FLOWX.AI

PLATFORM DEEP DIVE / Plugins / 📦 Custom Plugins / 📦 Documents plugin / Using the plugin / uploading-a-new-document

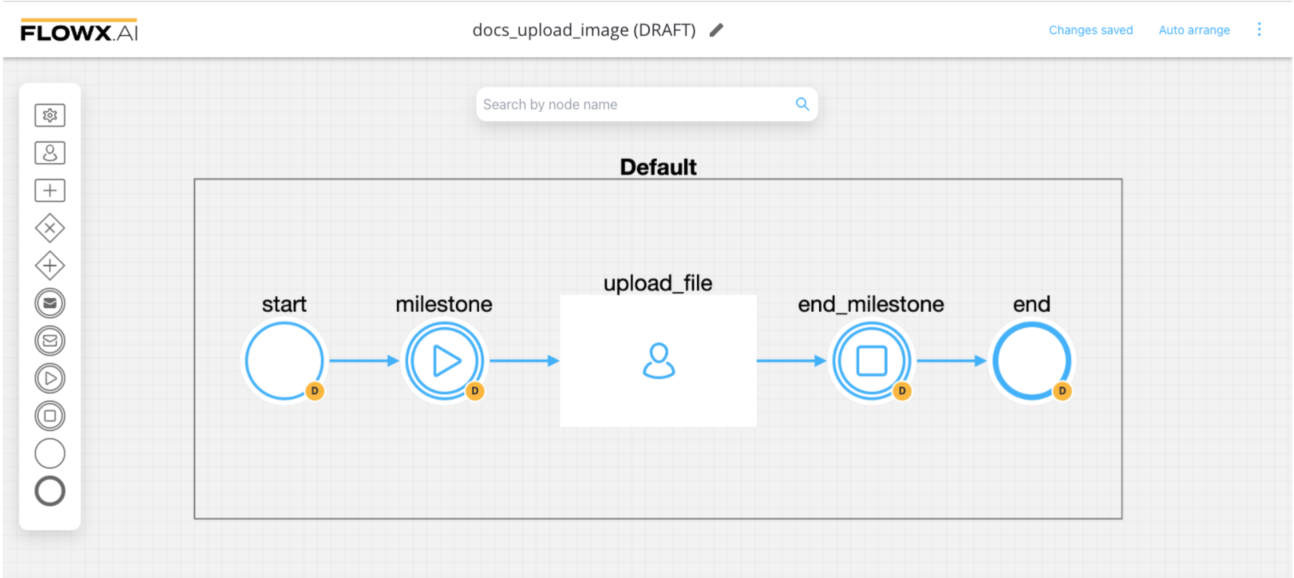
Contents

PLATFORM DEEP DIVE / Plugins / 📦 Custom Plugins / 📦 Documents plugin / Using the plugin / Uploading a new document

You can integrate document upload into a

The fallback content to display on prerendering
by adding a user task node with an **Upload action**. This allows users to interact with the process and choose which file to upload.

> **INFO**
>
> User task
>
> The fallback content to display on prerendering
> enable you to define and configure UI templates and actions for specific template config nodes, such as an upload file button.



To upload a document using a process, follow the next steps.

Defining the process

1. Create a process definition.
2. Add the necessary nodes, including **start/end nodes**, **start/end milestone nodes**, and a **user task node**.
3. Configure the user task node:
   - Configure the node settings.
   - Configure the upload action, including topics, document type, and folder.
   - (UI) Configure the upload button.

