**Azure AI ML Business Process Automation Lab using Power Platform and Azure cognitive services**

GIT URL : <https://github.com/CSALabsAutomation/azure-ai-ml-bpa-using-powerapps-lab>

**Prerequisites**

1. Create azure resources using the ARM Template

URL : <https://github.com/CSALabsAutomation/azure-ai-ml-bpa-using-powerapps-lab/blob/main/environments/env1/template.json>

1. Download “BPA\_Template\_v2\_Current.zip” file
2. Download “Calculation\_email\_approval\_template-BPA\_20230111144219.zip” file
3. Download “SendNotificationToFinanceTeam\_template\_20230112095308.zip” file
4. Download Training Documents
5. Download Test Documents
6. Download Postman collection files

**Importing Power Apps template and creating AI Model**

1. Open new tab and go to <https://make.preview.powerapps.com/>
2. Select “Apps” from the left navigation and Click on “Import Canvas app

Graphical user interface, text, application, email

Description automatically generated

1. Upload the “.zip” file then click on “Import”.

Graphical user interface, text, application, email

Description automatically generated

1. Click on “Open App”

Graphical user interface, text, application, email

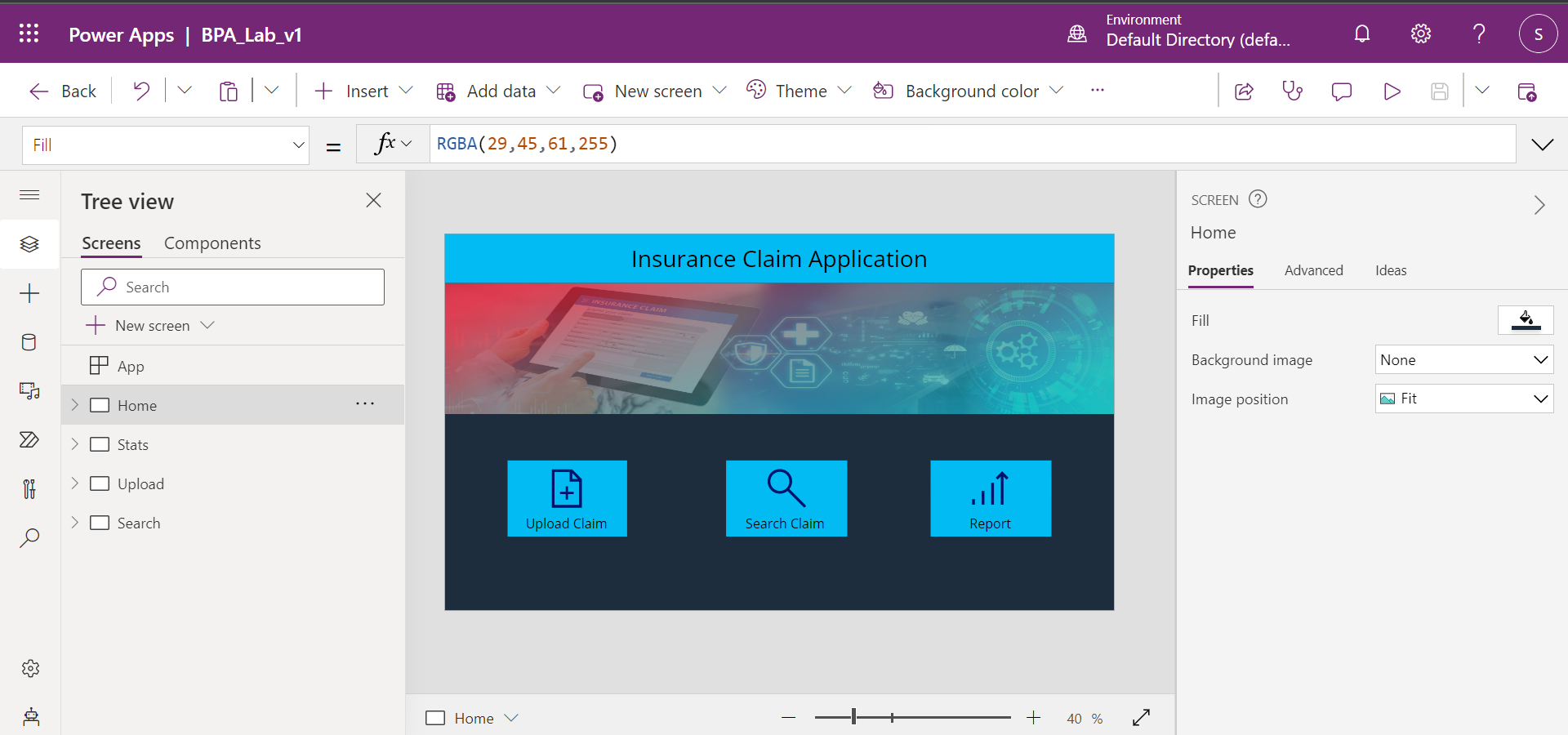
Description automatically generated

1. Click on “Get Started”

Graphical user interface, text, application, email, Teams

Description automatically generated

1. App will be open with pre built pages and controls



1. Click on Upload in the Tree view pane.

Graphical user interface

Description automatically generated

1. Select “Form Processor ” control on the screen. Click on “Select Model” from the properties section. We need to create a new model here for the app.



1. Selecting “New Model” will open a new tab. Select “Extract information from invoices”.

Graphical user interface, application

Description automatically generated

1. Select “Document processing custom model” then “Get started”.

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

1. Select “Structured documents” and click on Next

Graphical user interface, text

Description automatically generated

1. Click on add button to add fields. Select Field and click next. Enter “InvoiceID” and click on Done.

Graphical user interface, application

Description automatically generated

Graphical user interface, application, Teams

Description automatically generated

1. Repeat same for below fields.

* InvoiceDate
* Category
* ClaimType
* Amount

After adding all the fields click on next

Graphical user interface

Description automatically generated

1. Click on New collection.

Graphical user interface, text, application, email

Description automatically generated

1. Click on + Icon and then on “Add documents”. Upload 5 training documents that you have downloaded before from Training documents folder in the prerequisites.

Graphical user interface, application

Description automatically generated

Click on “Upload 5 documents“. After documents get uploaded click on “Done” and then “Next”

Graphical user interface, application, Teams

Description automatically generated

Graphical user interface, table

Description automatically generated with medium confidence

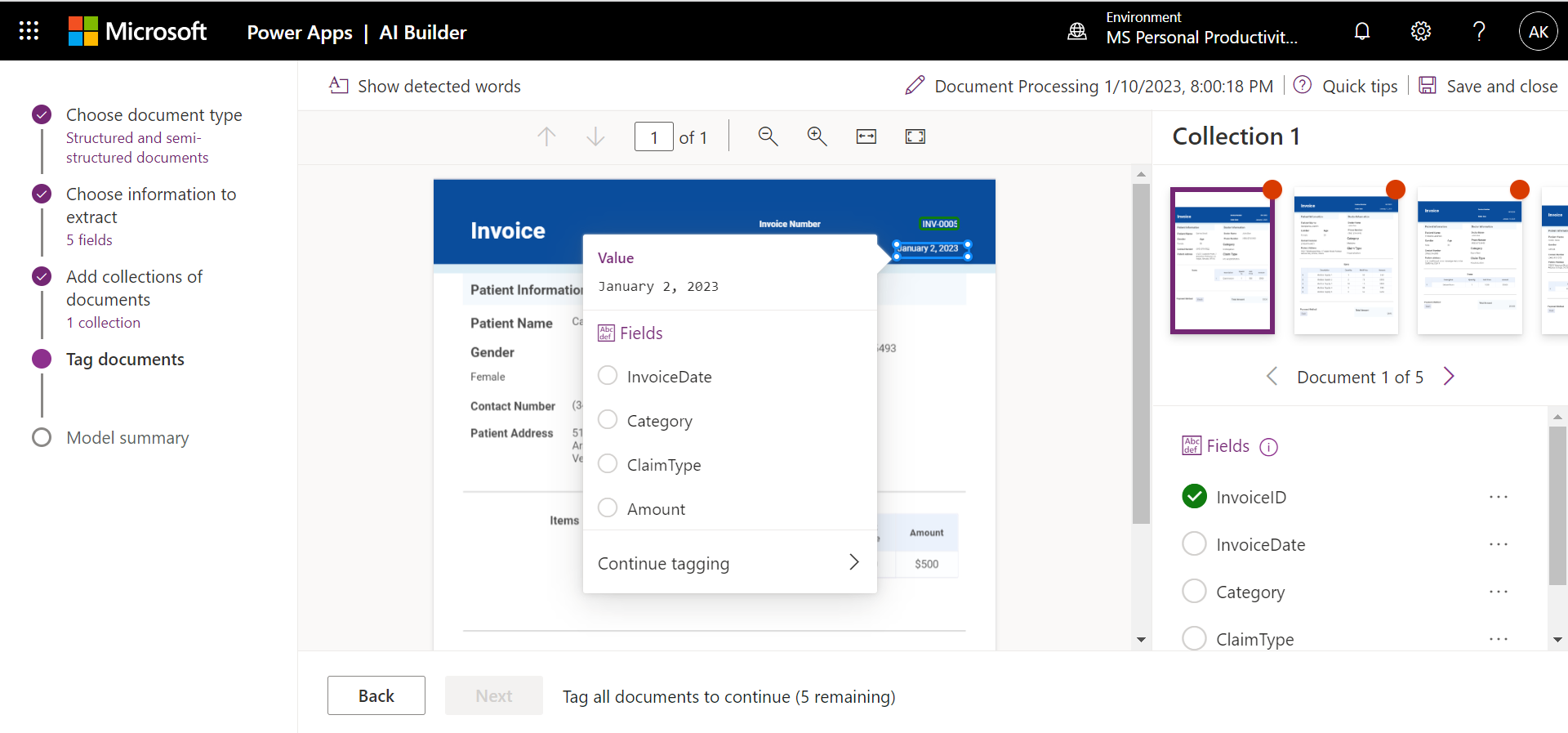
1. Now after document get uploaded, you need to mark the fields in the documents.

Mark invoiceID (INV00X) on document and select InvoiceID as field.

