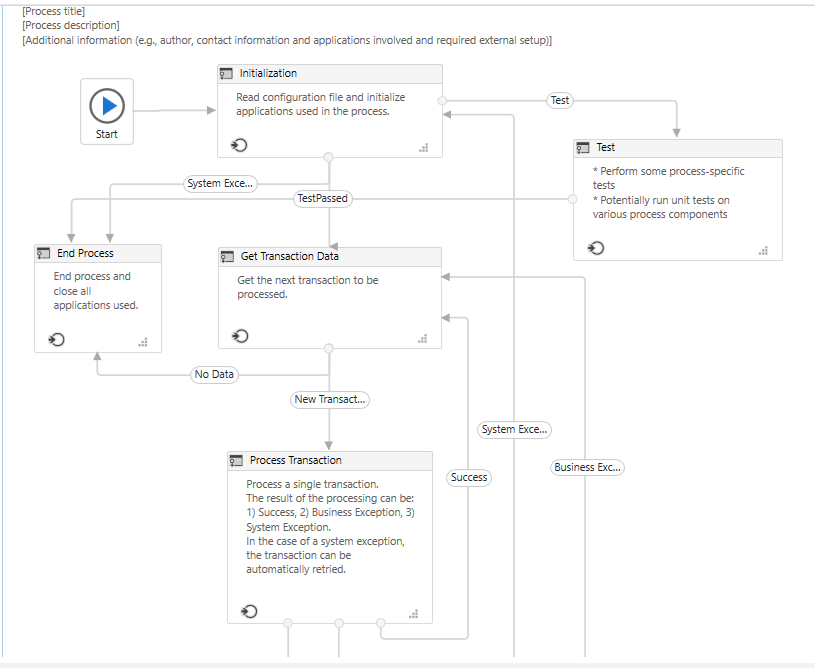
Latest UiPath Framework- Project Template

* [UiPath Project Template](#_ekx9im45gxt7)
* State Machine
  + [Test](#_8haxzxw98pcn)
  + [Added folder called “Test” that contains some test cases related to the framework’s workflows](#_1o07dv76cv0d)
* [ServiceNow Incident created only once after max retries reached](#_800z00ndd2i7)
* [Added GetQueueMaxRetries workflow to find out Max Retries of the Queue in Process state](#_20bmn6yycwm0)
* [Added KillProcessperUser](#_l2iw8r21hftv)
* [Improvements that are released in 21.10](#_vphgvporamz1)
  + [Added Orchestrator Queue Folder Name in the Config file and input argument in Main.xaml](#_2ycdcqrxiuar)
  + [Moved Process.xaml to the “Framework” folder](#_pl369e8xx9ww)
  + [Fixed issue of “Successful” status for a queue item in the event of a killed process](#_s6chwspjxd3y)
  + [Added maximum consecutive retry number to avoid consuming the queue for persistent errors](#_ny4i7divh4p7)
  + [In case the maximum consecutive retry number is reached, the Job is now marked as failed (Faulted)](#_c275651ljcmd)
  + [Logging screen resolution in init Section](#_nkrq2y78bcwj)
  + [Added retry mechanism for Set Transaction Status and](#_7acvusqj1g7h)
  + [Get Transaction Item activities in case of error](#_7acvusqj1g7h)
  + [Process stops in Init Phase in case an asset not found in Orchestrator](#_hpfzp979d2jv)

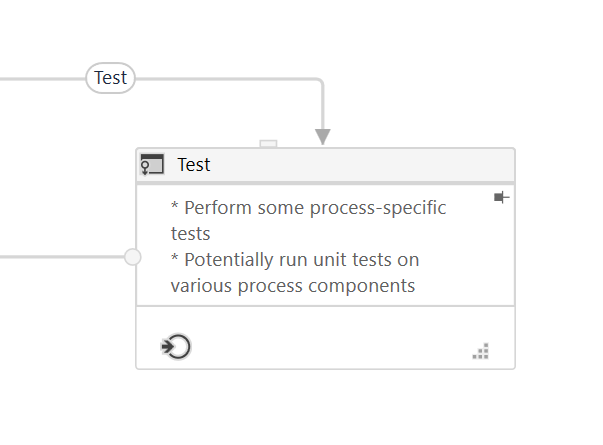
## **UIPath Project Template:**

* If starting a new project, there is a blank project template that you can use which has a high-level state machine, project configuration, logging, exception handling and system tests built into it

