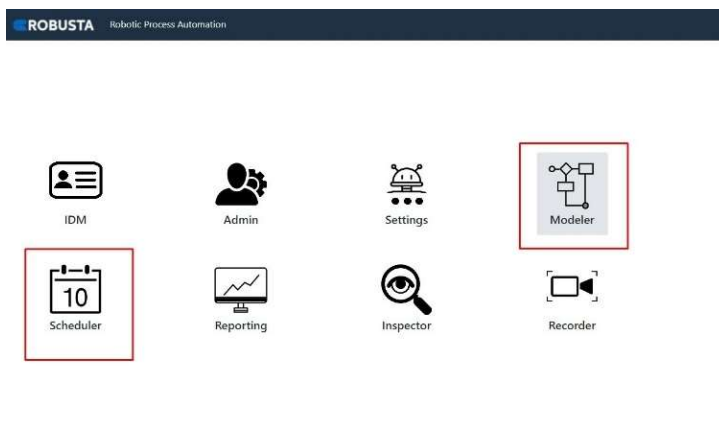


# Introduction to sample process design with Robusta RPA

In this paper we will show you the use of the Robusta RPA design studio and its interfaces by designing a simple RPA (Robotic Process Automation) process.

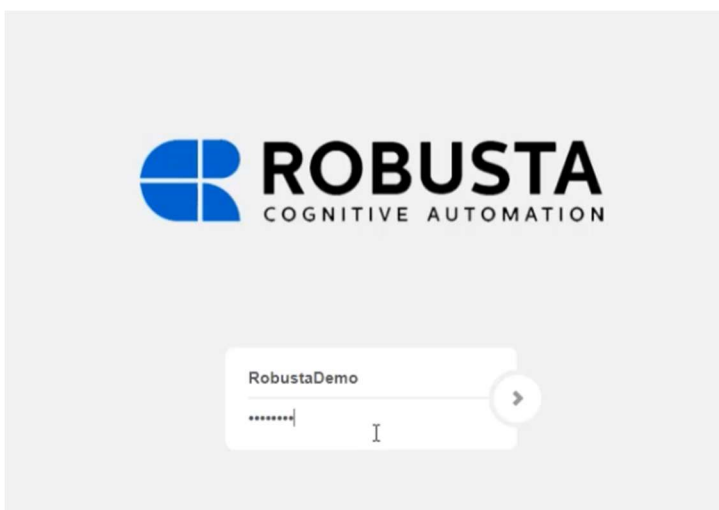
- In this sample process, we will open the Youtube website and wait for 5 seconds and then close it.



**\*Modeler(Design Studio) :** Robusta design studio is the main component in which business process are trained to robots. It offers process design with no code approach.

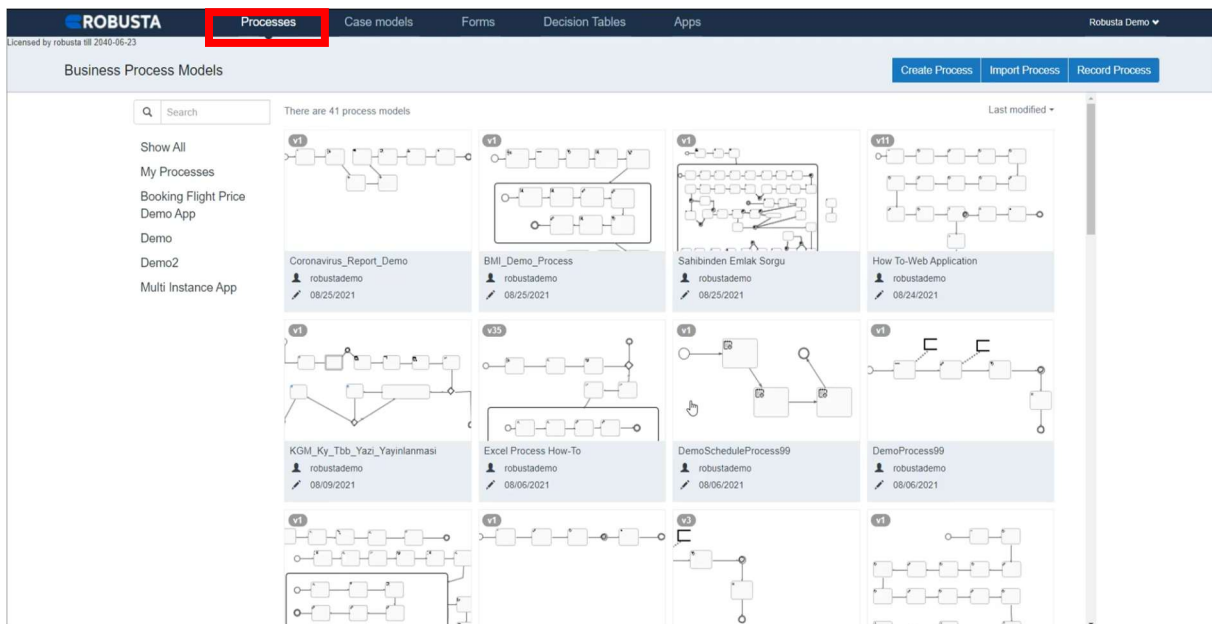
**\*Scheduler :** With scheduler you can manage robots time based, prioritize processes among robots.