Graphical user interface, application

Description automatically generated

Mark Date on document and select InvoiceDate as field

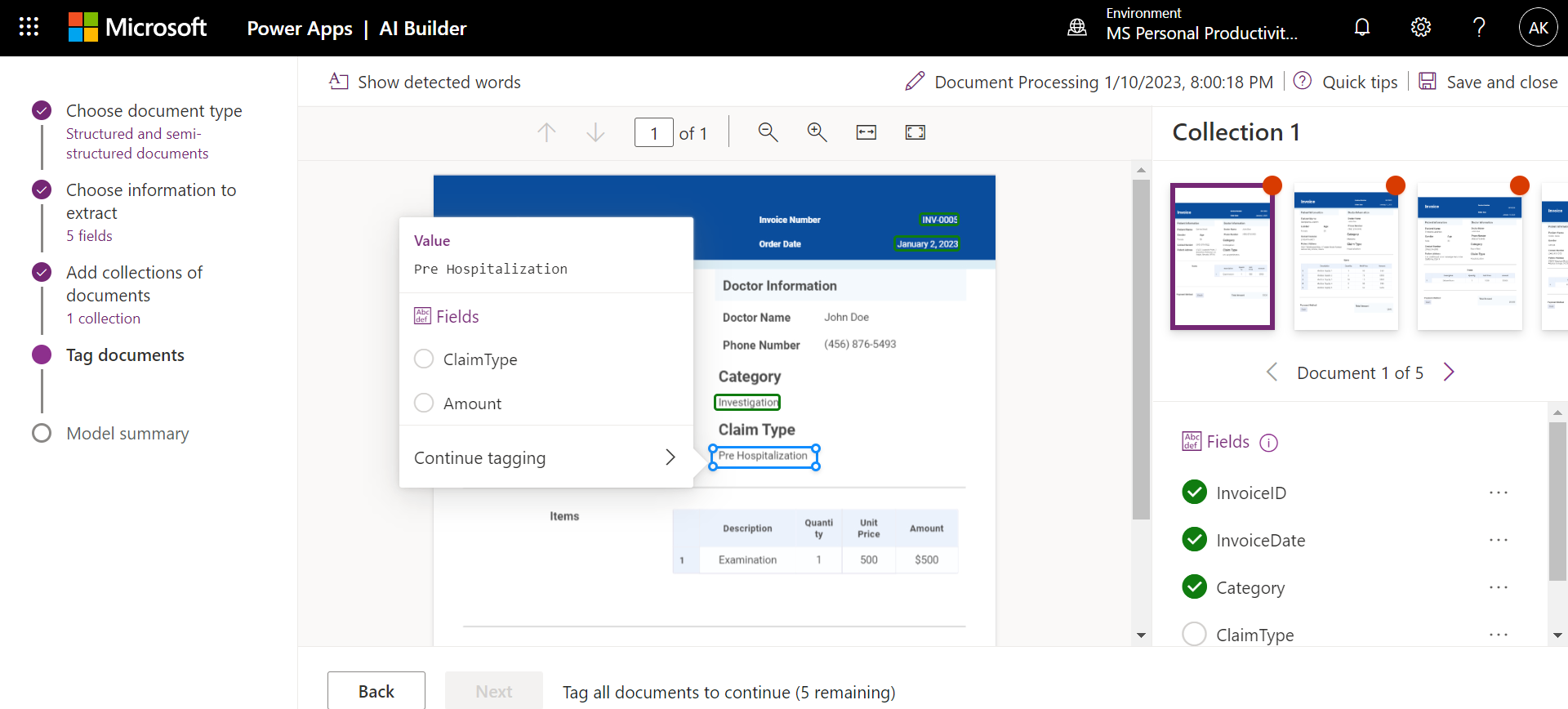


Mark Category data on document and select Category as field

Graphical user interface, application

Description automatically generated

Mark Claim type data on document and select ClaimType as field

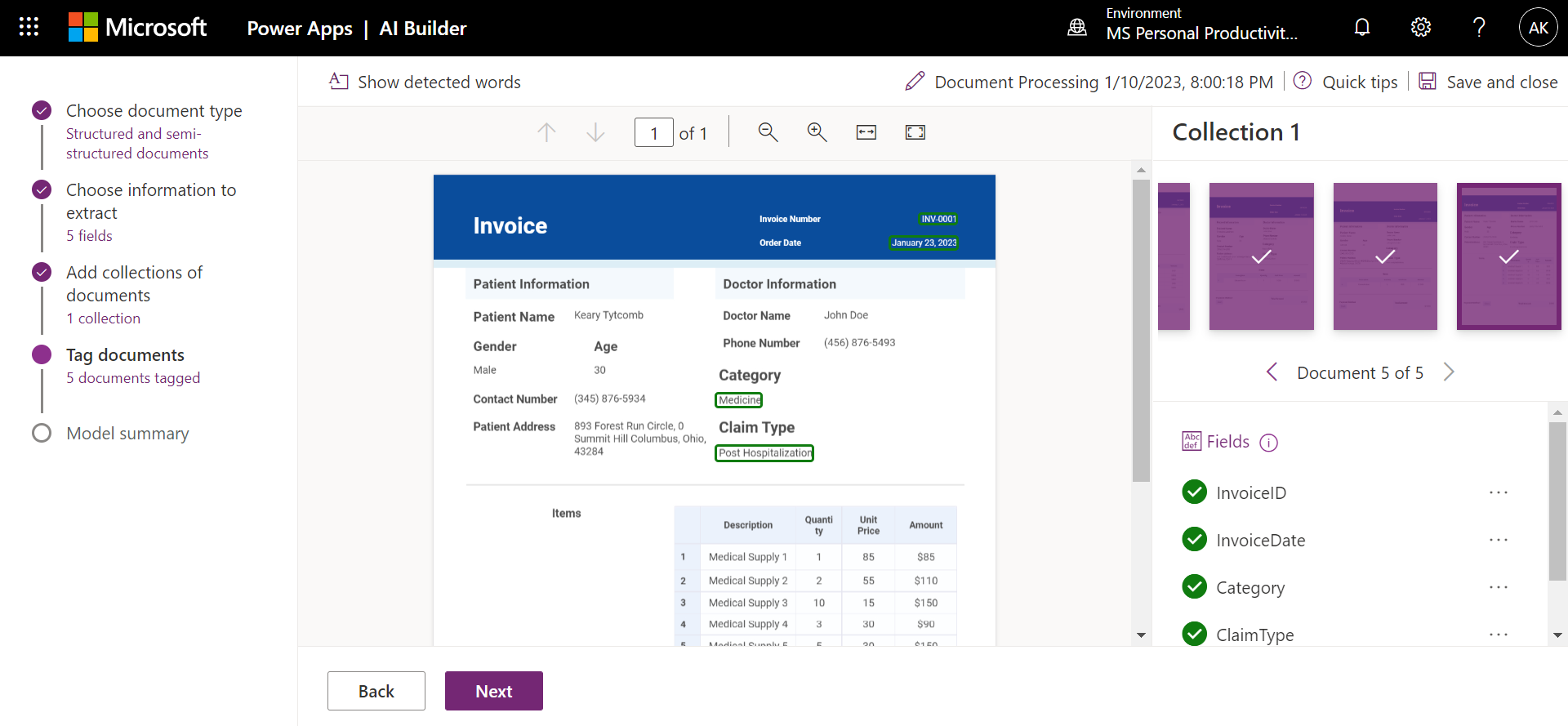


Mark Amount data on document and select Amount as field

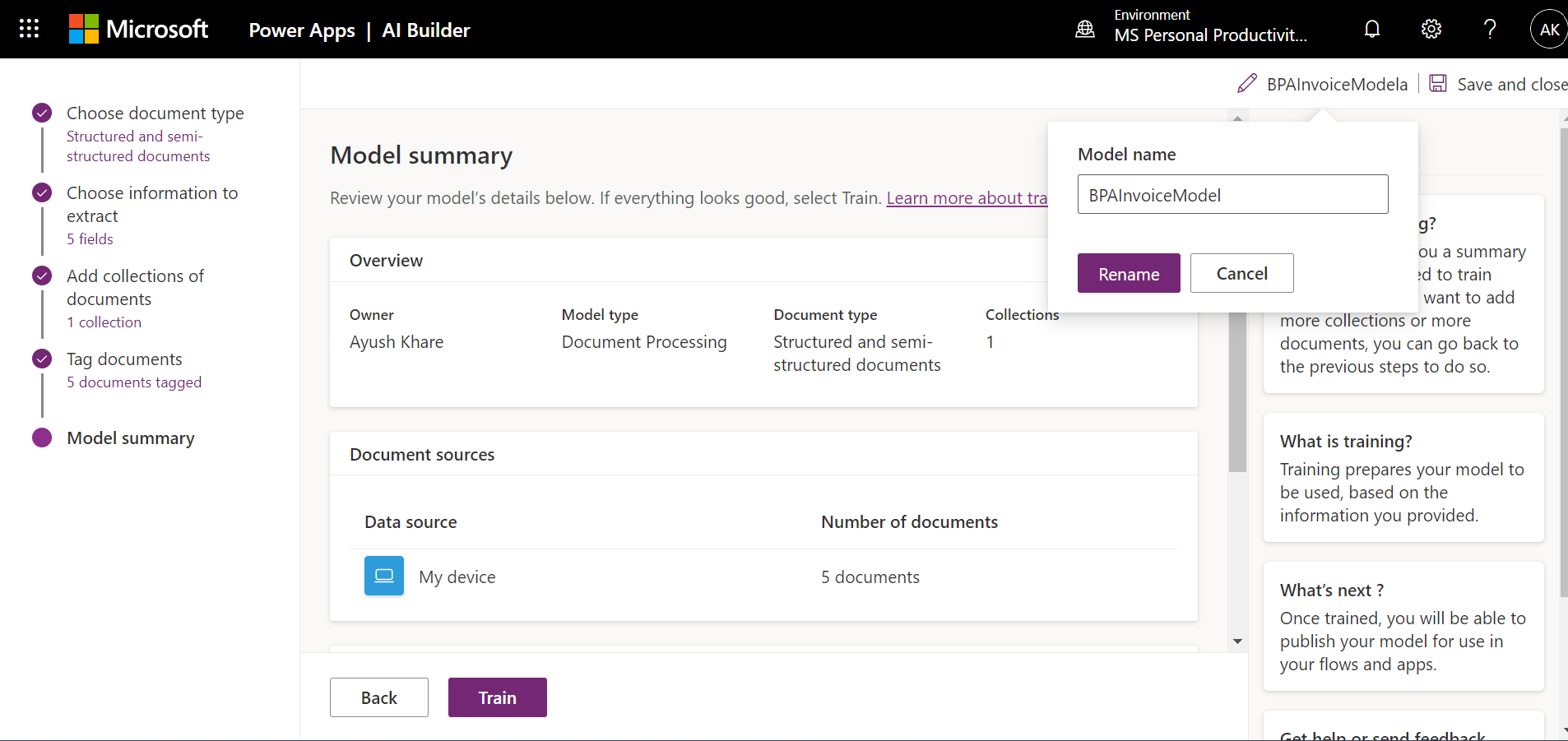
Graphical user interface

Description automatically generated

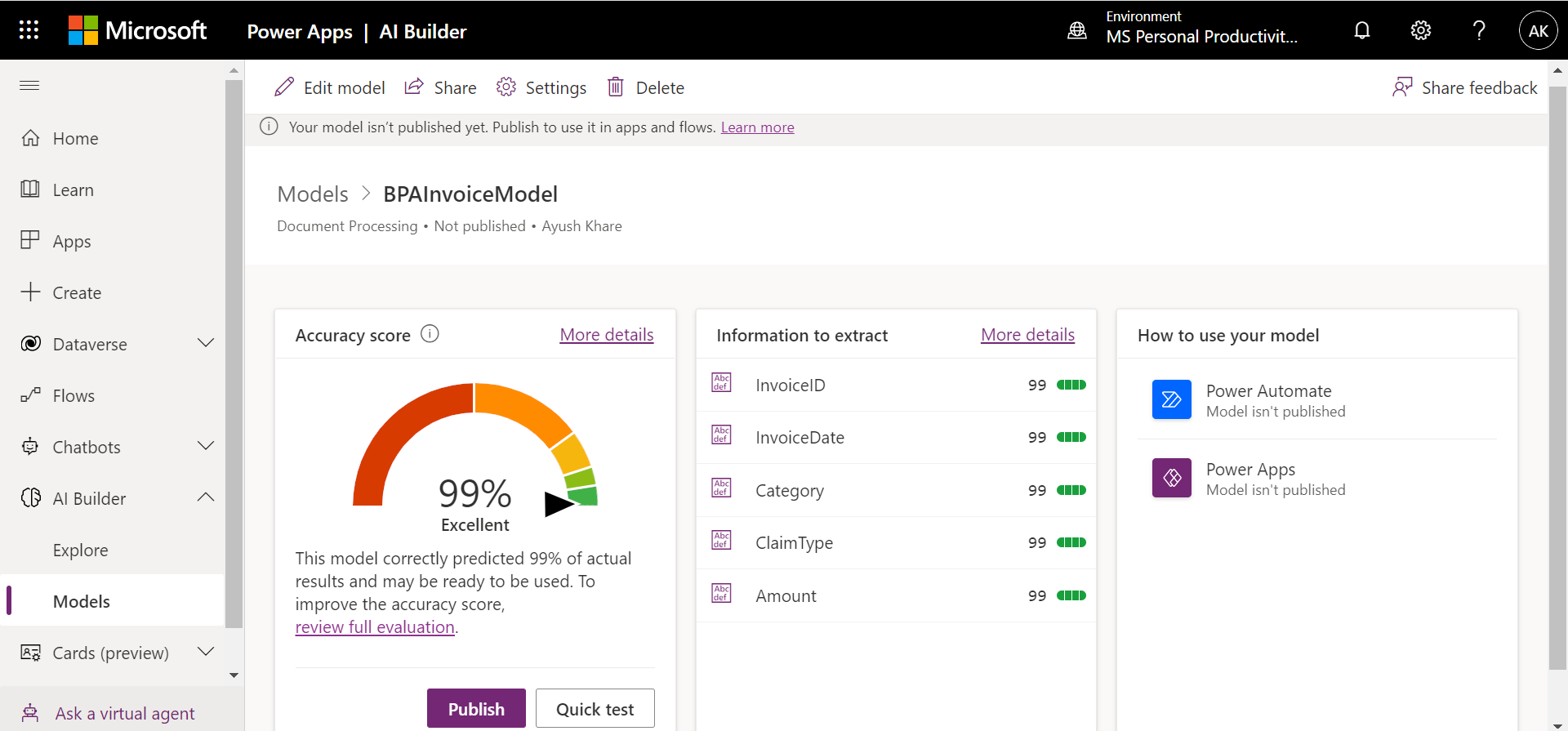
1. Now repeat same for other 4 documents and click on Next.



Now Rename the Model to “BPAInvoiceModel” then click on Train and wait for 2 mins.



1. Click on “Publish”



1. Go back to PowerApp tab where app is open. And select the newly created AI Model.

Assign same AI Model to other 3 Form Processor control.

Graphical user interface, text

Description automatically generated

1. Go to Power Automate section in left navigation and select Add flow. Then create from blank. Procced with the actions as mentioned in the attached template.

Note : instructions to be provided to the user for creating the flow not template.