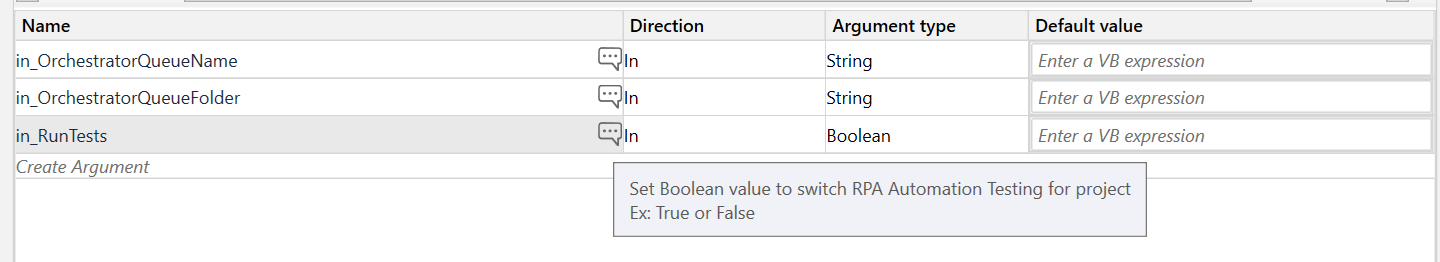
The Process looks something like this



# **Test State:**



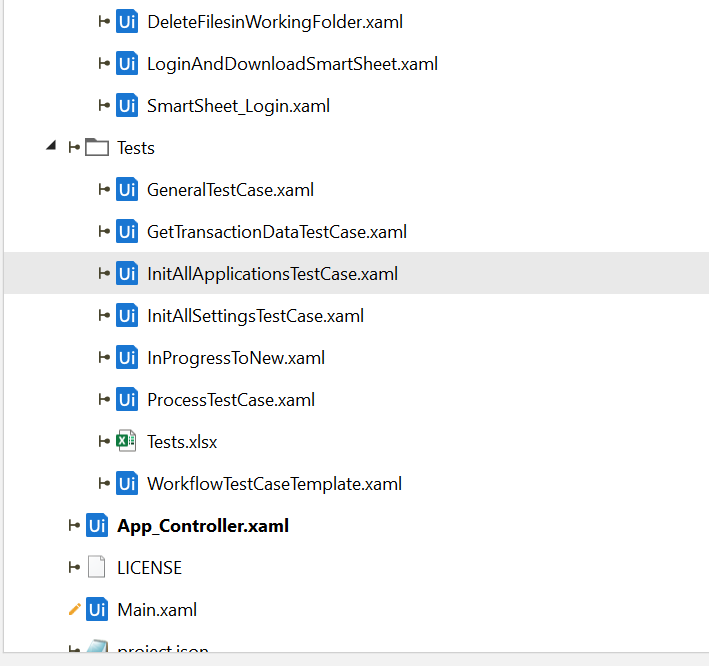
When the project is configured to run tests instead of the main process, this state will be called before closing applications and finishing. From here, unit tests and any additional automated test workflows should be called.