[Community Edition](#)  
[Modeler\(Design Studio\) and Scheduler](#)

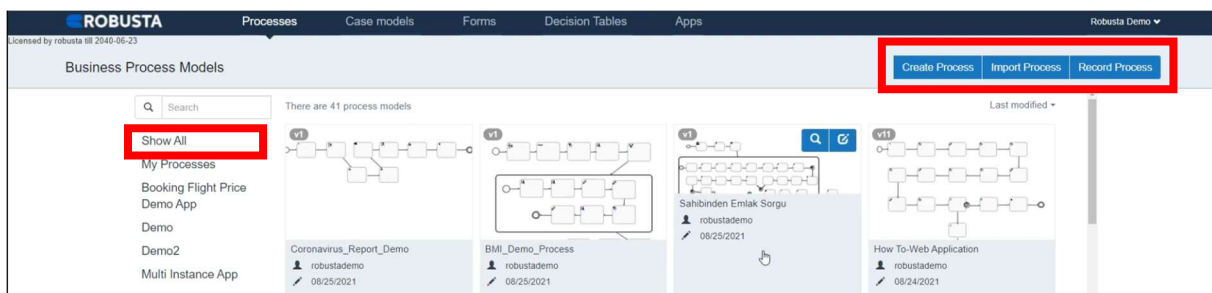


**\*Use the same username and password used to connect Robusta orchestrator.**

- The Processes tab that appears when you log in contains the menus to create, import, search and edit processes. On this page, you can see that all previously designed processes are listed.



- You can use the "Show All" option to list all processes by removing any applied filter. The Create Process button in the upper right corner allows to create a new process, and the Import Process button to import the processes that we have exported from other Robusta Orchestrator environments.



- To create a process, you need to click Create Process button, fill necessary fields and then, click Create New Model button. In the "Model name" field, we give our process a short descriptive name. This name is displayed by end users on the process start and monitoring pages and can be updated at any time.
- In the "Model key" field, we enter a key value for our process. When calling our process as a subprocess from another process, we use the "Model key" value. If the "model key" value changes, the processes that call this process must also be updated. That's why we don't usually change the "Model key" value after we create our process. "Model key" is not displayed by end users on process start and monitoring pages. Please be aware that special characters and spaces are not allowed when naming the model key.
- In the Priority field, we can assign a priority value to our process by entering a positive number. Doing so, we can determine which of the jobs waiting in the queue will be processed first. If value "1" is assigned high assumes highest priority process, while a larger number should be entered for lower priority processes. In this process, we will leave this field with the default value of 50.
- In the Human effort field, we can enter the information about how long it takes a human to perform the operations in our process.

# Create a new business process model

You need to give a name for the new model and you may want to add a description at the same time

Model name

Model key

Priority

Human effort (in minutes. E.g. 4.5)

