FLOWX.AI

BUILDING BLOCKS / Process Designer

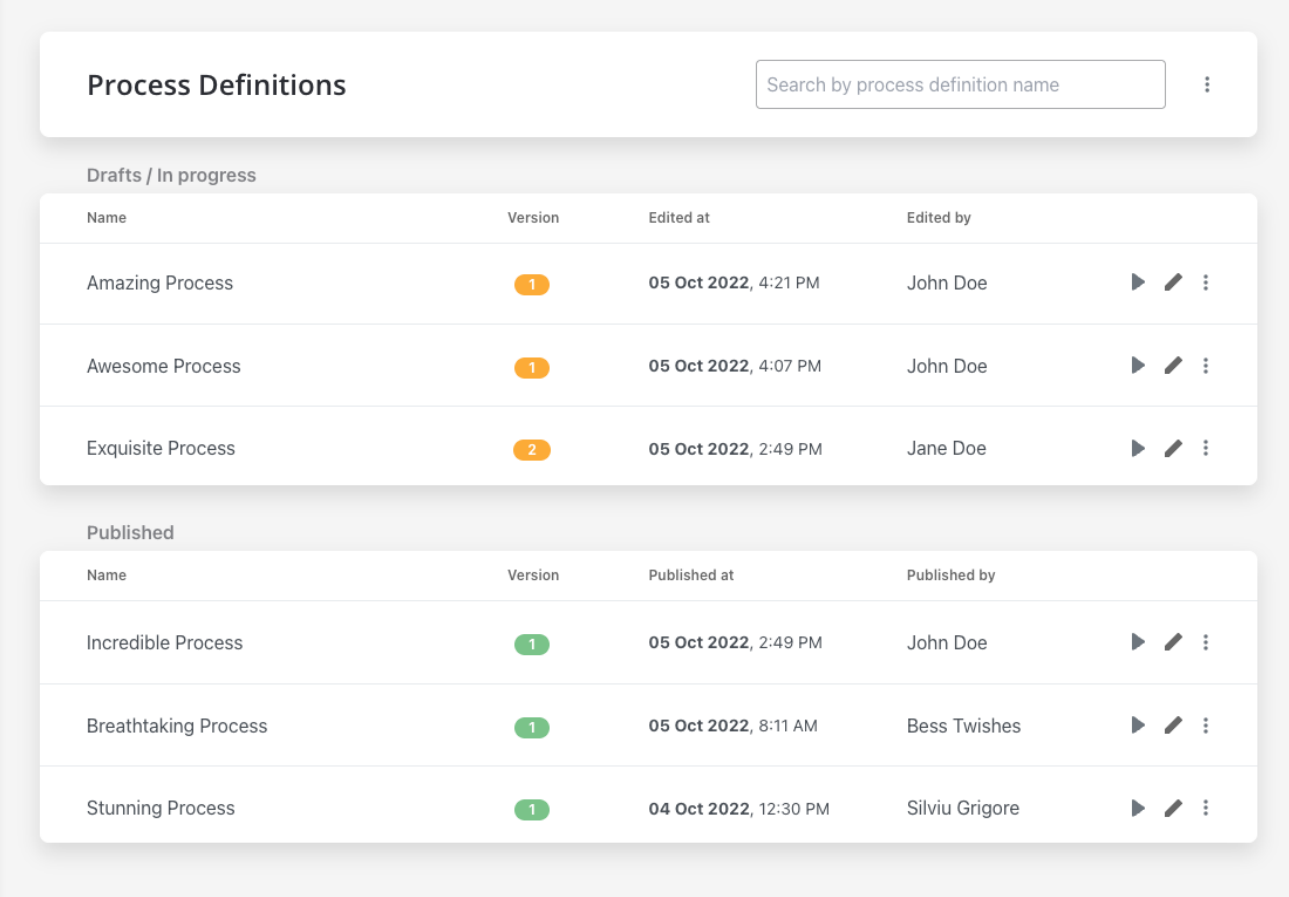
Contents

- BUILDING BLOCKS / Process Designer / Process definition
  - History
    - Versions
    - Audit log
  - Data model
    - Attributes type
    - Data model reference
    - Sensitive data
    - Reporting
    - Generating data model
  - Swimlanes
  - Settings
    - General
    - Permissions
    - Task management
- BUILDING BLOCKS / Process Designer / Active process / Process instance
  - Overview
  - Checking the Process Status
  - Understanding the Process Status Data
    - Process menu
    - Color coding
  - Starting a new process instance
  - Troubleshooting possible errors
- BUILDING BLOCKS / Process Designer / Active process / Failed process start
  - Exceptions
    - Exceptions data

    - Exceptions type
- BUILDING BLOCKS / Process Designer / Subprocess
    - Configuring & starting subprocesses
    - Executing subprocesses

BUILDING BLOCKS / Process Designer / Process definition

The core of the platform is the process definition, which is the blueprint of the business process made up of nodes that are linked by sequences.



Once a process is defined and set as published on the platform, it can be executed, monitored, and optimized. When a business process is started, a new instance of the definition is created.

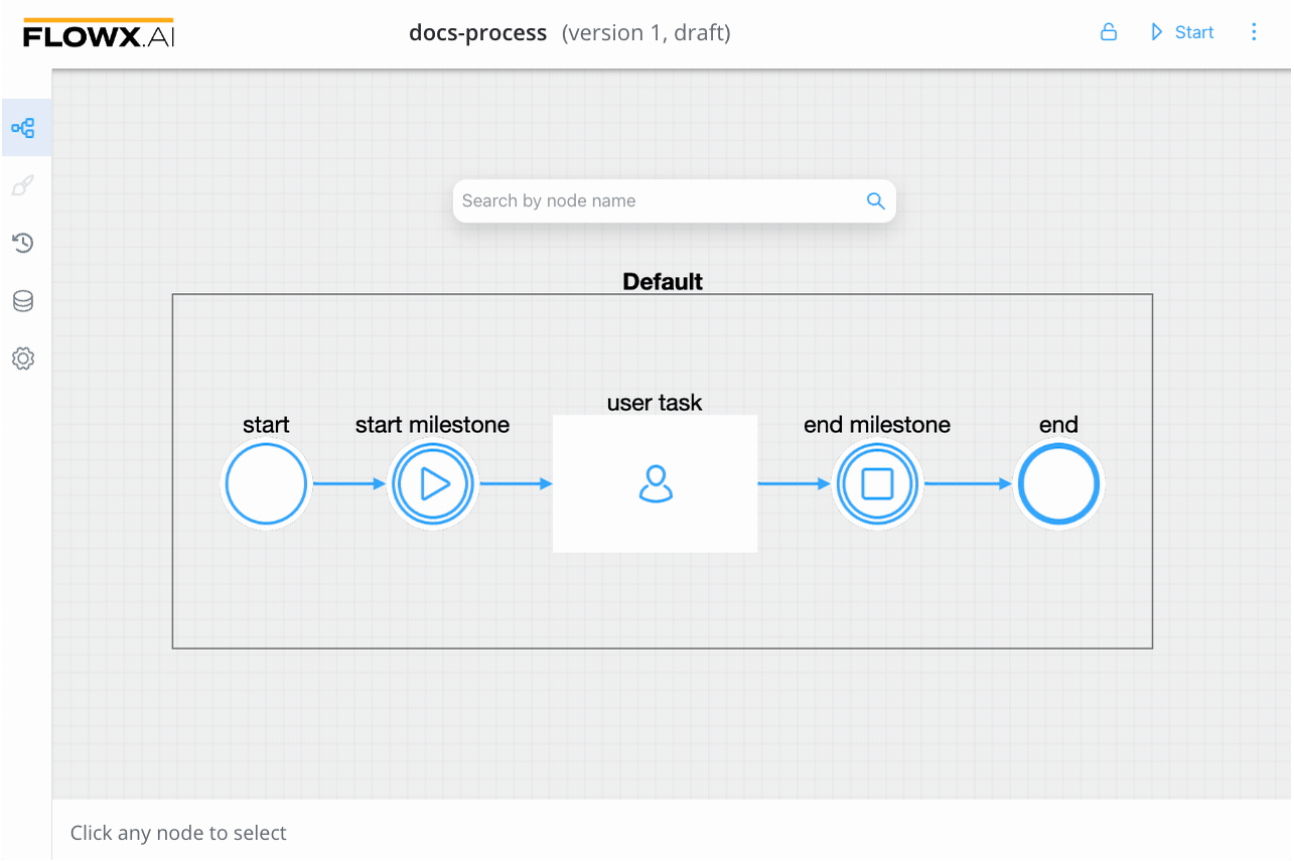
» Process instance

» Failed process start

History

In the **History** tab, you will find a record of all the modifications and events that have occurred in the process.

- **Versions** - provides information on who edited the process, when it was modified, and the version number and status
- **Audit log** - provides a detailed record of events and changes



Versions

In the **Versions** tab you will find the following details:

- Last edited on - the last time when the process was modified
- Last edited by - the last person who modified a process
- Version - version number
- Status - can be either **Published** or **Draft**

> **HINT**
>
> ! Published processes cannot be modified (they must be deprecated to be set as **Draft** before editing them).

- View process - clicking on the eye icon will redirect you to the process definition

Audit log

In the **Audit log** tab you will find the following items:

