Demo Orchestrator via REF

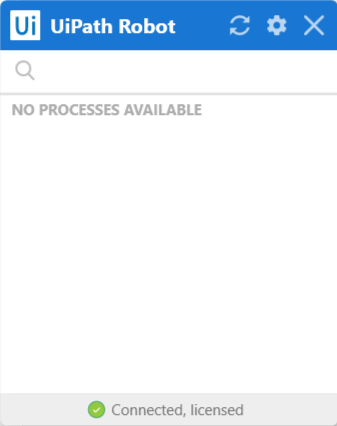
# Opdracht

Excel-bestand (factuur) rij per rij uploaden in Orchestrator zodat iedere rij een TransactionItem is. Hierna het GetTransactiondata proces gebruiken en hierna elk TransactionItem is een txt bestand laten schrijven.

Doel is om te laten zien hoe je data kan uploaden en afhalen van Orchestrator.

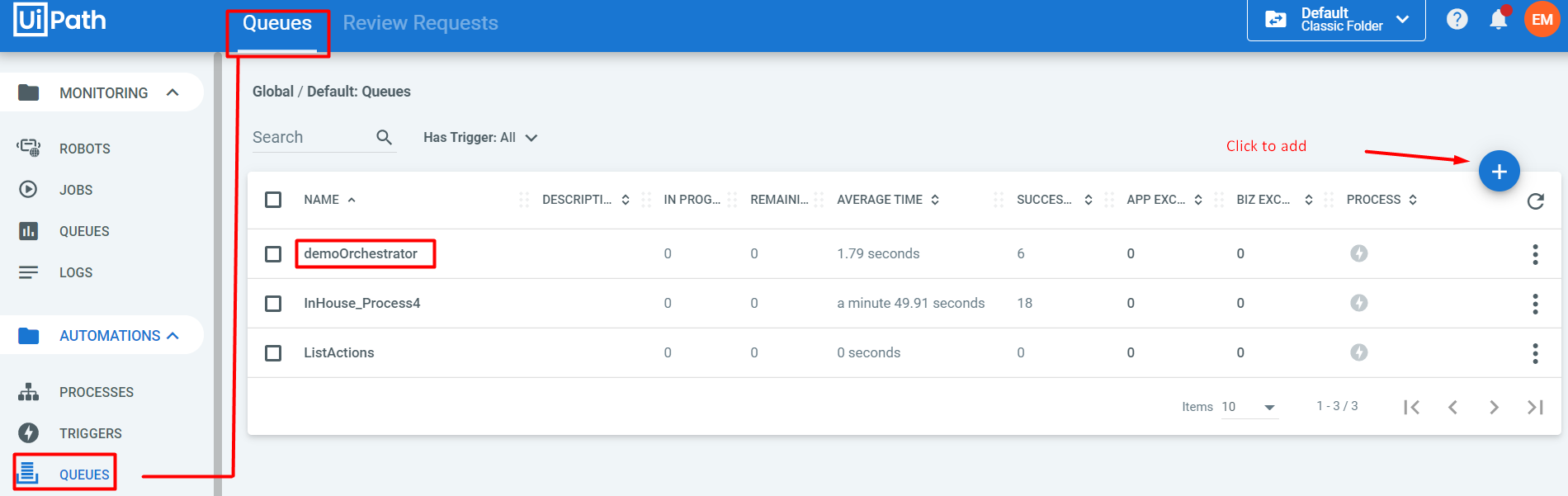
**BELANGRIJK: om te testen:**

1. **zorg ervoor dat de robot aan staat en geconnecteerd is met Orchestrator +**
2. **Run het gehele bestand als je wilt testen. (niet enkel Run File)**