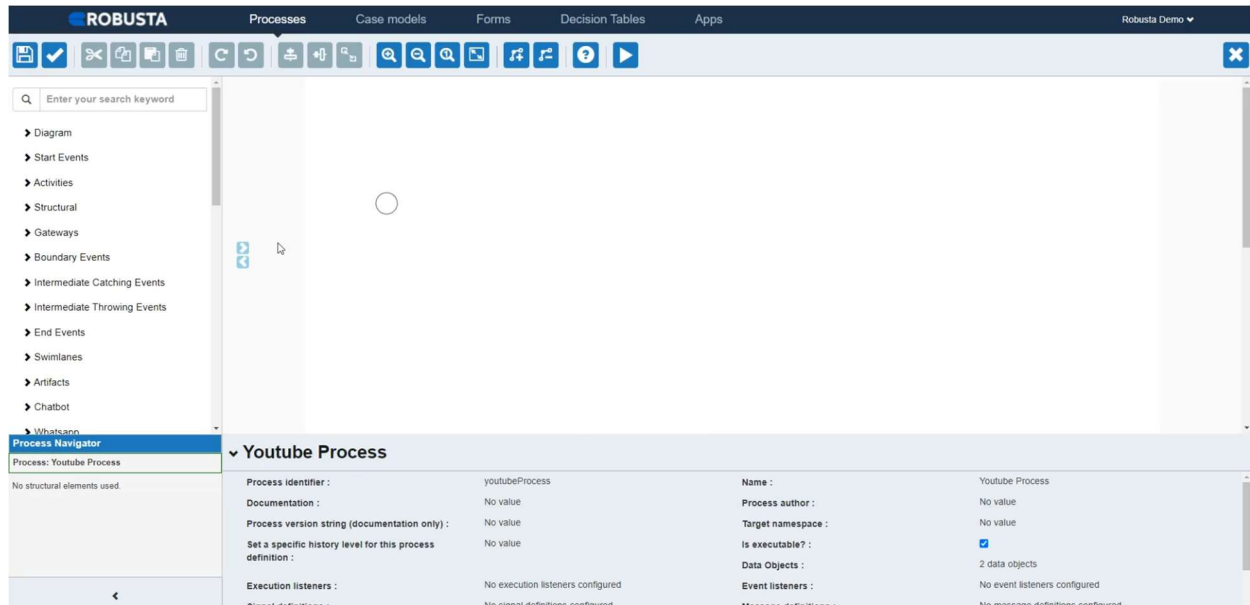
Description

I

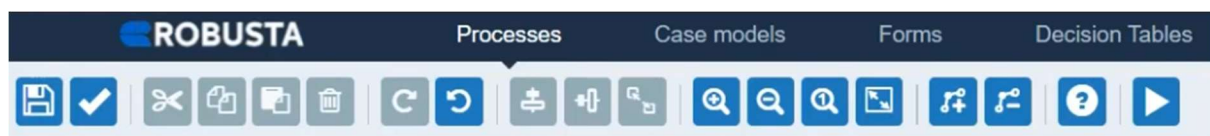
Cancel

Create new model

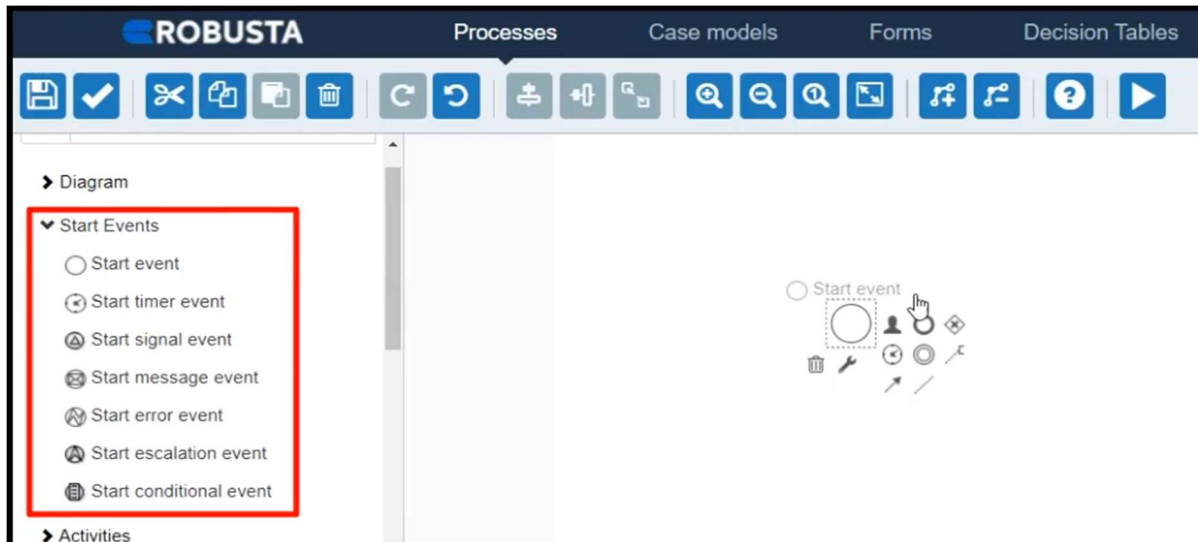
- The following screenshot shows process design studio. On the left pane, you see BPM (Business Process Mapping) and RPA (Robotic Process Automation) components all together. On the right side, you see the canvas where we generate process flow.



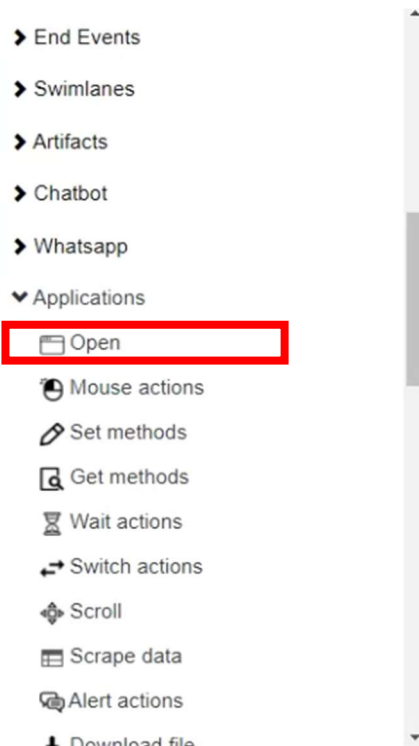
- At the top, you can see the toolbar with shortcuts. By using the buttons in this section, you can perform operations such as cutting, copying, pasting, deleting the activities in the process flow, undoing what you have done, zooming in and out of the design area. Also, you can run your processes by clicking the "Run" button here for testing purposes.



- Processes should include only one start activity and at least one end activity. When you need to add it, you can add this activity to your process by drag and drop of “Start event” activity under the Start Events section.



- As the next activity in our process, we will use the Open activity under the Applications section to open the Youtube website. You just need to drag and drop a selected activity onto the canvas.



- To define the flow, you need to connect the activities to each other by using arrows. For this, we select the Start Event activity and drag the arrow icon from the shortcut activities displayed on the right and hover over the Open activity. When green appear around the activity, it is ok to drop the arrow.



- After the drag and drop, some mandatory fields (marked with \*) have to be filled. For example, the values for Open activity under 'Applications' may be set as given at the following table:

Name	Open : Youtube
*Application Name	openYoutube
Type	CHROME
*Url	<a href="https://www.youtube.com/">https://www.youtube.com/</a>

**P.S:** In this field special character constraints is important, only letters and numbers are recommended

**Open : Youtube**

Id :	No value	Asynchronous :	<input checked="" type="checkbox"/>
Name :	Open : Youtube	Documentation :	No value
* Application name :	openYoutube	* Type :	CHROME
Application reference :	No value	* Url :	<a href="https://www.youtube.com/">https://www.youtube.com/</a>
Wait until load :	<input checked="" type="checkbox"/>	Download directory :	No value
Timeout (sec) :	No value	Profile directory :	No value
Maximize :	<input checked="" type="checkbox"/>		

- In the next step, we want to wait 5 seconds after the Youtube website is opened and then close it. So, we will use a timer activity. You can find the timer activity in the shortcuts. Just drag and drop it to the space.



- Since we want to wait 5 seconds after the website is opened, let's write PT5S in the Time duration field.

The screenshot shows the BPMN editor interface. The workflow now includes a start event, an 'Open: Youtube' activity, and a timer activity. A blue speech bubble explains the time duration format:

- PT5S >> 5 seconds
- PT5M >> 5 minutes
- PT1H >> 1 hour

Below the workspace, the configuration panel for the 'Youtube Search Process' is visible. The 'Time duration (e.g. PT5M)' field is highlighted with a red box and contains the value 'PT5S'.

Id :	No value	Name :	
Documentation :	No value	Execution listeners :	No execution listeners configured
Time cycle (e.g. R3/PT10H) :	No value	Time date in ISO-8601 :	
Time duration (e.g. PT5M) :	PT5S		

- In the next step, we will add the Close activity in the Applications section to close the opened website. In this activity, we select the reference name of the application that we want to close from the list.