Git URL : <https://github.com/CSALabsAutomation/azure-ai-ml-bpa-using-powerapps-lab/blob/main/environments/env1/Artifacts/Calculation_email_approval_template-BPA_20230111144219.zip>

Flow Name : Calculation\_email\_approval

Graphical user interface, application

Description automatically generated

**Configuring Azure Cognitive Search**

1. ACS will be created by the ARM template. Instruction to be provided for creating the index

Add below fields.

* InvoiceID
* InvoiceDate
* Category
* ClaimType
* Amount

Select all the attributes:

**Graphical user interface, application, Word

Description automatically generated**

**Creating Custom Connectors using the Postman collection**

1. Download the Postman collection from the Git Repo.

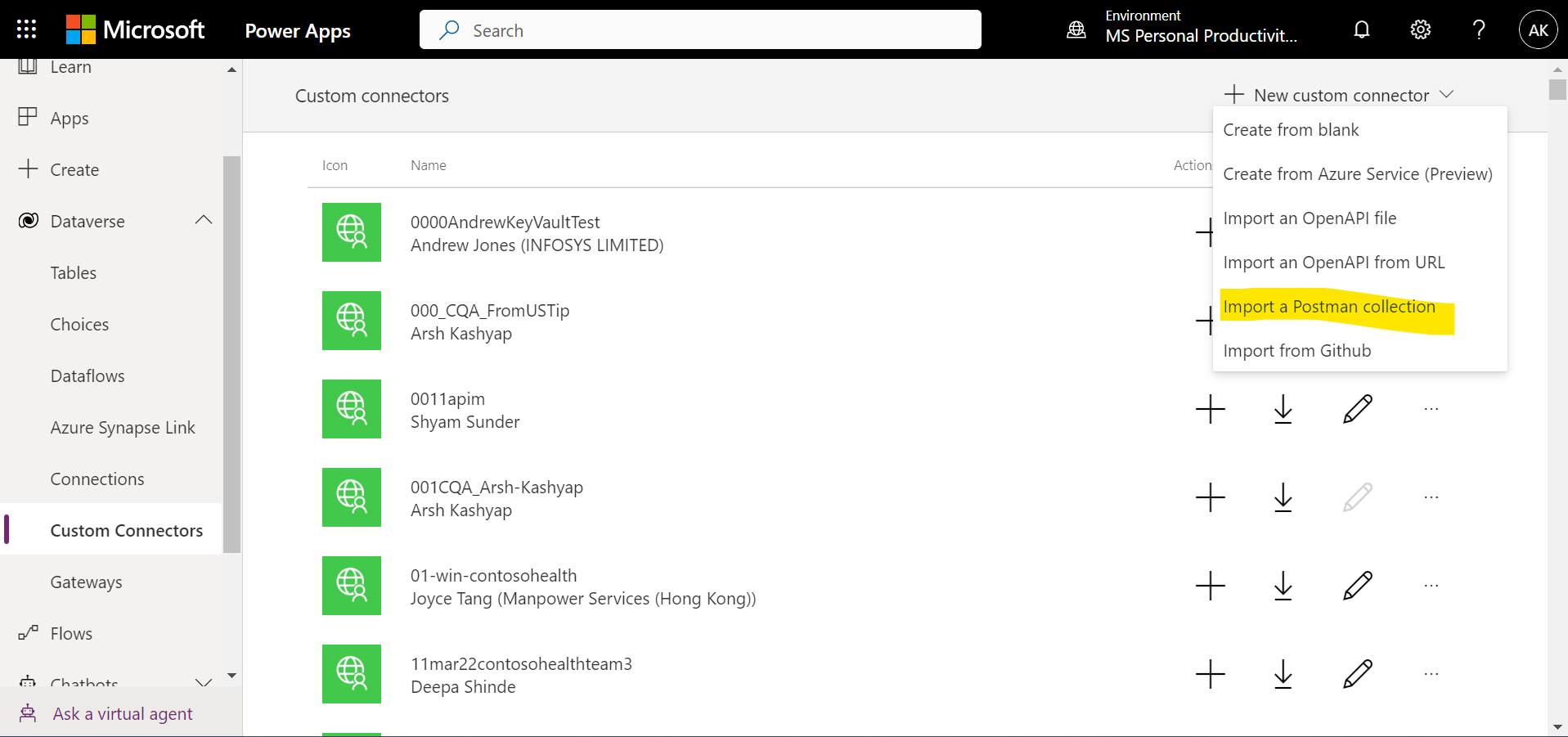
Git Repo URL :

Search : <https://github.com/CSALabsAutomation/azure-ai-ml-bpa-using-powerapps-lab/blob/main/environments/env1/Artifacts/ACSBPA_Search.postman_collection.json>

Upload : <https://github.com/CSALabsAutomation/azure-ai-ml-bpa-using-powerapps-lab/blob/main/environments/env1/Artifacts/ACSBPA_Upload.postman_collection.json>

Note : Update the API Key and the ACS URL after downloading the file

1. Open Power Apps and create a new custom connector using downloaded postman files.



1. Connector Upload

* Import
* In Body provide Title to each parameter and set Is Required as “No”

