

POWER APPS

USER GUIDE

SIMBA Chain's brings blockchain
to Microsoft Power Apps.

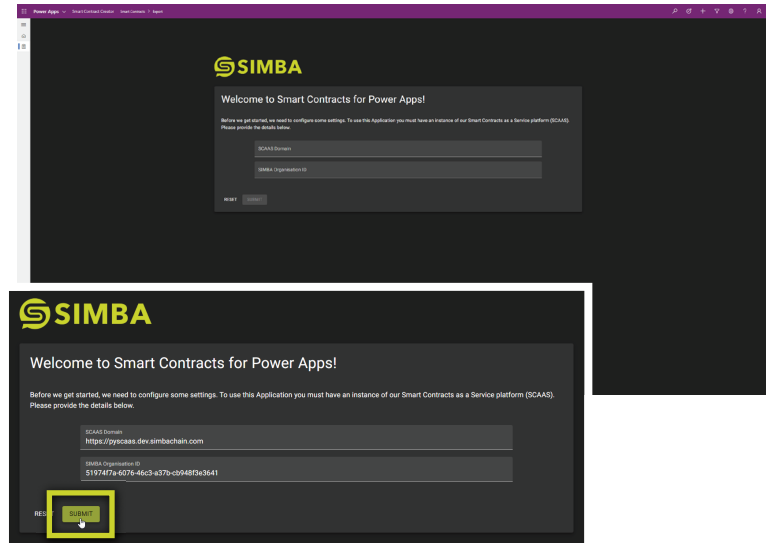
CONFIGURE YOUR SETTINGS	3
CREATING A SMART CONTRACT	4
STEP 1: CHOOSE AN ENTITY	4
STEP 2: CHOOSE SOME ATTRIBUTES	5
STEP 3: NAME SOME STUFF!	5
USING YOUR CONTRACT IN A FLOW	6
CREATING THE CUSTOM CONNECTOR	6
GENERAL INFORMATION	7
SECURITY	8
DEFINITION AND TEST	9
ESTABLISHING A CONNECTION	9
USING THE CONNECTOR IN A FLOW	10
USING THE FLOW IN A POWER APP	14

CONFIGURE YOUR SETTINGS

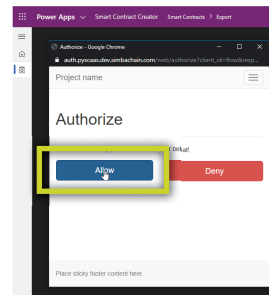
Upon loading the app for the first time, you will be greeted by a welcome page where you will be asked to configure some settings.

Here you need to set your SCAAS Domain and your SIMBA Chain organisation's ID in the respective text fields. You can press the reset button at any time to reset the fields.

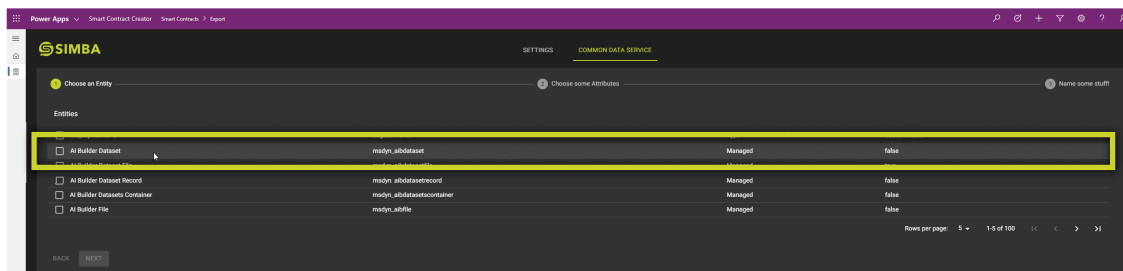
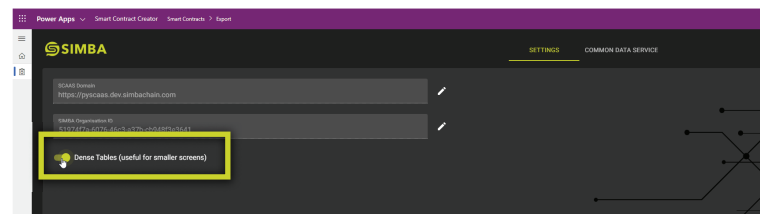
Once you have done that, press the **submit** button.



The app will then reload, and a pop up will appear asking for some permissions, press the **allow** button to proceed to the application.



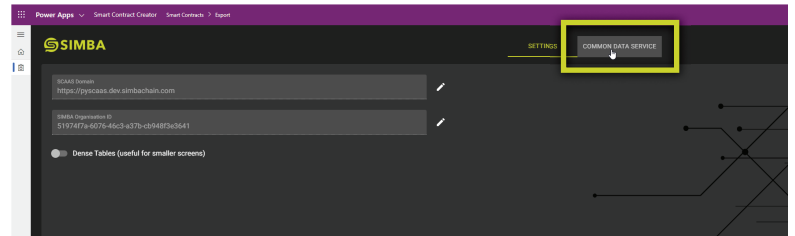
Now you are through to the main application, you can view your configuration in the **Settings** tab, here there is one more setting that you can configure to your liking. The **dense tables switch** alters the density of the tables.



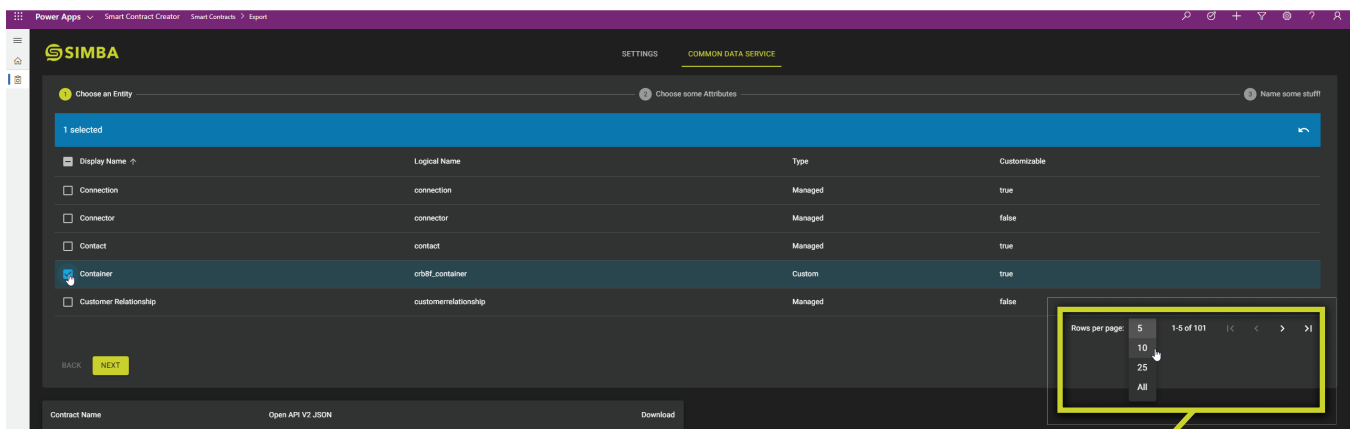
When the switch is checked, the table rows become narrower, allowing you to fit more on the screen. Try it out!

CREATING A SMART CONTRACT

Now that your settings are configured, navigate to the Common Data Service Tab by clicking Common Data Service towards the top of the screen. Here you will be greeted by a three step form.