- Timestamp
- User
- Subject
- Event
- Subject Identifier
- Version
- Status

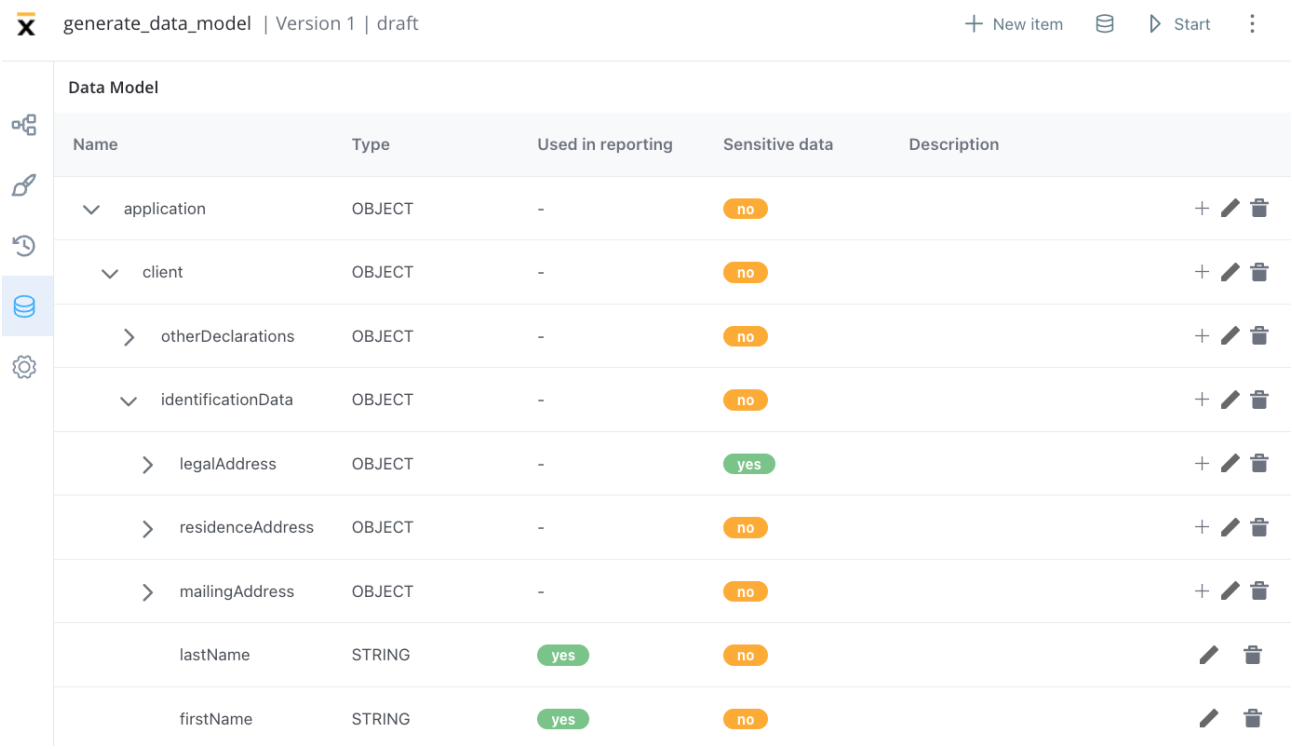
> **INFO**
>
> Some items in the Audit log are filterable, making it easy to track changes in the process.

» Audit

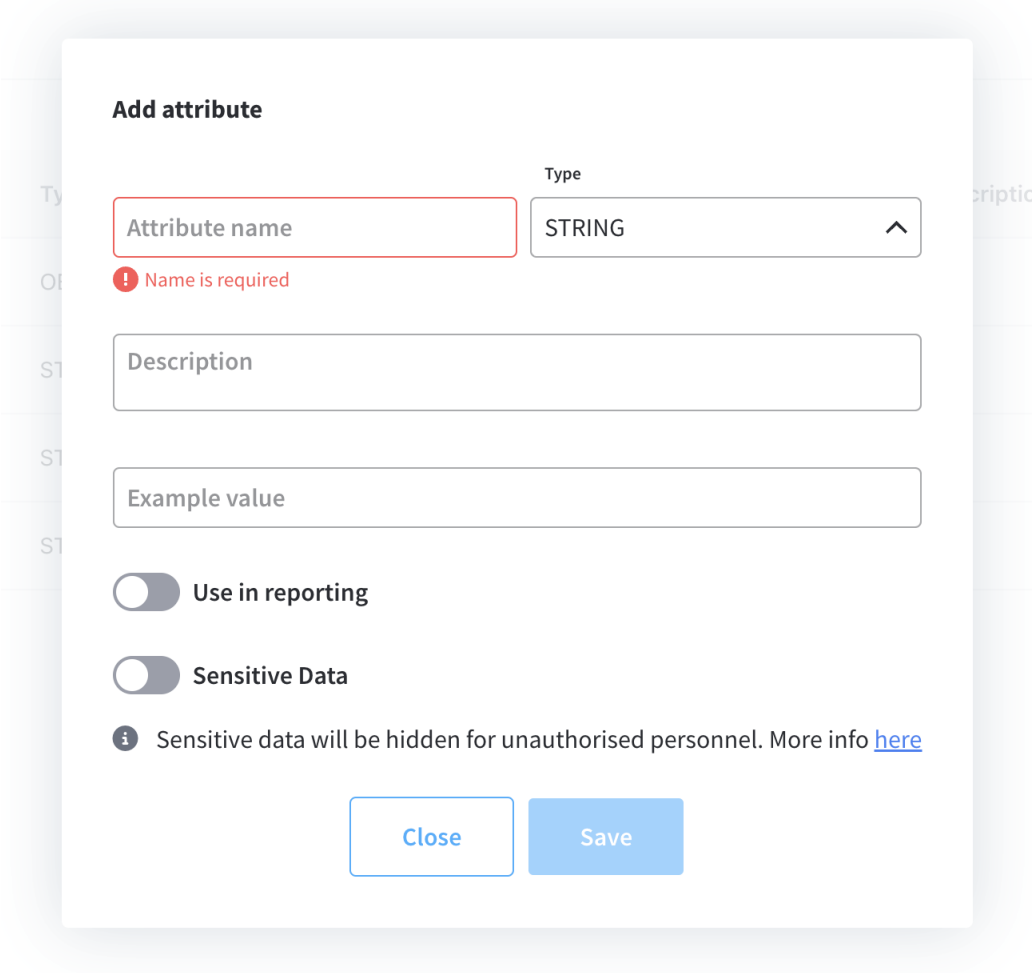
Data model

In the Data Model, you can add new key-pair values, which enables you to use shortcuts when adding new keys using the UI Designer, without having to switch

back and forth between menus.



Attributes type



The Data Model supports the following attribute types:

- STRING
- NUMBER
- BOOLEAN
- OBJECT

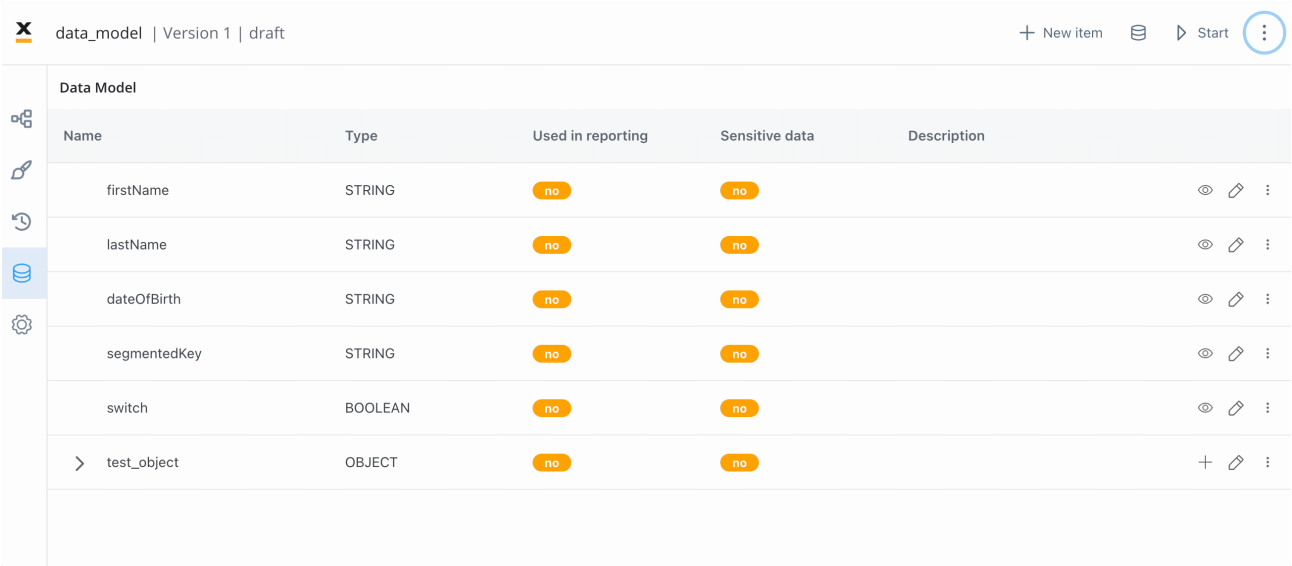
- ARRAY
  - ARRAY OF STRINGS
  - ARRAY OF NUMBERS
  - ARRAY OF BOOLEANS
  - ARRAY OF OBJECTS
  - ARRAY OF ENUMS
- ENUM

> **INFO**
>
> When you export or import a **process definition**, the data model will be included.

Data model reference

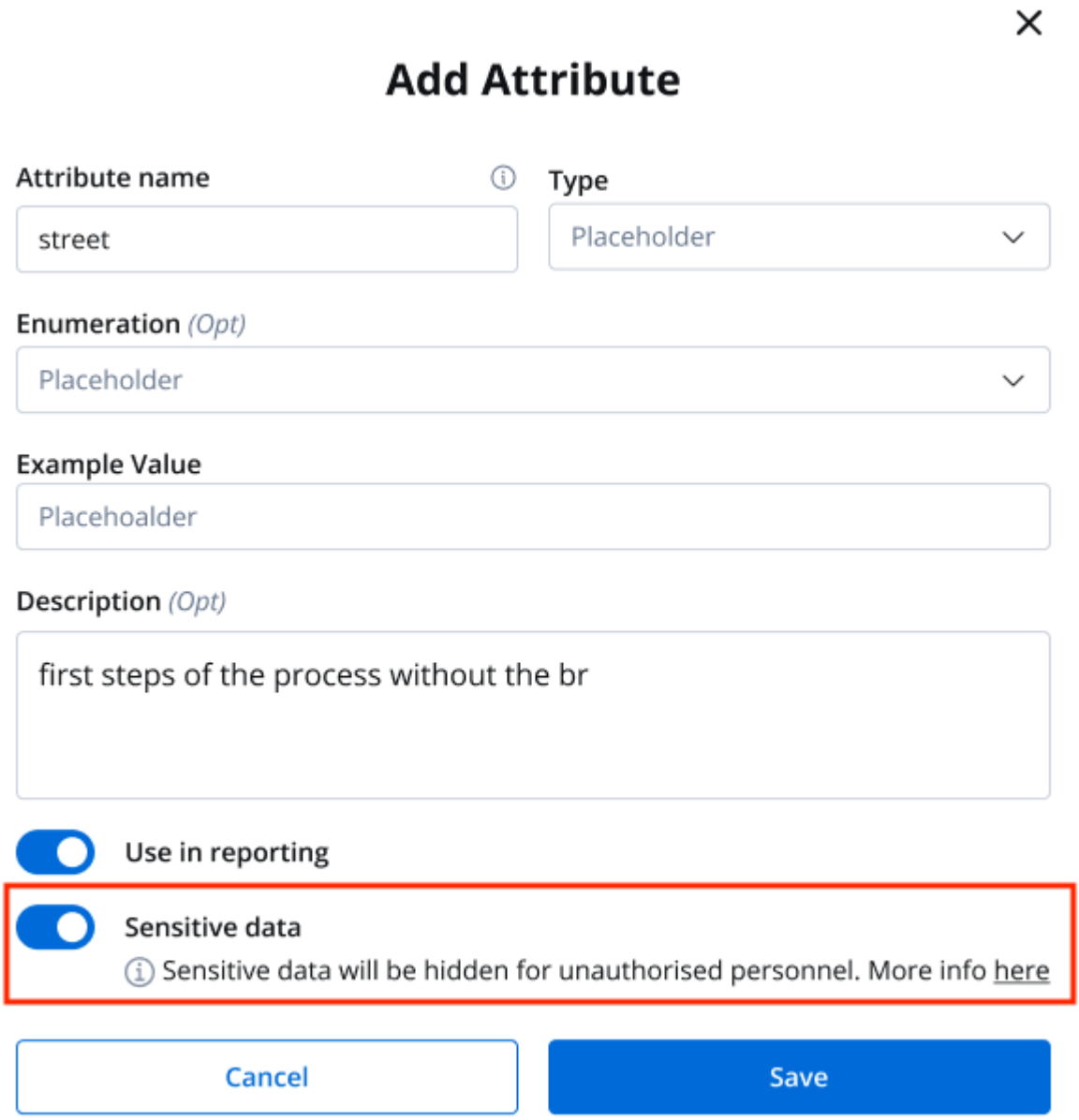
You can use data model reference feature to view attribute usage within the data model. You can now easily see where a specific attribute is being used by accessing the "View References" feature. This feature provides a list of process keys associated with each attribute and displays possible references, such as UI Elements.

