Leveraging advanced technologies like Microsoft Power Platform can significantly enhance your supply chain operations. In this blog, we will explore how to create a Copilot for supply chain management that identifies the best vendor based on vendor details stored in Dataverse. This Copilot will utilize parameters like lead time, quality score, and unit price, and employ AI prompts to make intelligent vendor recommendations.

**Overview**

Our supply chain Copilot will:

1. **Fetch Vendor Details from Dataverse**: Retrieve data for multiple vendors, each providing the same product with different parameters.
2. **Use AI to Identify the Best Vendor**: Implement AI-powered analysis to find the most suitable vendor based on the retrieved data and parameters like lead time, quality score, and unit price.

**Step-by-Step Guide to Building the Supply Chain Copilot**

**Step 1: Set Up Dataverse for Vendor Data**

To begin, we need to organize our vendor details in Dataverse. Create a table named **“Vendor Details”** with the following columns:

* **Material Name**: The name of the material or product (e.g., Aluminium Sheets, Copper Wires).
* **Vendor Name**: The name of the vendor supplying the product.
* **Unit Price (Dollars)**: The cost per unit of the product.
* **Quality Score**: A score representing the quality of the product (typically on a scale of 1-100).
* **Lead Time (Days)**: The time taken by the vendor to deliver the product.

**Step 2: Populate the Dataverse Table**

Enter the vendor details into the Dataverse table. We have the below data for the demo

1.pngA screenshot of a computer

Description automatically generated

Step 3: Creating the Copilot in Copilot Studio

Head over to <https://copilotstudio.microsoft.com/> and click on “Create”

1\_3.pngA screenshot of a computer

Description automatically generated

This will provide us the option to create a copilot based on an existing template on create a blank copilot from scratch, Lets select “New copilot”

1\_4.pngA screenshot of a computer

Description automatically generated

This will take us to the page where we can

1. Describe the copilot functionality and provide any specific instructions to the copilot.
2. Once done, click on “Create” to provision the copilot.

2.pngA screenshot of a computer

Description automatically generated

## Step 4: Enable Generative selection of topics

The copilot is now created. We can then make the needed configuration changes

1. Click on “Edit”, edit the copilot details like name, icon and description.
2. Click on “Settings” to enable the Generative selection of topics so that without relying on triggers, the topics will be auto selected based on user conversation resulting in a much smoother user experience

3.pngA screenshot of a computer

Description automatically generated

To enable the automatic detection of topics from user interaction:

1. Click on Generative AI
2. Select “Generative(preview)”
3. Click on “Save” to update the settings
4. Click on Close icon to go back to the home page of this custom copilot

4\_1.pngA screenshot of a computer

Description automatically generated

Step 5 : Create Topics

Now let’s go ahead and create the topics that will automatically redirect the conversation flow to appropriate topics based on the question user posts.

1. Click on “Topics” from the navigation menu.

To add the topic, we can either go with the option to create a blank topic or use Copilot to create the topic with initial set of prepopulated conversation nodes based on the topic description that we provide.

1. Let’s Click on “Add a Topic” and
2. Select “Create from description with Copilot”.

4.pngA screenshot of a computer

Description automatically generated

Let’s provide the below topic description details in the pop up that opened when we clicked the Add topic button previously.

Then, Click on Create, which will provision the topic skeleton based on the provided description.

