



Dataverse for Teams in a Day

Lab 03

Power Virtual Agents for Teams

Contents

Contents.....	2
Overview	4
Exercise 1: Create a chatbot in Teams	5
Task 1: Discover and add the Power Virtual Agents app to Teams.....	5
Task 2: Create a new chatbot.....	7
Task 3: Navigating between other parts of Teams and your existing chatbot	11
Exercise 2: Personalize the greeting	13
Task 1: Edit and personalize the greeting topic.....	13
Exercise 3: Create a topic.....	17
Task 1: Switch off the lesson topics.....	17
Task 2: Create a new topic for questions about home office setup.....	18
Task 3: Test your chatbot	28
Exercise 4: Create a flow using the embedded flow experience	31
Task 1: Get the schema name for your Dataverse for Teams table.....	31
Task 2: Create a new Topic	33
Task 3: Create a flow to get a row from a Dataverse for Teams table	36
Task 4: Connect your flow with outputs to your chatbot.....	44
Task 5: Create a flow to submit a request for an asset.....	45
Task 6: Connect your flow with inputs to your chatbot.....	53
Task 7: Test your bot	56
Exercise 5: Publishing and sharing your chatbot	58
Task 1: Publish your bot and open it in Teams	58
Task 2: Share your chatbot with other members of your team	60
Task 3: Share your chatbot with your organization.....	63

Task 4: Approve the chatbot as admin.....	67
Summary.....	70
Lab Survey	70
Terms of Use	71

Overview

The estimated time to complete this lab is 60 minutes.

Power Virtual Agents empowers subject matter experts to build intelligent conversational bots, using a guided, no-code graphical interface. Leveraging Power Automate, you can call flows from your topics, connecting your bot to other data sources, so that it can take action on behalf of the user.

Power Virtual Agents is available as an app in Microsoft Teams, enabling you to create an internal chatbot. This is an ideal solution if you want to create a chatbot to answer common internal questions posed by employees, particularly for areas like IT and human resources where there are a lot of frequently asked questions. (Note that if you want to create a chatbot for customers, to deploy on your website or other channels, you can use the full version of Power Virtual Agents.)

In this lab, we will be adding the Power Virtual Agents app to Microsoft Teams, and creating a chatbot for the employees of Contoso to get information about frequently asked questions for home office setup, and to use the bot to request home office assets. The chatbot will be able to submit an asset request on behalf of the user, and you will see the request in the same Dataverse for Teams table that you created in Lab 01.

- **Exercise 1: Create a chatbot in Teams**

Explore the Power Virtual Agents app for Teams and create a bot.

- **Exercise 2: Personalize the greeting**

Edit the greeting topic to configure how the bot introduces itself to a user.

- **Exercise 3: Create a topic**

Create a topic for the bot to answer common questions from employees about setting up their home office.

- **Exercise 4: Create a flow using the embedded flow experience**

Set up the chatbot to interact with the Dataverse for Teams tables created in Lab 01 so that the user can submit a request for home office equipment using the chatbot. To do this we will be using the embedded flow experience inside the Power Virtual Agents app in Teams.

- **Exercise 5: Publishing and sharing your chatbot**

Publish the bot so that it can be used by yourself, others in your Team and the organization.