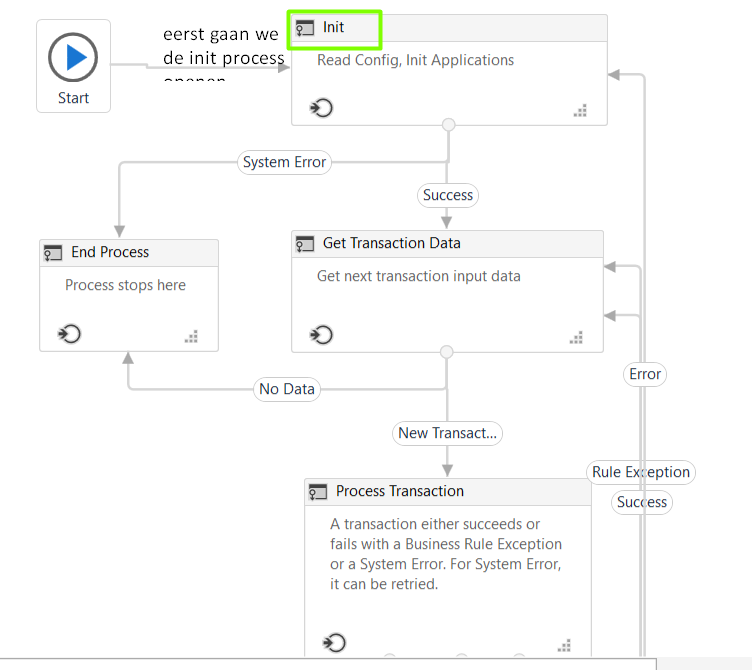


# Stappen

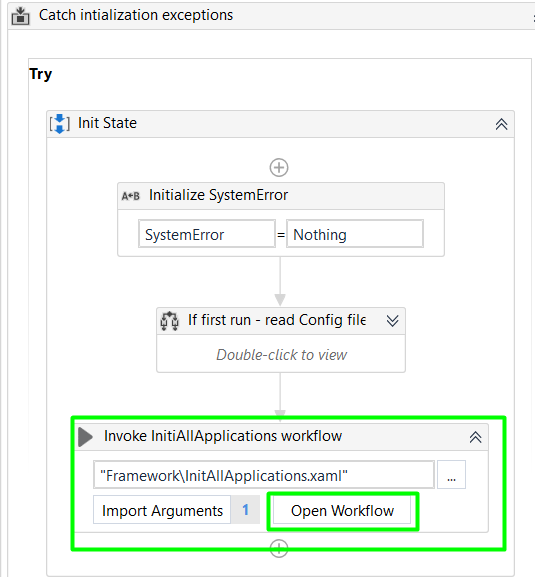
* Maak een QueuItem in Orchestrator
  + Automations
    - Queues 🡪 Add
      * demoOrchestrator
      * max retries = 1 of naar je eigen keuze
* Voeg QueueName toe aan de Config file
  + OrchestratorQueueName 🡪 onder value 🡪 “demoOrchestrator”



* Open een REFramework.

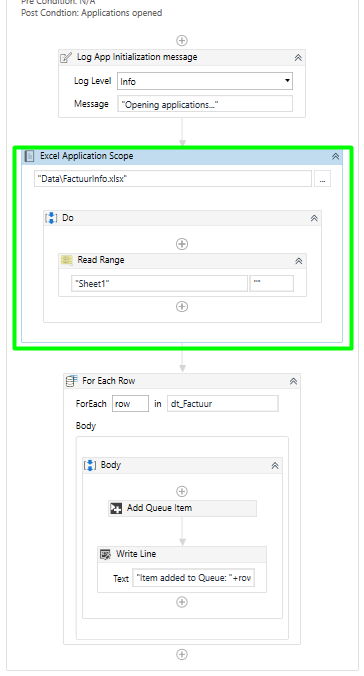


* In Main.xaml ga naar Init 🡪 open de workflow van InitAllApplications

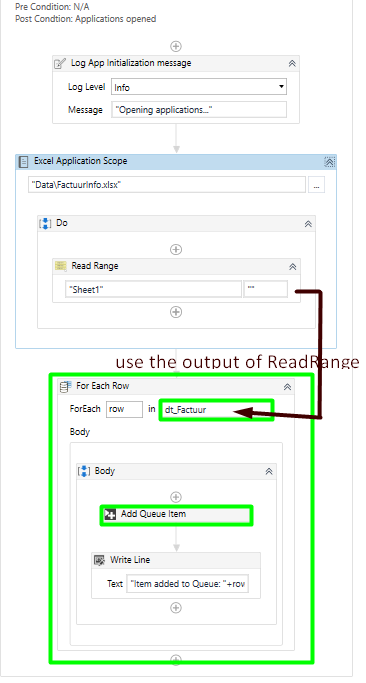


* + Eerst Excel readen via Excel Application Scope activity
    - Selecteer de gewenste factuur
    - In Do
      * Read range
        + Properties