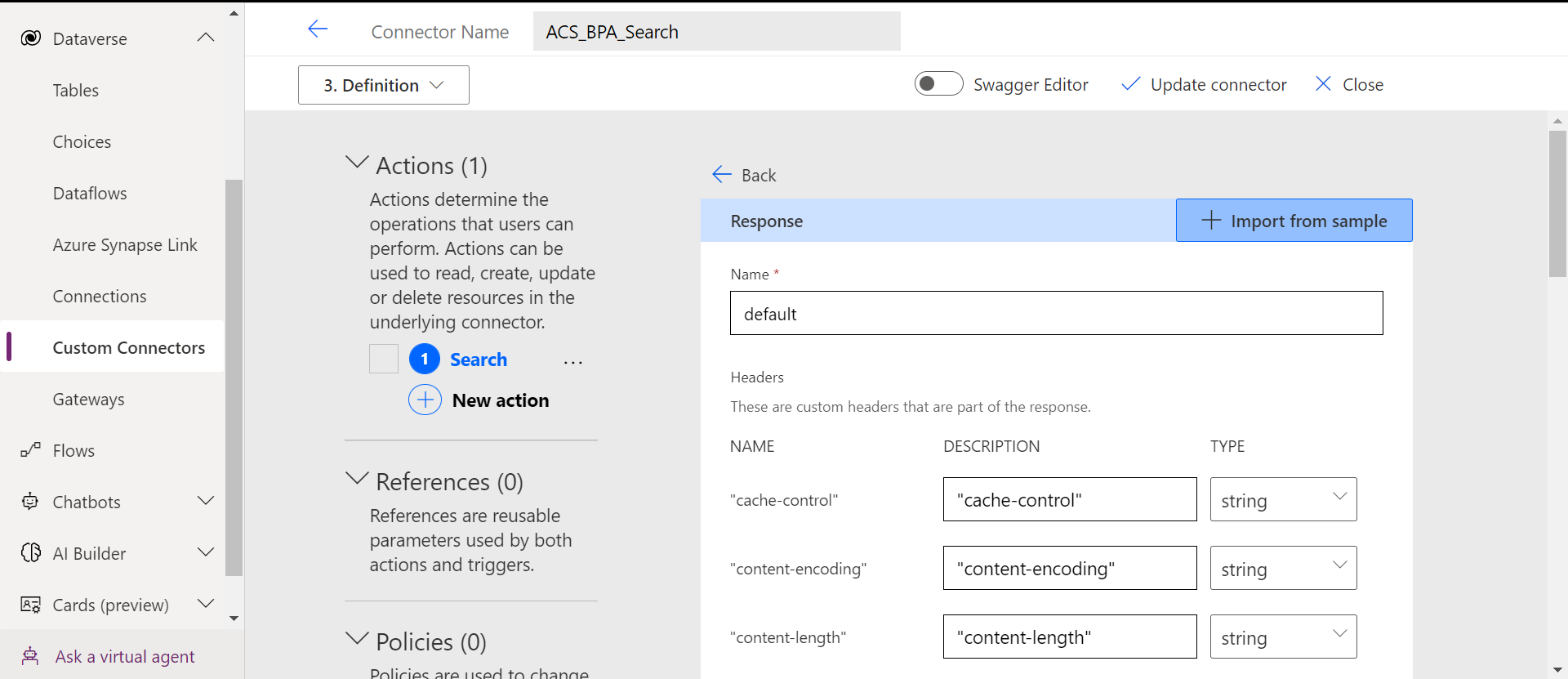
Graphical user interface, text, application, email

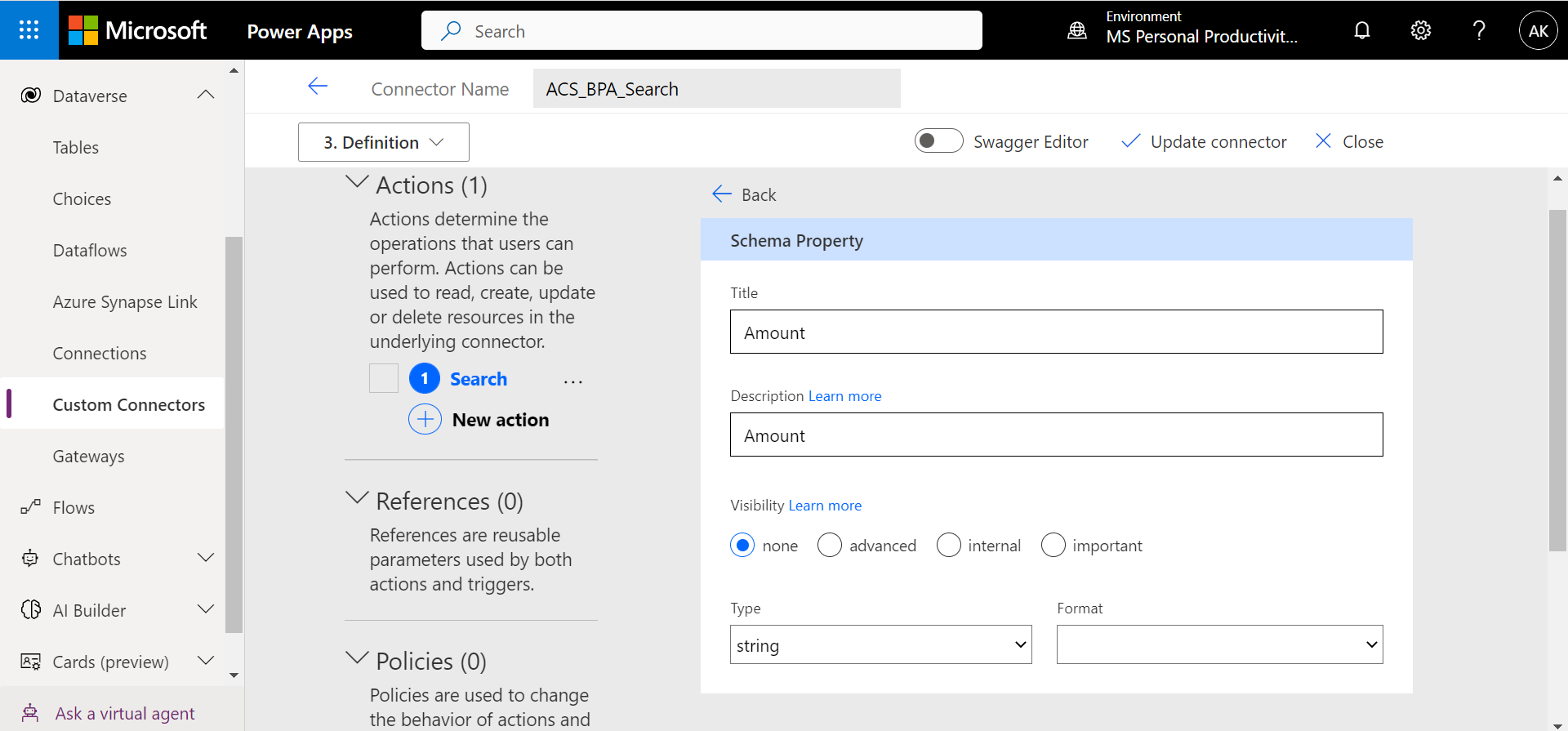
Description automatically generated

* Test

1. Connector search

* Import
* Update the response using the sample import and provide Title to each parameter





* Test

1. Goto PowerApp Upload screen again and Use below code for OnSelect of the Submit button.

Set(varAmount1,Right(Label6\_1.Text,Len(Label6\_1.Text) - 1));

Set(varAmount2,Right(Label6\_5.Text,Len(Label6\_5.Text) - 1));

Set(varAmount3,Right(Label6\_6.Text,Len(Label6\_6.Text) - 1));

Set(varAmount4,Right(Label6\_7.Text,Len(Label6\_7.Text) - 1));

Set(vartab,Table(

{'@search.action':"upload",id:"1", InvoiceID: Label6.Text, InvoiceDate: Label6\_8.Text, Category: Label6\_9.Text, ClaimType: Label6\_10.Text, Amount: varAmount1},

{'@search.action':"upload",id:"2", InvoiceID: Label6\_2.Text, InvoiceDate: Label6\_11.Text, Category: Label6\_14.Text, ClaimType: Label6\_15.Text, Amount: varAmount2},

{'@search.action':"upload",id:"3", InvoiceID: Label6\_3.Text, InvoiceDate: Label6\_12.Text, Category: Label6\_16.Text, ClaimType: Label6\_17.Text, Amount: varAmount3},

{'@search.action':"upload",id:"4", InvoiceID: Label6\_4.Text, InvoiceDate: Label6\_13.Text, Category: Label6\_18.Text, ClaimType: Label6\_19.Text, Amount: varAmount4}

));

ACSBPA\_Upload.Upload({value:vartab});

Calculation\_email\_approval.Run(varAmount1,varAmount2,varAmount3,varAmount4);

Notify("Claim Submitted",NotificationType.Success);

Navigate(Home);

1. Add the data sources from upload and search custom connector.

Graphical user interface, text, application

Description automatically generated

**Search screen setup in Power Apps**

1. Open the Search screen and Use below code for OnSelect of the Search button

ClearCollect(

HotelCol,

ACSBPA\_Search.Search(

"2021-04-30-Preview",

TextInput1.Text,

"application/json",

"UfRV0s0Ej3nGijCfFepzMCNbYmNFBWAEbeyPgt4xDSAzSeBIbNjl"

).value

);