Configuring the process definition

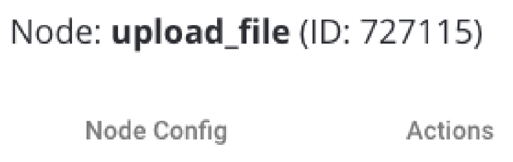
User task node

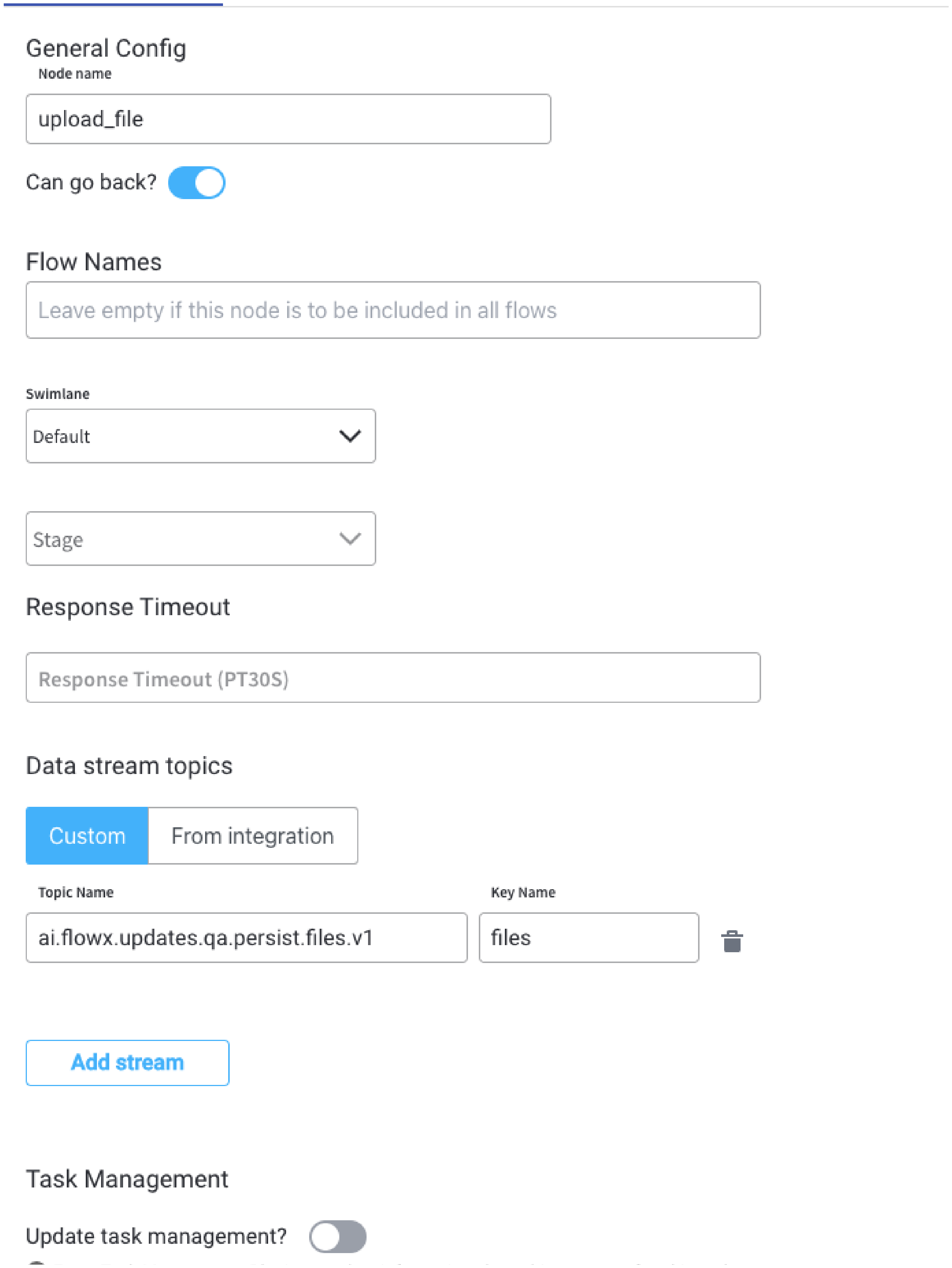
Node Config

- **Swimlane**: Choose a swimlane (if there are multiple swimlanes in the process) to restrict access to specific user roles. If there's only one swimlane, the value is "Default".
- **Stage**: Assign a stage to the node.
- **Topic Name**: Specify the topic name where the process engine listens for the response. This topic should be added to the platform and match the topic naming rule for the engine to listen to it. The default value is `ai.flowx.updates.qa.persist.files.v1`, extracted from `KAFKA_TOPIC_DOCUMENT_PERSIST_IN`.

> ⚠ **CAUTION**
>
> A naming pattern must be defined in the
>
> The fallback content to display on prerendering
> configuration to use the specified topics. It's important to ensure that all events starting with the configured pattern are consumed by the Engine. For example, the `KAFKA_TOPIC_PATTERN` is the topic name pattern where the Engine listens for incoming
> The fallback content to display on prerendering
> events.

- **Key Name**: This key will hold the result received from the external system. If the key already exists in the process values, it will be overwritten.