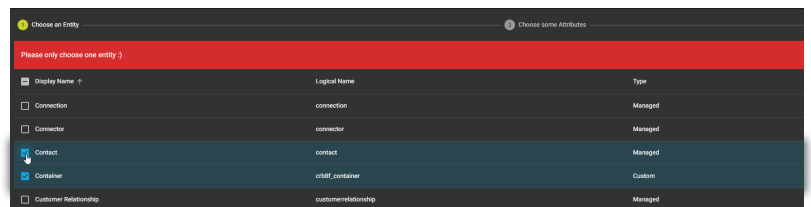
STEP 1: CHOOSE AN ENTITY



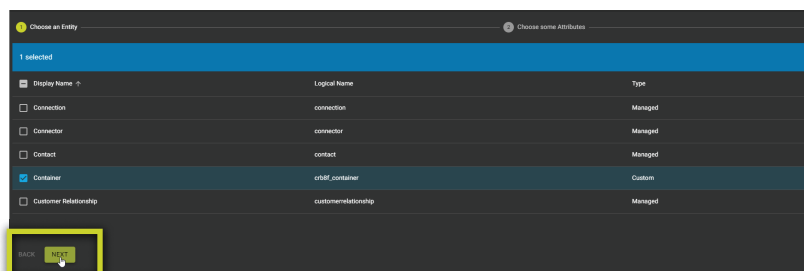
Here you should be greeted by a table titled Entities. This table contains all of your common data service entities, which should be familiar. All there is to do here is click on the entity that you wish to use to create your smart contract, either via the checkbox, or simply clicking the row. You can undo your selection at any time by clicking the undo icon at the top right of the table. Here I've chosen a custom Container entity.

If you cannot find the entity on the first page, then use the pagination buttons to navigate through the table, until you find it. If you want to see more entities on each page, then the Rows per page select box will allow you to choose between 5, 10, 25 or all the rows.

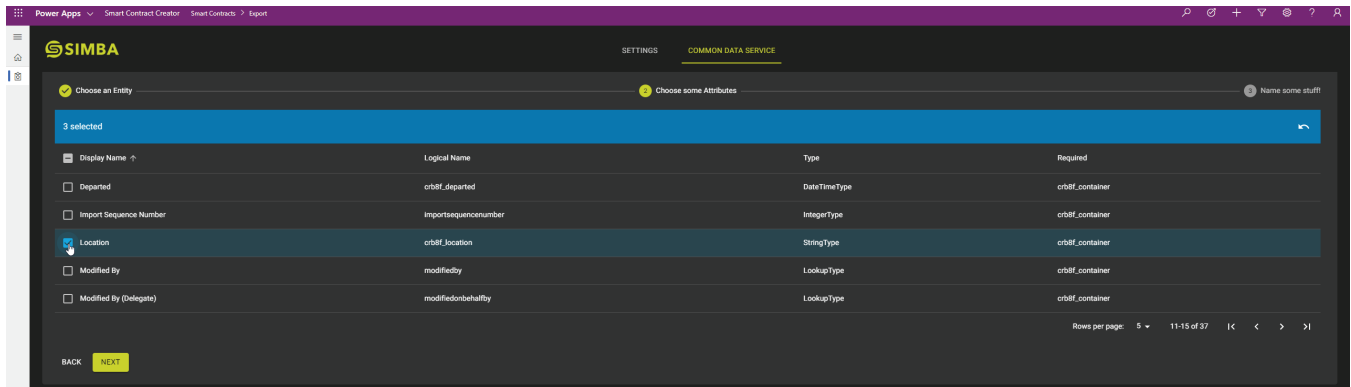
Note: you can only choose one entity at the moment.



Once you've chosen your entity, click **next** to proceed to step 2.

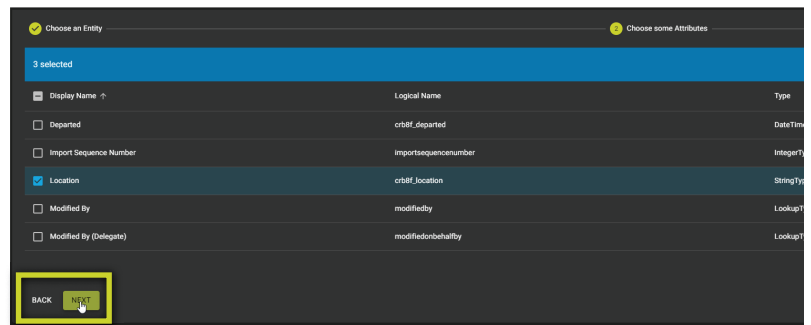


STEP 2: CHOOSE SOME ATTRIBUTES

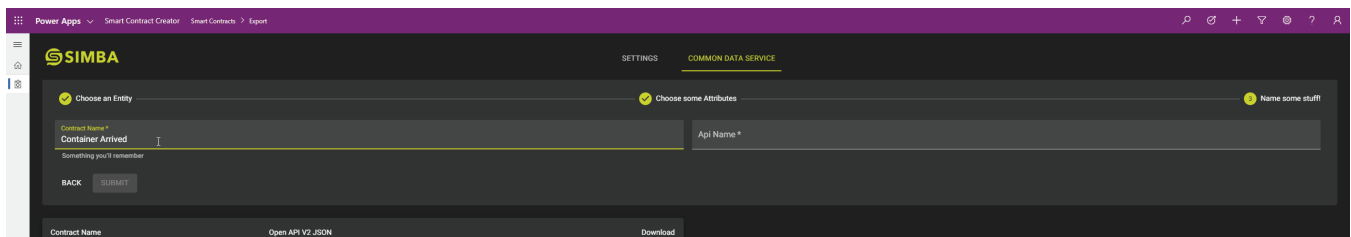


Here you should see a table titled Attributes. This table contains all the attributes for your selected entity. Once again simply click the checkbox or the row of the attributes that you want as part of your smart contract. These will form the “fields” in your contract, and you are allowed to choose more than one this time. You can undo your selections at any time by clicking the undo icon at the top right of the table, or by simply clicking the checkbox or row again. Here I’ve chosen 3 attributes.

Once you’ve chosen your attributes, click next to proceed to step 3.



STEP 3: NAME SOME STUFF!

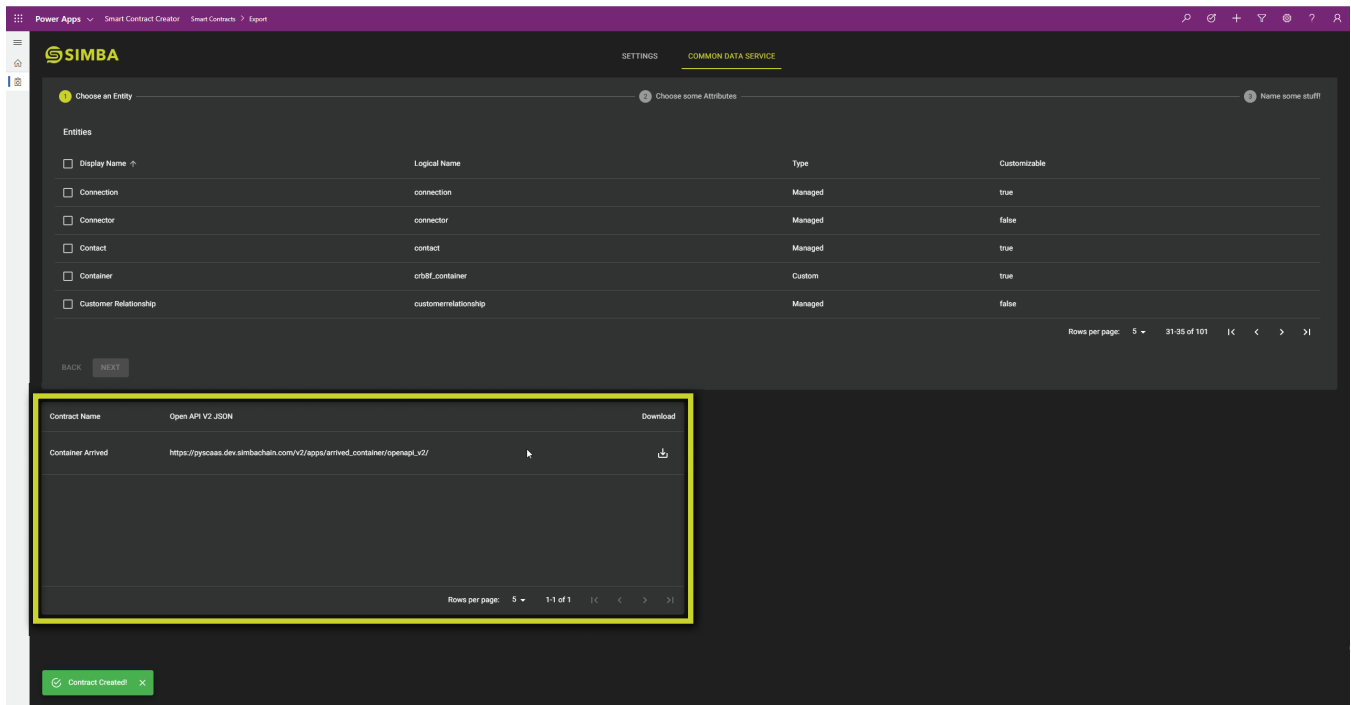


Here you are greeted with two text inputs, Contract Name, and API Name. Contract name is a user facing name for your contract, it’s for you, so name it something you can remember.