1. Set HotelCol as the data source for the Gallery.

**Graphical user interface

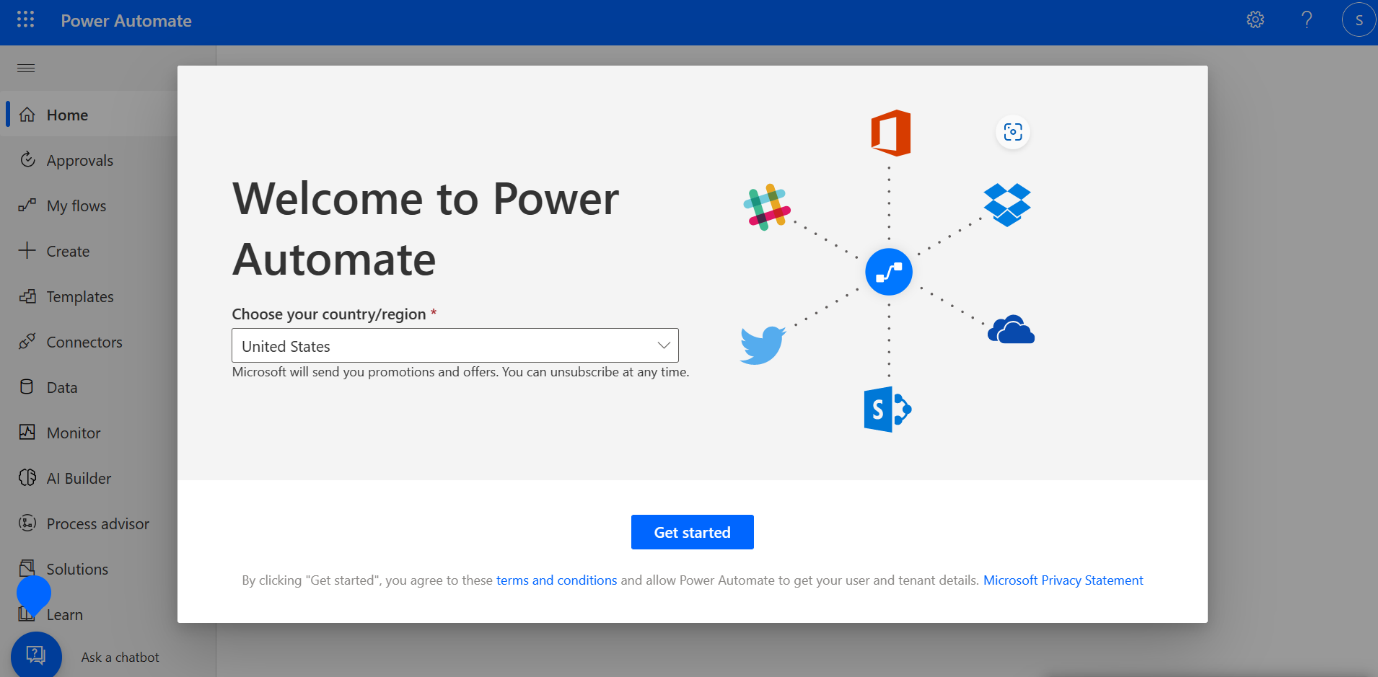
Description automatically generated**

**Configuring Subscriber Power Automate and Event Grid**

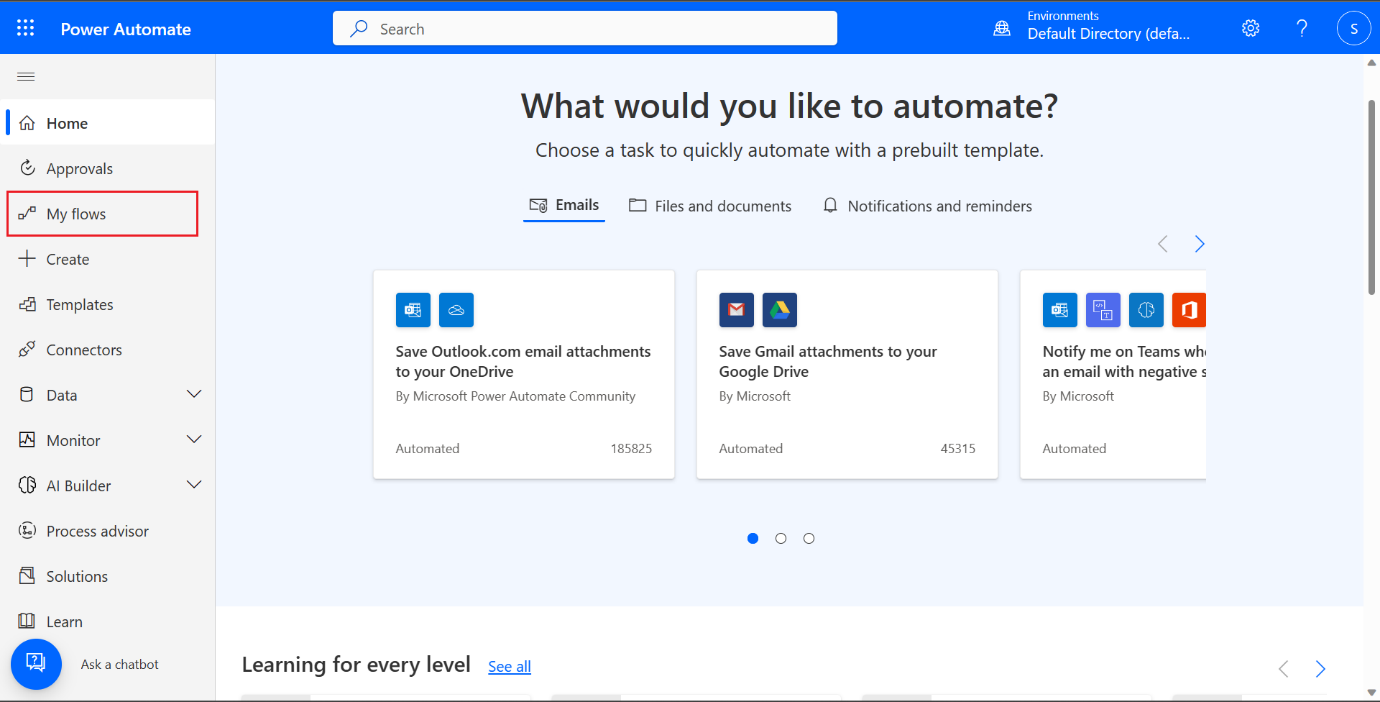
1. Login to the *Power Automate* portal and click *Get Started*.

Flow Template URL :

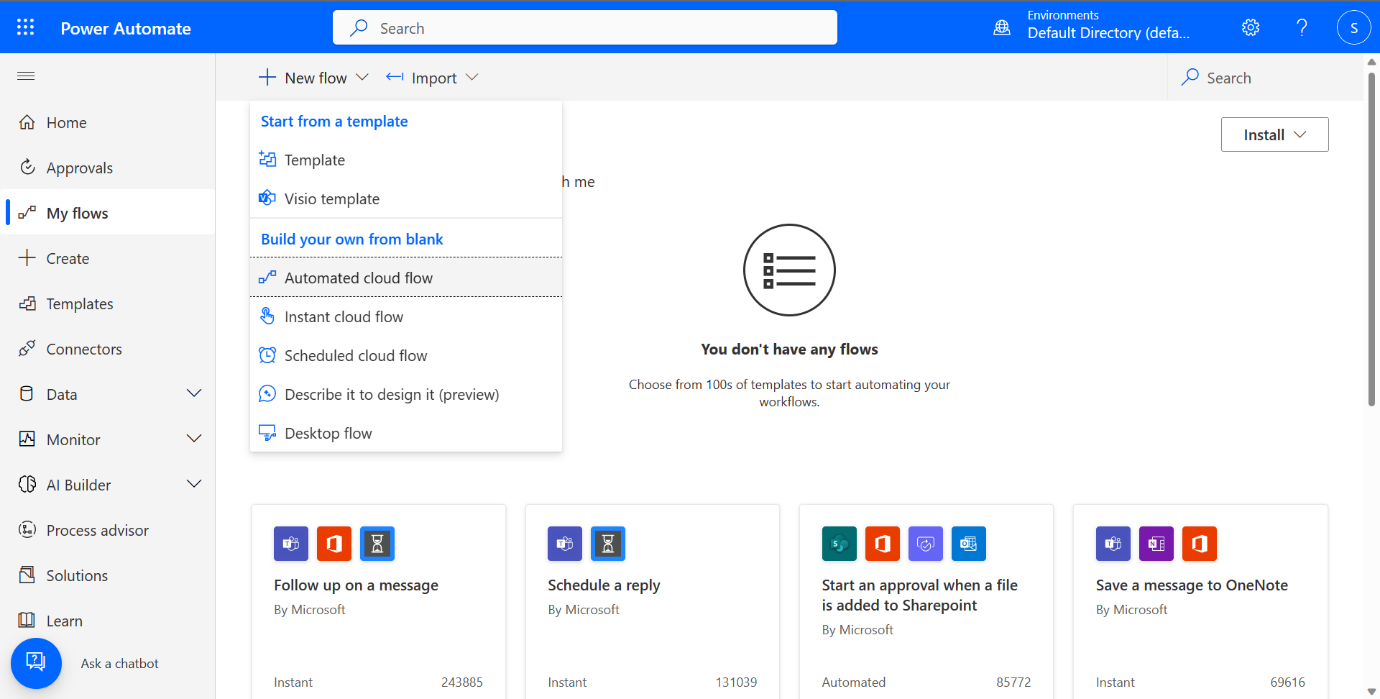
<https://github.com/CSALabsAutomation/azure-ai-ml-bpa-using-powerapps-lab/blob/main/environments/env1/Artifacts/SendNotificationToFinanceTeam_template_20230112095308.zip>

[](https://github.com/CSALabsAutomation/azure-ai-ml-bpa-using-powerapps-lab/blob/main/steps/event_grid_subscription/assets/01.png)

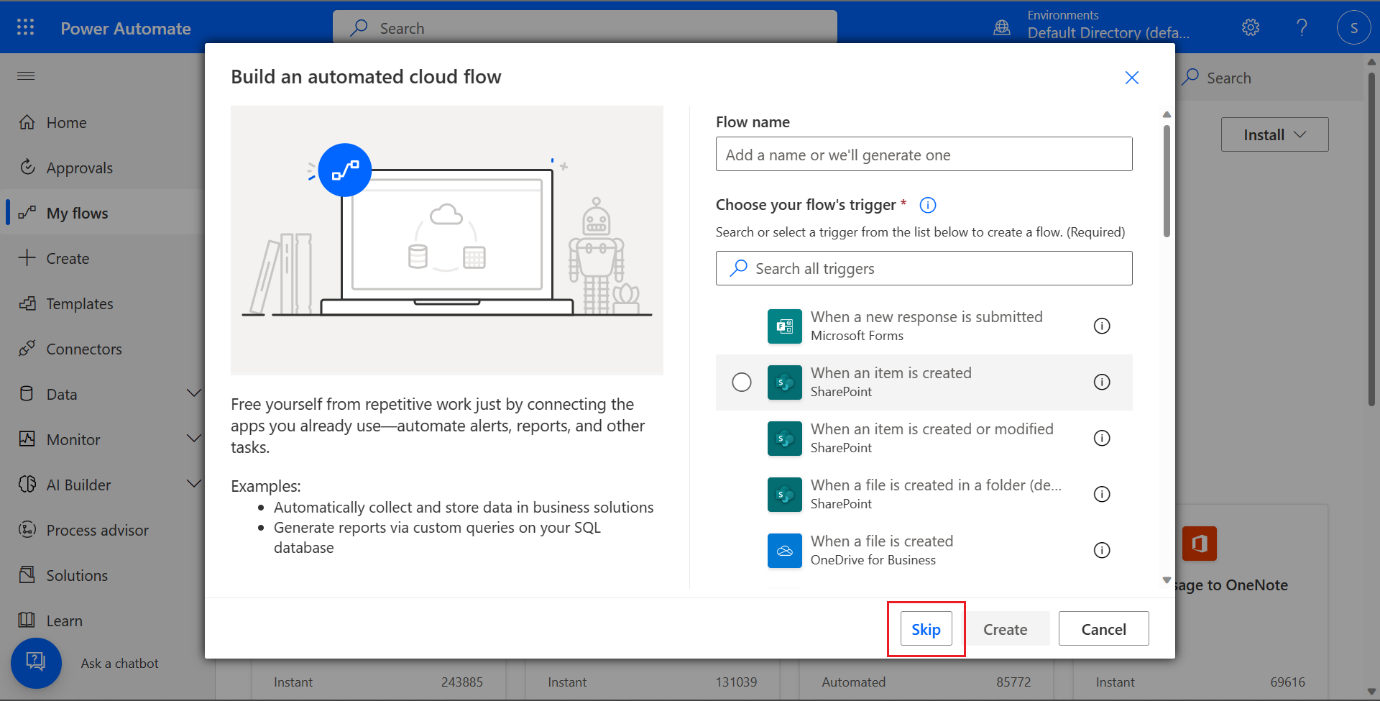
1. Click *My flows* from the left navigation bar.