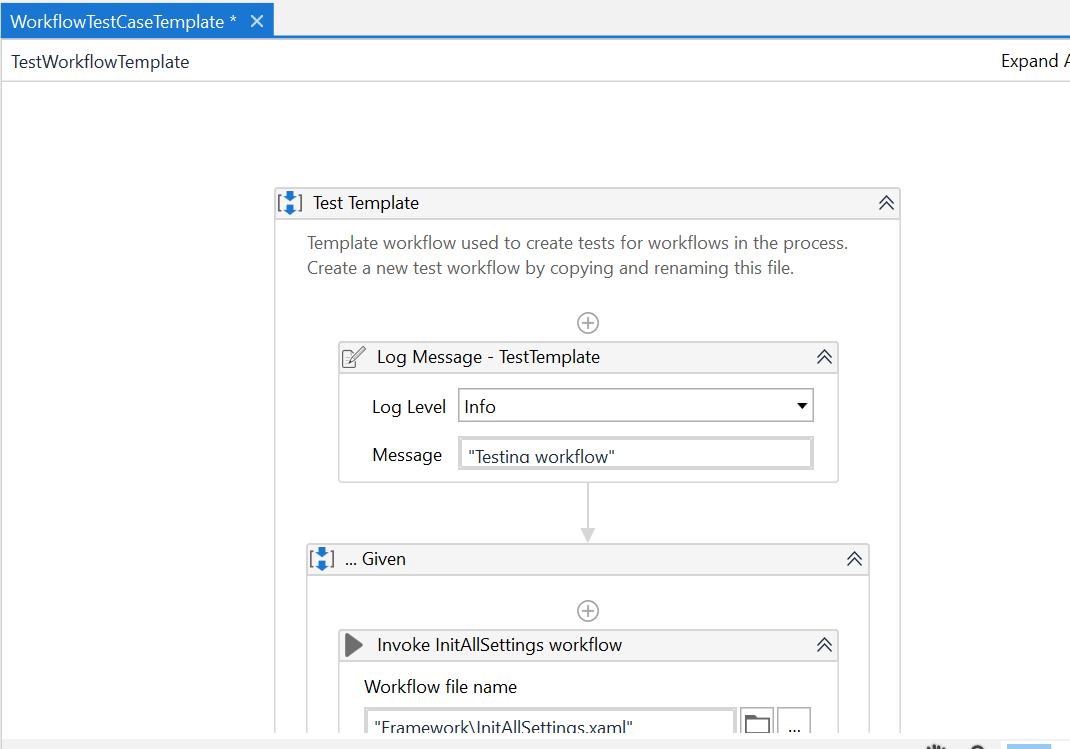


## **in\_RunTests:** This parameter should be set to true or false in order to do unit testing or sanity testing

* Majorly used in UAT or production environments with minimal test cases defined in Test state
* The main aim of this testing is to detect early major issues.
* This is designed to check system stability and conformance to requirements.

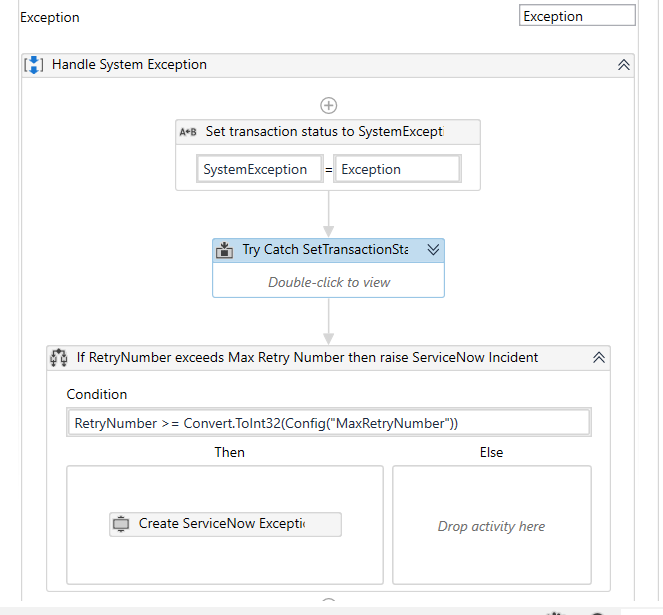
## In the REFramework folder, there is a folder called “Test” that contains some test cases related to the framework’s workflows