In output Datatable 🡪 dt\_Factuur (variabele aanmaken)



* + Hieronder voeg de activiteit For Each Row toe



* + - ForEach row in dt\_Factuur(output of ReadRange)
    - Body
      * Add Queue Item activity
        + Zodat de data naar Orchestrator wordt geupload
        + Properties 🡪 Input

QueueName 🡪 “in\_Config(“OrchestratorQueueName”).ToString”

Dit zorgt ervoor dat de juiste Queue wordt aangesproken in Orchestrator



ItemInformation

Factuurnummer: IN: String: row(“FactuurNummer”).ToString

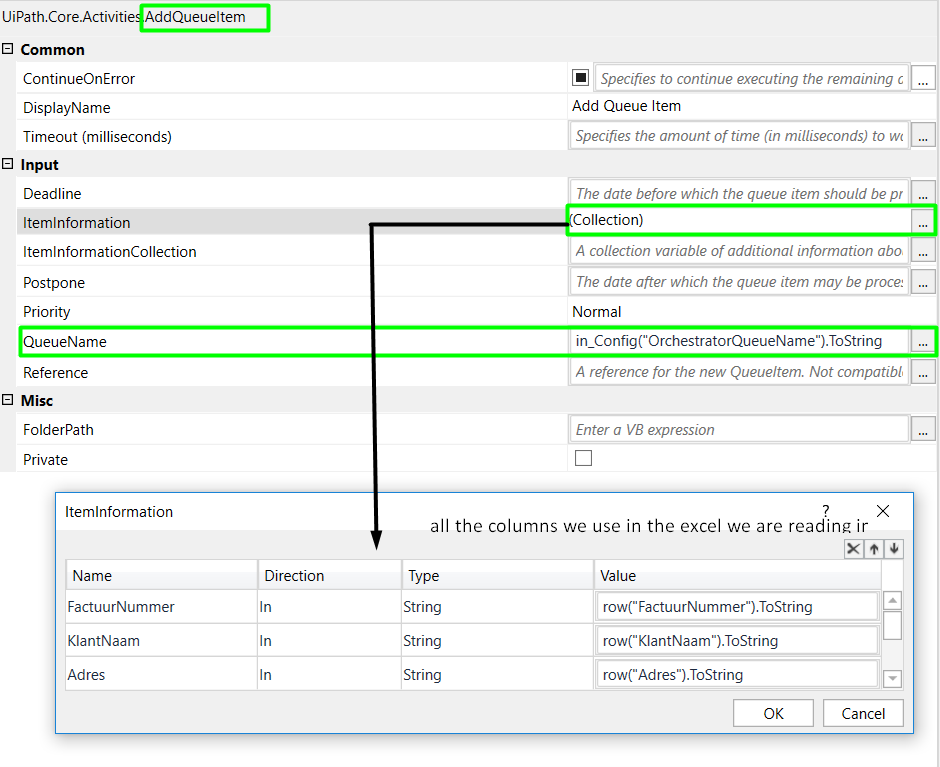
KlantNaam: IN: String: row(“KlantNaam”).ToString

Adres: IN: String: row (“Adres”).ToString

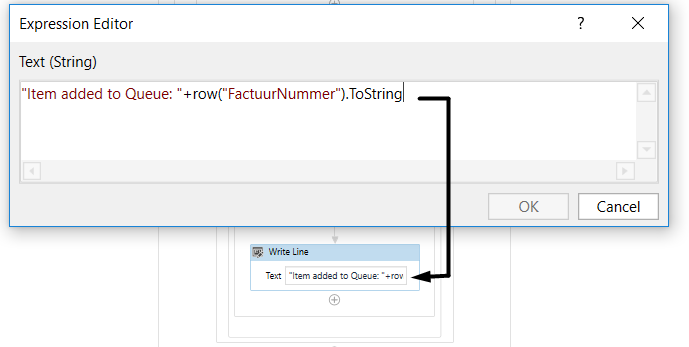
*Indien je de hele table wilt gebruiken*