API Name on the other hand, forms part of the smart contract api url that will be generated. Therefore, we don’t allow spaces or special characters, as indicated by the placeholder text.

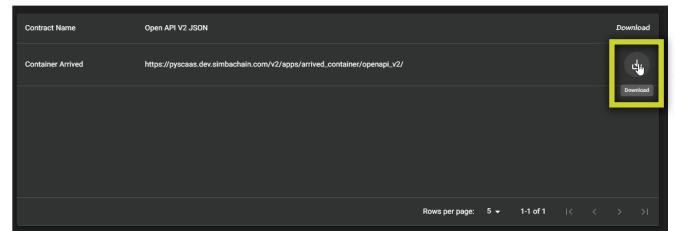


USING YOUR CONTRACT IN A FLOW



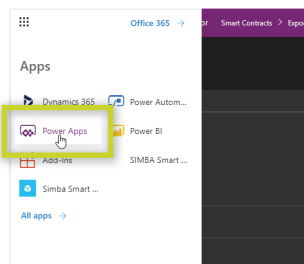
Once the server has responded, you will find your new smart contract in the bottom table.

Look for your contract name, and then press the respective download button on the tables right hand side. This will download an open api v2 json file named [ContractName].json, and it can be found in the default downloads directory for your operating system. We will use this file to create a custom connector for our newly created smart contract API

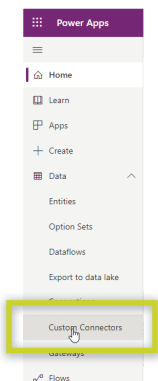


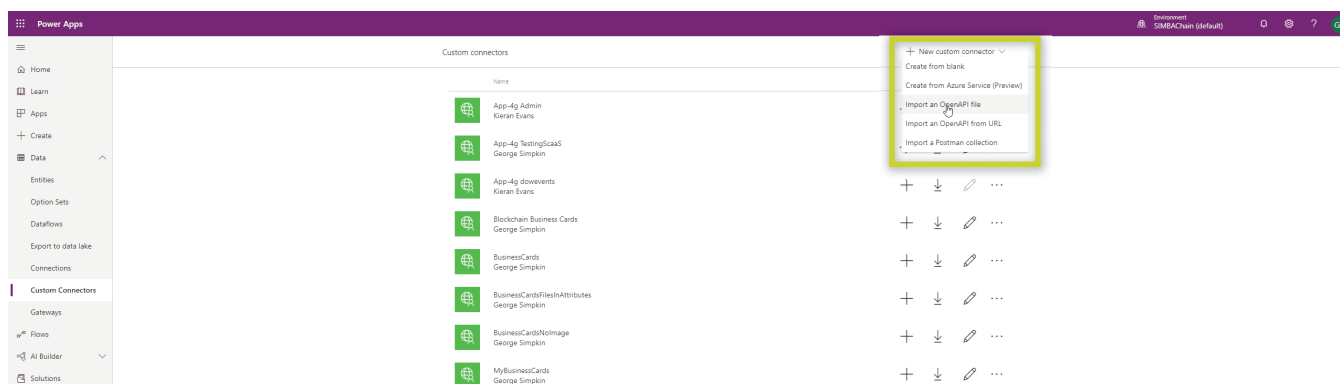
CREATING THE CUSTOM CONNECTOR

To begin, first navigate to the Power Apps home page.



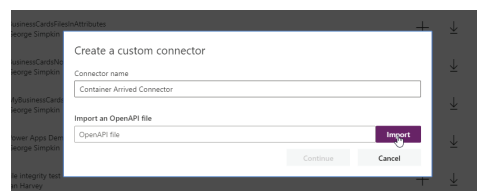
Now click the Data drop down and navigate to custom connectors and click.



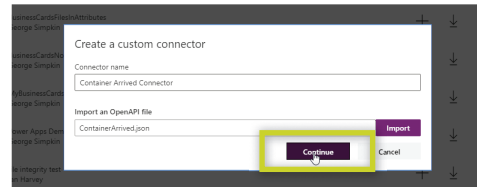


Next click on the New Customer Connector drop down menu at the top right, and select Import an OpenAPI file.

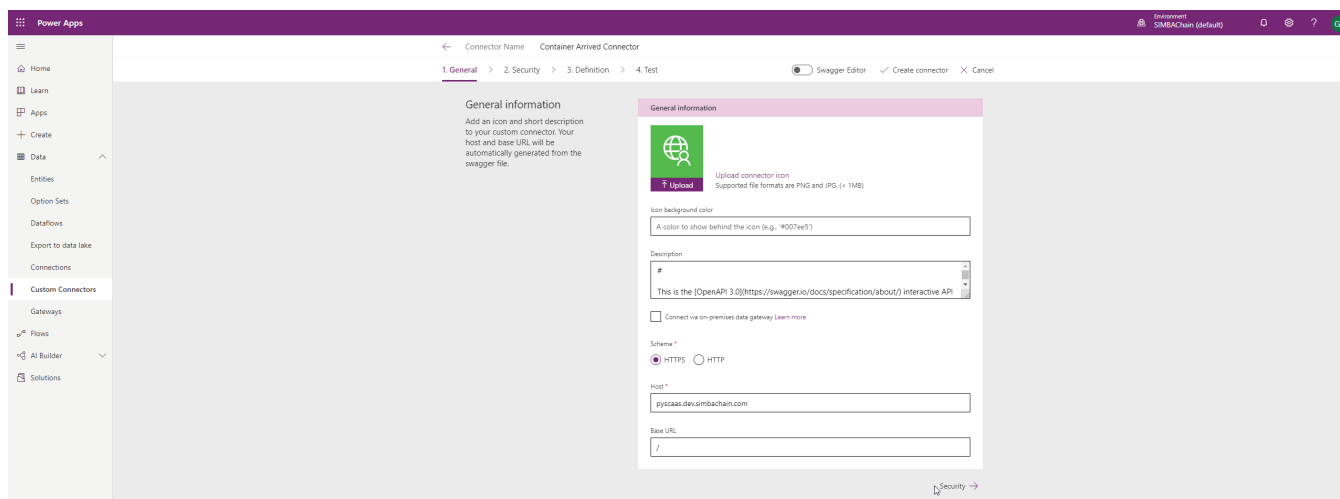
Now simply name your connector and import the open API file we downloaded earlier, remember it should be in your default downloads directory.



Press Continue

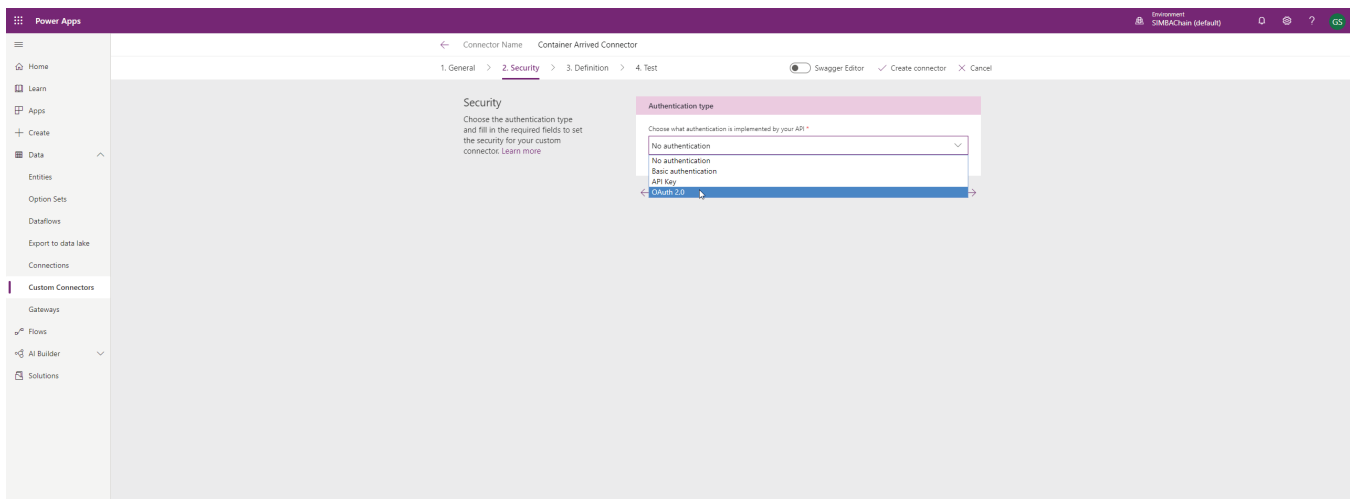


You should now be on a four step form to configure the connector.