5.pngA screenshot of a computer

Description automatically generated

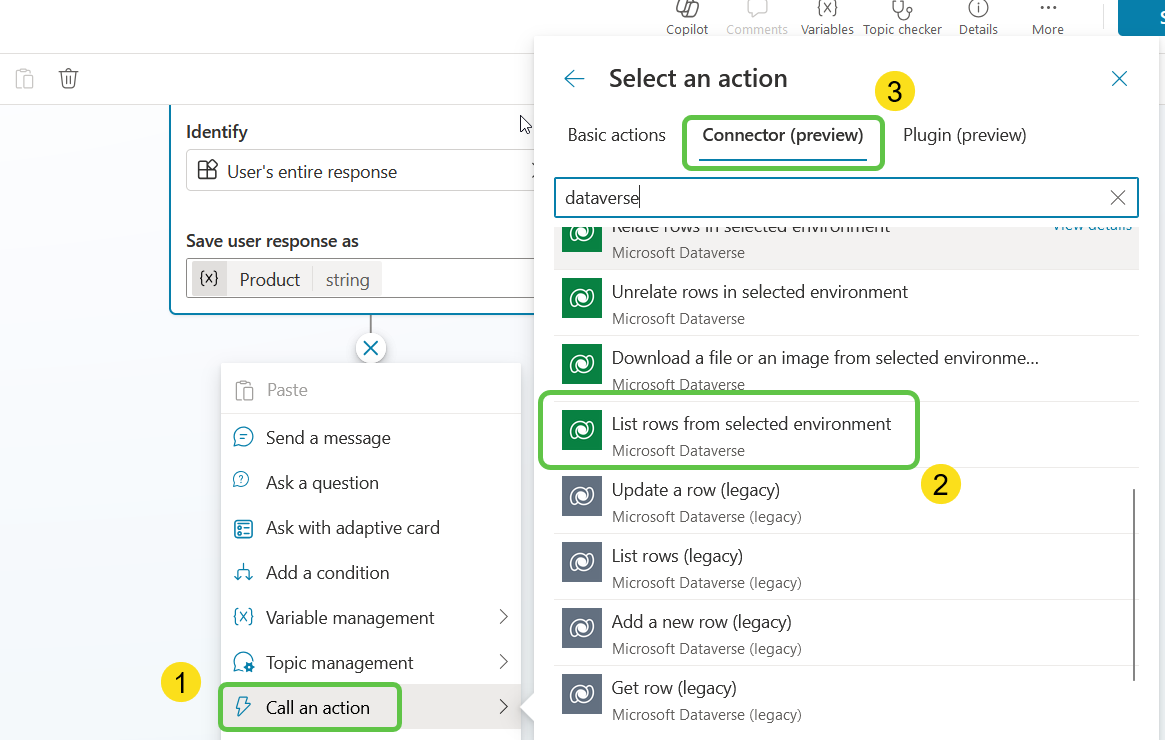
Thus we have the basic topic created with an automatic trigger as well as a question to the user which are generated using the description provided. We can now add more conversation nodes.

6.pngA screenshot of a computer

Description automatically generated

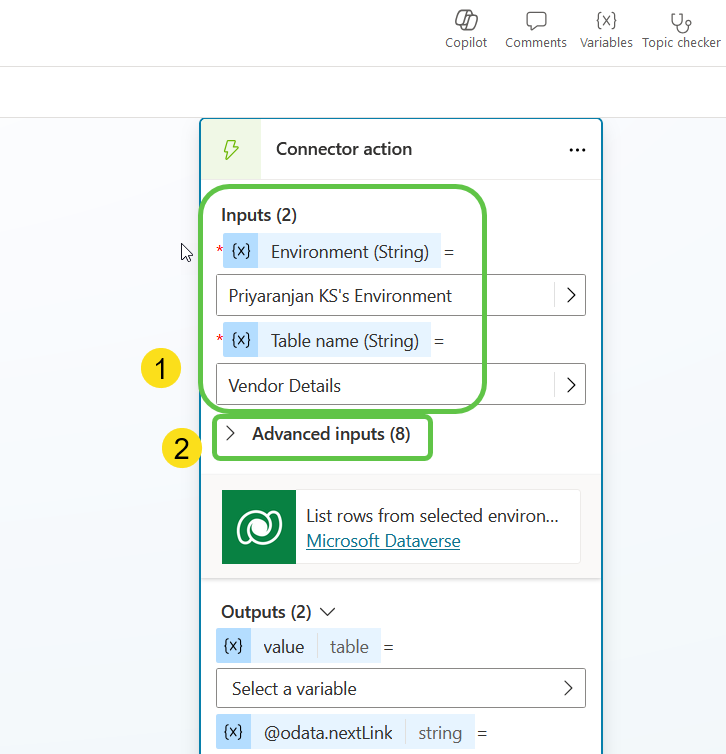
We now need to fetch the vendor details from the Dataverse table for which we will add the Dataverse connector action by :