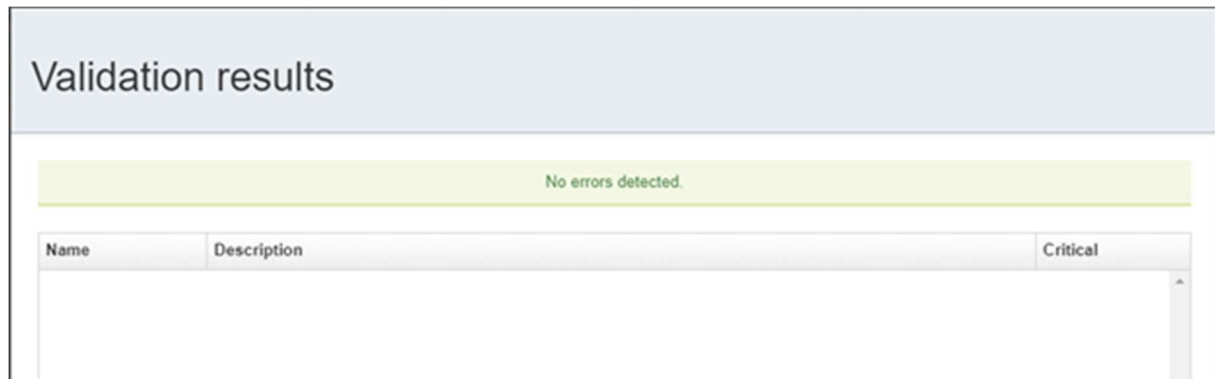
The screenshot shows the Camunda BPMN editor interface. On the left, the 'Process Navigator' pane lists various actions, with 'Close' highlighted under the 'fx Functions' category. The main canvas displays a process flow starting with an 'Open: Youtube' task, followed by a 'Close' task. The 'Close' task is currently selected, and its configuration panel is visible at the bottom. In this panel, the 'Application name' field is set to '\${openYoutube}', which is highlighted with a red rectangle. Other fields like 'Id', 'Name', 'Asynchronous', and 'Documentation' are also visible.

Name	Close
*Application Name	\${openYoutube}

- Finally, let's use an End Event to complete our process flow and validate and save our process.

This screenshot shows the final step of the process flow configuration. The 'Close' task is now followed by an End Event (represented by a thick circle). The 'Close' task's configuration panel remains open, with the 'Application name' field still set to '\${openYoutube}'. The process flow is now complete, starting from the initial event, through the 'Open: Youtube' task, then the 'Close' task, and finally reaching the End Event.

- After you completed the process flow, validate the model by clicking validate (✓) button.

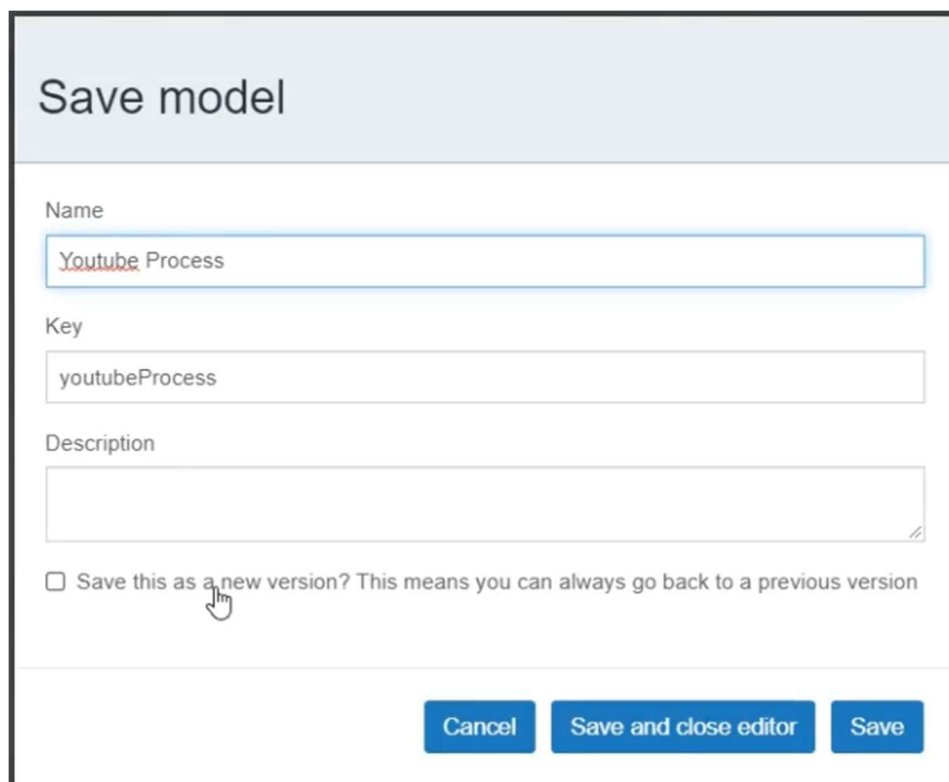


Validation results

No errors detected.

Name	Description	Critical
------	-------------	----------

- After you validated the process flow, save the model by clicking save (💾) button. If we select the "Save this as a new version" option here, and save our process, we will create a new version by preserving the last saved version of our process. Then, we will have the chance to revert to an old version whenever we want.



Save model

Name  
Youtube Process

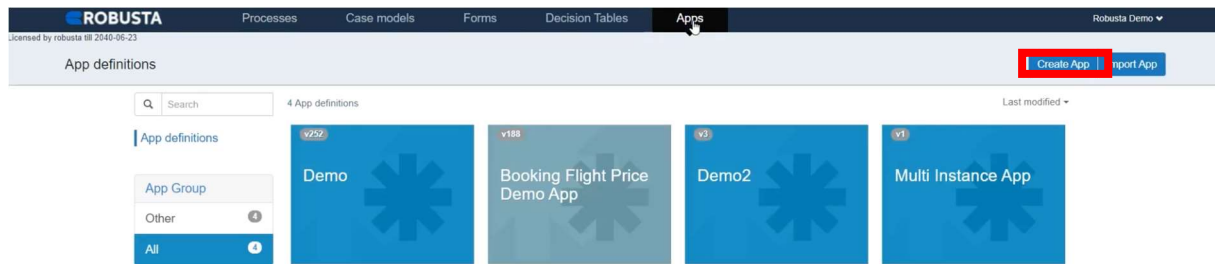
Key  
youtubeProcess

Description

☐ Save this as a new version? This means you can always go back to a previous version

Cancel Save and close editor Save

- After saving a process, this process should be included in an App to be run. In Apps tab, you will find the menus to create, import, edit apps.



- To create a App, you need to click Create App button, fill necessary fields and then, click Create New App Definition button.

## Create a new app definition

You need to give a name for the new app definition and you may want to add a description at the same time.