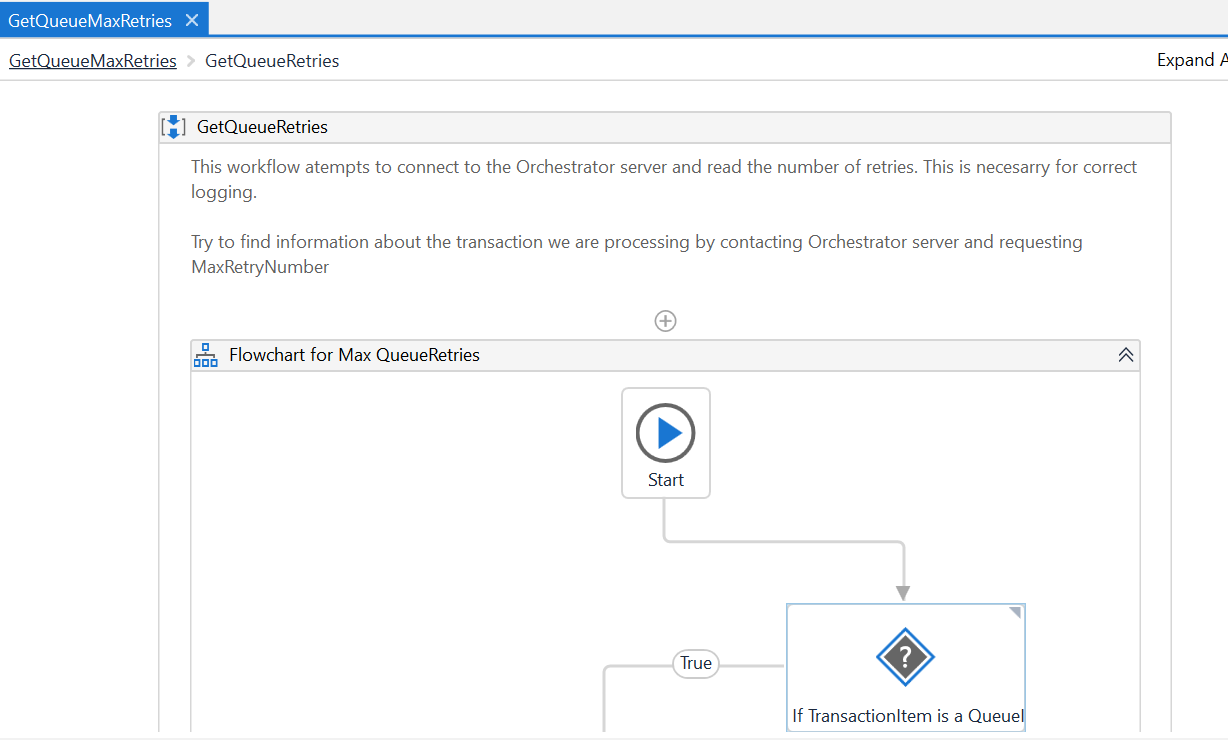
## ServiceNow Incident gets created only after MaxRetryNumber is reached

* This condition is located at Exception block of Process.xaml



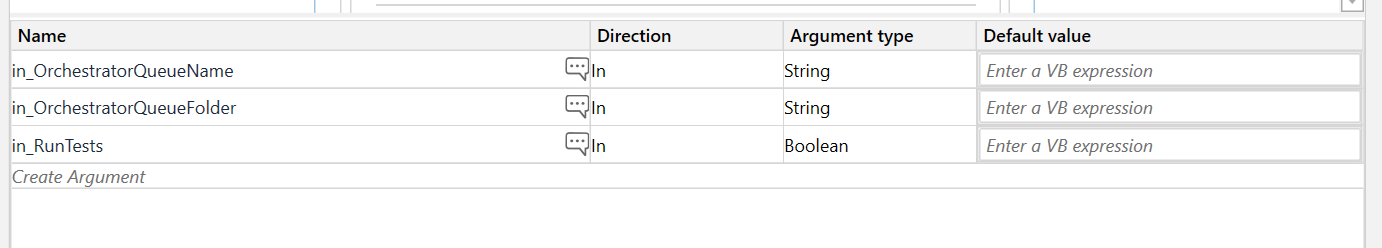
## GetQueueMaxRetries.xaml

* This workflow attempts to connect to the Orchestrator server and read the number of retries. This is necessary for correct logging
* The MaxRetryNumber of Orchestrator queue is stored in io\_Config("MaxRetryNumber")

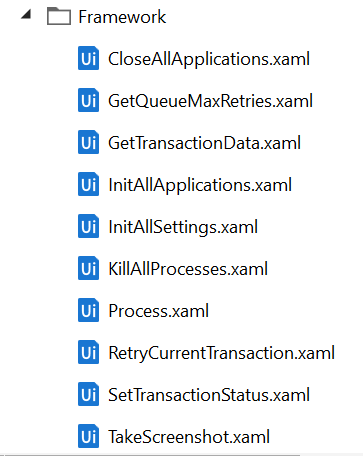


# **Here are the improvements that were released in 21.10:**

## Added [Orchestrator](https://docs.uipath.com/orchestrator/docs/introduction) Queue Folder Name in the Config file and as input argument in Main.xaml