Node: **upload_file** (ID: 727115)



General Config

Node name: upload_file

Can go back? ●

Flow Names: Leave empty if this node is to be included in all flows

Swimlane: Default

Stage

Response Timeout: Response Timeout (PT30S)

Data stream topics

Topic Name: ai.flowx.updates.qa.persist.files.v1

Key Name: files

Task Management

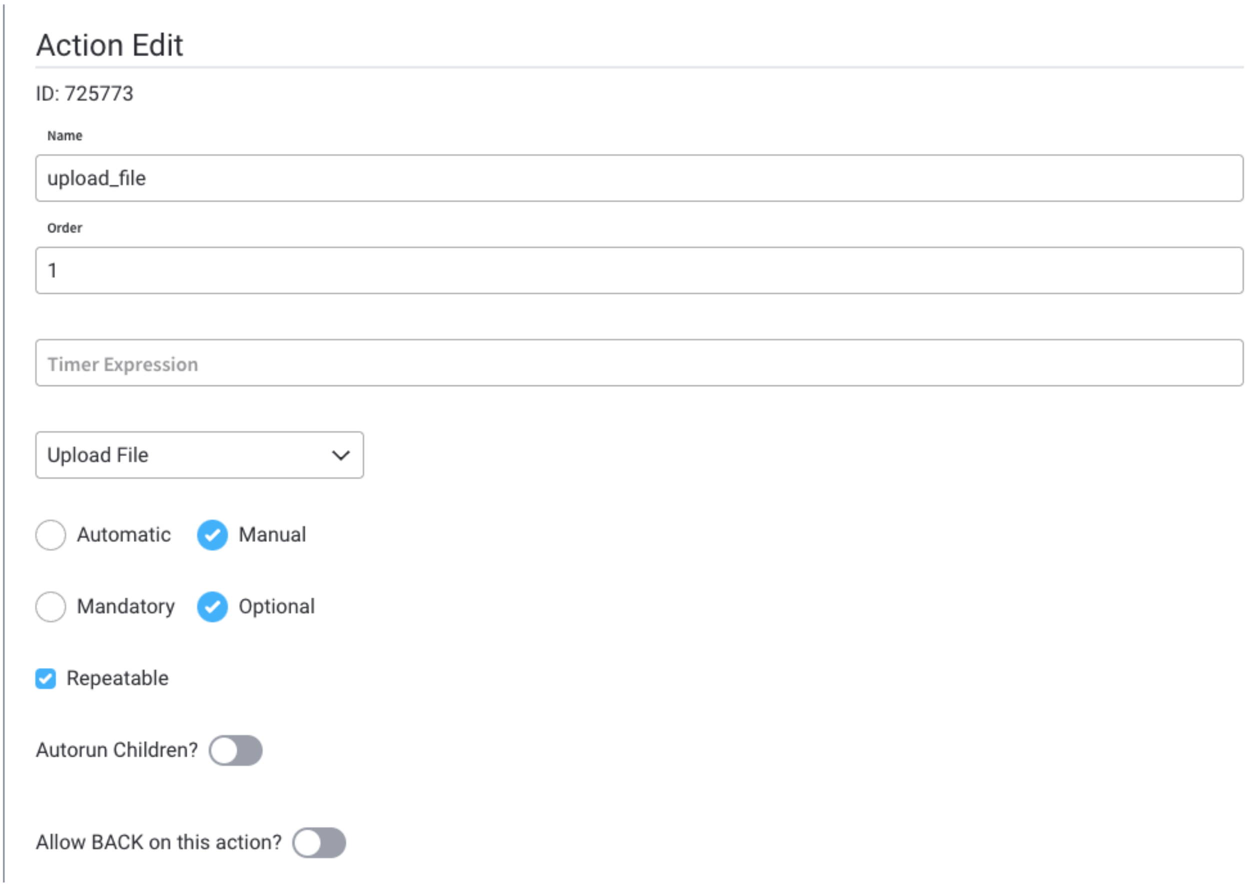
Update task management? ○

ⓘ Force Task Management Plugin to update information about this process after this node.

Actions

Actions edit

- **Action Type**: Set it to Upload File.
- **Trigger Type**: Choose Manual to allow user-triggered action.
- **Required Type**: Set it as Optional.
- **Reputable**: Check this option if the action can be triggered multiple times.
- **Autorun Children**: When enabled, the child actions defined as mandatory and automatic will run immediately after the parent action is finalized.