1. Select “Call an action”
2. Select the “Connector” tab,
3. Click on “List rows from selected environment”

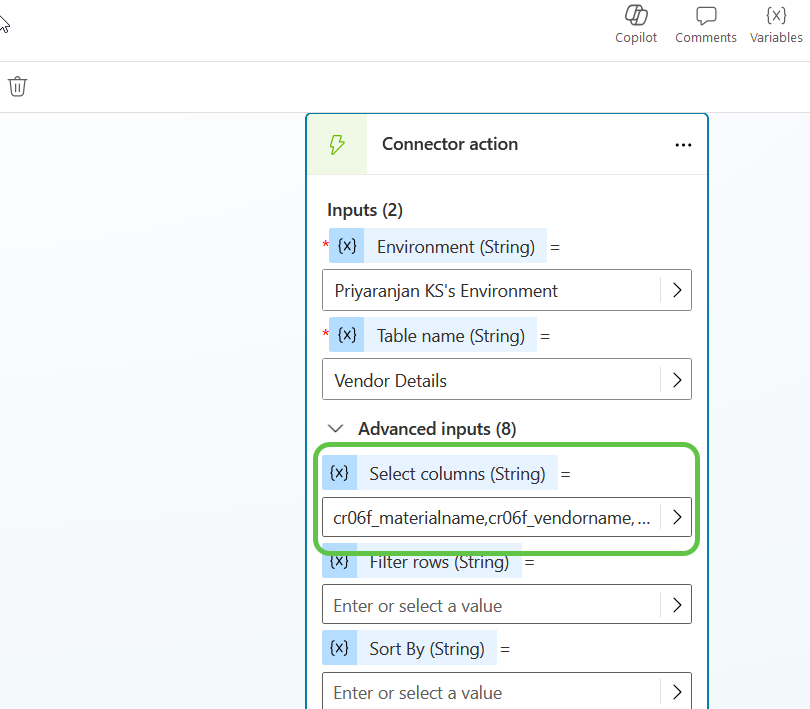
36.png

This will add the dataverse connector action to the Copilot canvas and lets configure the inputs for this action by :

1. Mentioning the Environment and Table name where Vendor Details Table is present
2. Click on Advanced inputs to filter the data that is returned from the table

37.png

Specify the dataverse table columns which should be present in the returned output. We will specify the Material name, Vendor name, Leadtime, Unit Price and Quality Score columns

38.png

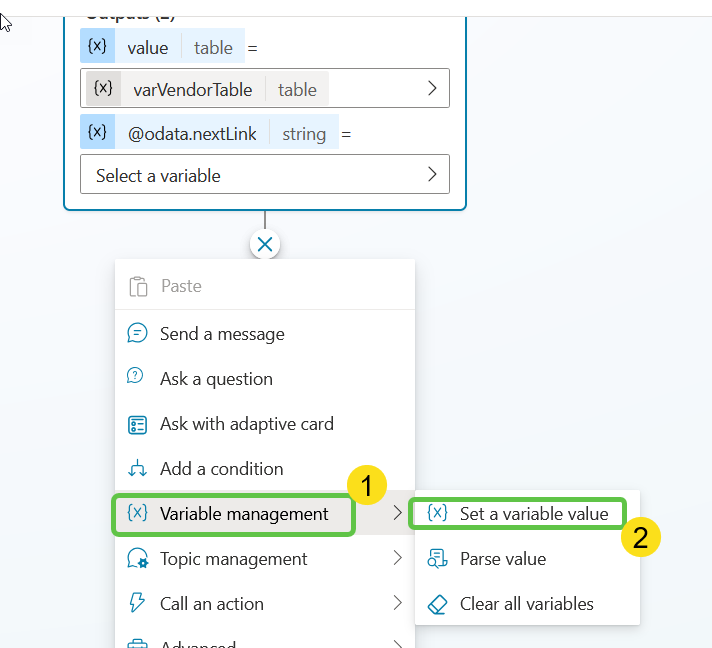
Note : You can get the logical name of the columns used in the “Select columns” field from the dataverse table by following the below path :

10.pngA screenshot of a computer

Description automatically generated

The output of the Dataverse Table Connector(varVendorTable) will contain lots of system columns as well . We will need to format the table to filter and ensure only the needed columns are present.