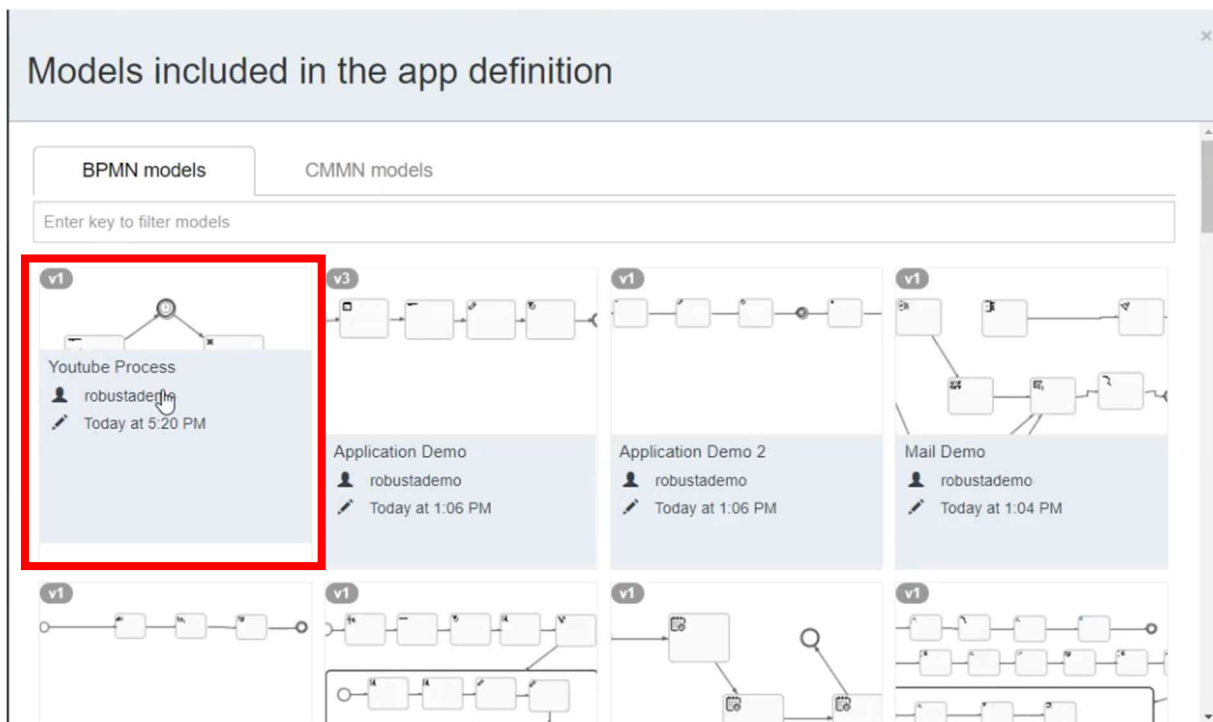
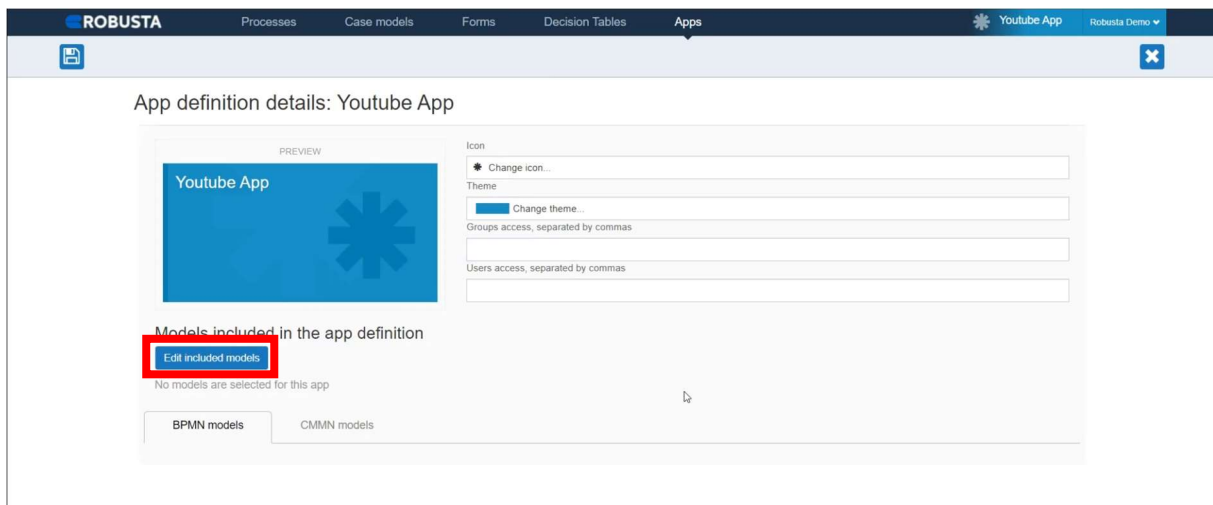
App definition name


App Group (Not Required)

App definition key

Description

- Click 'Edit included models' button and select the process to be included in your app(check blue mark when selected). Then, click any space out of popup window to close it.



- Let's save our process by clicking the save button (  ) at the top left. At the same time, since we need to publish the App we created, let's click the Save button after selecting the Publish option.

### Save app definition

You need to give a name for the new app definition and you may want to add a description at the same time.

App definition name

App Group (Not Required)

App definition key

Description

☒ Publish? Note that if publish is enabled, this app definition will be versioned and the workflow app will be updated if existing already.

Cancel Save and close editor **Save**

- You can also publish an App by clicking the “Publish” button after clicking the App you created at the bottom of the Apps page.