GENERAL INFORMATION

For this step, make sure HTTPS is selected for the Scheme, and then enter your domain name (the same one from your settings) as the host. Leave base URL as /.



SECURITY

For authentication type choose OAuth 2.0

For Identity Provider choose Generic Oauth 2

In the Client Id field enter a client id for interacting with a web api for your authorization server. In the Client Secret field, enter the client secret for that particular client id.

For Authorization URL enter `https://auth.[Domain]/web/authorize`
e.g. `https://auth.pyscaas.example.com/web/authorize`

For token url input `https://auth.[Domain]/v1/oauth/tokens`
e.g. `https://auth.pyscaas.example.com/v1/oauth/tokens`.
For refresh url, simply input your token url

And finally for scope input `read_write`

Authentication type

Choose what authentication is implemented by your API *

OAuth 2.0

Edit

OAuth 2.0

Identity Provider

Generic Oauth 2

Client id *

api

Client secret *

Authorization URL *

https://auth.pyscaas.dev.simbachain.com/web/authorize

Token URL *

https://auth.pyscaas.dev.simbachain.com/v1/oauth/tokens

Refresh URL *

https://auth.pyscaas.dev.simbachain.com/v1/oauth/tokens

Scope

read_write

Redirect URL

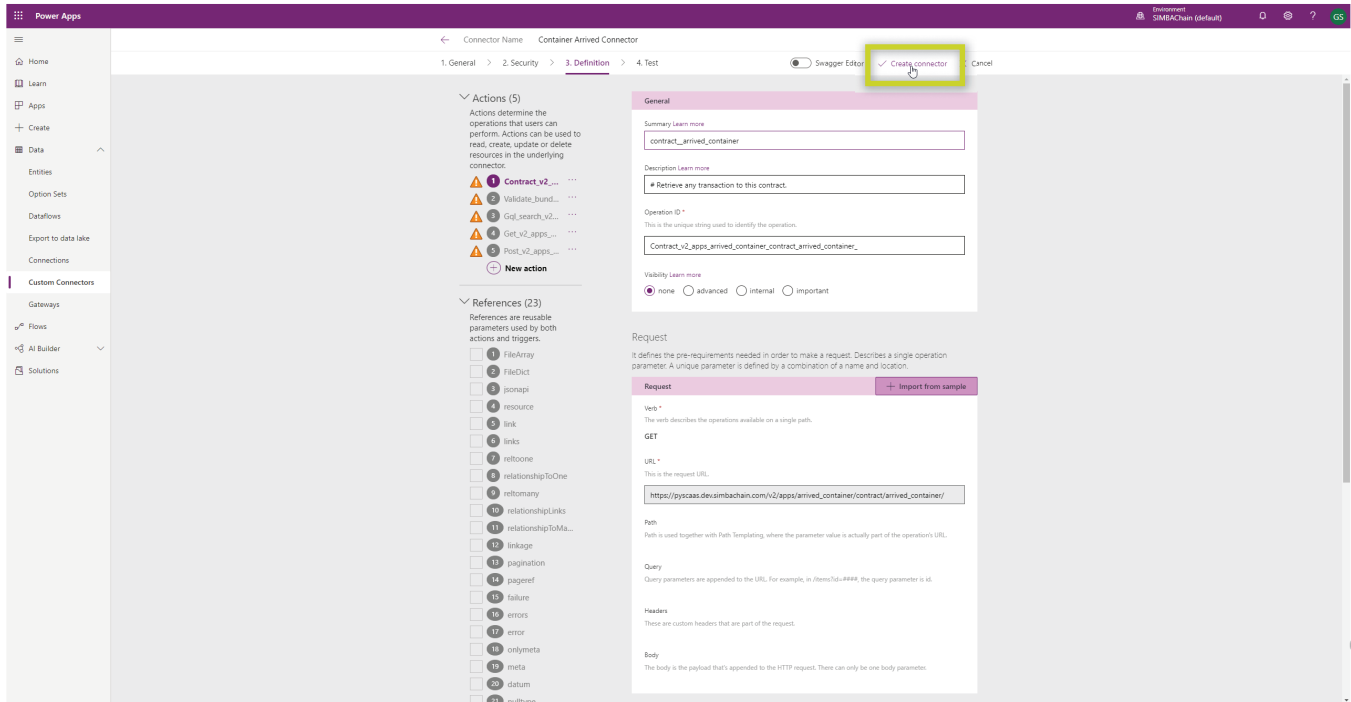
Save the custom connector to generate the redirect URL

Edit

General
Definition

DEFINITION AND TEST

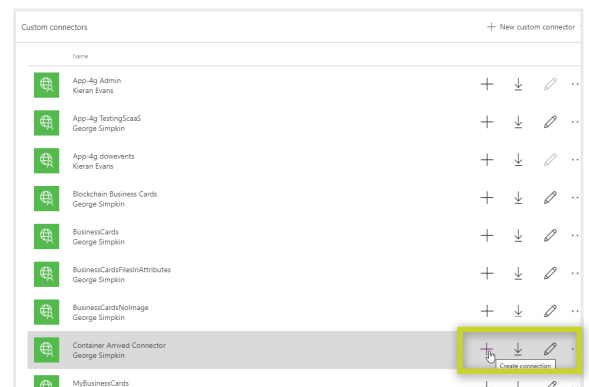
There is nothing you need to do for these sections, but feel free to have to look over and make sure everything looks in good order. Once you're satisfied press update connector in the top right.



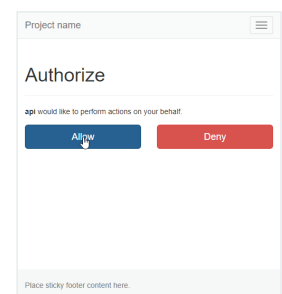
ESTABLISHING A CONNECTION

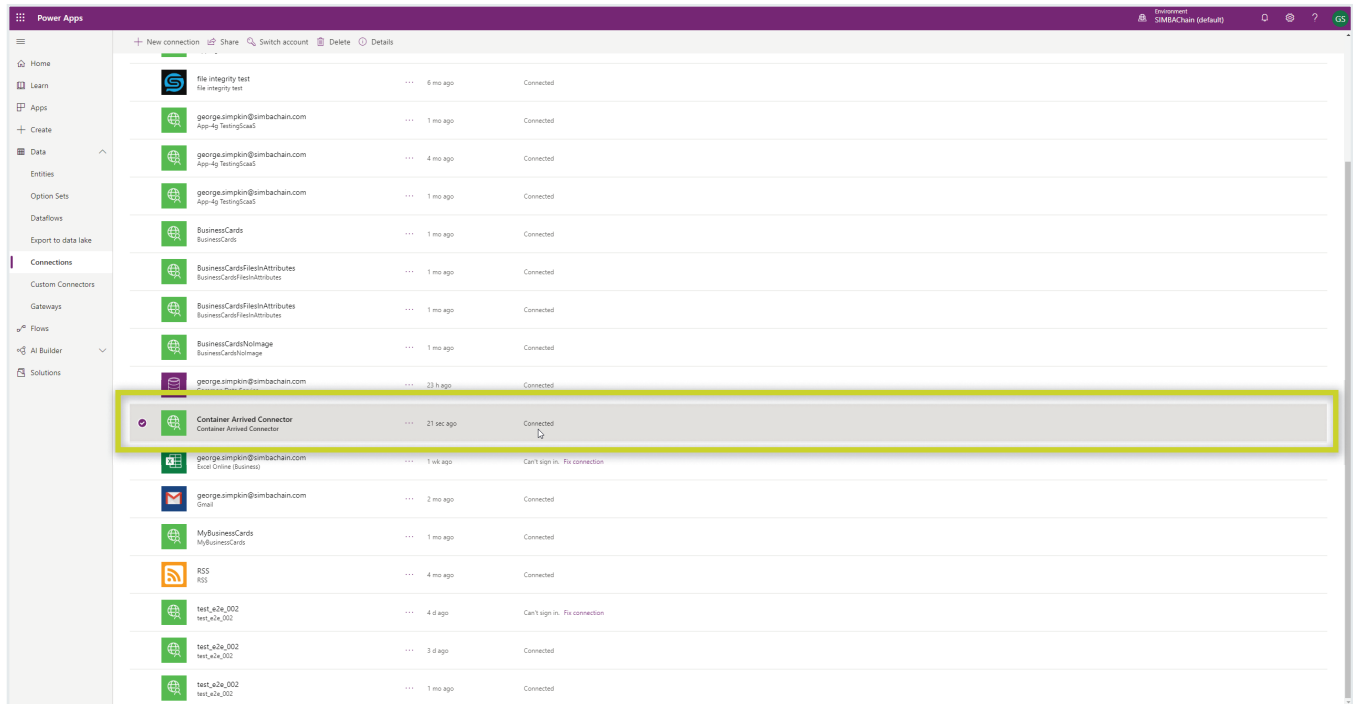
Once you've pressed close to exit the custom connector creation wizard, you should be back on the custom connectors screen.