To do this let’s initialize a variable to hold the output of Dataverse connector

39.png

We will then add the below formula which will filter the output to create a subset of the table and store it in the variable. It does this by looping through the previous Dataverse connector output and fetching only the columns that we have mentioned in the expression

```

ForAll(

    Topic.varVendorTable,

    {

        MaterialName: ThisRecord.cr06f\_materialname,

        VendorName: ThisRecord.cr06f\_vendorname,

        LeadTime: ThisRecord.cr06f\_leadtimedays,

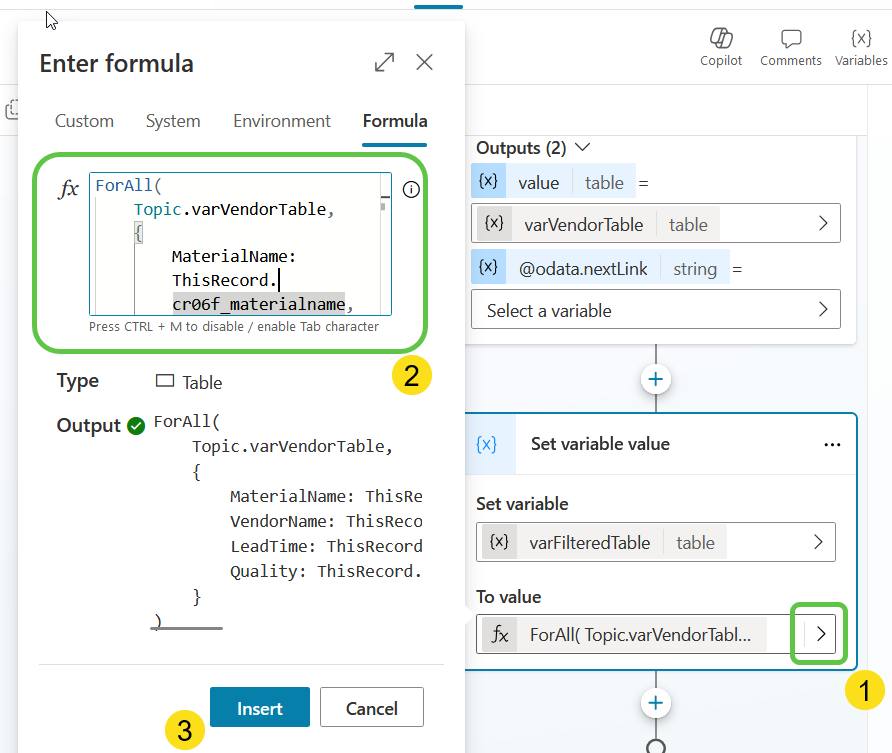
        Quality: ThisRecord.cr06f\_qualityscore,UnitPrice:ThisRecord.cr06f\_unitpricedollars

    }

)

```

40.png



If we were to test and output this variable value in the test pane, we will get the output as :

```json

[{“LeadTime”:“5”,“MaterialName”:“Aluminum Sheets”,“Quality”:“92”,“UnitPrice”:“35”,“VendorName”:“PureAluminum”},

{“LeadTime”:“6”,“MaterialName”:“Aluminum Sheets”,“Quality”:“90”,“UnitPrice”:“37”,“VendorName”:“MetalSheetsCo”},

{“LeadTime”:“7”,“MaterialName”:“Aluminum Sheets”,“Quality”:“91”,“UnitPrice”:“33”,“VendorName”:“HighGradeAluminum”},

{“LeadTime”:“10”,“MaterialName”:“Copper Wires”,“Quality”:“93”,“UnitPrice”:“60”,“VendorName”:“ElectroCopper”},

{“LeadTime”:“9”,“MaterialName”:“Copper Wires”,“Quality”:“92”,“UnitPrice”:“62”,“VendorName”:“PureCopperSupplies”}