**ROBUSTA** Processes Case models Forms Decision Tables **Apps** Robusta Demo

v2 Youtube App [Show all definitions](#) [Import App](#) **Publish** [App Editor](#)

Created by robustademo  
Last updated by robustademo - Today at 5:24 PM

This app definition has no description. Modify the app definition properties to add one

App definition details: Youtube App

PREVIEW

Youtube App

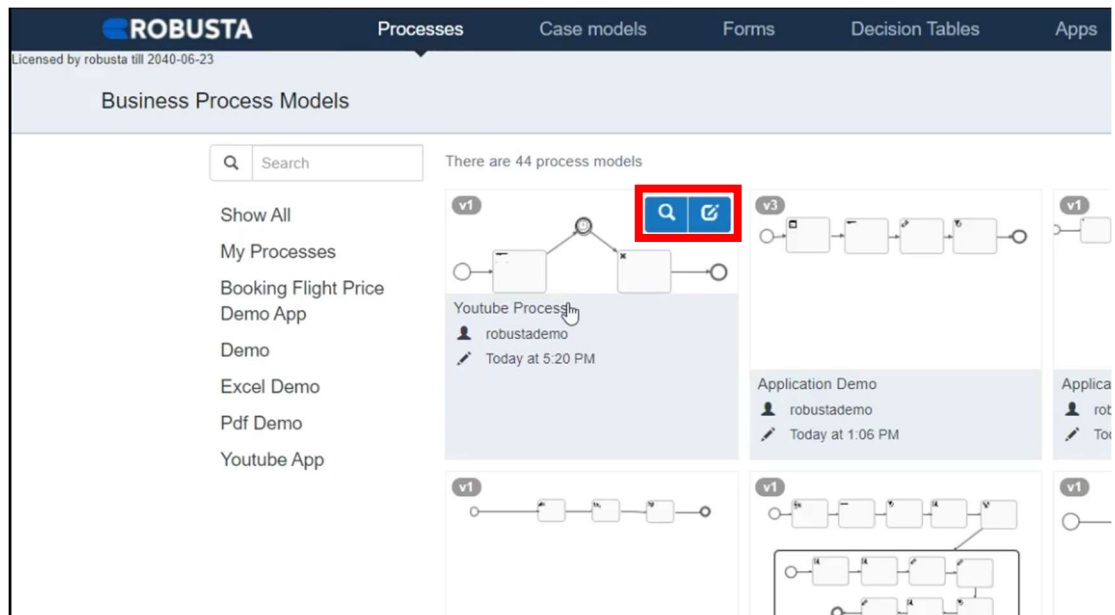
Models included in the app definition

BPMN models CAMUN models

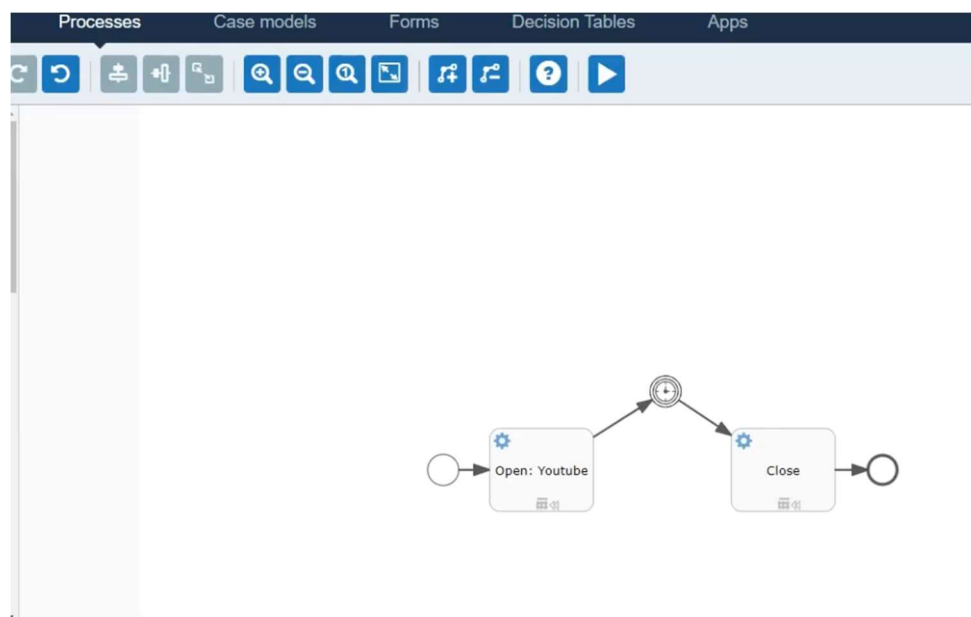
v1

Youtube Process

- Let's go back to the Processes tab. We see our process in the first place since it is the last updated process. When we hover over our process, two buttons are displayed at the top right. By pressing the button on the right, we can access the process design editor directly. By clicking the button on the left or clicking anywhere on the box, we can view the details of our process.



- To run the process, select your process and click play (▶) button.



- Select your app and your worker on the popup screen. Then click “Run Process” button. Once you see ‘Process Successfully scheduled’ message on the screen, your process is scheduled will be picked by Scheduler to be sent to your worker. Also it is important to select “App Publish Required” check box, to be sure newly Published version will be Run.