For UI Elements, the references include the element label, node name, and UI Element key. Additionally, the context of the reference is provided, showing the node name and the UI element type along with its label. Users can conveniently navigate to the context by clicking the provided link to the node's UI page.



Sensitive data

To protect your data and your customer's data, you can hide data that could be visible in the process details or in the browser's console. You can now also secret data for a specific key.



Reporting

The **Use in Reporting** tag is used for keys that will be used further in the reporting plugin.

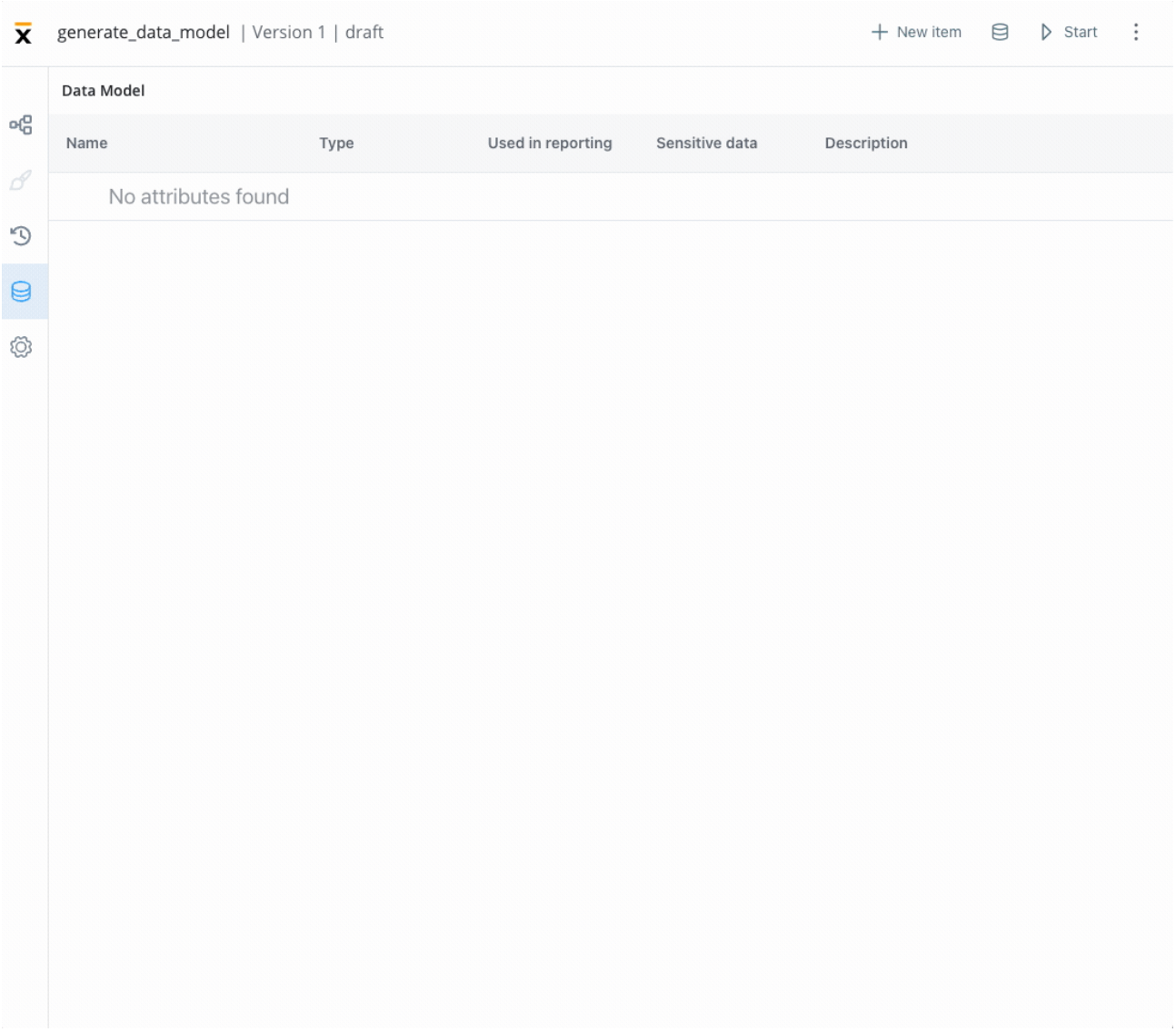
» Reporting

Generating data model

A data model can be generated using data values from a process instance. This can be done by either merging the data model with an existing one or replacing it entirely.

To generate a data model, follow these steps:

1. Open **FLOWX.AI Designer**.
2. Go to the **Definitions** tab and select the desired **process definition**.
3. Select the **Data Model** tab and then click **Generate data model** button.
4. Add the **process instance** of the process from which you want to generate the data model.
5. Choose whether to **replace** the existing data model or **merge** it with the new one.
6. Click the **Load Data** button to display the data model body.
7. Finally, click **Save** button to save the generated data model.



By generating a data model, you can ensure that your data is structured and organized in a way that is appropriate for your business needs. It can also help you to identify any inconsistencies or errors in the data, allowing you to correct them before they cause problems down the line.

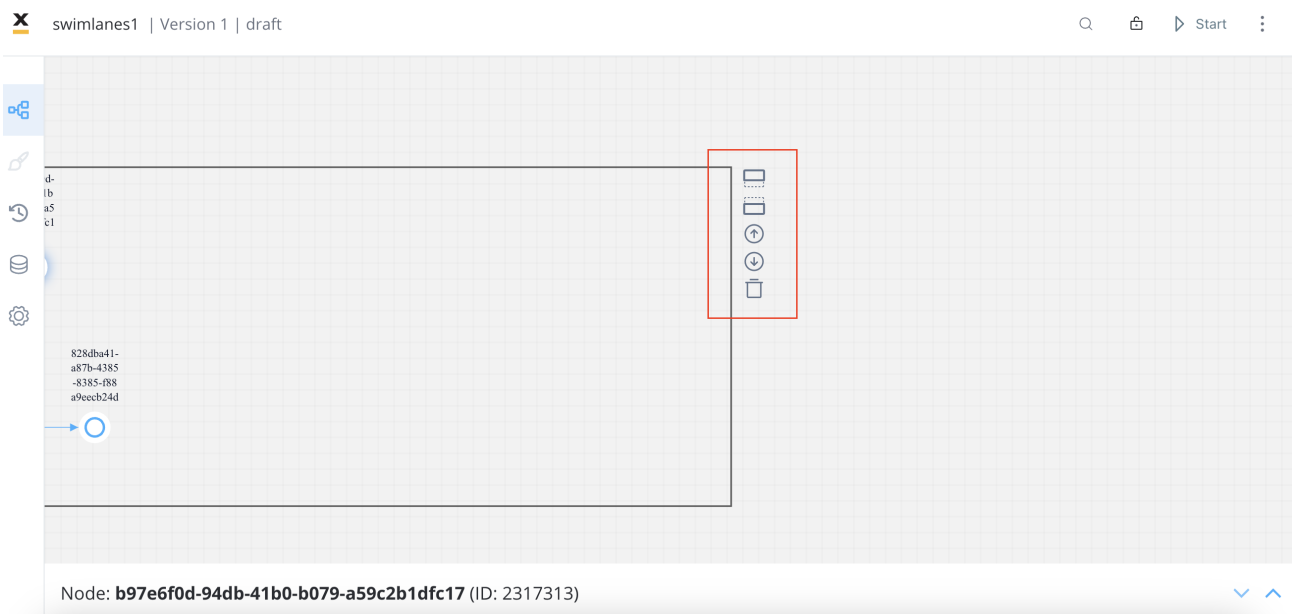
Swimlanes

Swimlanes offer a useful method of organizing process nodes based on process participants. By utilizing swimlanes, you can establish controlled access to specific process nodes for particular user roles.

Adding new swimlanes

To add new swimlanes, please follow these steps:

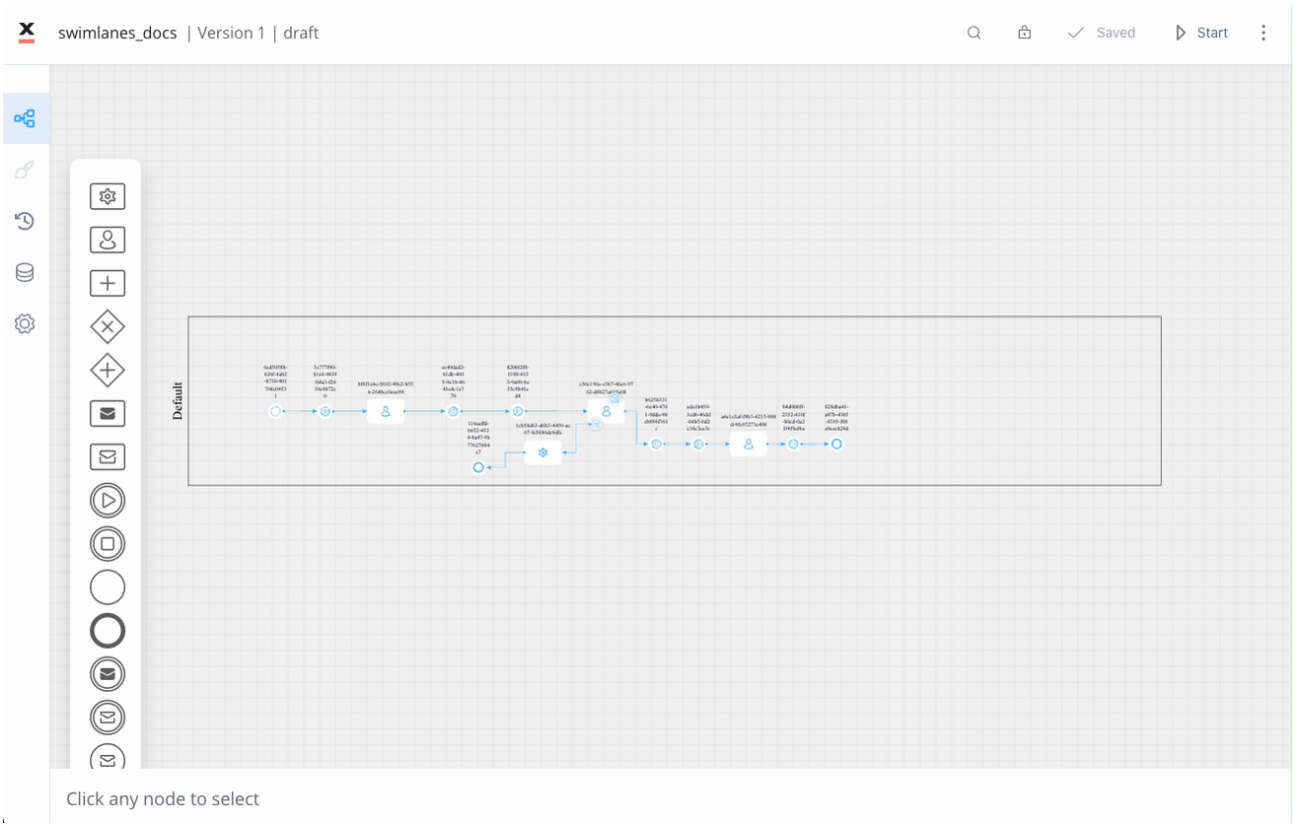
1. Access the **FLOWX.AI Designer**.
2. Open an existing process definition or create a new one.
3. Identify the default swimlane and select it to display the contextual menu.



INFO

With the contextual menu, you can easily perform various actions related to swimlanes, such as adding or removing swimlanes or reordering them.

4. Choose the desired location for the new swimlane, either below or above the default swimlane.
5. Locate and click the **add swimlane icon** to create the new swimlane.



For more details about user roles management, check the following section:

» User roles management - Swimlanes

For more details about setting up user role-based access on process definitions, check the following section:

» Configuring access roles for processes

Settings

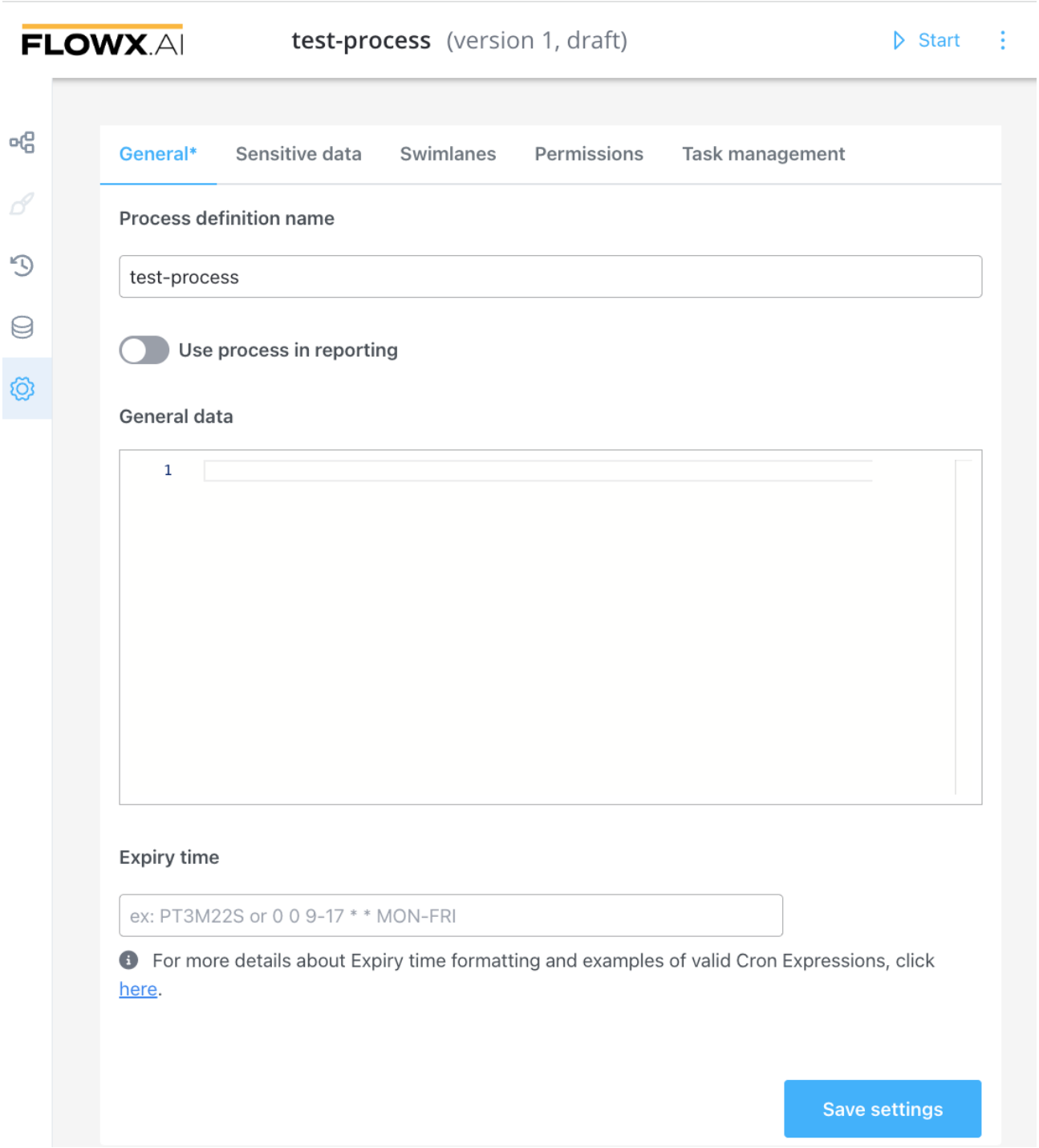
General

In the General settings, you can edit the process definition name, include the process in reporting, set general data, and configure expiry time using Cron Expressions and ISO 8601 formatting.

- **Process definition name** - edit process definition name
- **Use process in reporting** - if switched on, the process will be included in reporting
- **Use process in task management** - if switched on, tasks will be created and displayed in the Task manager plugin, more information here
- **General data** - data that you can set and receive on a response
- **Expiry time** - a user can set up a `expiryTime` function on a process, for example, a delay of 30s will be set up like: `30 16 11 4 7 1`

For more information about **Cron Expressions** and **ISO 8601** formatting, check the following section;

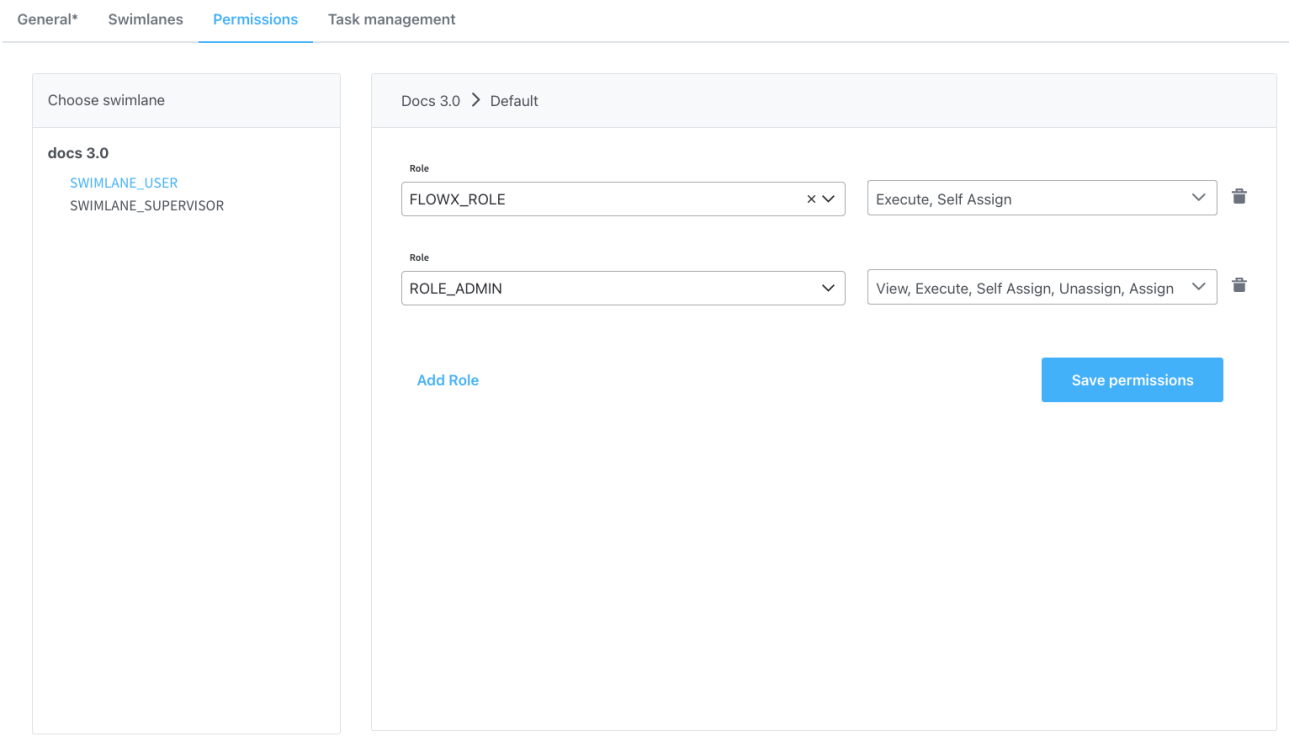
» Timer Expressions



Permissions

After defining roles in the identity provider solution, they will be available to be used in the process definition settings panel for configuring swimlane access.

When you create a new swimlane, it comes with two default permissions assigned based on a specific role: execute and self-assign. Other permissions can be added manually, depending on the needs of the user.