[](https://github.com/CSALabsAutomation/azure-ai-ml-bpa-using-powerapps-lab/blob/main/steps/event_grid_subscription/assets/02.png)

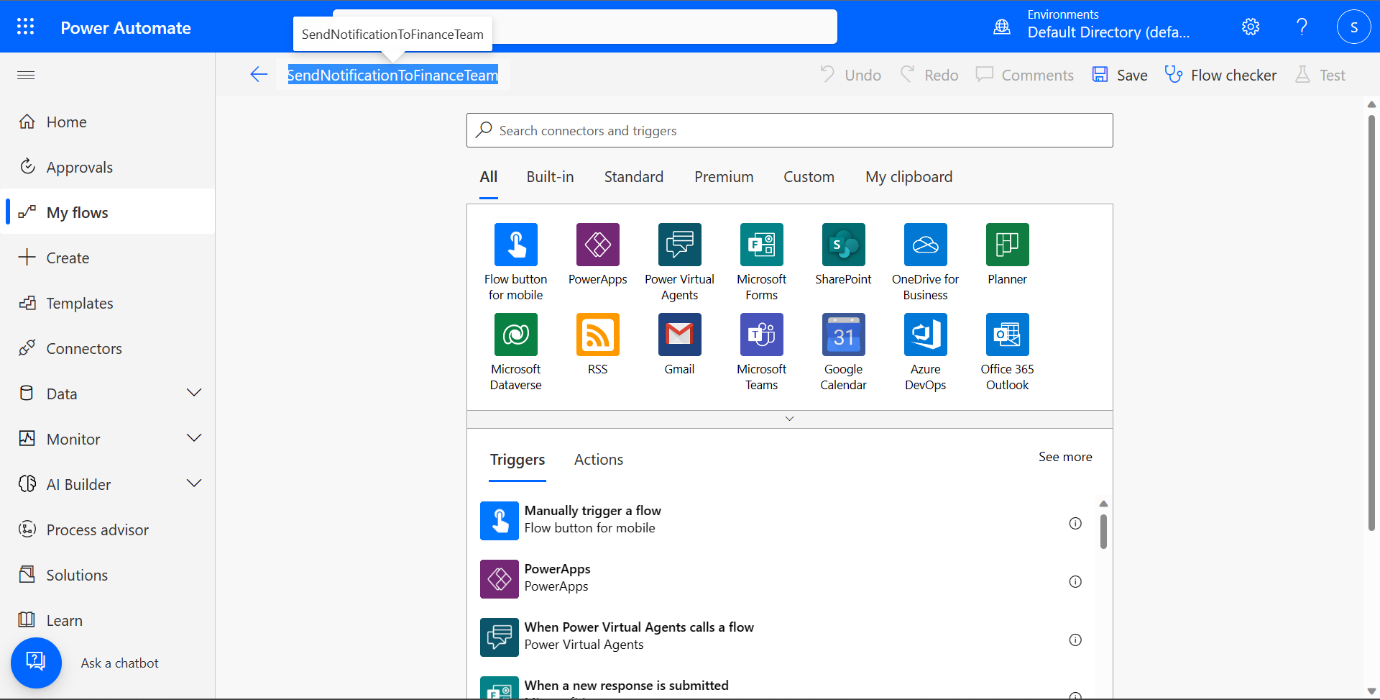
1. Create a new flow by clicking *New flow*. From the dropdown, choose *Automated cloud flow*.

[](https://github.com/CSALabsAutomation/azure-ai-ml-bpa-using-powerapps-lab/blob/main/steps/event_grid_subscription/assets/03.png)

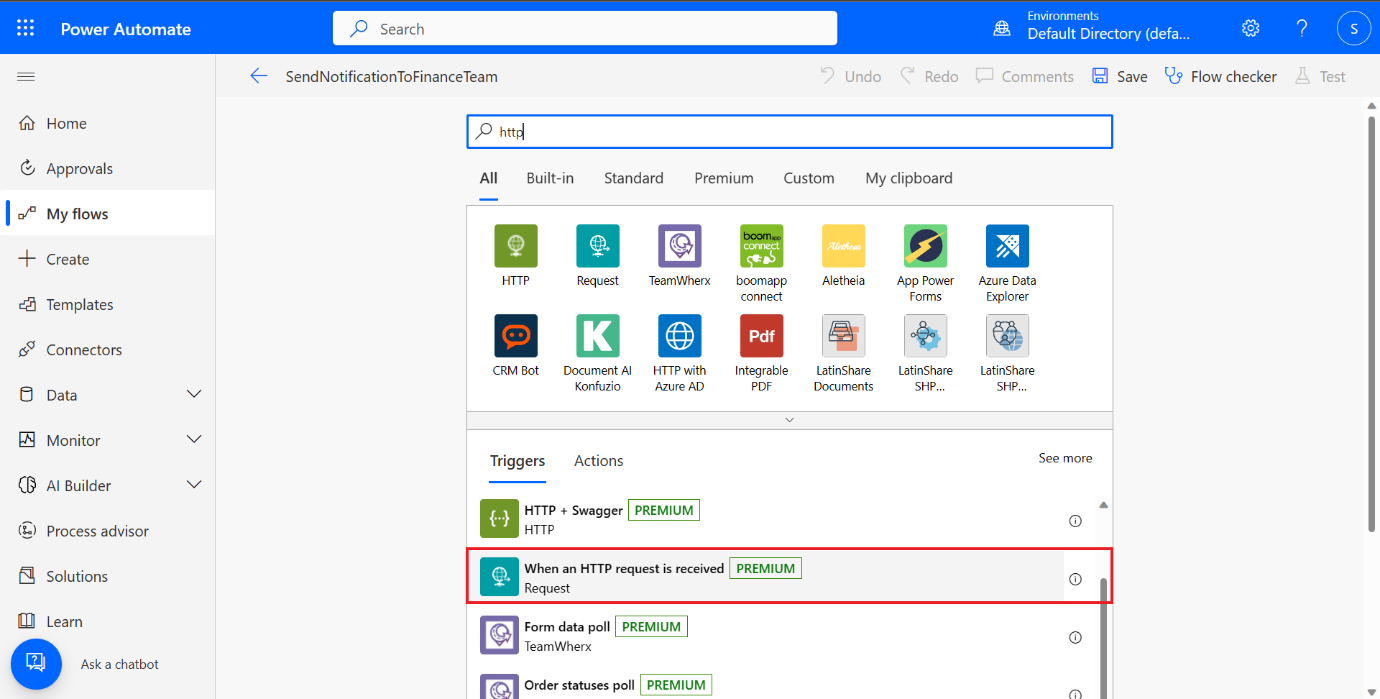
1. A dialog appears, click *Skip* which takes you to the flow editor.

[](https://github.com/CSALabsAutomation/azure-ai-ml-bpa-using-powerapps-lab/blob/main/steps/event_grid_subscription/assets/04.png)

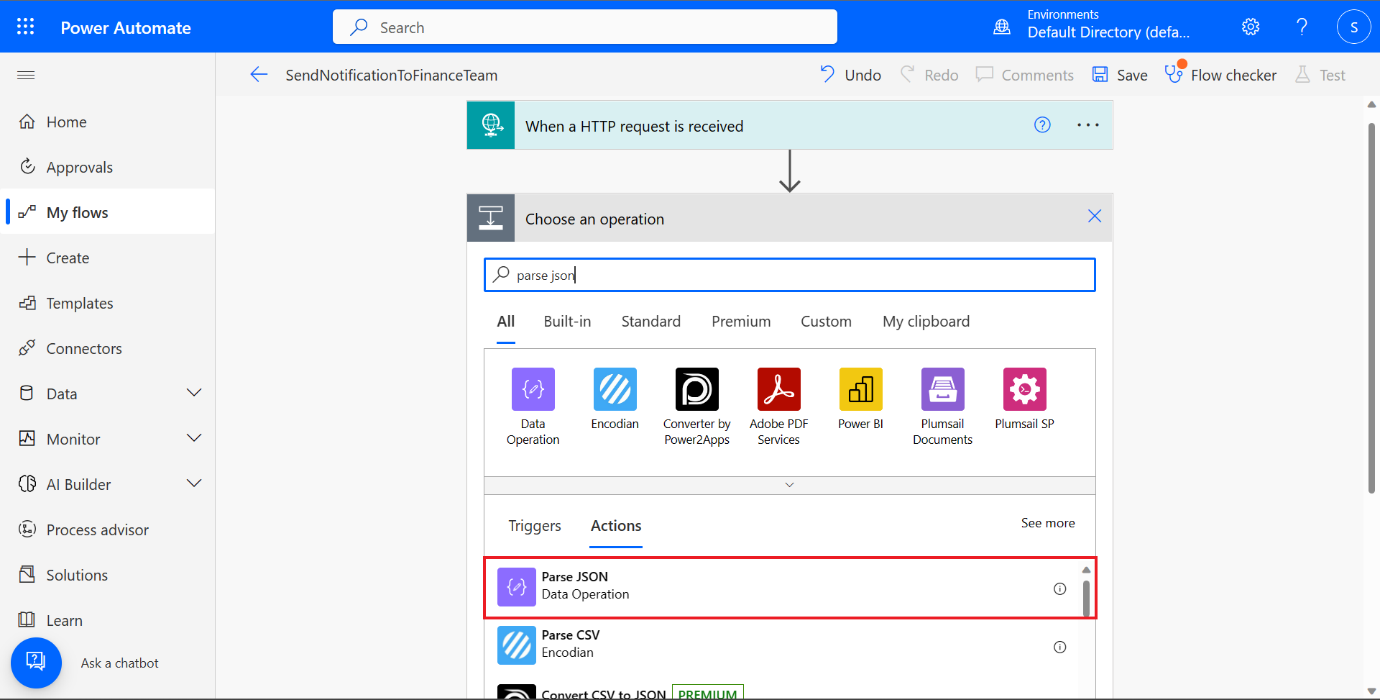
1. You can name your flow. Set *SendNotificationToFinanceTeam* as the name to your flow.