### Schedule & Run

Application

Youtube App

Worker

robustademo@robusta.com.tr

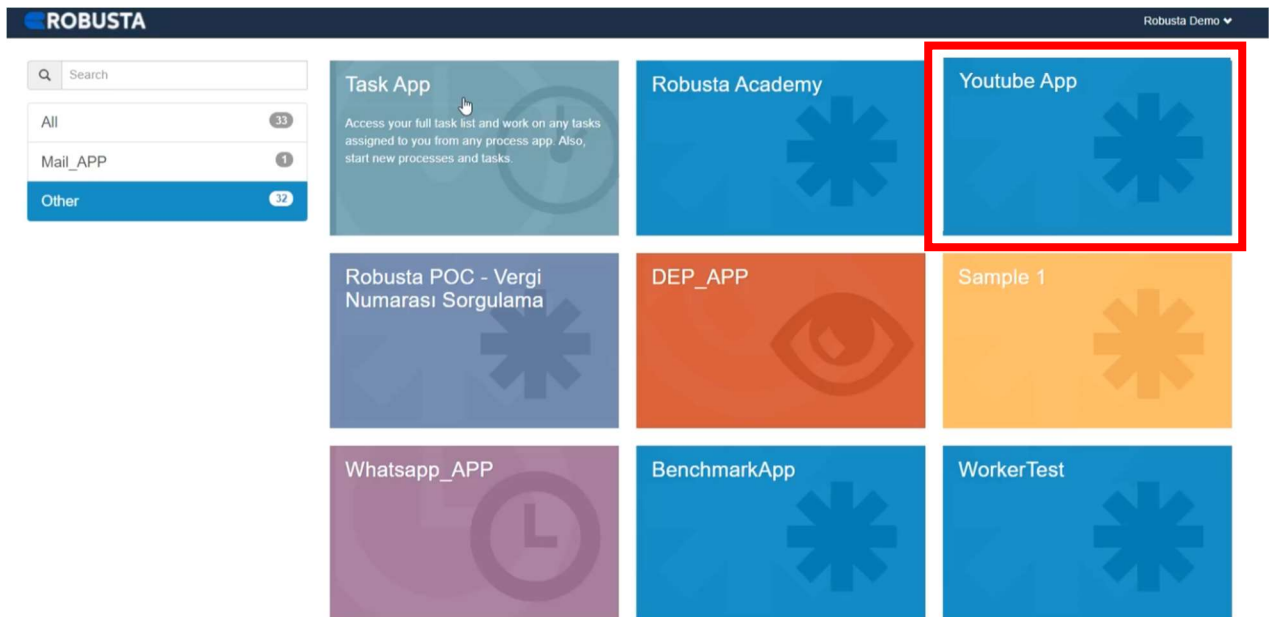
☒ App Publish Required

Close

Run Process

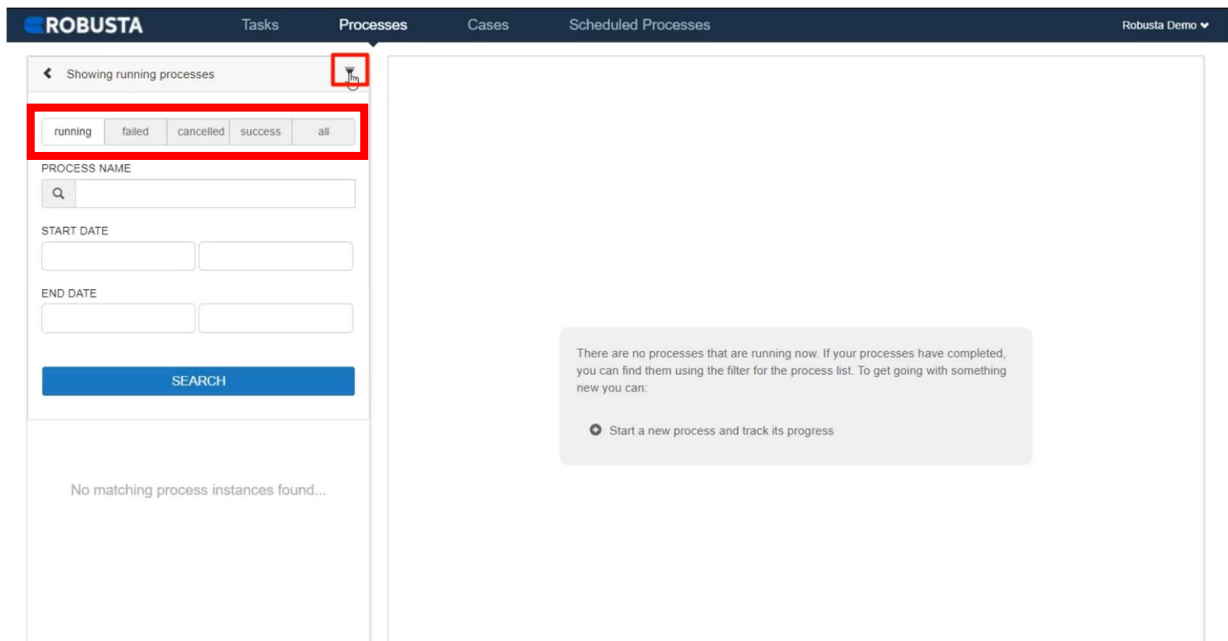
# Monitor Scheduled Processes with Robusta Scheduler

- You can open Scheduler module. Then, select your app to monitor scheduled processes in this app.