```

We will do one more formatting of this output to serialize this json into a readable format <Material Name>: <Vendor Name>:<Lead Time Days>:<Quality Score>:<Unit Price Dollars>

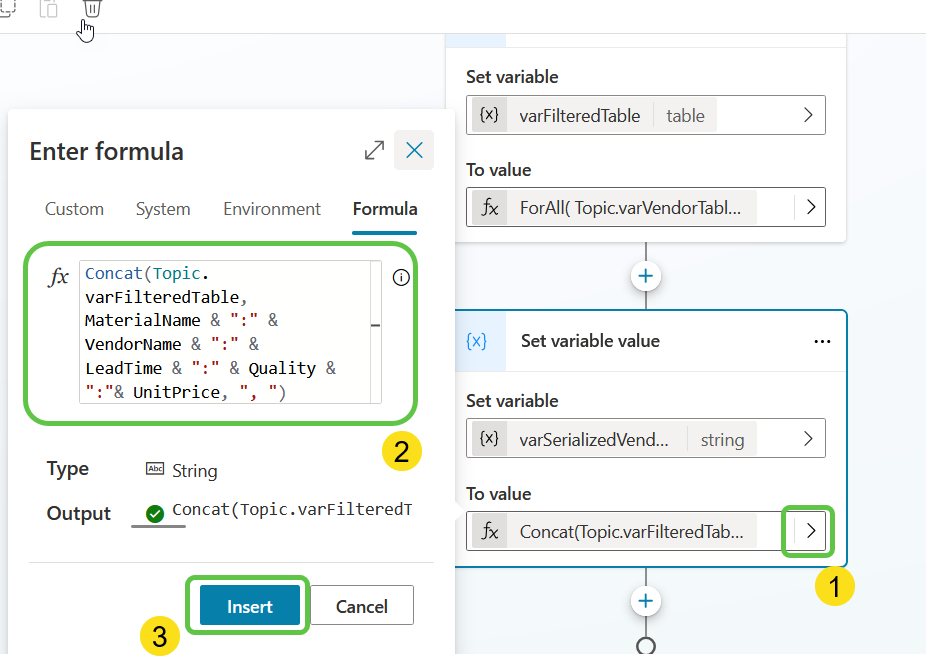
To do this, lets add another variable and set its formula to

```

Concat(Topic.varFormattedTable, MaterialName & ":" & VendorName & ":" & LeadTime & ":" & Quality & ":"& UnitPrice, ", ")

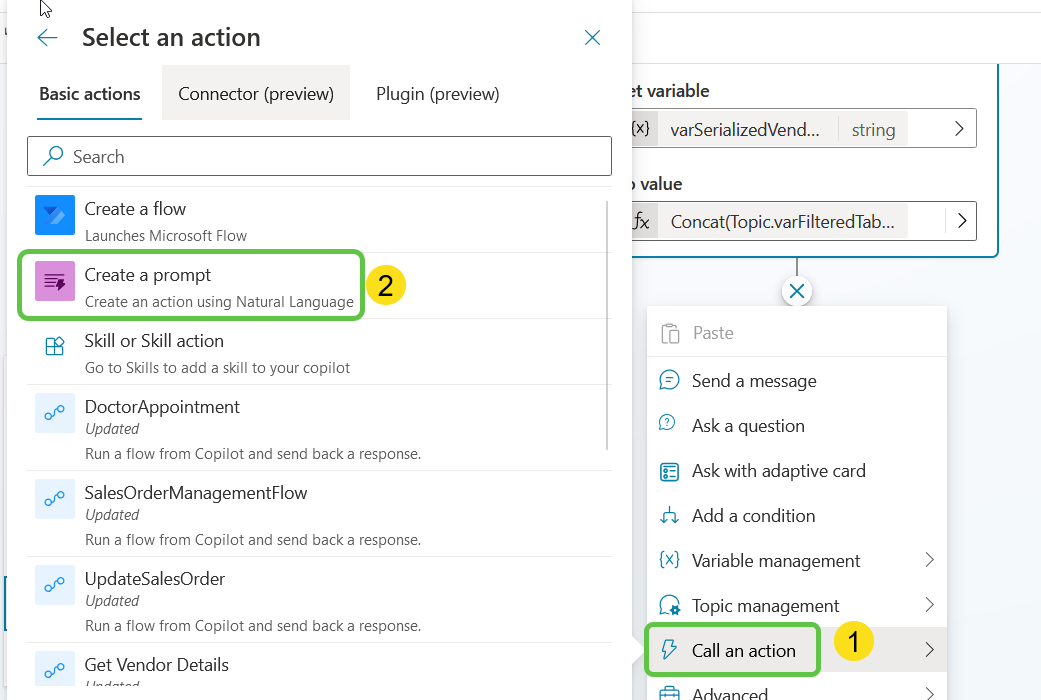
```