*Object: IN: Object: {“Factuurnummer”, “KlantNaam”, “Adres”}*

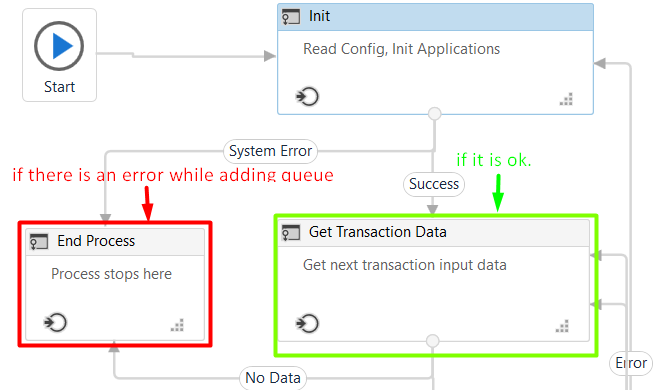
*“Convert to Json”*



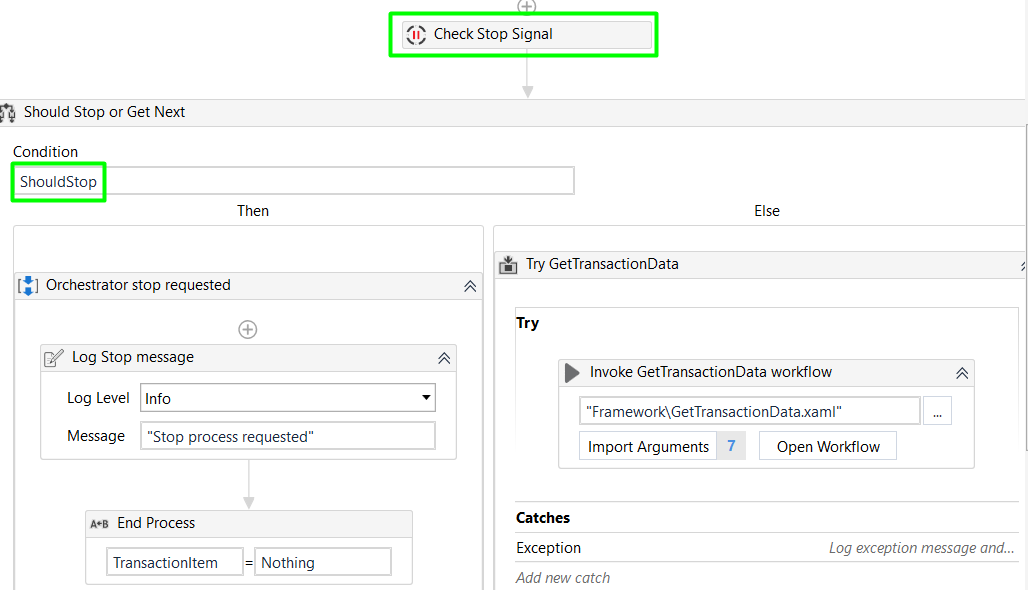
* + - * WriteLine
        + “Item added to Queue” + row(“FactuurNummer”).ToString