To create a connection, find your connector in the list and press the + symbol



If a menu pops up press continue, then you should see an authorization screen, press allow.

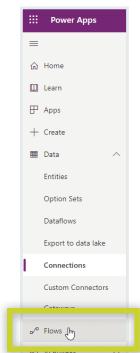




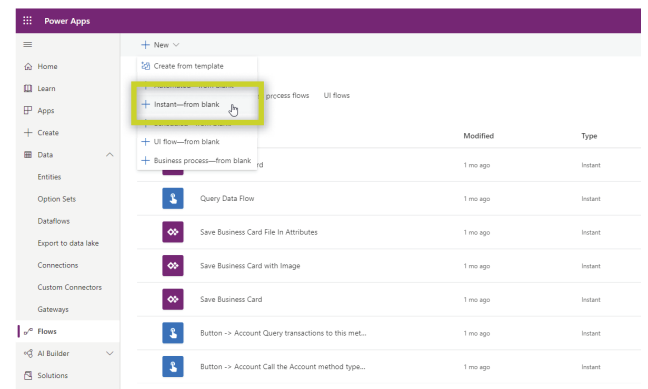
Now if you navigate to Connections via the left hand side drawer, you should be able to find your connector, and it should say you are connected

USING THE CONNECTOR IN A FLOW

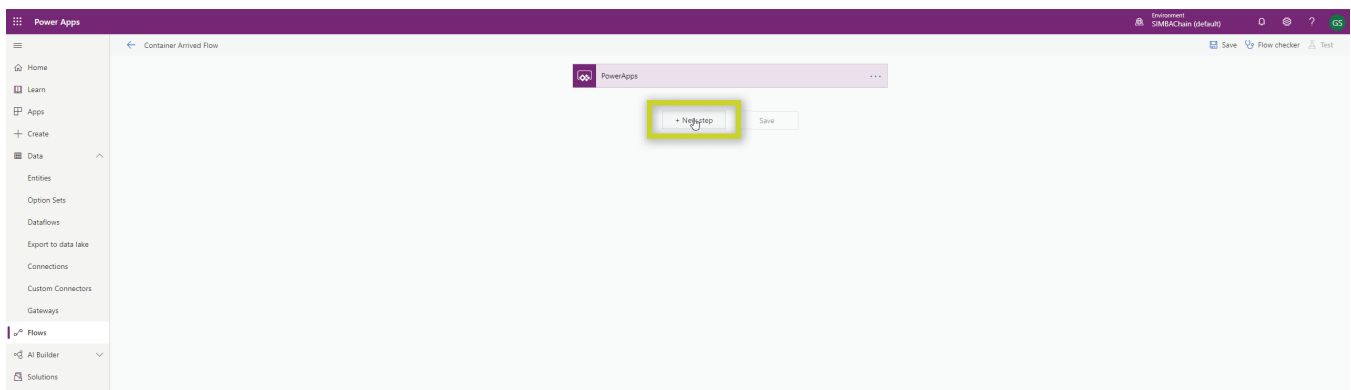
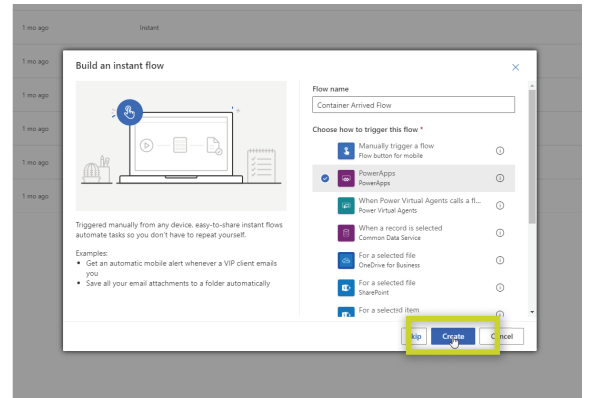
To use our new connector in a flow navigate to Flows in the drawer on the left hand side of the screen.



Then click on the new dropdown and select which type of flow you wish to use. For our example I will be using an Instant Flow, which is the most appropriate type for use from a Power App. If you wanted a more automated flow then choose Automated, and it will run in the background.

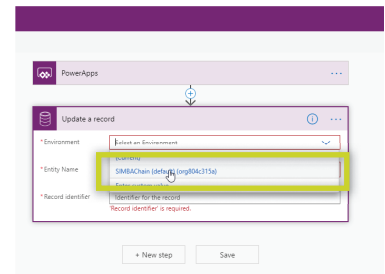


Once you've chosen your flow, name it via the text field and choose what trigger you want to use. For this example I will use a PowerApps trigger, which will allow us to trigger the flow from within a Power App.

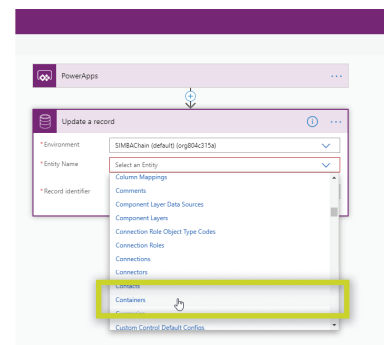


Now you should be on the PowerApps Flow creation page. To begin press the new step button.

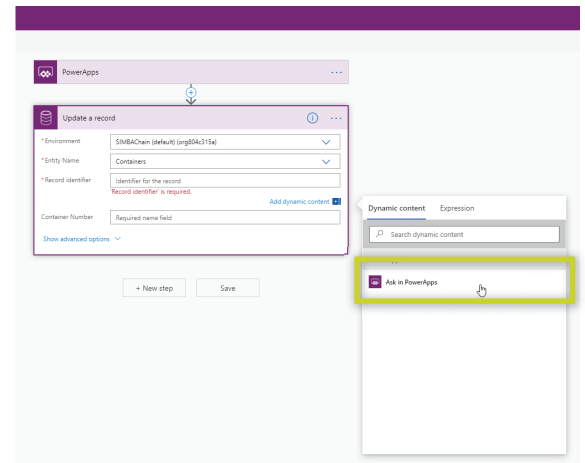
Next choose your environment, here I've used our default organization environment.