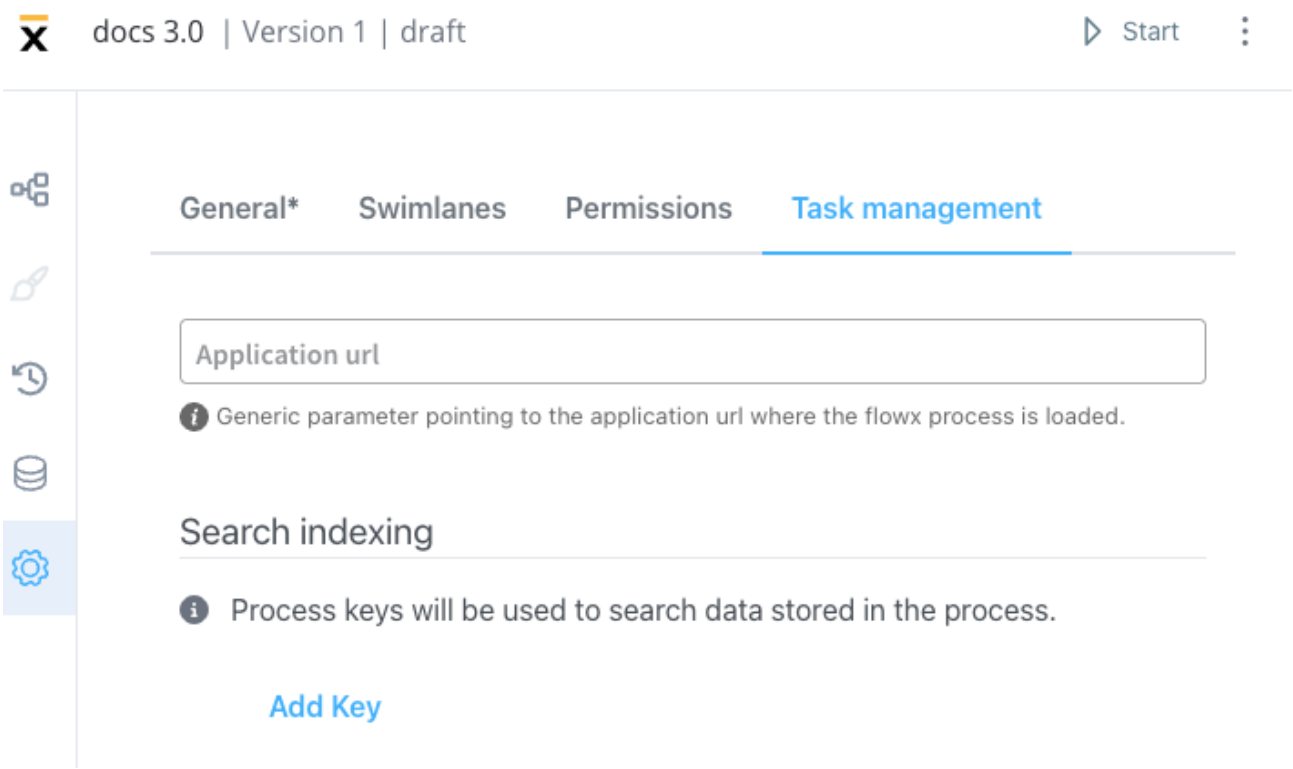


» Configuring access rights for processes

Task management

The Task Management plugin offers a business-oriented view of the process you defined in the Designer and allows for interactions at the assignment level. It also

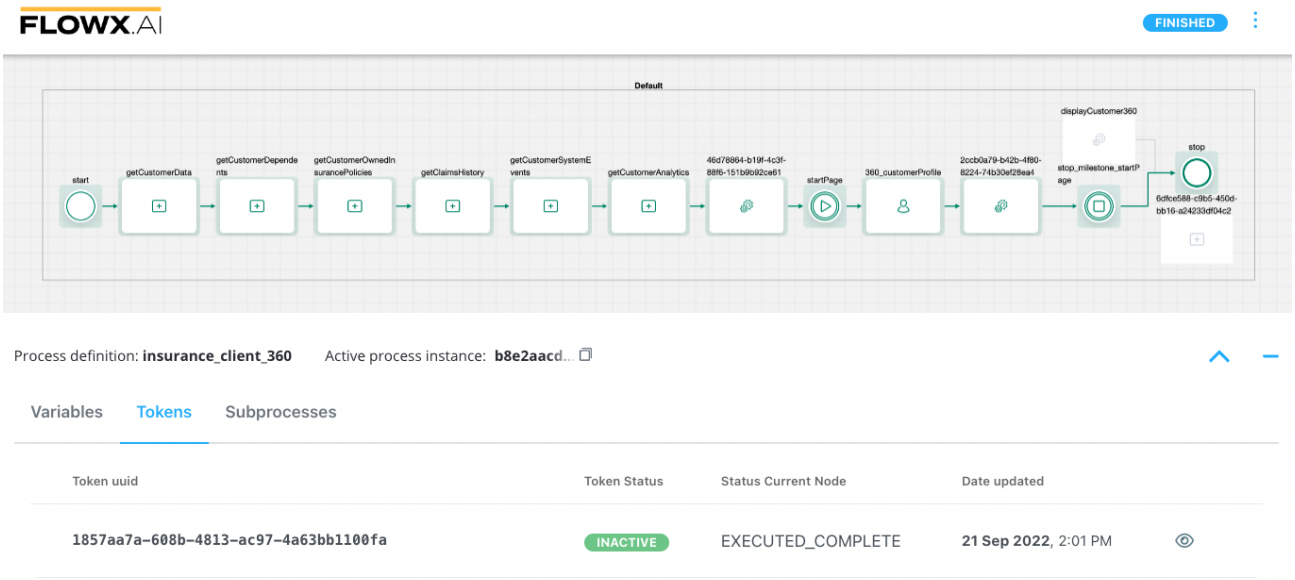
includes a generic parameter pointing to the application URL where the Flowx process is loaded and uses process keys to search data stored in the process.



Was this page helpful?

BUILDING BLOCKS / Process Designer / Active process / Process instance

A process instance is a specific execution of a business process that is defined on the FLOWX.AI platform. Once a process definition is added to the platform, it can be executed, monitored, and optimized by creating an instance of the definition.



Overview

Once the desired processes are defined in the platform, they are ready to be used. Each time a process needs to be used, for example each time a customer wants to request, for example, a new credit card, a new instance of the specified process definition is started in the platform. Think of the process definition as a blueprint for a house, and of the process instance as each house of that type being built.

The **FLOWX Engine** is responsible for executing the steps in the process definition and handling all the business logic. The token represents the current position in the process and moves from one node to the next based on the sequences and rules defined in the exclusive gateways. In the case of parallel gateways, child tokens are created and eventually merged back into the parent token.

Kafka events are used for communication between FLOWX.AI components such as the engine and integrations/plugins. Each event type is associated with a Kafka

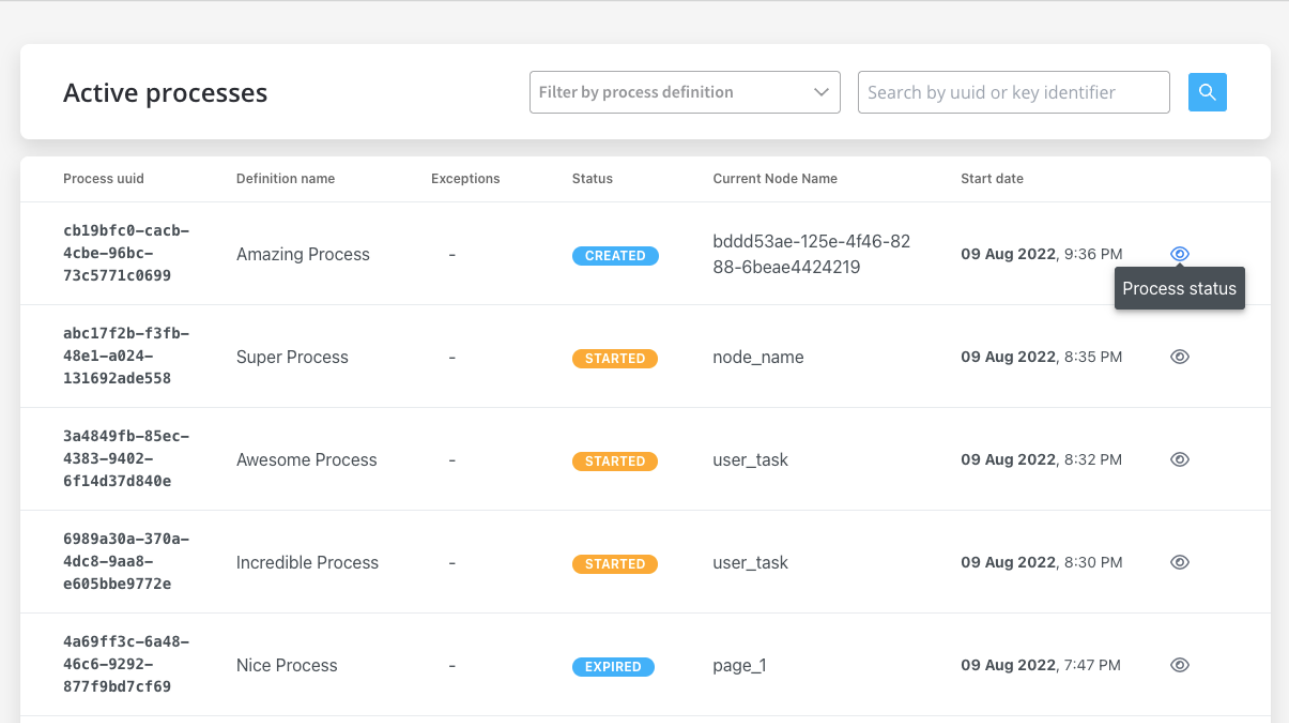
topic to track and orchestrate the messages sent on Kafka. The engine updates the UI by sending messages through sockets.

» More about Kafka

Checking the Process Status

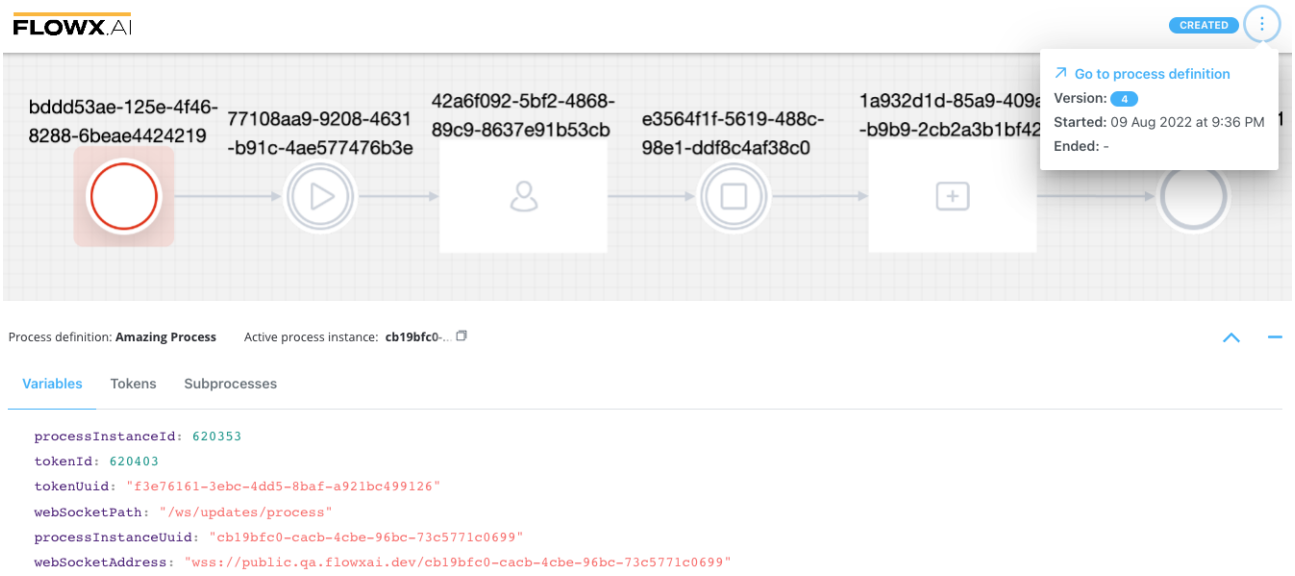
To check the status of a process or troubleshoot a failed process, follow these steps:

1. Open **FLOWX Designer**.
2. Go to **Processes** → **Active Process** → **Process instances**.
3. Click **Process status** button.