Exercise 1: Create a chatbot in Teams

Task 1: Discover and add the Power Virtual Agents app to Teams

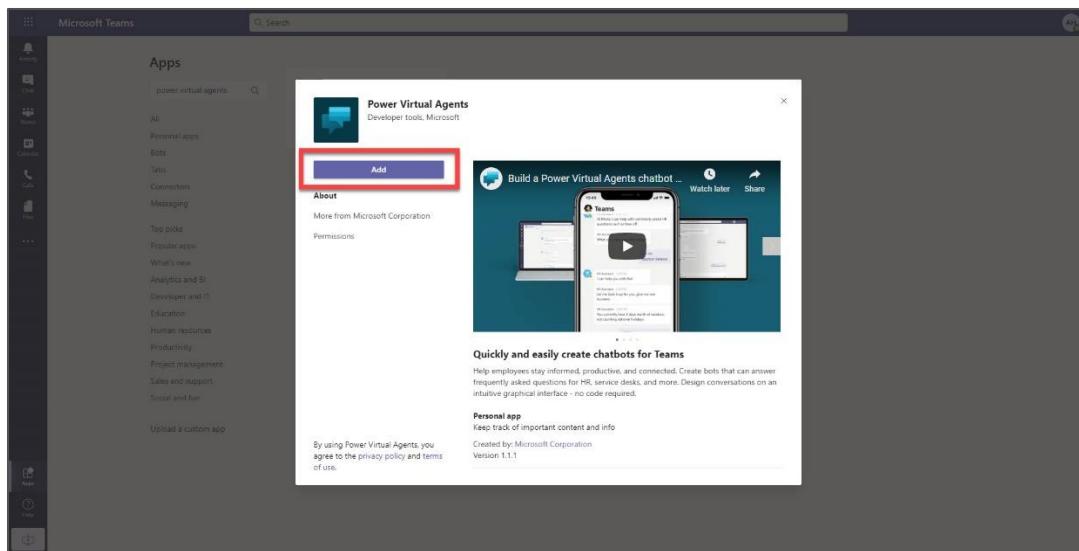
1. Click on the **Apps** icon on the left toolbar, and then search for Power Virtual Agents.

The screenshot shows the Microsoft Teams 'Apps' page. On the left, there's a vertical sidebar with icons for Activity, Chat, Home, Calendar, Calls, Files, and Apps (which is highlighted with a red box). The main area has a search bar at the top. Below it, there's a section titled 'Get more done with apps!' featuring logos for Eloops, Zeplin, SurveyMonkey, and TeamViewer. The 'All' filter is selected in the search dropdown. The results are divided into sections: 'What's new' and 'All apps'. In the 'All apps' section, the 'Power Virtual Agents' app by Microsoft Corporation is visible, showing its icon, name, and a brief description: 'Help employees stay informed, productive, and connected. Create bots that can answer frequently asked questions for HR, service...'. Other apps listed include ServiceDesk Plus Cloud, Workstreams.ai, Ment.io, Zoho Projects, Forms, Polly, YouTube, Jira Cloud, Azure Boards, Power BI, Trello, Flow, Communities, Testportal, Calendar Pro, and Who.

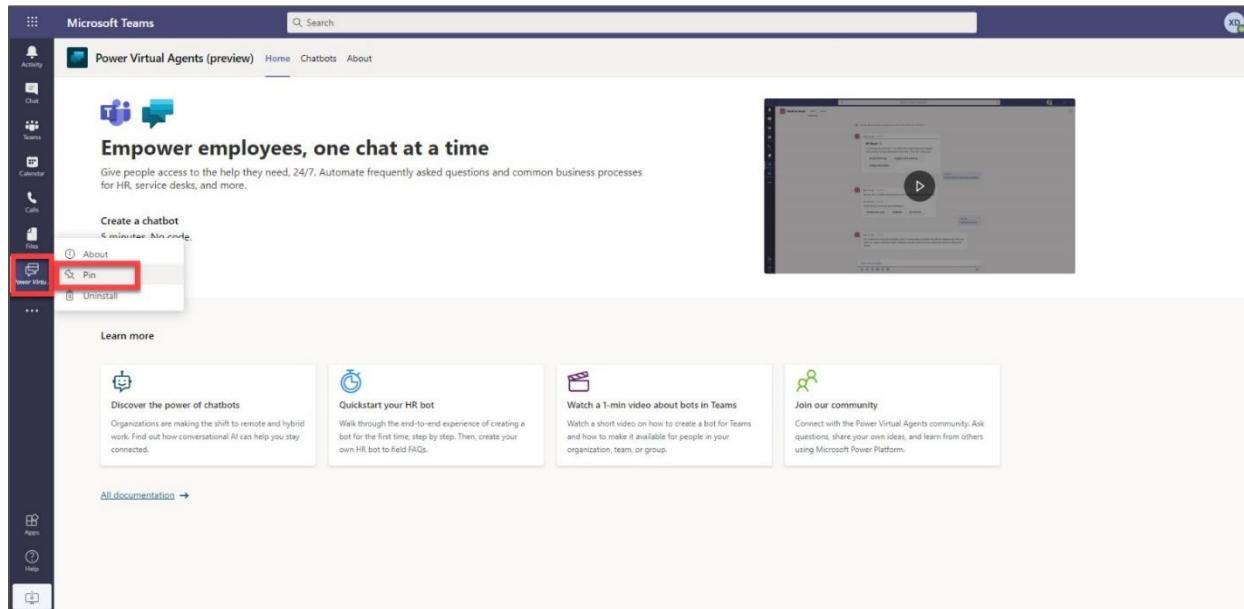
2. Click on the Power Virtual Agents App when you find it in the search results.