****

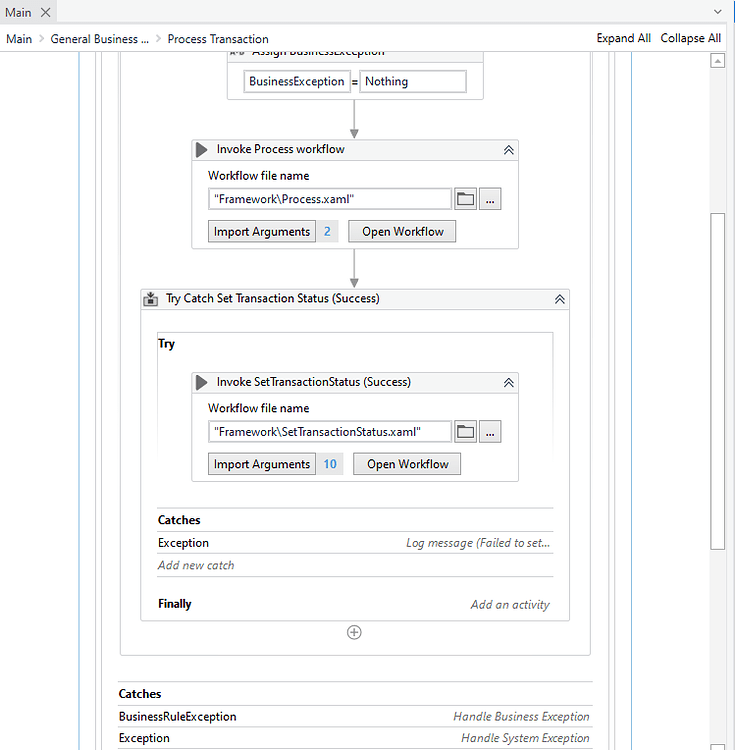
## Moved Process.xaml to the “Framework” folder



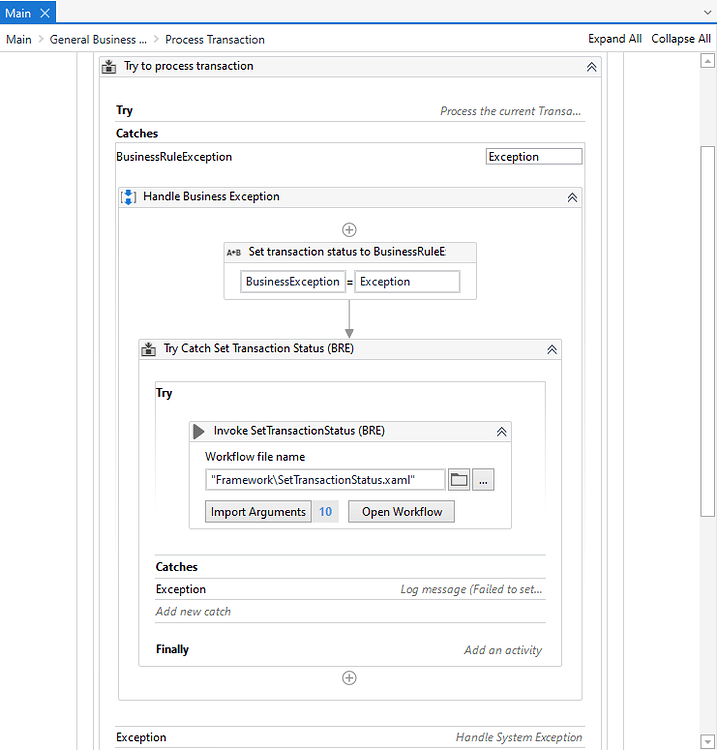
## Fixed an issue which consisted in setting a “Successful” status for a queue item in the event of a killed process:

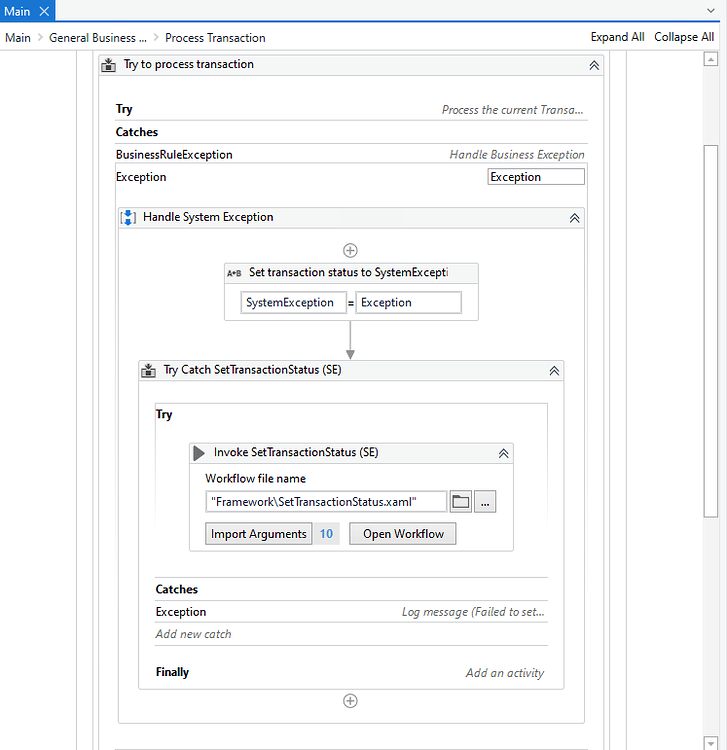
* We removed the invocation of Framework\SetTransactionStatus.xaml from inside the Finally block.

## An invocation of Framework\SetTransactionStatus.xaml (surrounded by a Try Catch activity) was added after the invocation of Framework\Process.xaml, for the successful cases, in the same Try block.