Action Edit

ID: 725773

Name: upload_file

Order: 1

Timer Expression

Upload File

○ Automatic ● Manual

○ Mandatory ● Optional

☑ Repeatable

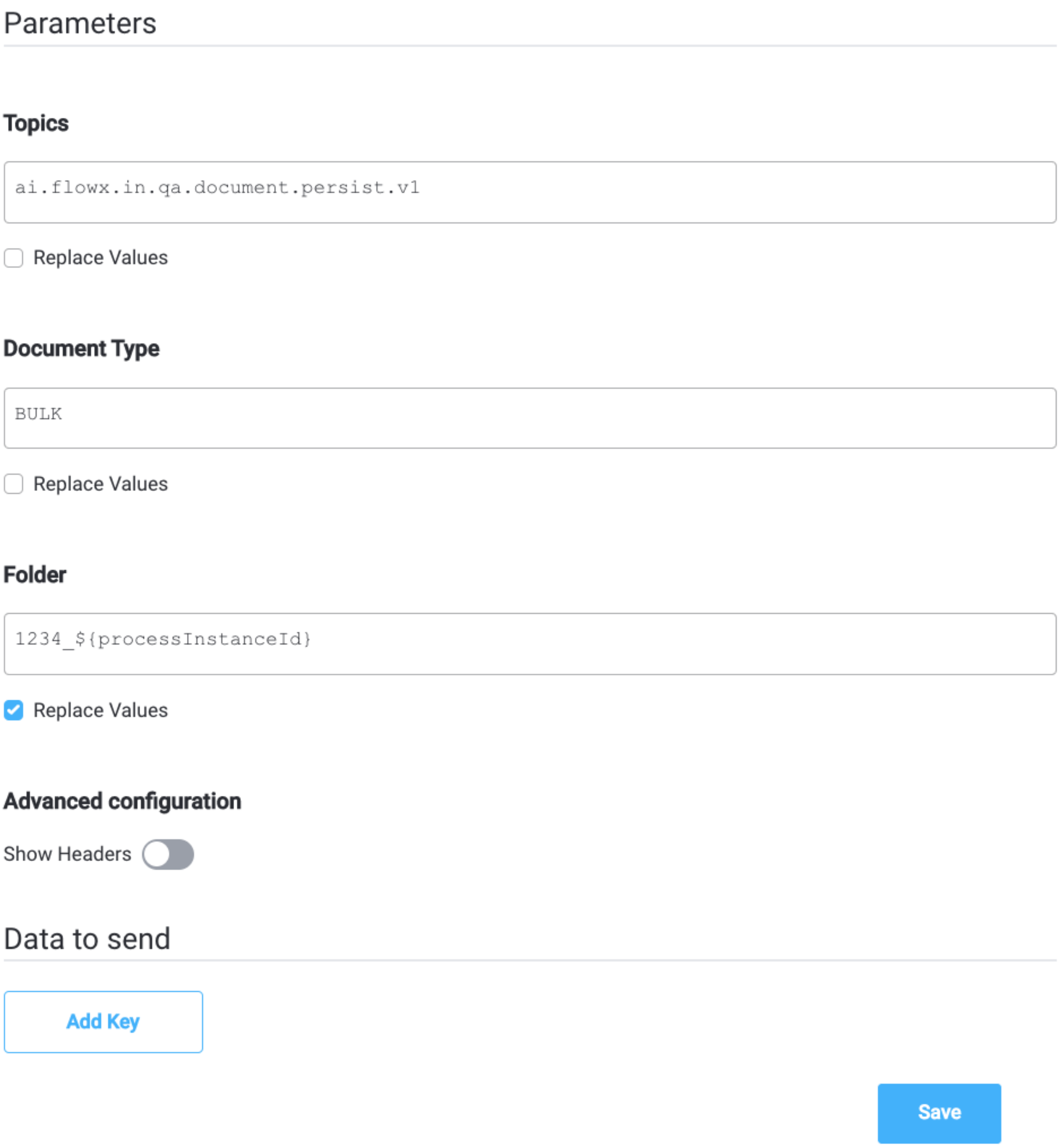
Autorun Children? (off)

Allow BACK on this action? (off)

Parameters

- **Topics**: Set it to `ai.flowx.in.document.persist.v1`, extracted from `KAFKA_TOPIC_DOCUMENT_PERSIST_IN`.
- **Document Type**: Set it to BULK.
- **Folder**: Allows you to configure a value by which the file will be identified in the future.
- **Advanced Configuration (Show Headers)**: Represents a JSON value that will be sent in the headers of the Kafka message.

