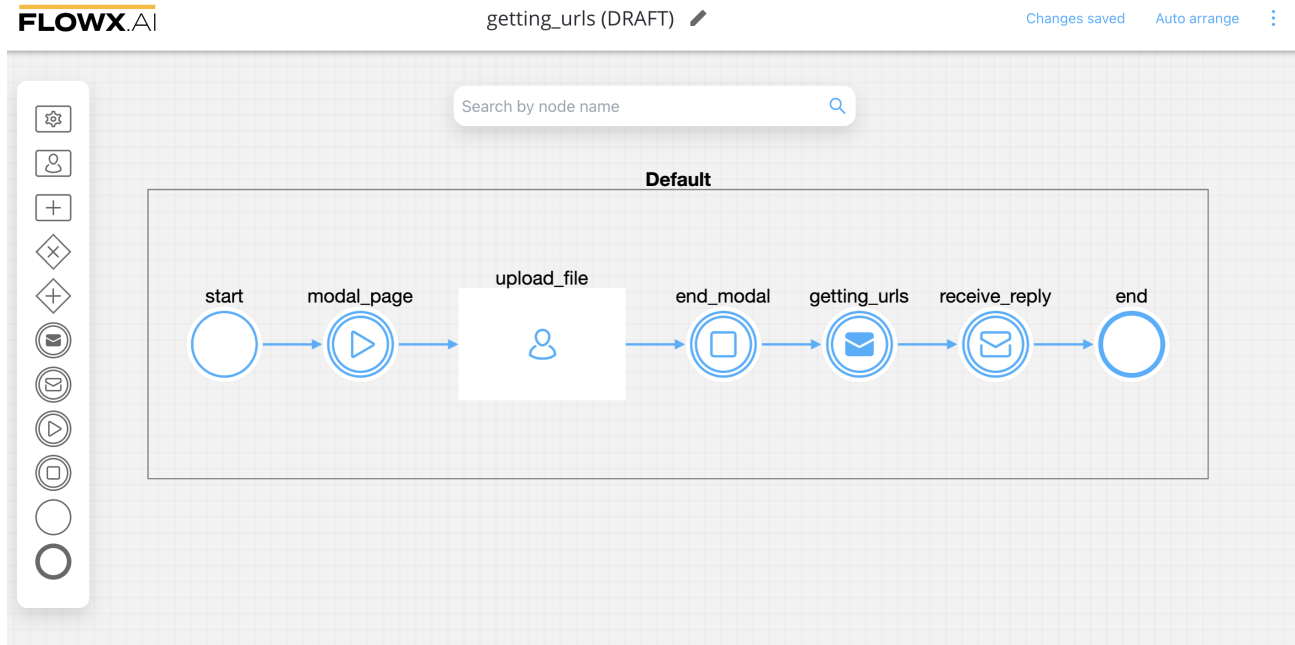
## PLATFORM DEEP DIVE / Plugins / Custom Plugins / Documents plugin / Using the plugin / Getting URLs for documents

In certain scenarios, you may need to obtain URLs that point to uploaded documents to be used by other integrations. This requires adding a custom action to your process that requests the URLs from the Documents Plugin.

### Sending the request

To retrieve document URLs and use them, for example, in the Notification Plugin to attach them to emails, follow the next steps:

1. Create a process and include the following nodes:
  - a **Kafka Send Event Node**,
  - a **Kafka Receive Event Node**
  - a **User Task Node**
  - **Start / End <ilestone** Nodes to **create a modal**



2. Configure the **User Task Node** and add an **Upload Action** to it.

## Action Edit

ID: 769898

Name

upload\_file\_action

Order

1

Timer Expression

Upload File

☐ Automatic ☒ Manual

☒ Mandatory ☐ Optional

☒ Repeatable

Autorun Children? ☐

Allow BACK on this action? ☐

Save

### 3. Configure the parameters for the **Upload Action**:

## Parameters

### Topics

☐ Replace Values

### Document Type

☒ Replace Values

### Folder

☒ Replace Values

### Advanced configuration

Show Headers ☒

```
1 {"processInstanceId": "${processInstanceId}"}
```

Save

## ! INFO

For more details on uploading a document and configuring an upload action, refer to the following sections:

**Upload document**

**Upload action**

4. Configure the Kafka Send Event Node by adding a **Kafka Send Action** and specifying the **Kafka topic** to send the request to:

## Parameters

Custom

From integration

### Topics

ai.flowx.in.ga.document.urls.v1

5. Fill in the body of the request message for the action:

### Message

```
1 {  
2   "types": [  
3     "${processInstanceId}", "${processInstanceId}"  
4   ]  
5 }
```

### Advanced configuration

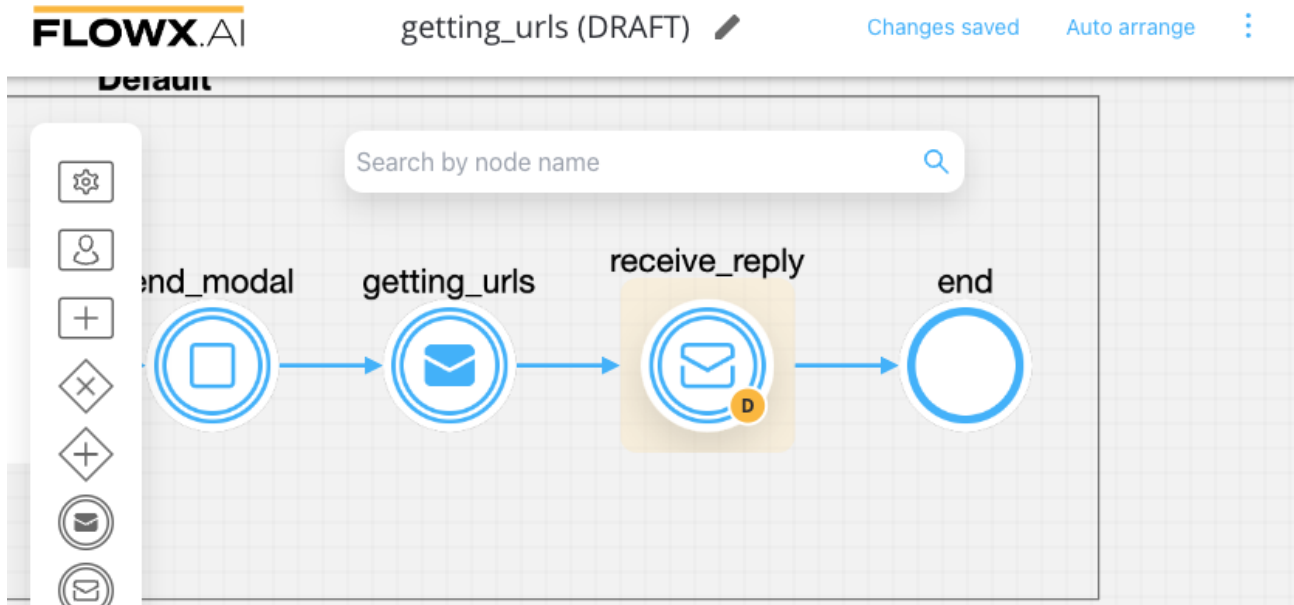
Show Headers ☒

```
1 {"processInstanceId": "${processInstanceId}"}
```

Save

- `types` - a list of document types

6. Configure the **Kafka Receive Event Node** by adding the kafka topic on which the response will be sent.

Node: **receive\_reply** (ID: 766487)

## Node Config

## Swimlane

Default



Stage



## Response Timeout

Response Timeout (PT30S)

## Data stream topics

Custom

From integration

Topic Name

ai.flowx.updates.qa.documen

Key Name

receiveReply



Save



**! INFO**

Kafka topic names can be set by using environment variables:

- `KAFKA_TOPIC_DOCUMENT_GET_URLS_IN` - `ai.flowx.in.qa.document.urls.v1` - the topic that listens for the request from the engine
- `KAFKA_TOPIC_DOCUMENT_GET_URLS_OUT` - `ai.flowx.updates.qa.document.urls.v1` - the topic on which the engine will expect the reply

The example topic names above are from an internal testing environment.

When setting topics for other environments, follow this pattern:

```
ai.flowx.updates.{{environment}}.document.urls.v1.
```

**! CAUTION**

The Engine listens for messages on topics with specific naming patterns. Ensure that your outgoing topic name matches the pattern configured in the Engine.

## Receiving the reply

### Audit log details

**Event:** process instance, message receive, 27 Oct 2022 at 12:26 PM

**Url:** ai.flowx.updates.qa.document.urls.v1

**Body:**

```
1 [{"success":true,"fullName":"1234_771853/4752_771853.pdf",  
  "fileName":"1234_771853/4752_771853","fileExtension":"pdf",  
  "url":"http://minio:9000/  
qualitance-dev-paperflow-qa-process-id-771853/1234_771853/  
4752_771853.pdf?X-Amz-Algorithm=AWS4-HMAC-SHA256&  
X-Amz-Credential=minio%2F20221027%2Fus-east-1%2Fs3%2Faws4_r  
equest&X-Amz-Date=20221027T092616Z&X-Amz-Expires=604800&  
X-Amz-SignedHeaders=host&  
X-Amz-Signature=76885166e179263cfabaf00d6cd57ca38d08d31f8f0  
502b3d89d160183c92b56"}]
```

The response body is expected to contain the following values:

```
[  
  {  
    "success": true,
```

```
    "fullName": "1234_771853/4752_771853.pdf",  
    "fileName": "1234_771853",  
    "fileExtension": "pdf",  
    "url": "  
<http://SOME_URL/1234_771853/4752_771853.pdf?X-Amz-  
Algorithm=SOME_ALGORITHM&X-Amz-Credential=SOME_CREDENTIAL&X-  
Amz-Date=20210223T113621Z&X-Amz-Expires=604800&X-Amz-  
SignedHeaders=host&X-Amz-Signature=>"  
  }  
]
```

- **success:** A boolean indicating whether the document exists and the URL was generated successfully.
- **fullName:** The full name of the document file, including the directory path.
- **fileName:** The name of the document file without the extension.
- **fileExtension:** The extension of the document file.
- **url:** The full download URL for the document.

**Was this page helpful?**