[](https://github.com/CSALabsAutomation/azure-ai-ml-bpa-using-powerapps-lab/blob/main/steps/event_grid_subscription/assets/05.png)

1. Search **http** and select *When an HTTP request is received* trigger.

[](https://github.com/CSALabsAutomation/azure-ai-ml-bpa-using-powerapps-lab/blob/main/steps/event_grid_subscription/assets/06.png)

1. Add your next step by clicking *New step*. Search **parse json** and select the *Parse JSON* action.

[](https://github.com/CSALabsAutomation/azure-ai-ml-bpa-using-powerapps-lab/blob/main/steps/event_grid_subscription/assets/07.png)

1. Set the *Content* to *Body*, which is one of the output of the previous step that you can find under the *Dynamic content* tab. **Paste the schema provided as provided below :**

{

"type": "array",

"items": {

"type": "object",

"properties": {

"id": {

"type": "string"

},

"subject": {

"type": "string"

},

"eventType": {

"type": "string"

},

"data": {

"type": "object",

"properties": {

"RaisedAmount": {

"type": "string"

},

"AppovedAmount": {

"type": "string"

}

}

},

"dataVersion": {

"type": "string"

},

"metadataVersion": {

"type": "string"

},

"eventTime": {

"type": "string"

},

"topic": {

"type": "string"

}

},

"required": [

"id",

"subject",

"eventType",

"data",

"dataVersion",

"metadataVersion",

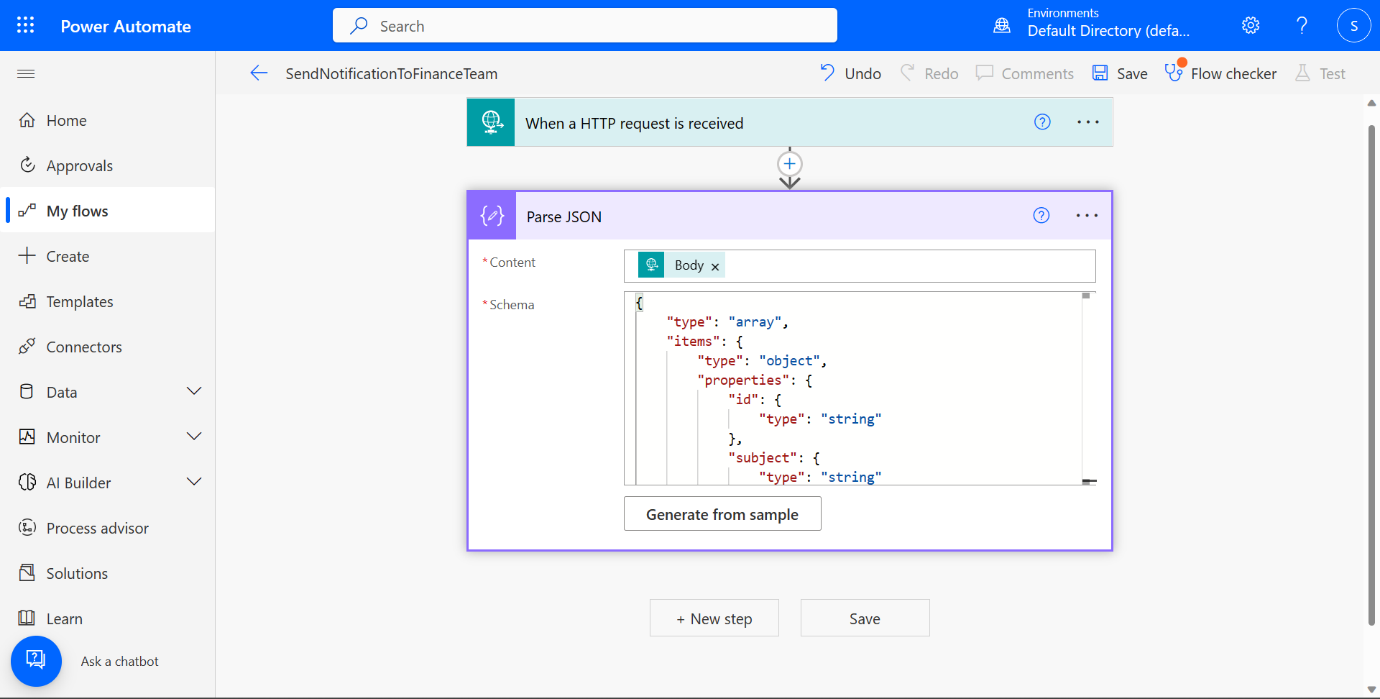
"eventTime",

"topic"

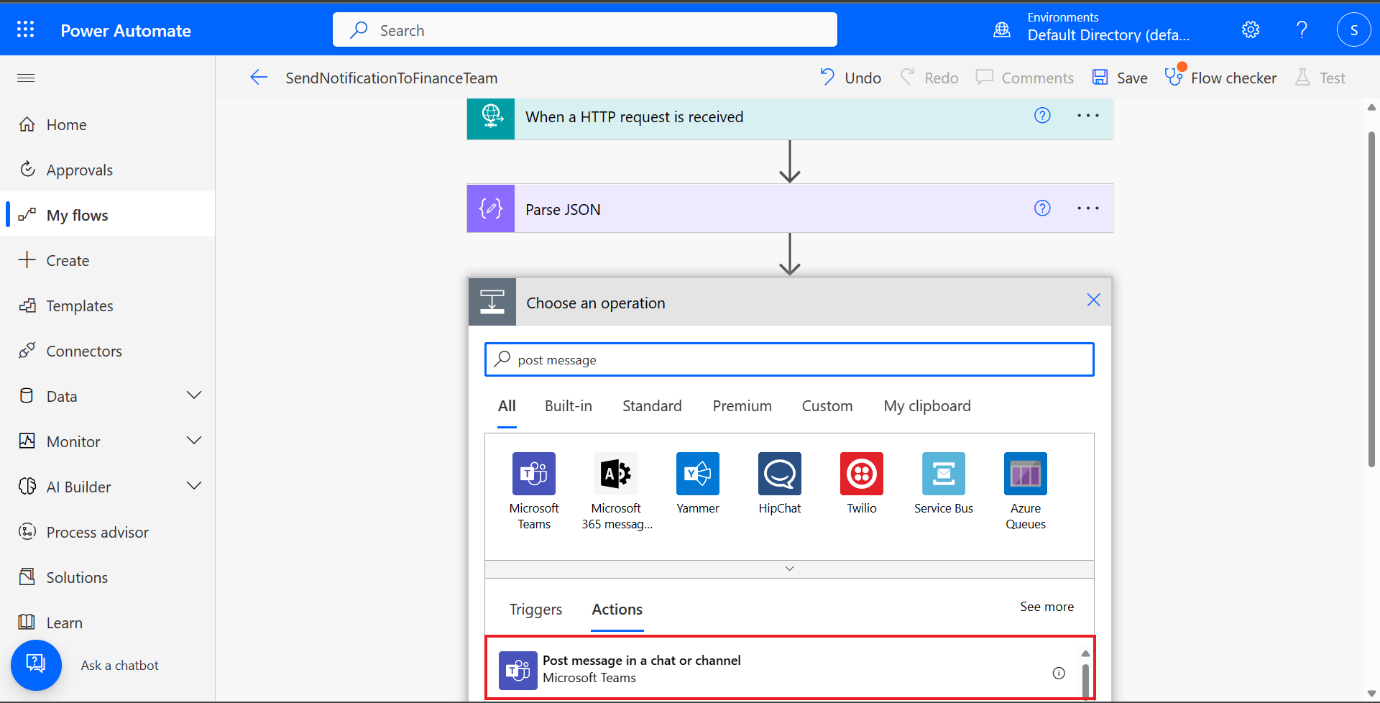
]

}

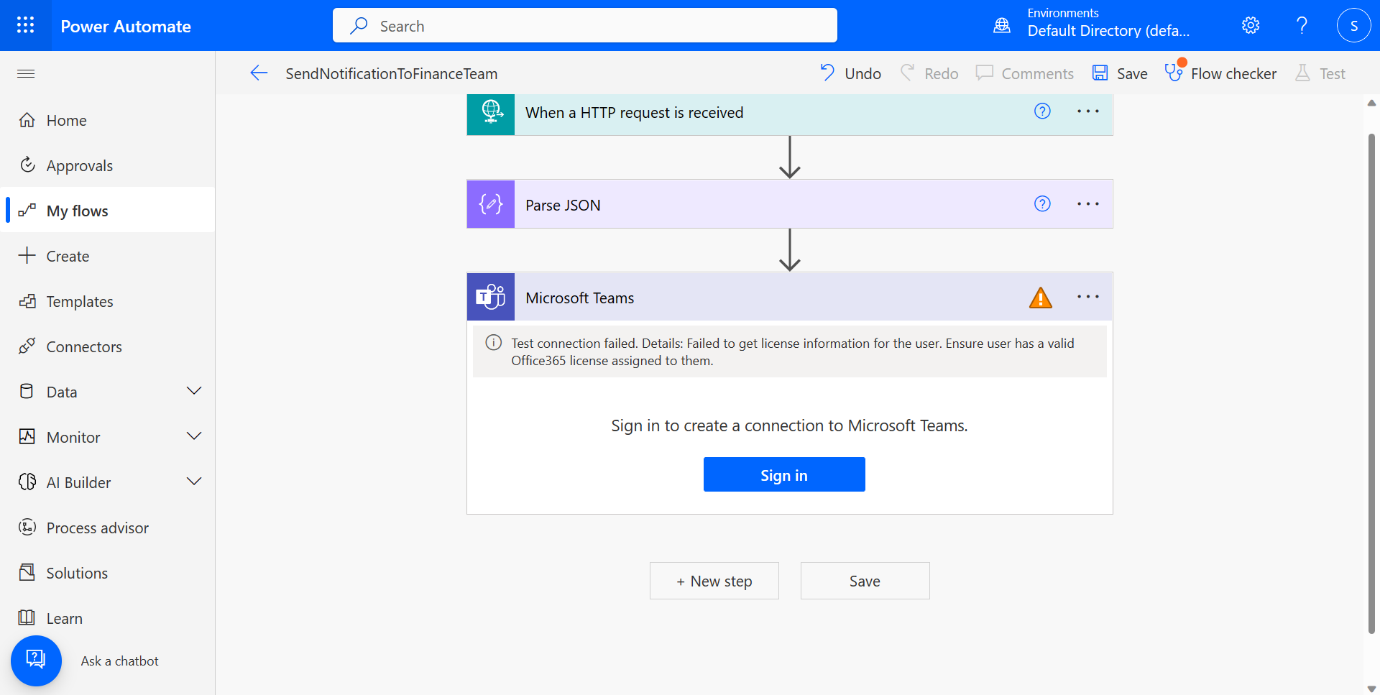
}

[](https://github.com/CSALabsAutomation/azure-ai-ml-bpa-using-powerapps-lab/blob/main/steps/event_grid_subscription/assets/08.png)

1. Add your next step by clicking *New step*. Search **post message** and select the *Post message in a chat or channel* action.

[](https://github.com/CSALabsAutomation/azure-ai-ml-bpa-using-powerapps-lab/blob/main/steps/event_grid_subscription/assets/09.png)

1. Click *Sign in* to create a connection to Teams. Use your personal/work account to sign in and send a teams message to Finance Team.

[](https://github.com/CSALabsAutomation/azure-ai-ml-bpa-using-powerapps-lab/blob/main/steps/event_grid_subscription/assets/10.png)

1. Copy the HTTP request URL

Graphical user interface, text, application, email

Description automatically generated

1. Open azure portal, goto Event Grid Topic (resource created by ARM template) and create new Event Subscription
2. Name the event subscription and use Webhook as event type.

Provide the HTTP request url in the Subscriber Endpoint.

Graphical user interface, text, application

Description automatically generated

**Run and test the App**

1. Save and Publish the app then click on Run Button
2. Goto Upload screen and upload 4 files from the Test documents using 4 Form Processing control and click on Submit Button.