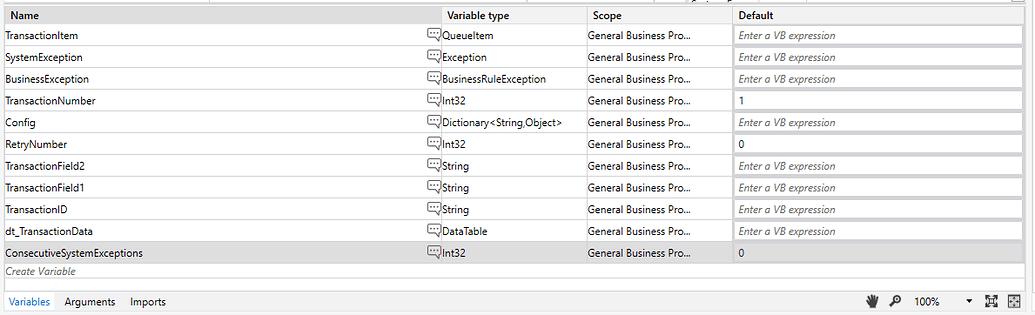


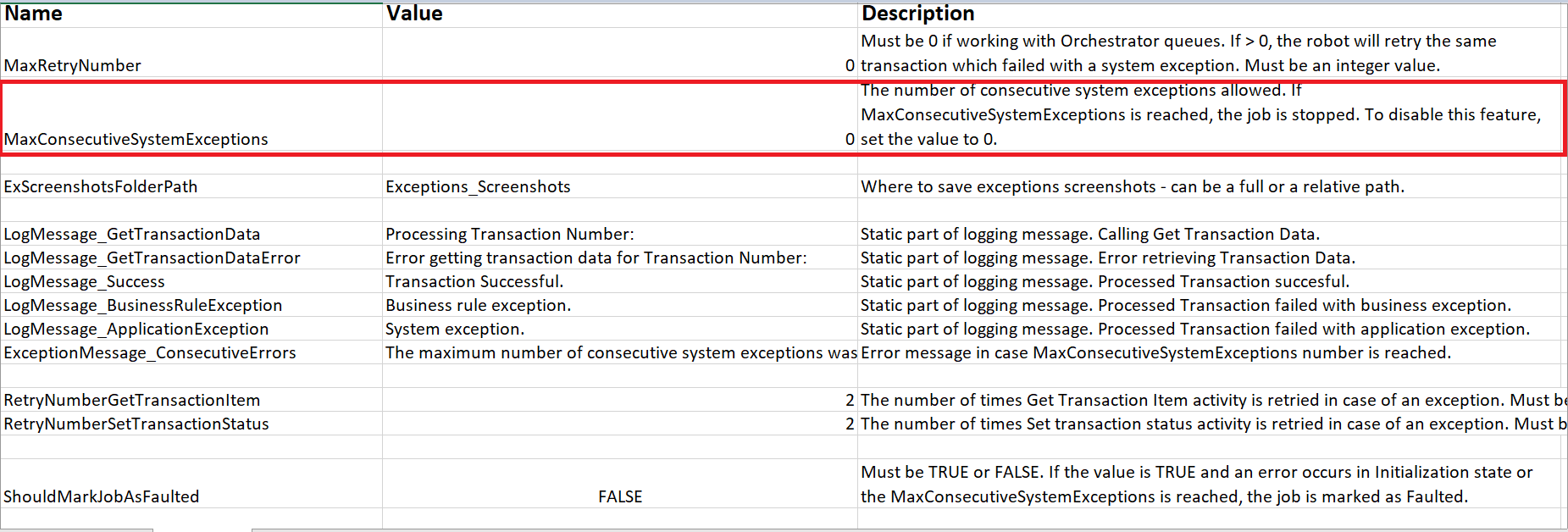
## On the Catch section, an invocation of Framework\SetTransactionStatus.xaml (surrounded by a Try Catch activity) was added for each type of exception (System Exception or Business Rule Exception)



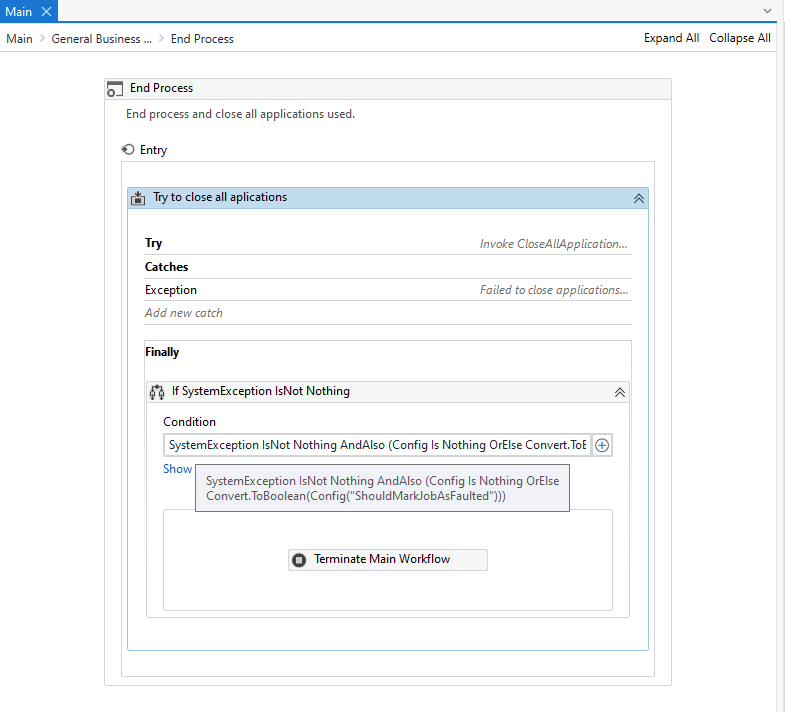


## Added a maximum consecutive retry number to avoid consuming the queue for persistent errors. There is a new global variable called ConsecutiveSystemExceptions as well as a constant value in config, called MaxConsecutiveSystemExceptions.