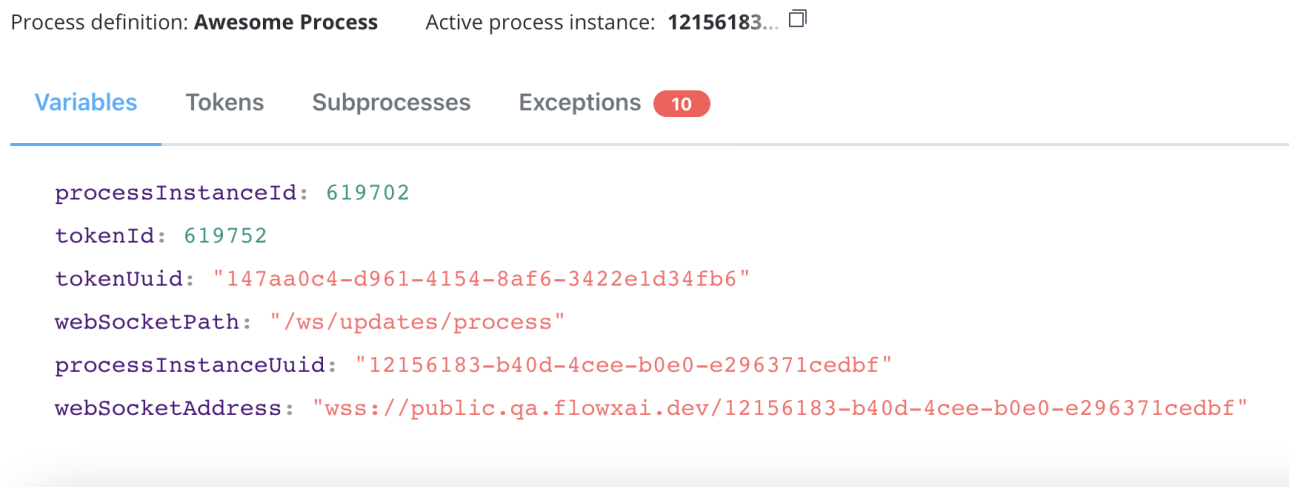
Understanding the Process Status Data

The process status data includes the following:

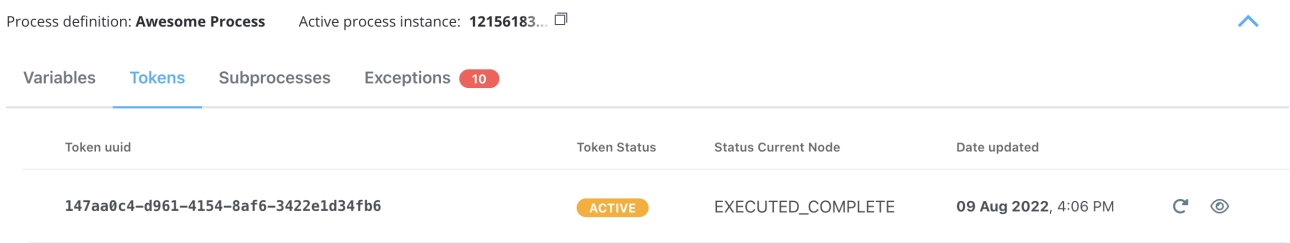


- **Status** - status of the process instance, possible values:
  - CREATED - the status is visible if there is an error in the process creation. If there is no error, the "Started" status is displayed.
  - STARTED - indicates that the process is currently running
  - DISMISSED - the status is available for processes with subprocesses, it is displayed when a user stops a subprocess
  - EXPIRED - the status is displayed when the "expiryTime" field is defined in the process definition and the defined time has passed.
  - FINISHED - the process has successfully completed its execution
- **Process definition** - the name of the process definition
- **Active process instance** - the UUID of the process instance, with a copy action available

- **Variables** - displayed as an expanded JSON



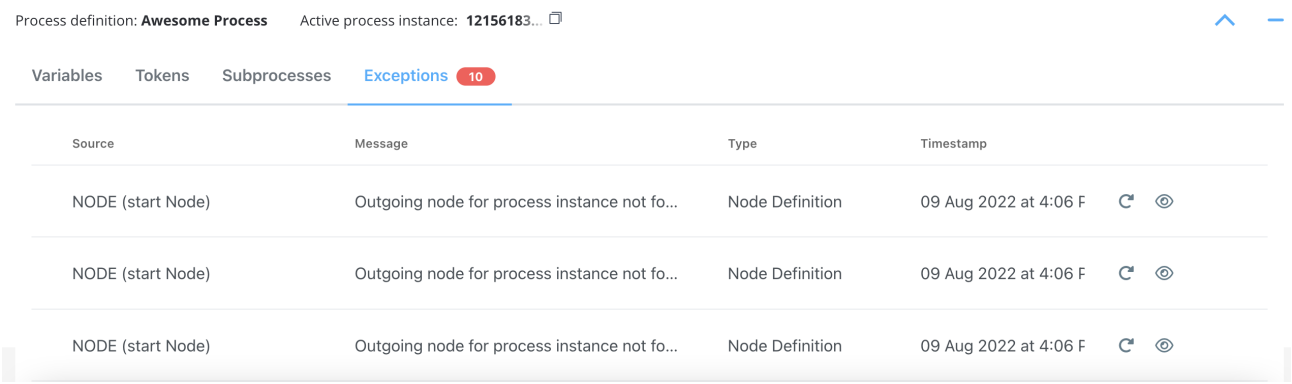
- **Tokens** - a token represents the state within the process instance and describe the current position in the process flow



> **INFO**
>
> For more information about token status details, here.

- **Subprocesses** - ! displayed only if the current
  The fallback content to display on prerendering
  generated a subprocess instance

- **Exceptions** - errors that let you know where the process is blocked, with a direct link to the node where the process is breaking for easy editing

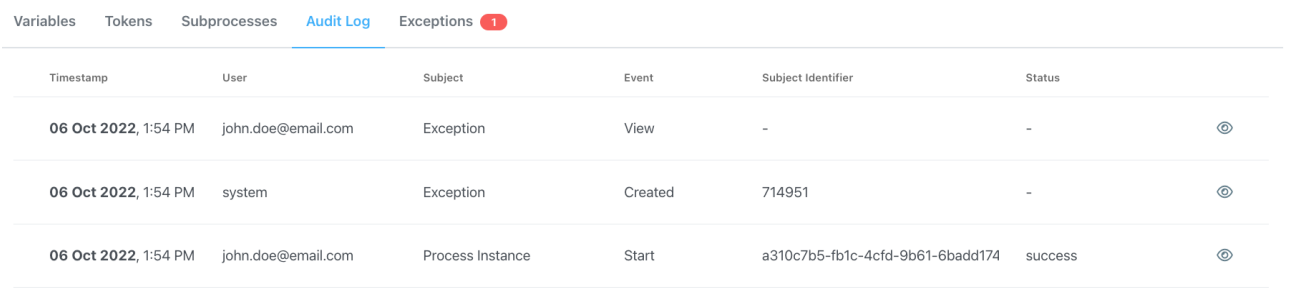


INFO

For more information on token status details and exceptions, check the following section:

» Failed process start

- **Audit Log** - the audit log displays events registered for process instances, tokens, tasks, and exceptions in reverse chronological order by timestamp

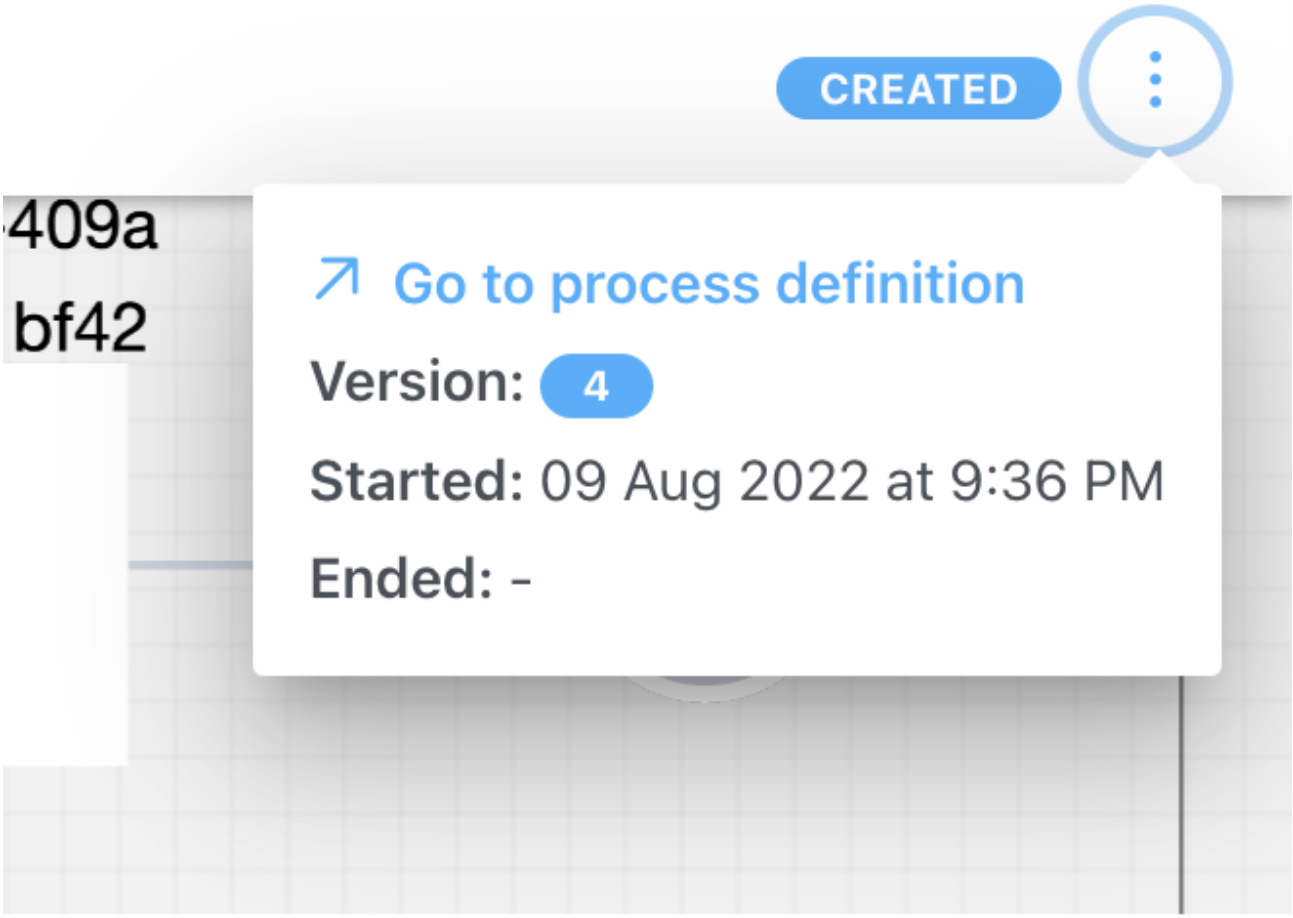


» Audit

Process menu

In the breadcrumb menu (top-right corner), you can access the following:

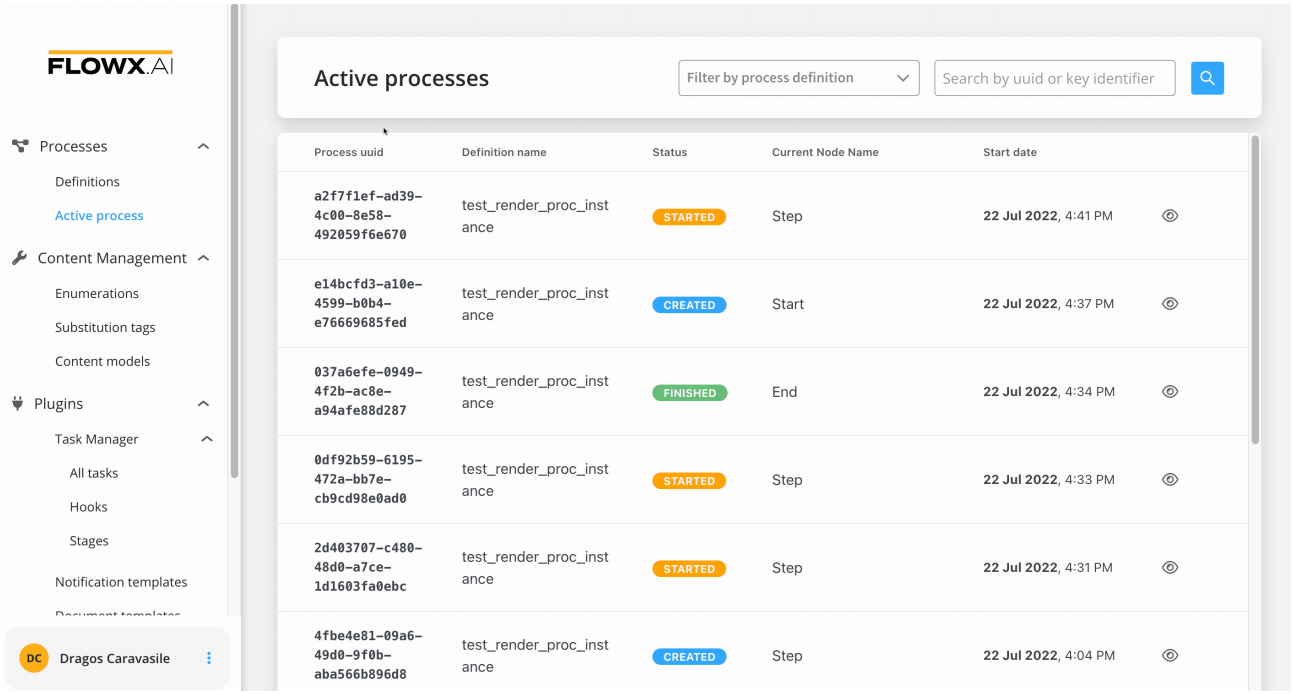
- **Go to process definition** - opens the process for editing
- **Version** - version of the process definition
- **Started** - timestamp for when the process instance started
- **Ended** - timestamp for when the process instance ended



Color coding

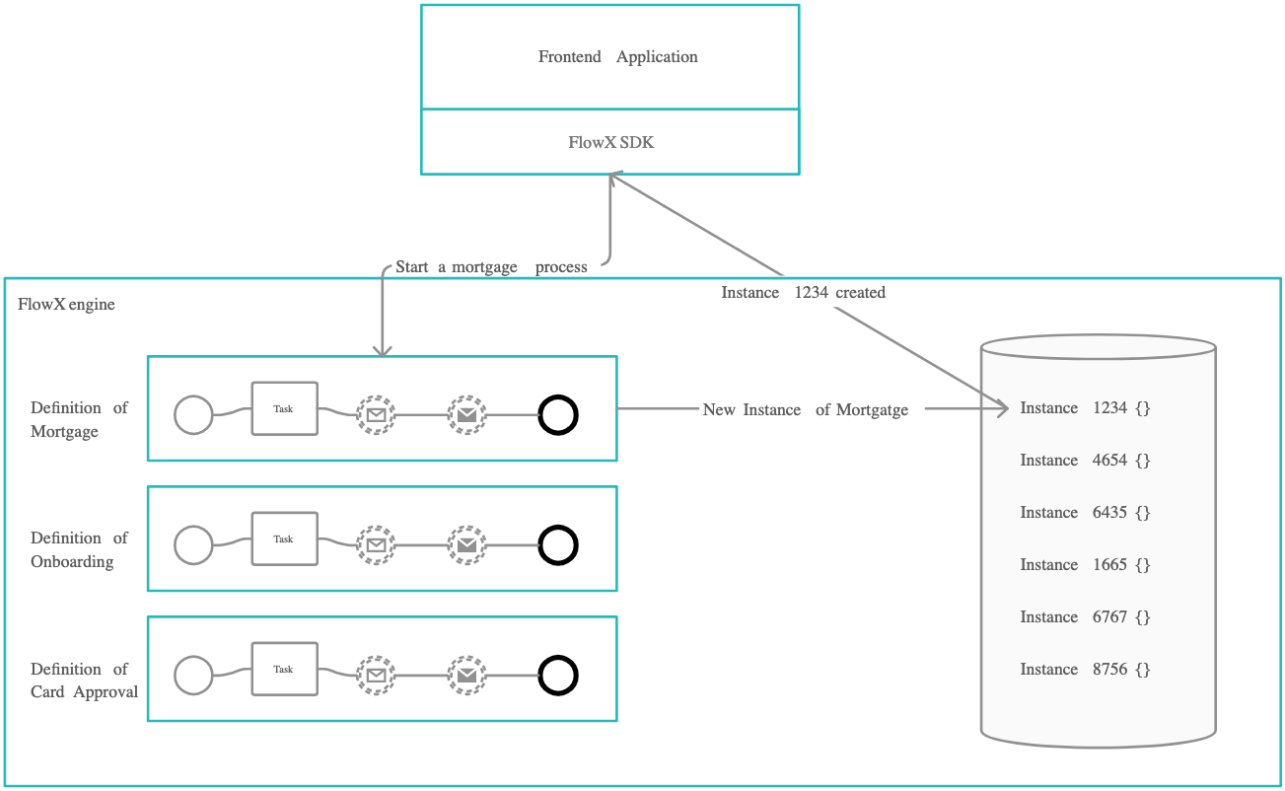
In the **Process Status** view, some nodes are highlighted with different colors to easily identify any failures:

- **Green** - nodes highlighted with green mark the nodes passed by the token
- **Red** - the node highlighted with red marks the node where the token is stuck (process failure)



Starting a new process instance

To start a new process instance, a request must be made to the FLOWX Engine. This is handled by the web/mobile application. The current user must have the appropriate role/permission to start a new process instance.



To be able to start a new process instance, the current user needs to have the appropriate role/permissions:

» Configuring access roles for processes

When starting a new process instance, we can also set it to inherit some values from a previous process instance.

Troubleshooting possible errors

If everything is configured correctly, the new process instance should be visible in the UI and added to the database. However, if you encounter issues, here are

some common error messages and their possible solutions: Possible errors include:

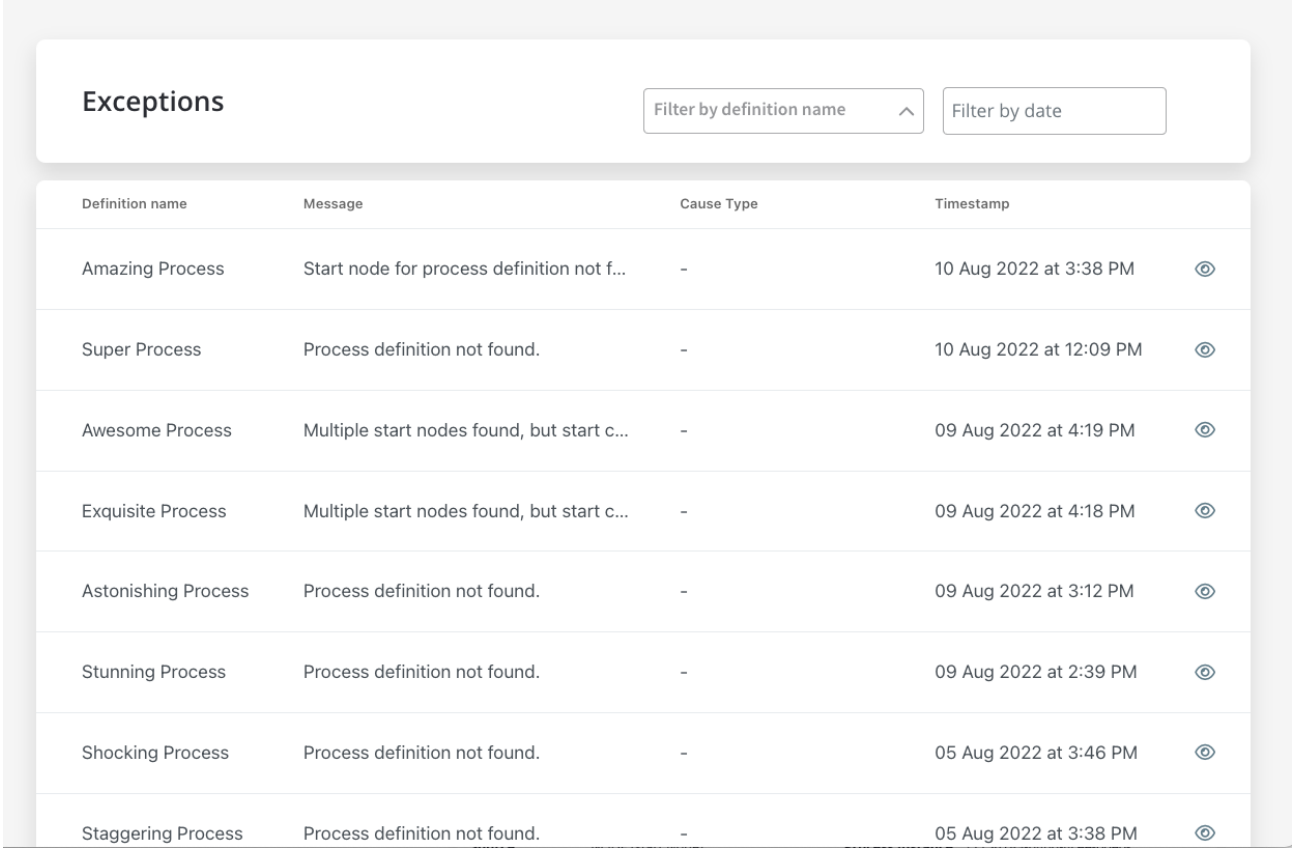
Error Message	Description
"Process definition not found."	The process definition with the requested name was not set as published.
"Start node for process definition not found."	The start node was not properly configured.
"Multiple start nodes found, but start condition not specified."	Multiple start nodes were defined, but the start condition to choose the start node was not set.
"Some mandatory params are missing."	Some parameters set as mandatory were not included in the start request.
`HTTP code 403 – Forbidden`	The current user does not have the process access role for starting that process.
`HTTP code 401 – Unauthorized`	The current user is not logged in.