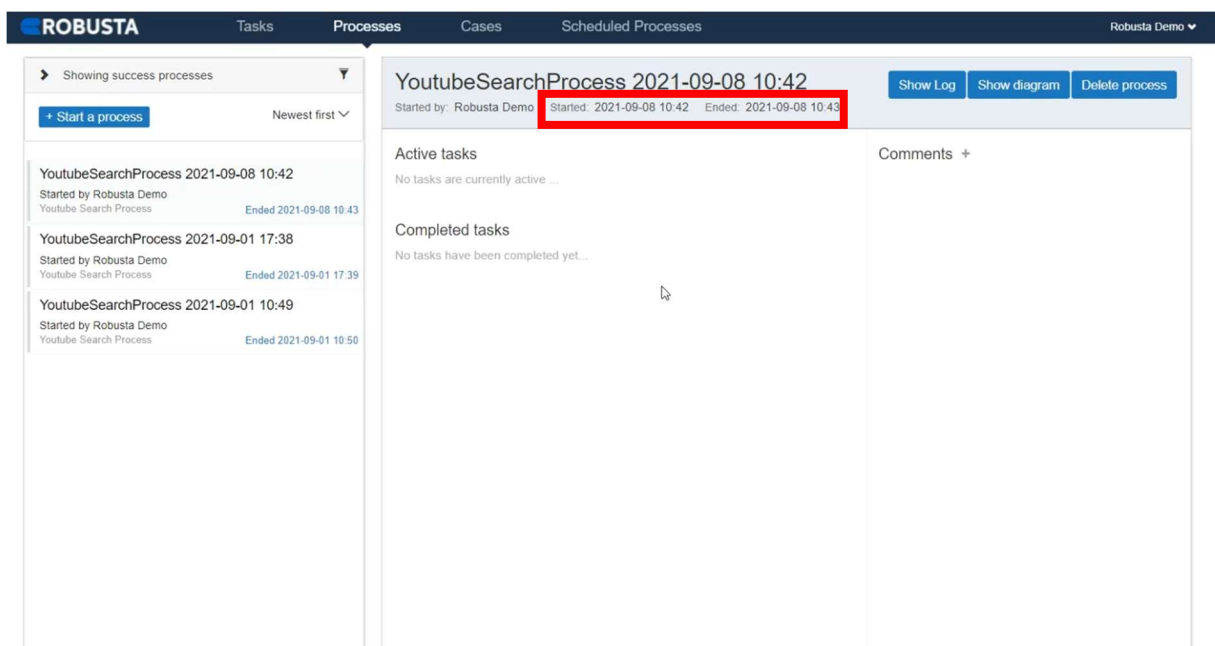


- By pressing the filtering field on the Processes page, you can list your processes and filter by success status.





- Here we can see the start date and end date of the process. Also we can see the process did not have any errors.




ROBUSTA Tasks Processes Cases Scheduled Processes Robusta Demo

PROCESS NAME  
PARAMETERS  
STATUS  
START DATE  
END DATE  
SCHEDULED DATE  
☐ Only Recurring  
SEARCH

Start Date: Filter

Found 1 matching processe(s), showing 1 to 10

Status	Waiting Reason	UserName	StartDate	EndDate	WorkerName	Show Logs	Show Diagram
SUCCEEDED		Robusta	2021/09/01 10:49:32	2021/09/01 10:50:13	robustademo@robusta.com.tr		

- When you click Show Diagram icon, the flow diagram is shown, and the path followed is seen as blue. Also, the values of the variables defined in the process is available at this page.

Open: Youtube

Close

Variables Executions Log Expression Script

Execution 2474c713-0af9-11ec-b471-005056b399e2

Type	Name	Value
string	initiator	robustademo
string	minBpPriority	50
string	humanEffort	0
string	__worker	robustademo@robusta.com.tr
string	openYoutube	1630482584472

- Now let's enter an invalid URL(<https://www.youtube.com/>) into the open activity to test, so the process gets an error, and we can see the error message on the processes page.

The top part of the screenshot shows a process diagram with two main steps: 'Open: Youtube' and 'Close'. The 'Open: Youtube' step is expanded, showing its configuration details.

Open: Youtube			
Id :	No value	Asynchronous :	<input checked="" type="checkbox"/>
Name :	Open: Youtube	Documentation :	No value
* Application name :	openYoutube	* Type :	CHROME
Application reference :	No value	* Uri :	<input type="text" value="https://www.youtube.com/"/>
Wait until load :	<input checked="" type="checkbox"/>	Download directory :	No value
Timeout (sec) :	No value	Profile directory :	No value
Maximize :	<input checked="" type="checkbox"/>		

- As you can see, the process got an error, and an error message appeared at the bottom of the processes page.

The screenshot shows the Robusta interface with a list of processes on the left and the details of the most recent process on the right.

**Processes List:**

- YoutubeSearchProcess 2021-09-09 09:35 (Started by Robusta Demo, Ended 2021-09-09 09:36)
- YoutubeSearchProcess 2021-09-08 10:42 (Started by Robusta Demo, Ended 2021-09-08 10:43)
- YoutubeSearchProcess 2021-09-01 17:38 (Started by Robusta Demo, Ended 2021-09-01 17:39)
- YoutubeSearchProcess 2021-09-01 10:49 (Started by Robusta Demo, Ended 2021-09-01 10:50)

**YoutubeSearchProcess 2021-09-09 09:35 Details:**

Active tasks: No tasks are currently active ...

Completed tasks: No tasks have been completed yet...

**Failure Reason:**

```
WA-0004[Response not success...unknown error:
net::ERR_CONNECTION_REFUSED (Session info: chrome=92.0.4515.159) Build info:
version: '3.9.1', revision: '63f7b50', time: '2018-02-07T22:25:02.294Z' System info:
host: 'DESKTOP-0NC5QOQ', ip: '10.253.240.21', os name: 'Windows 10', os arch:
'amd64', os version: '10.0', java version: '1.8.0_221' Driver info:
org.openqa.selenium.chrome.ChromeDriver Capabilities {acceptInsecureCerts: false,
browserName: chrome, browserVersion: 92.0.4515.159, chrome:
{chromedriverVersion: 91.0.4472.19 (1bf021f248676..., userDataDir:
C:\Users\iozgur\AppData\Local\...}, goog:chromeOptions: {debuggerAddress:
localhost:55721}, javascriptEnabled: true, networkConnectionEnabled: false,
pageLoadStrategy: normal, platform: WINDOWS, platformName: WINDOWS, proxy:
Proxy(), setWindowRect: true, strictFileInteractability: false, timeouts: {implicit: 0,
pageLoad: 300000, script: 30000}, unhandledPromptBehavior: dismiss and notify,
webauthn: extension:largeBlob: true, webauthn:virtualAuthenticators: true} Session ID:
f9a50bfe9acc96f45055c7a63eb2f7af
```

- When you click the Show Diagram icon, we see the Open : Youtube box is not blue. This indicates that the process failed. As you can see, the error message is here as well.

Process diagram | Root

Variables Executions Log

Execution 2e0f7bbd-1138-11ec-b0a1-005056b399e2

Type	Name	Value
string	initiator	robustademo
string	minBpPriority	50
string	humanEffort	0
string	__worker	murtazaasci@robusta.com.tr
string	FAILURE_REASON	WA-0004 Response not success...unknown error: net:ERR_CONNECTION_REFUSED (Session info: chrome=92.0.4515.150) Build info: version: '2.0.4', revision: '8277b50', time: '2019-07-07T22:35:02.704'

As we experienced together we have learned the use of the Robusta RPA design studio and its interfaces by designing a simple RPA process .See you next time.