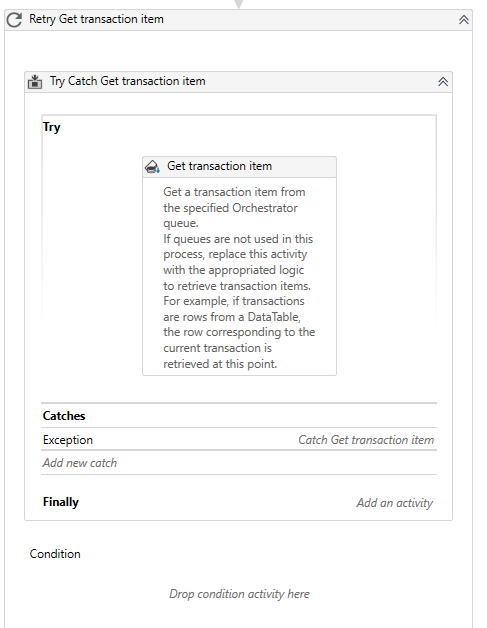


## In case the maximum consecutive retry number is reached, the Job is now marked as failed (Faulted). This behavior can be enabled by setting the boolean value ShouldMarkJobAsFaulted in Config to True (by default, it is set to False).

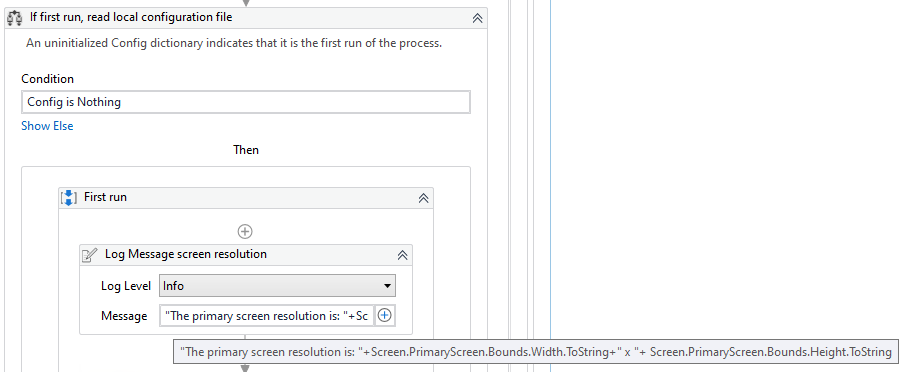


## Updated the failed queue items with the name and location of associated screenshots

## Added retry mechanism for Set Transaction Status and Get Transaction Item [activities](https://docs.uipath.com/activities/docs/introduction) in case of error

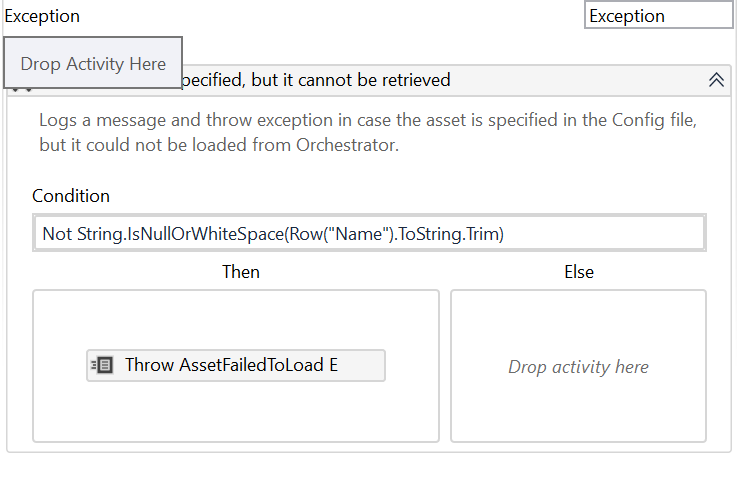


## In the Init section, we now log the screen resolution



## **Updated the asset management:**

* + Before this version, when an asset was declared in the Config file (Assets sheet), but it was missing in [Orchestrator](https://docs.uipath.com/orchestrator/docs/introduction), there was a log message with warning level. The process continued even if there were values missing for those keys and was prone for errors later in the process. With this new update, the framework now throws an exception when an asset is not found in [Orchestrator](https://docs.uipath.com/orchestrator/docs/introduction) and the process stops in the Init phase



## **KillProcessPerUser.xaml:**

* This code is used to Hard Kill all desired application processes in order to restore a clean environment and session.
* This xaml is called at Init state instead of KillProcess.xaml