This way, from the previous filtered table, we will concat the Material Name and details in the readable format which can be shared as an input to the AI Prompt



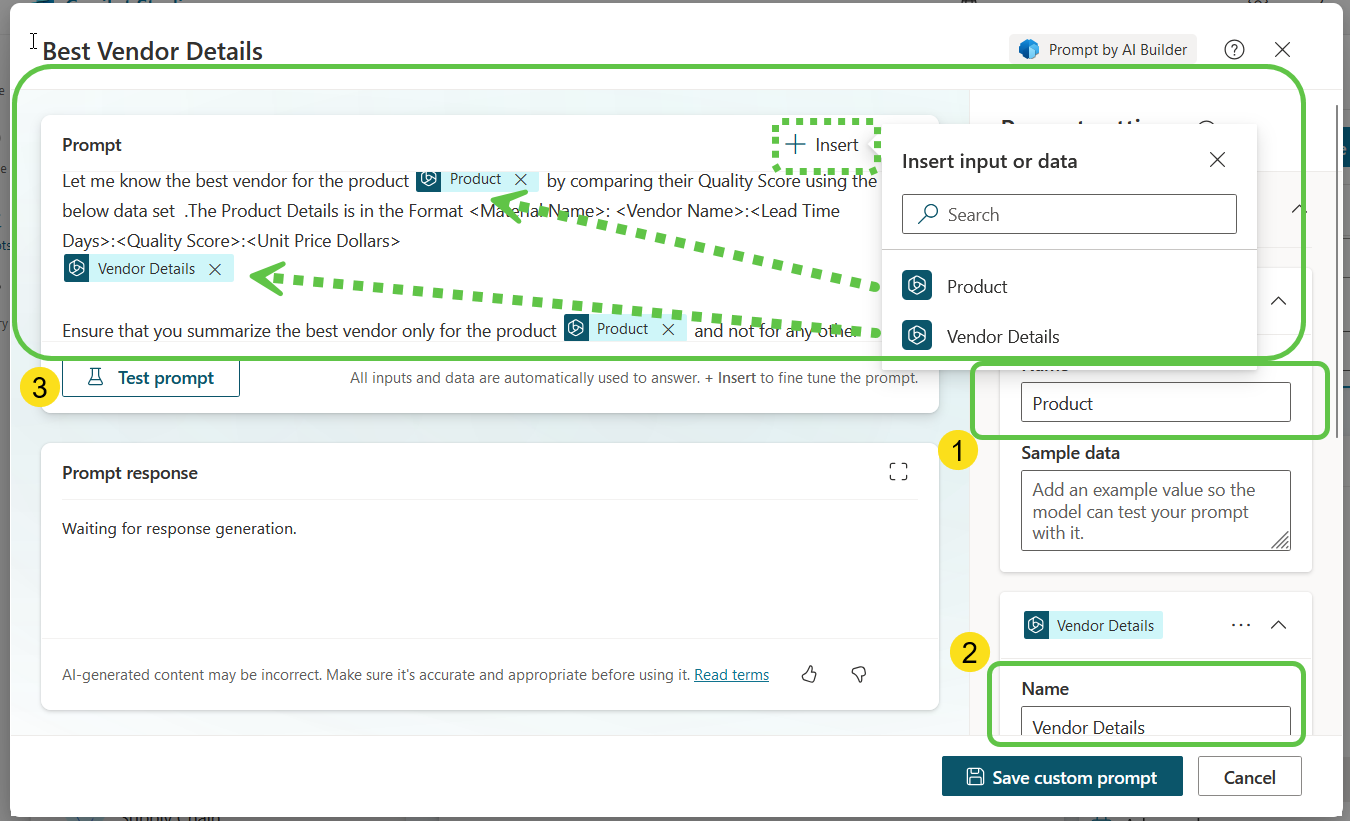
Now lets add the AI prompt using which we can find the best vendor for the product specified by the user. To do this :