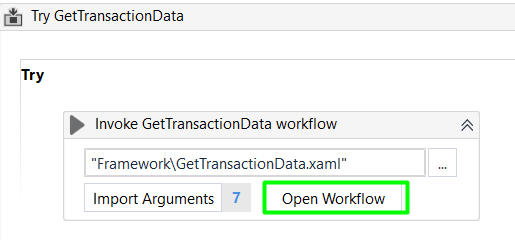


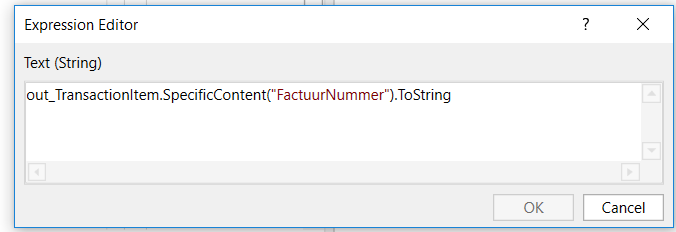
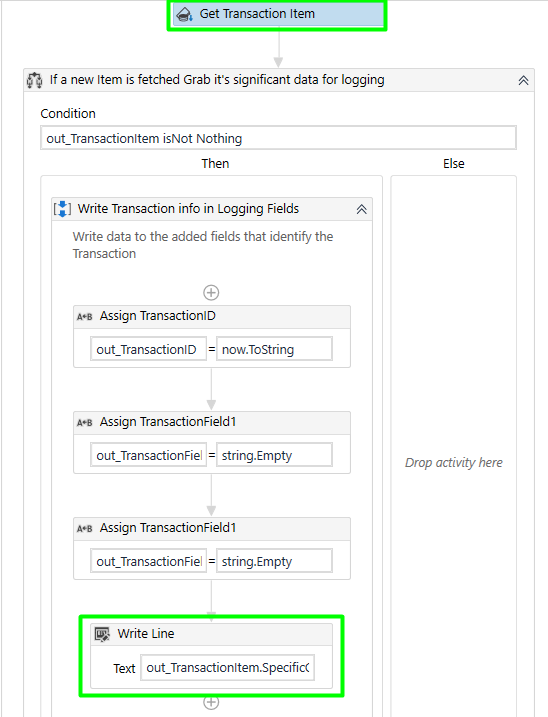
* In Main.xaml
  + Indien toevoegen aan de Queue gelukt is 🡪 Get Transaction Data;
  + Indien niet gelukt 🡪 End Process
  + Van hieruit neemt die iedere rij uit de Orchestrator en gebruikt deze om de Process Transaction workflow uit te voeren



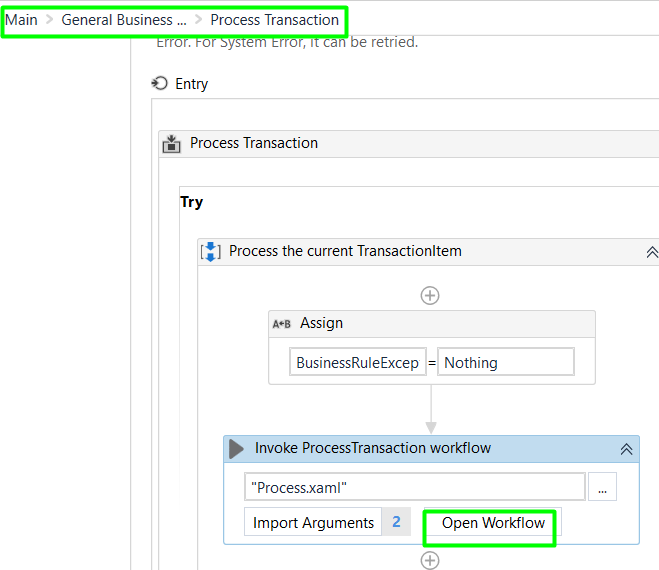
* Get Transaction Data openen
  + Indien Orchestrator aangeeft dat er een stopsignaal is 🡪 End Process
  + Anders 🡪 Try GetTransactionData



* + 
    - Open de workflow
      * Get Transaction Item = item (lees 1 rij) in de Queue van Orchestrator
      * Binnen deze workflow krijgt iedere item in de Queue een TransactionItem nummer mee
      * Voeg beneden in de If statement 🡪 then rij 🡪 **een writeline** toe
        + “out\_TransactionItem.SpecificContent(“FactuurNummer”).ToString”



* In Main.xaml 🡪 Open Process Transaction of Process.xaml openen



* + Hier worden alle processen voor de verwerking van de TransactionItems (lees QueueItems uit Orchestrator) geplaatst.
  + Voeg een sequence toe en noem deze “Open notepad”
    - Open application activity
      * Do
        + Type into activity toevoegen

In\_TransactionItem.SpecificContent(“FactuurNummer”).ToString + in\_TransactionItem.SpecificConent(“KlantNaam”).ToString