**A screenshot of a computer

Description automatically generated**

1. Approve the Claim from Teams or Outlook.
2. Go to Search screen and click on search button.

A screenshot of a computer

Description automatically generated

**Power BI Report Creation**

1. Copy the Search URL from ACS

[https://bpa-acs.search.windows.net/indexes/invoicedoc/docs?api-version=2021-04-30-Preview&search=\*](https://bpa-acs.search.windows.net/indexes/invoicedoc/docs?api-version=2021-04-30-Preview&search=*)

1. Download and open Power BI Desktop
2. GetData 🡪 Web
3. Use Search URL from ACS and headers

Replace the App Key

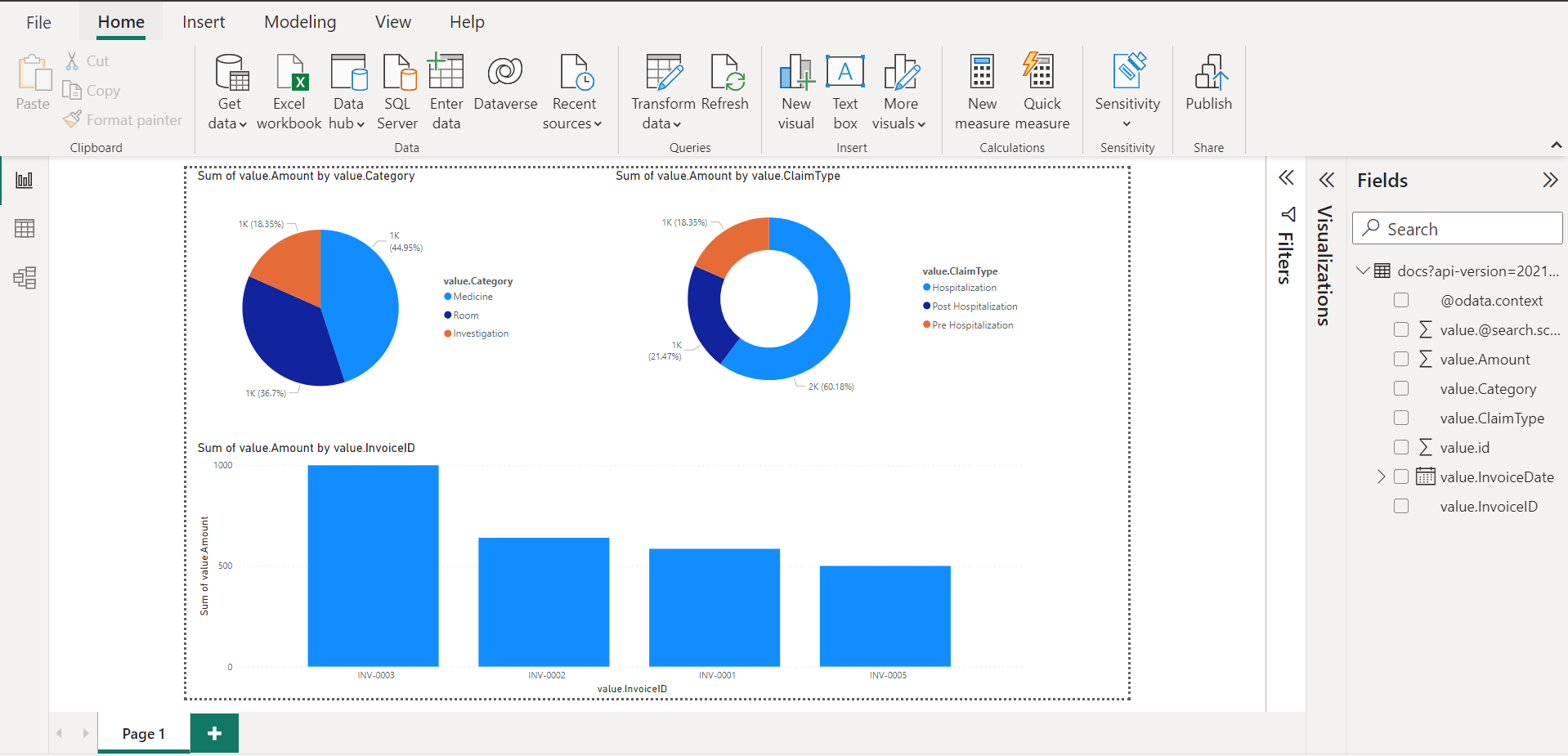
Graphical user interface, application

Description automatically generated

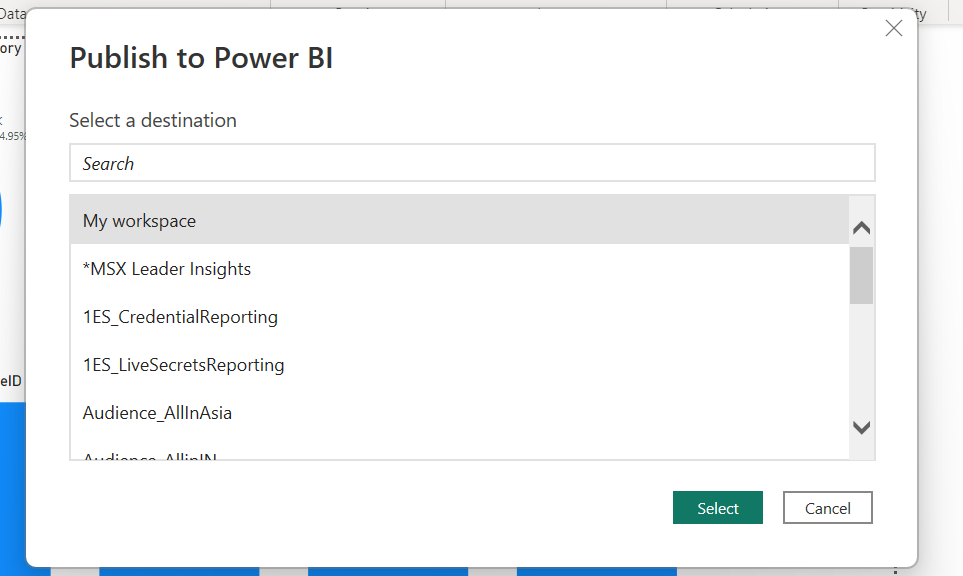
1. After Data is loaded generate below report.

Refer PBIX file

<https://github.com/CSALabsAutomation/azure-ai-ml-bpa-using-powerapps-lab/blob/main/environments/env1/Artifacts/claim.pbix>



1. Publish to Power BI Cloud

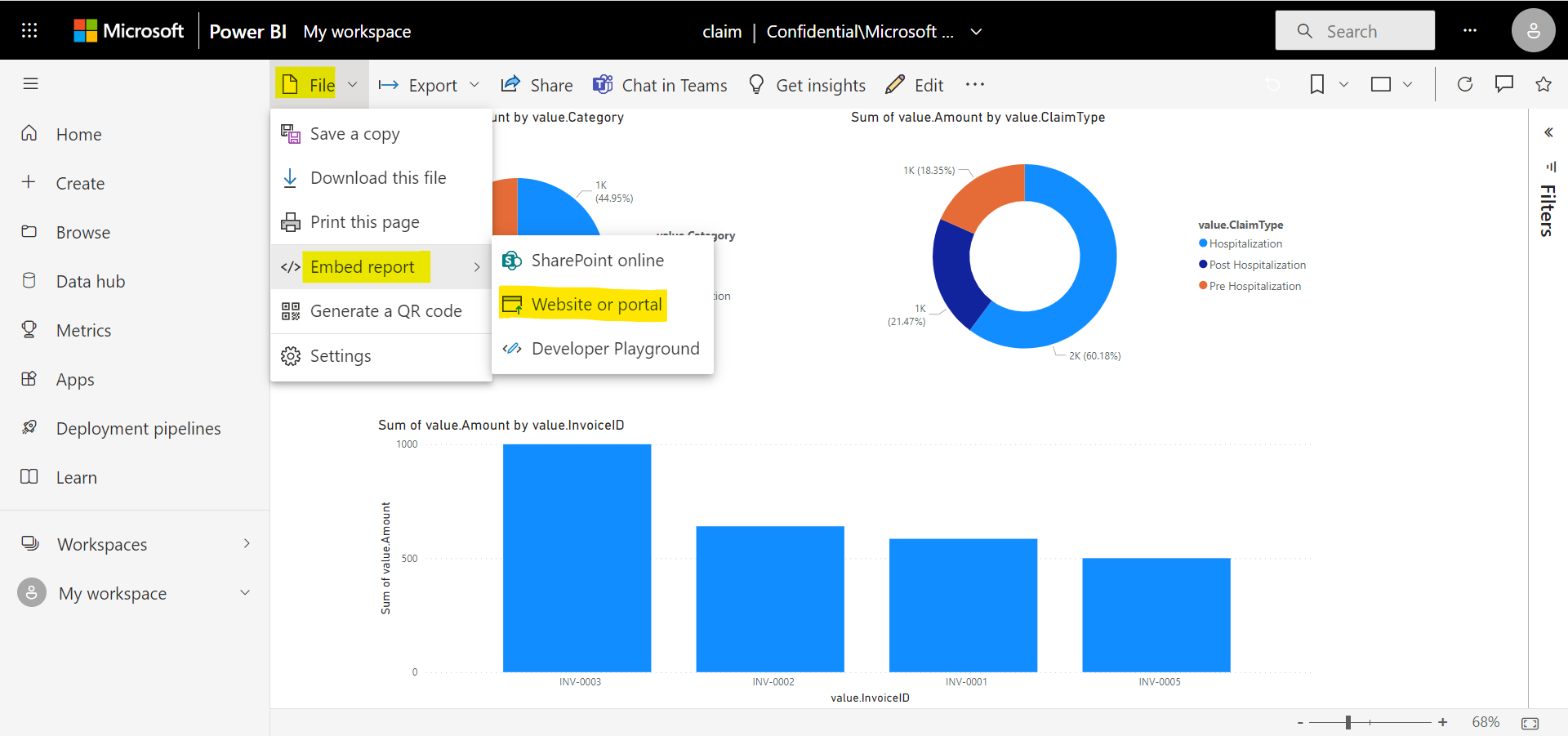


1. Click on “Open ‘claim.pbix’ in Power BI”

Graphical user interface, text, application

Description automatically generated

1. Goto -> File-> Embed Report -> Website or Portal



1. Copy the link to embed the content

Graphical user interface, text, application

Description automatically generated

**Power BI Report Integration in Power Apps**

1. Open Power Apps portal and Report Screen
2. Insert Power BI Tile

**Graphical user interface, application, Word

Description automatically generated**

1. Select Power BI Tile and update the “TileUrl” with the embed URL copied above from the Power BI

Graphical user interface, application

Description automatically generated

1. Save and Publish the App

**Pending Tasks**

1. Validations
2. Instructions