1. Select “Call an action” and
2. From “Basic Actions”, select “Create a prompt”

42.png

This will open up the Prompt AI Builder in a pop up where we can :

1. Create the Product Variable to which we will pass the user inputted product name for which he is finding the best vendor
2. Create the Vendor Details variable to which we will pass the Dataverse connector returned vendor details
3. In the Prompt section use the below prompt and add the variables dynamically using the “Insert” button which will allow us to add the Product and Vendor Details input variables into the prompt
4. Click on “Save custom prompt” which will make the prompt available in the copilot designer.



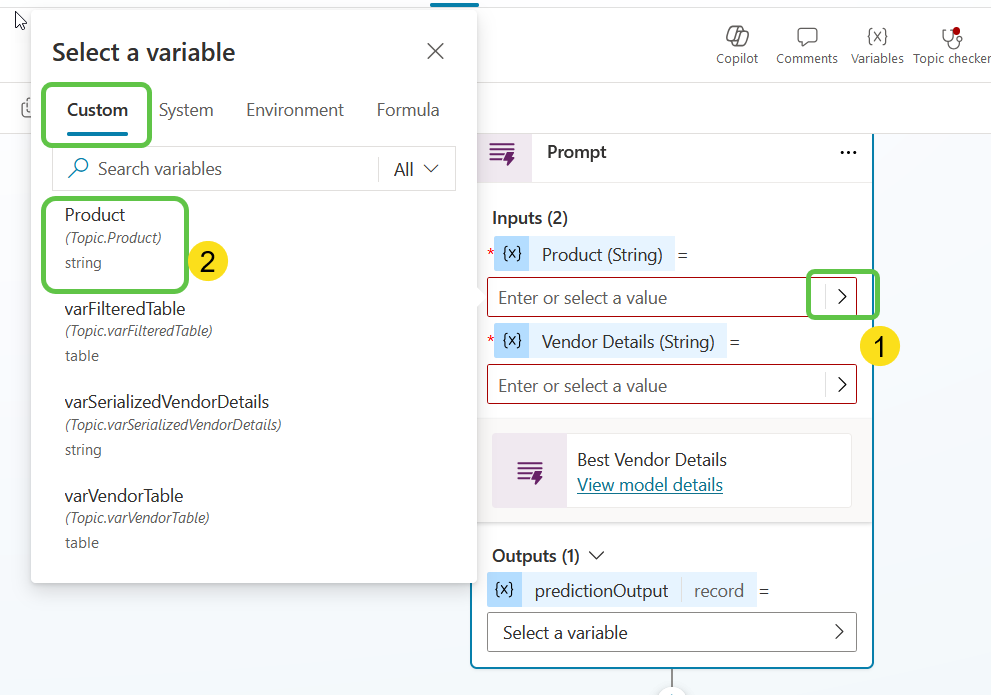
43.png

We will add the recently created prompt by

1. Selecting Call an action
2. From the Basic actions tab, select the “Best Vendor Details” prompt

43\_1.png

The prompt is now available, lets configure it by adding the first input parameter which is the user input Product Name