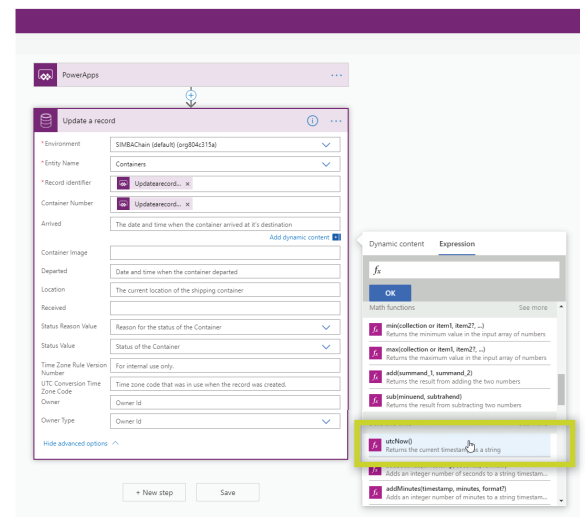
Now choose the entity that you used to create the contract, I used my custom Containers component, so I chose that, you may have chosen a different one, that's fine.



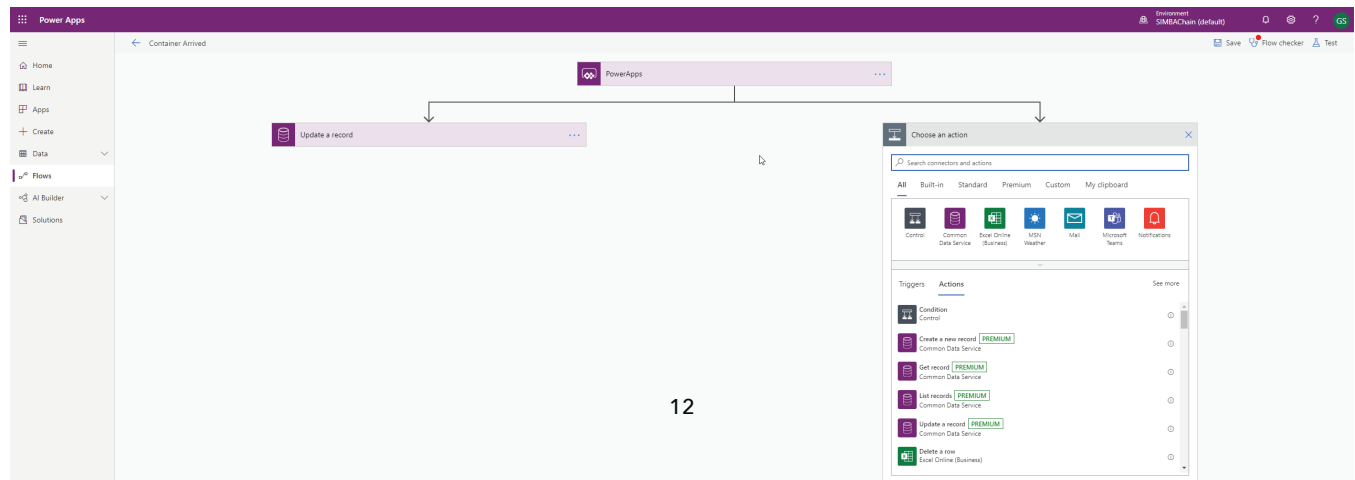
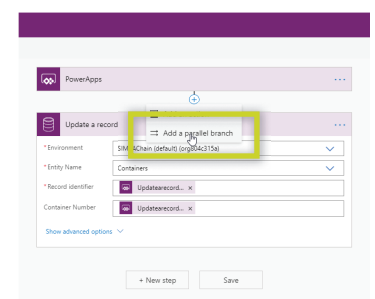
Now for the other fields, if you are going to trigger this flow from a PowerApp, then click on the field, and select Ask From Power Apps from the dynamic content.

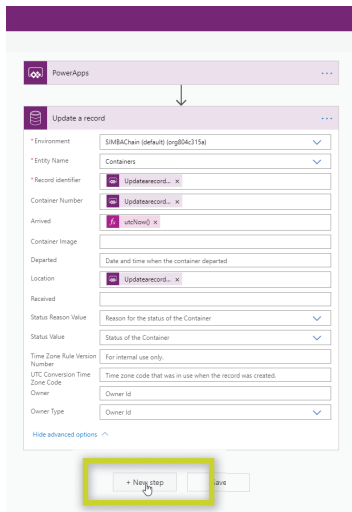


Here we have a datetime field, so I am using a utcNow() expression from the expression tab.

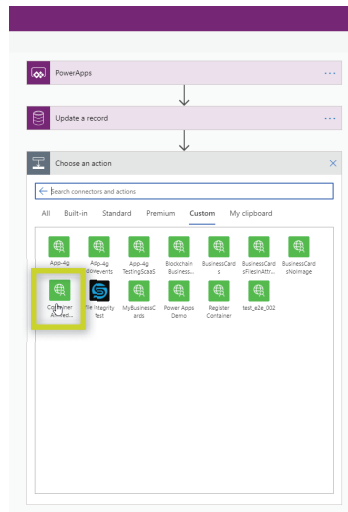


Next we need to add our contract. Since we have a timestamp in this example, I am going to run this flow sequentially, in order to ensure the timestamp is consistent both on the blockchain and in the entity record. It is also possible to run the two operations in parallel, choose whichever you think suits your needs. To run one in parallel simply click the + sign above the step you just created, "Update a record" in my case, and then click "Add a parallel branch".

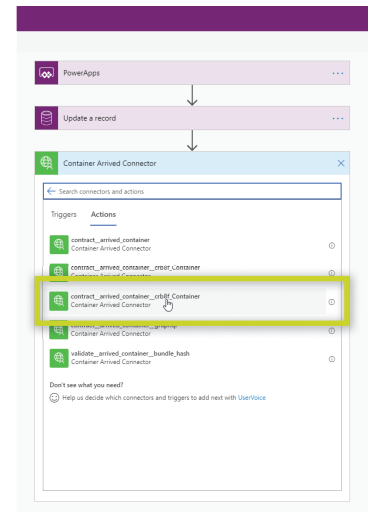




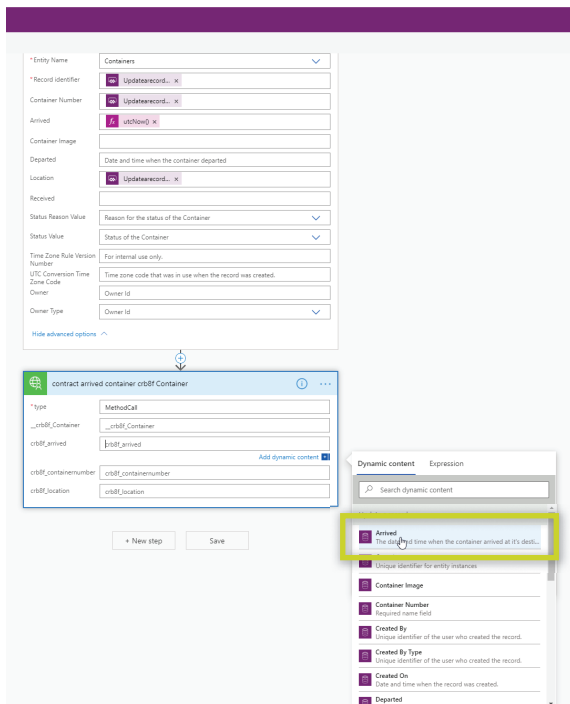
Click the new step button.