> **INFO**
>
> Kafka topic names can be customized by overwriting the following environment variables during deployment:
>
> - `KAFKA_TOPIC_DOCUMENT_PERSIST_IN` - default value: `ai.flowx.in.qa.document.persist.v1`
> - `KAFKA_TOPIC_DOCUMENT_PERSIST_OUT` - default value: `ai.flowx.updates.qa.document.persist.v1`
>
> The above examples of topics are extracted from an internal testing environment. When setting topics for other environments, follow this pattern: `ai.flowx.updates.{{environment}}.document.persist.v1`.



Parameters

Topics

ai.flowx.in.qa.document.persist.v1

☐ Replace Values

Document Type

BULK

☐ Replace Values

Folder

1234_${processInstanceId}

☑ Replace Values

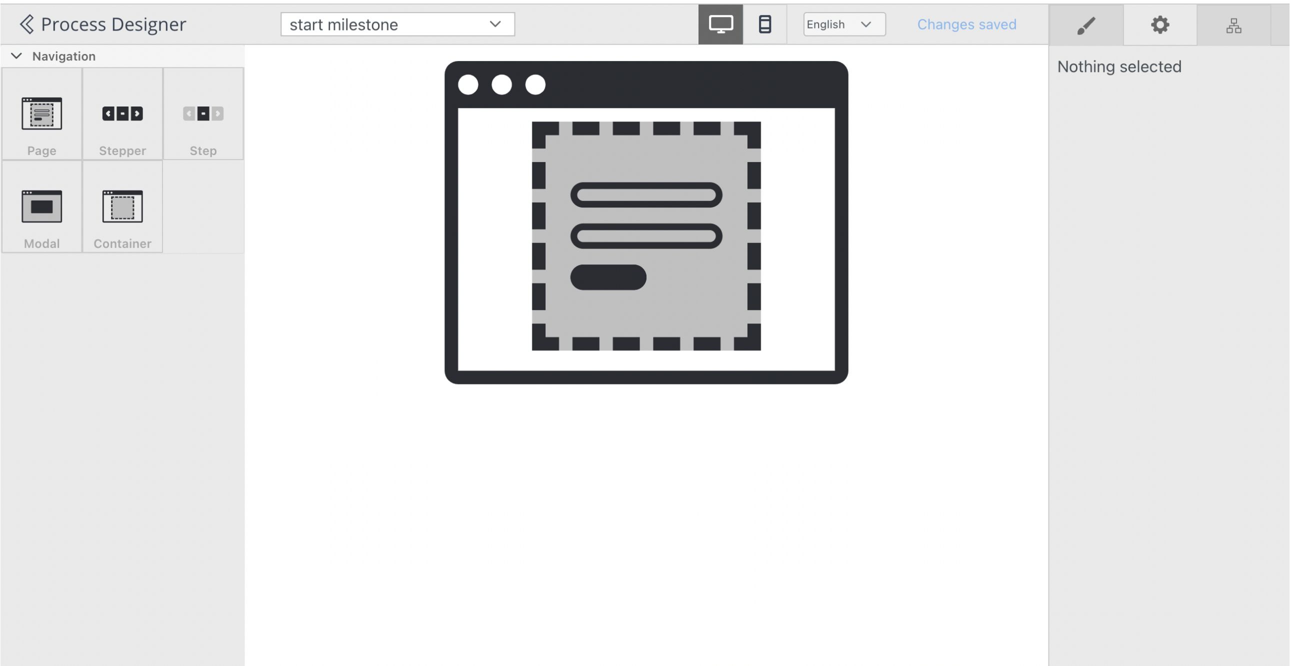
Advanced configuration

Show Headers ○

Data to send

Milestone nodes

You can configure start and end milestone nodes before and after the user task. Additionally, you can add a modal template (e.g., a **Page**) to the start milestone node to display a modal screen, as shown in the example above.