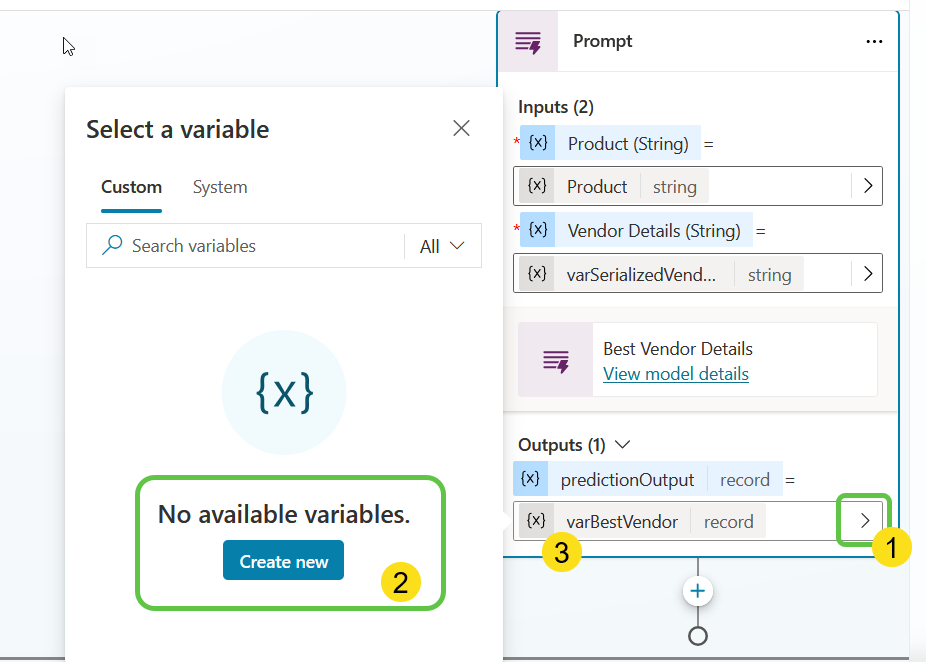
44.png 

Next, we will add the Dataverse returned output data which we had formatted to the Vendor Details field

45.pngA screenshot of a computer

Description automatically generated

Finally, we need to store the output record generated by the Prompt action in a variable for which we will create a new variable by the name varBestVendor

46.png 

Thus we have configured the Prompt action with the required input and it should ideally fetch us back the best vendor for the passed in product based on the Quality Score parameter.

Finally, we will show the best vendor output as a basic card back to the user for which we will select the “text” Property of the varBestVendor record which will contain the generative ai output.

47.pngA screenshot of a computer

Description automatically generated

Publish the Copilot to Team

We will finally publish the copilot and add it to the Teams Channel for which we we will enable the teams channel by selecting the Channels Tab -> Microsoft Teams -> Turn On Teams

28.pngA screenshot of a computer

Description automatically generated

Click on the Availability Options to select how to distribute the copilot within Teams

29.pngA screenshot of a computer

Description automatically generated

We have the option to publish the app to the app store which goes through an admin approval or we can download the zip file and upload as a personal app . For this demo, we will go publishing the app org wide.

30.pngA screenshot of a computer

Description automatically generated

Click on Submit for Admin approval and the admin can approve this from the teams admin center.

31.pngA screenshot of a computer

Description automatically generated

We can see that the copilot has become available for publishing in the teams admin center. Click on Publish

32.pngA screenshot of a computer

Description automatically generated

Lets now add the copilot to Teams so that we can interact and get results with it.

33.pngA screenshot of a computer

Description automatically generated

Thus we have initiated the conversation with the copilot and tried to find the best vendor for Copper Wires.

If we filter the Dataverse table for Copper Wire Vendors, we could see 3 records but AI prompt that we used have refined the results by comparing the Unit Price and Quality score to arrive at the best vendor

35.png

A screenshot of a computer

Description automatically generated

We can see that the copilot has picked the vendor details from Dataverse and using the AI Prompt, fetched the generative contextual answer for the best vendor by comparing the unit price and quality score

34.pngA screenshot of a chat

Description automatically generated

# Conclusion

In this blog, we've explored how to build a supply chain Copilot using Microsoft Power Platform and Dataverse to streamline the vendor selection process. By leveraging Dataverse for structured storage of vendor details and employing AI prompts for intelligent analysis, our Copilot, provides an efficient solution for identifying the best vendor based on critical parameters like lead time, quality score, and unit price.