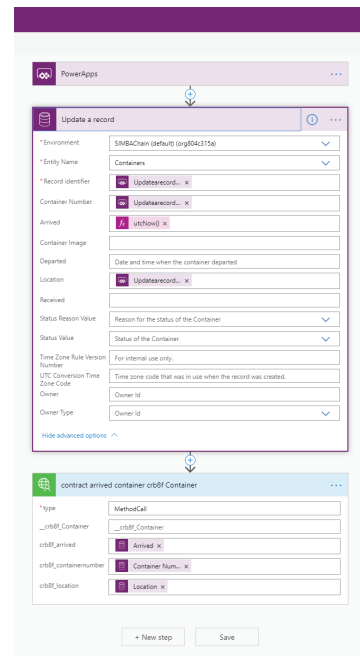
Navigate to the Custom tab and find your newly created custom connector, here ours is called *Container Arrived*



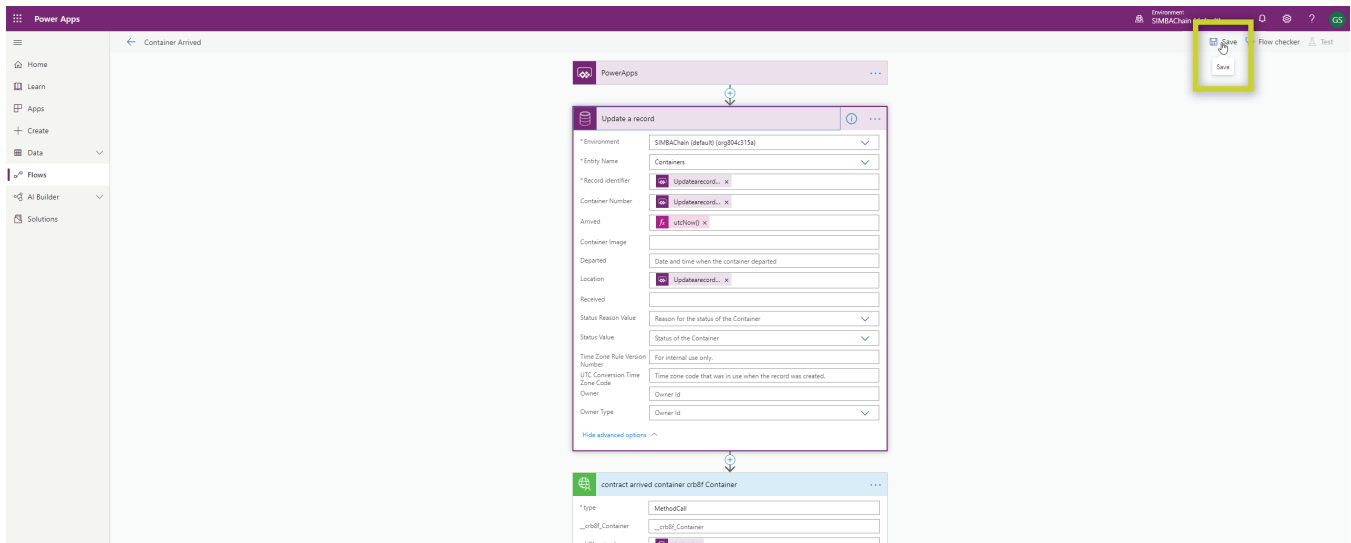
From the actions choose the second one named something similar to *contract_arrived_container__crb8f_Container*



Leave the type field as **MethodCall**, an **__crb8f_Container** can be left blank. For the remaining fields, choose from the dynamic content, here I've chosen the arrival field of the record I've updated.



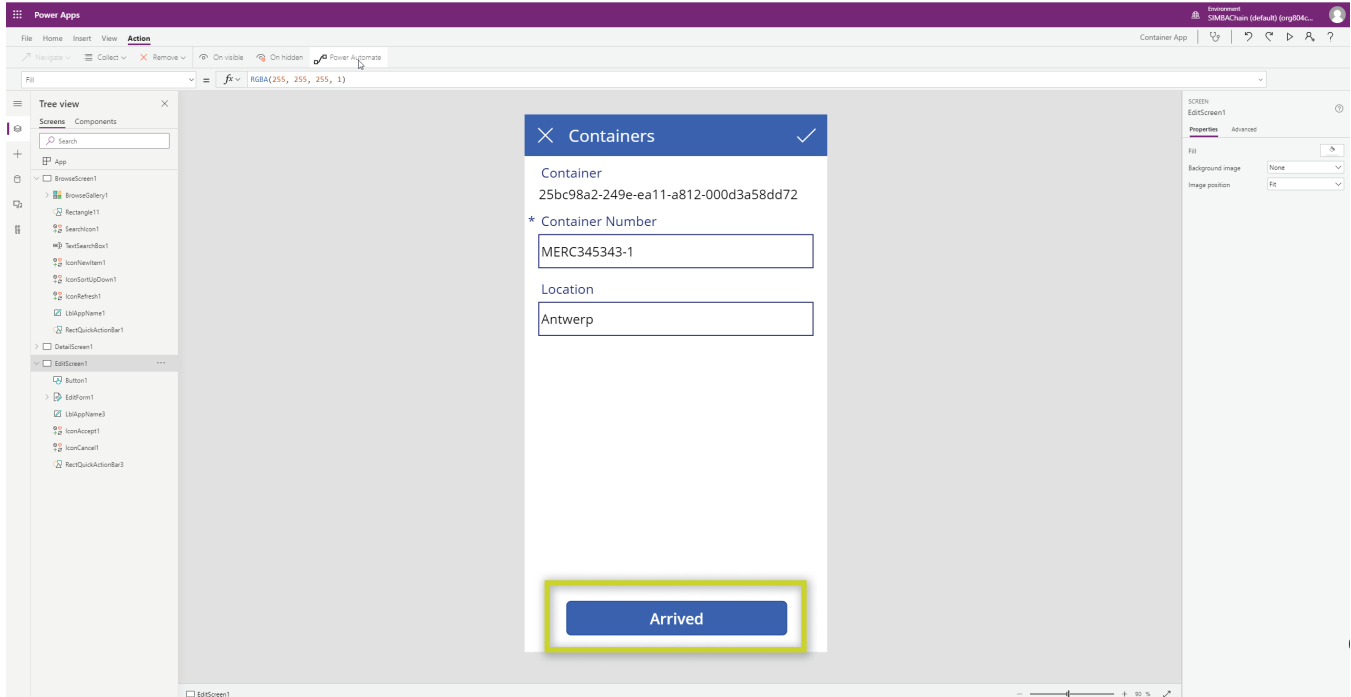
Repeat for all relevant fields. If running in a parallel branch, simply choose from the **Ask in Powerapps** dynamic content you generated earlier, e.g. *Updaterecord_containernumber*.



Once satisfied, press the save button in the top right, and exit the page via the arrow in the top left.

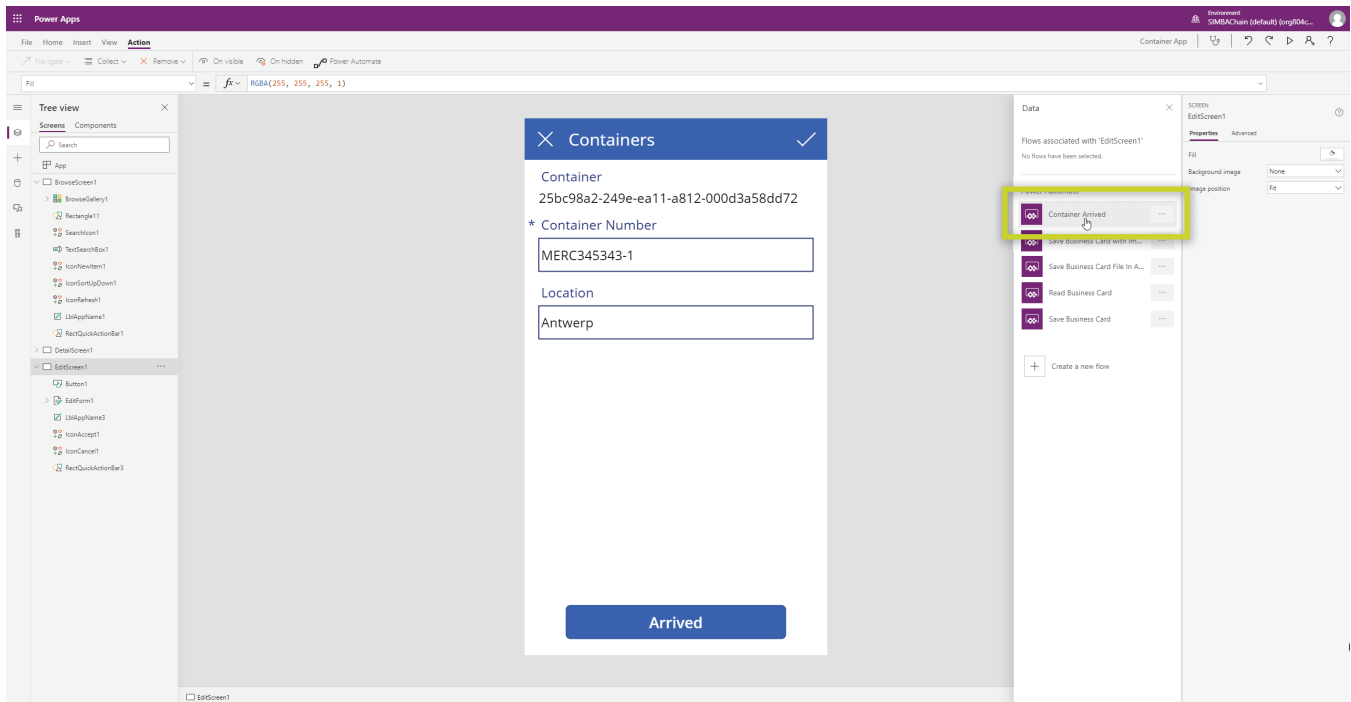
USING THE FLOW IN A POWER APP

To use the Flow in a Power App first navigate to the Power App within which you wish to trigger the flow.