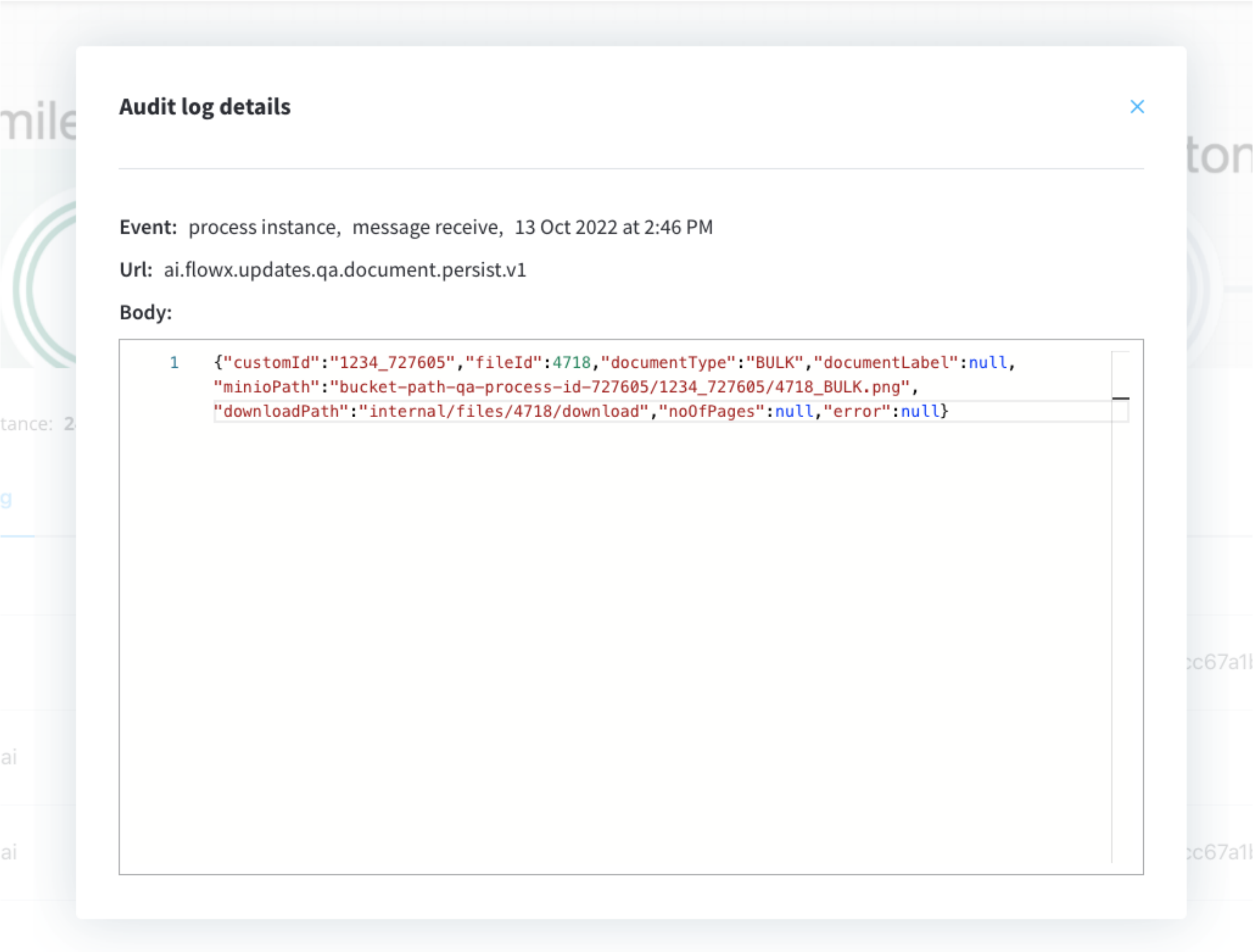
Receiving the reply

The reply body is expected to contain the following values:

- **customId**: The client ID.
- **fileId**: The ID of the file.
- **documentType**: The document type.
- **minioPath**: The path where the uploaded file is saved. It represents the location of the file in the storage system, whether it's a MinIO path or an S3 path, depending on the specific storage solution.
- **downloadPath**: The download path for the uploaded file. It specifies the location from where the file can be downloaded.
- **noOfPages**: The number of pages in the document.

INFO

You can view the response by accessing the **Audit log** menu.



Audit log details

Event: process instance, message receive, 13 Oct 2022 at 2:46 PM

Url: ai.flowx.updates.qa.document.persist.v1

Body:

```
{"customId":"1234_727605","fileId":4718,"documentType":"BULK","documentLabel":null,
"minioPath":"bucket-path-qa-process-id-727605/1234_727605/4718_BULK.png",
"downloadPath":"internal/files/4718/download","noOfPages":null,"error":null}
```

```
{
    "customId" : "1234_727605",
    "fileId" : 4718,
    "documentType" : "BULK",
    "documentLabel" : null,
    "minioPath" : "bucket-path-qa-process-id-
727605/1234_726254/4718_BULK.png",
        "downloadPath" : "internal/files/4714/download",
        "noOfPages" : null,
        "error" : null
    }
```

Was this page helpful?