Was this page helpful?

BUILDING BLOCKS / Process Designer / Active process / Failed process start

Exceptions

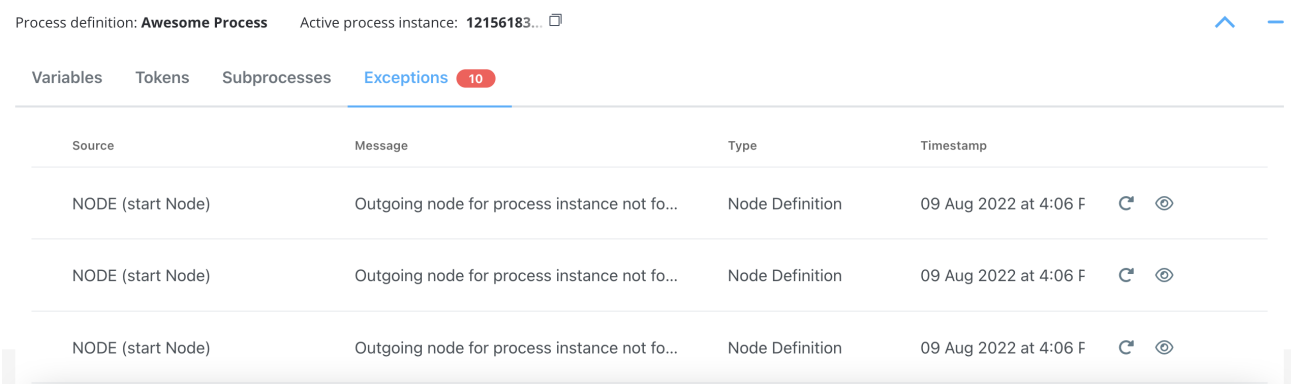
Exceptions are types of errors meant to help you debug a failure in the execution of a process.



Exceptions can be accessed from multiple places:

- **Failed process start** tab from **Active process** menu in FLOWX Designer

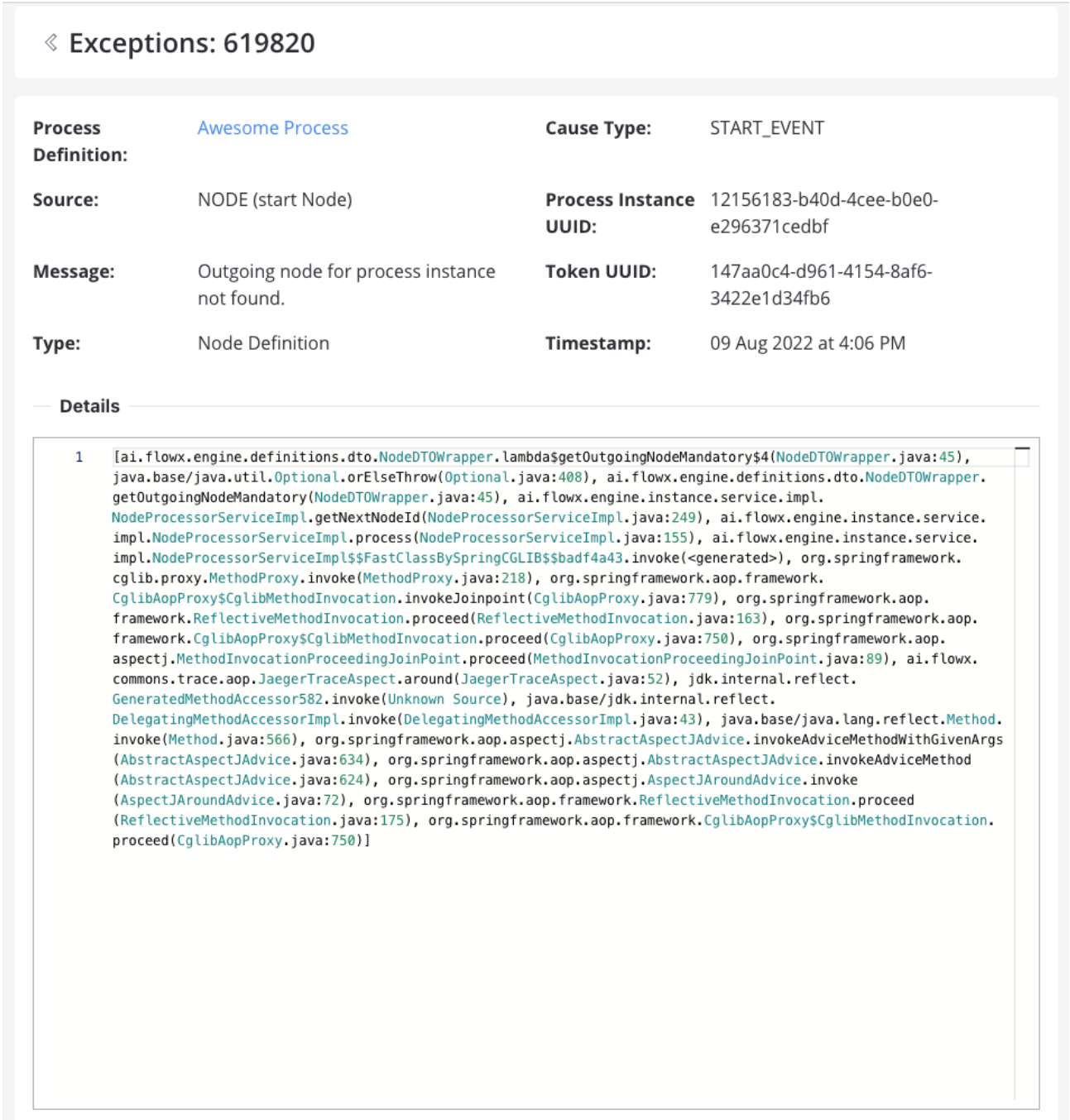
- **Process Status** view, accessible from **Process instances** list in FLOWX Designer



> **CAUTION**
>
> If you open a process instance and it does not contain exceptions, the **Exceptions** tab will not be displayed.

Exceptions data

When you click **view** button, a detailed exception will be displayed.



- **Process Definition** - the process where the exception was thrown
- **Source** - the source of the exception (see the possible type of sources below)
- **Message** - a hint type of message to help you understand what's wrong with your process

- **Type** - exception type
- **Cause Type** - cause type (or the name of the node)
- **Process Instance UUID** - process instance unique identifier
- **Token UUID** - token unique identifier
- **Timestamp** - default format: `yyyy-MM-dd'T'HH:mm:ss.SSSZ`
- **Details** - stack trace (a **stack trace** is a list of the method calls that the process was in the middle of when an **Exception** was thrown)

Possible sources:

- Action
- Node
- Subprocess
- Process Definition

Exceptions type

Based on the exception type, there are multiple causes that could make a process fail. Here are some examples:

Type	Cause
Business Rule Evaluation	when executing action rules fails for any reason
Condition Evaluation	when executing action conditions

Type	Cause
Engine	when the connection with the database fails when the connection with Redis fails
Definition	misconfigurations: process def name, subprocess parent process id value, start node condition missing
Node	when an outgoing node can’t be found (missing sequence etc)
Gateway Evaluation	when the token can’t pass a gateway for any reason, possible causes: • missing sequence/node • failed node rule
Subprocess	exceptions will be saved for them just like for any other process, parent process ID will also be saved (we can use this to link them when displaying exceptions)

Was this page helpful?

BUILDING BLOCKS / Process Designer / Subprocess

Sub-processes are smaller process flows that can be triggered by actions in the main process. They can also inherit some process parameter values from the parent process and send their results back to the parent process when they are completed. The subprocesses will communicate with the front-end apps using the same connection details as their parent process.

They can be started in two ways:

- **asynchronous** - they will execute alongside the parent process since the parent process does not need to wait for the sub-process to end
- **synchronous** - the parent process will wait until the sub-processes are finished before advancing

Configuring & starting subprocesses

The sub-processes will be designed in the same way as the main process, by using the FLOWX Designer.

They can be started by a parent process in one of two ways:

- by using a StartSubprocess action inside any of the task nodes in the process
- by adding a custom Subprocess Run node type in the process

In both cases, by default, the sub-process will inherit all the parent process parameter values. It can be configured to inherit only some parameter values from

its parent. The available action parameters for this are:

- *paramsToCopy* - choose which of the keys from the parent process parameters to be copied to the sub-process
- *withoutParams* - choose which of the keys from the parent process parameters are to be ignored when copying parameter values from the parent process to the sub-process

Sub-processes can have an action configured on them which will append their results to the parent process parameter values.

Executing subprocesses

The sub-processes can be started in async or sync mode, by setting a specific action parameter, named *startedAsync,* on the action that triggers the subprocess.

If the subprocesses are started in sync mode, they will notify the parent process when they are completed and the parent process will handle receiving the process data from the child and resuming its flow.

Was this page helpful?