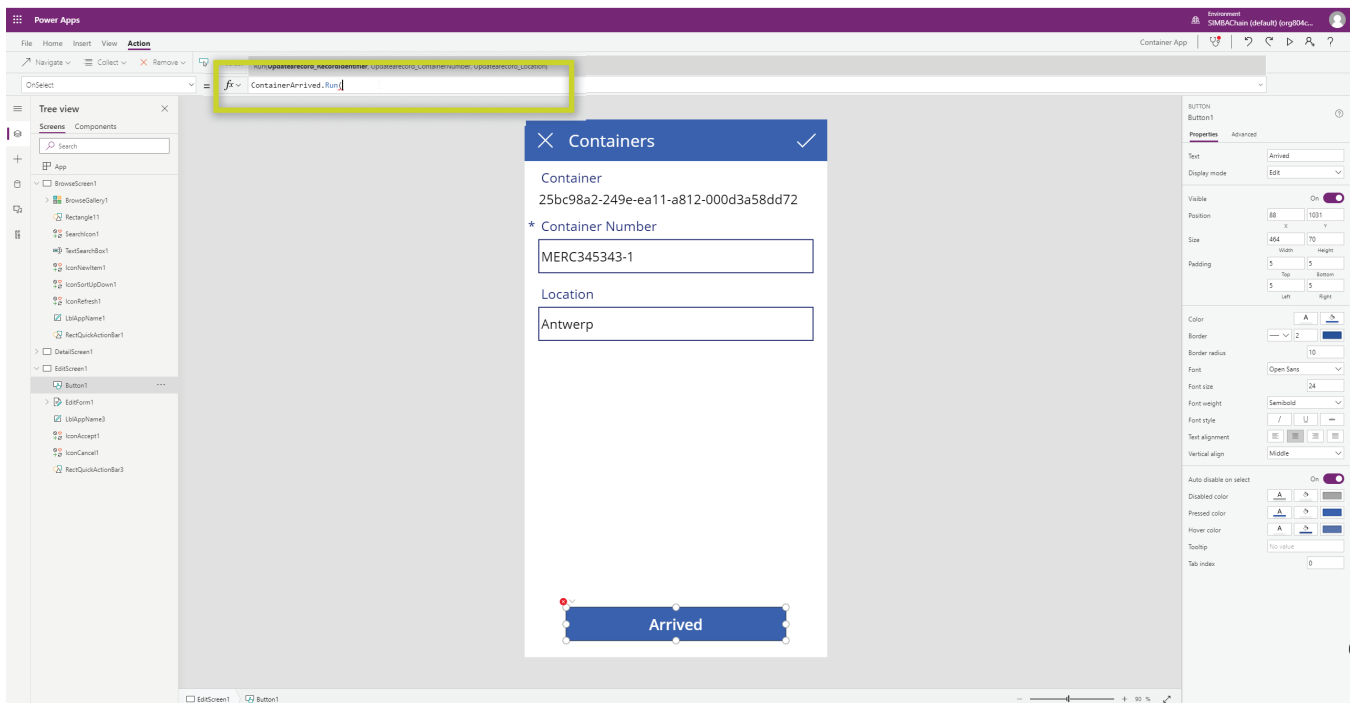


Once in your app, choose a component which you wish to use to trigger the flow, this will usually be a button of some description, here I'm using a button labeled "Arrive".

Next, navigate to Action and click on Power Automate.

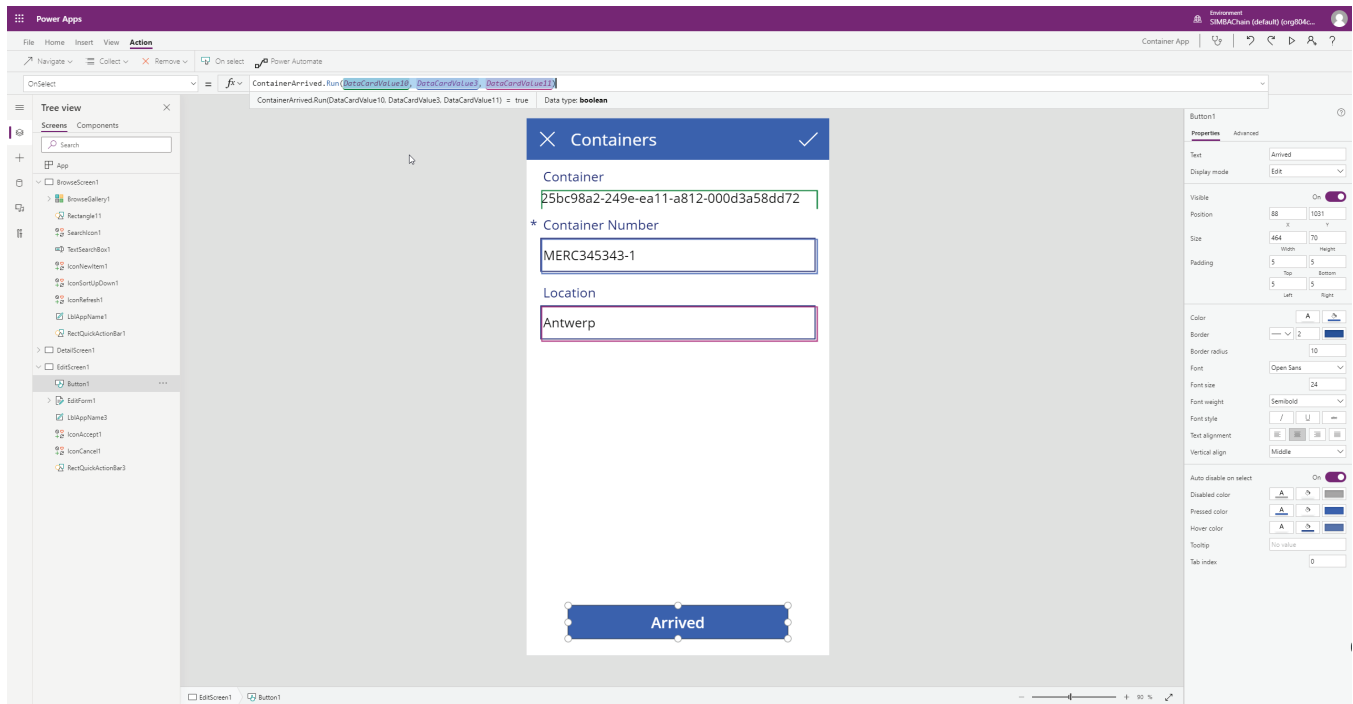


A menu will appear on the right hand side containing your available flows. Select the flow you wish to use and it will be bound to the button. Here I've chosen Container Arrived.



Once it's finished loading, you should see an incomplete function has been loaded into the function bar `fx` which should be similar to **ContainerArrived.Run()**

Clicking in the function bar, a tooltip should appear telling you what data fields to include for each of the inputs. This is mapping to the **Ask in Powerapps** dynamic content from earlier.



Fill out the function. An easy way to know what input is what is to look at the tree view on the left hand side and simply type the name of the respective input into the function in the correct place. In our example, **DataCardValue16** is the **Record Identifier**, **DataCardValue3** is the **Container Number** Input and **DataCardValue15** is the **Location** Input.

Now you're all set! Don't forget to save before you exit. If you want to test immediately, you can click the play button in the top right.



SIMBA Chain enables seamless utilization and integration of blockchain technology to bolster trust, security, and risk mitigation for enterprise and government

🌐 simbachain.com

☎ 574-914-4446

✉ info@simbachain.com