This screenshot shows the Microsoft Teams 'Apps' page again, but this time the search bar contains 'power virtual agents'. The results list the 'Power Virtual Agents' app by Microsoft Corporation, which is highlighted with a red box. The app's description is partially visible: 'Help employees stay informed, productive, and connected. Create bots that can answer frequently asked questions for HR, service...'. The rest of the page structure is identical to the first screenshot, with the 'Apps' icon in the sidebar also highlighted with a red box.

3. You will see a pop up with information about the Power Virtual Agents App for Microsoft Teams. Click on the **Add** button.

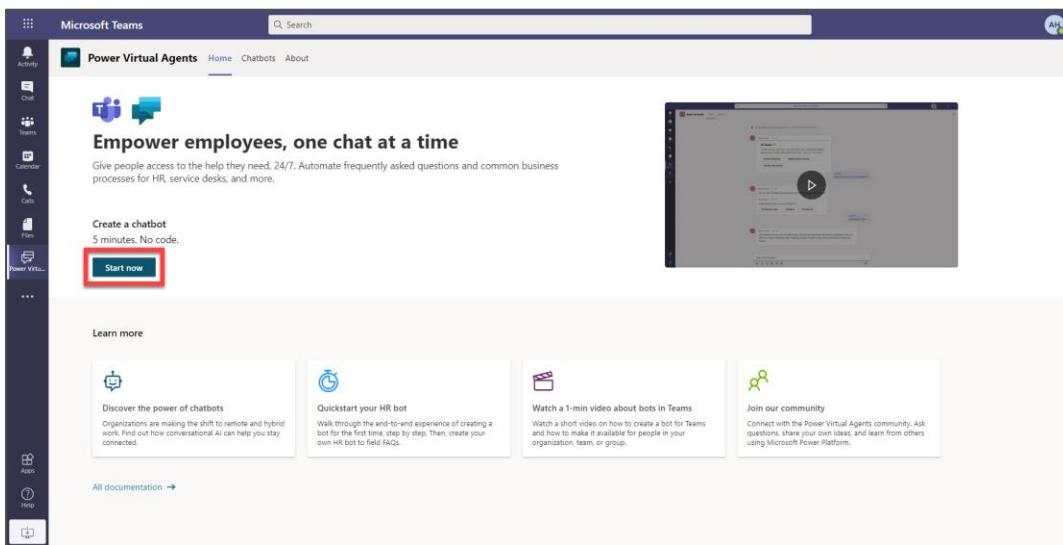


4. You will now see the Power Virtual Agents app in the left-hand navigation bar. Right click on that icon and select **Pin**. This pins the app to the navigation bar, making it easier to return to it when you need to.



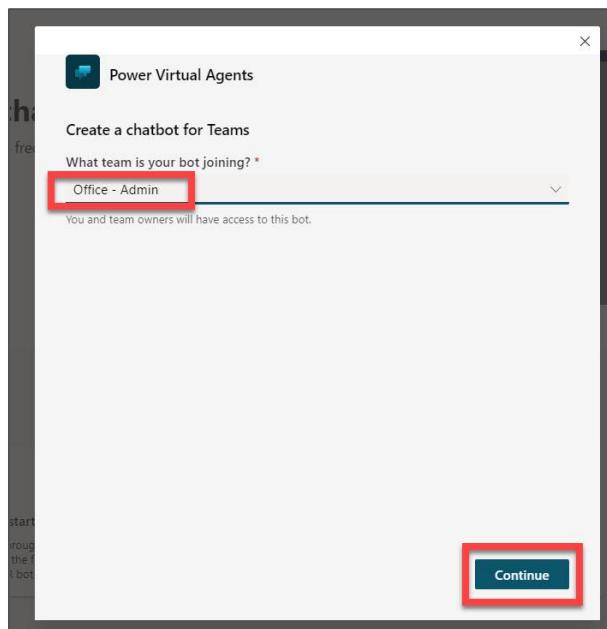
Task 2: Create a new chatbot

1. Click on the **Start now** button.

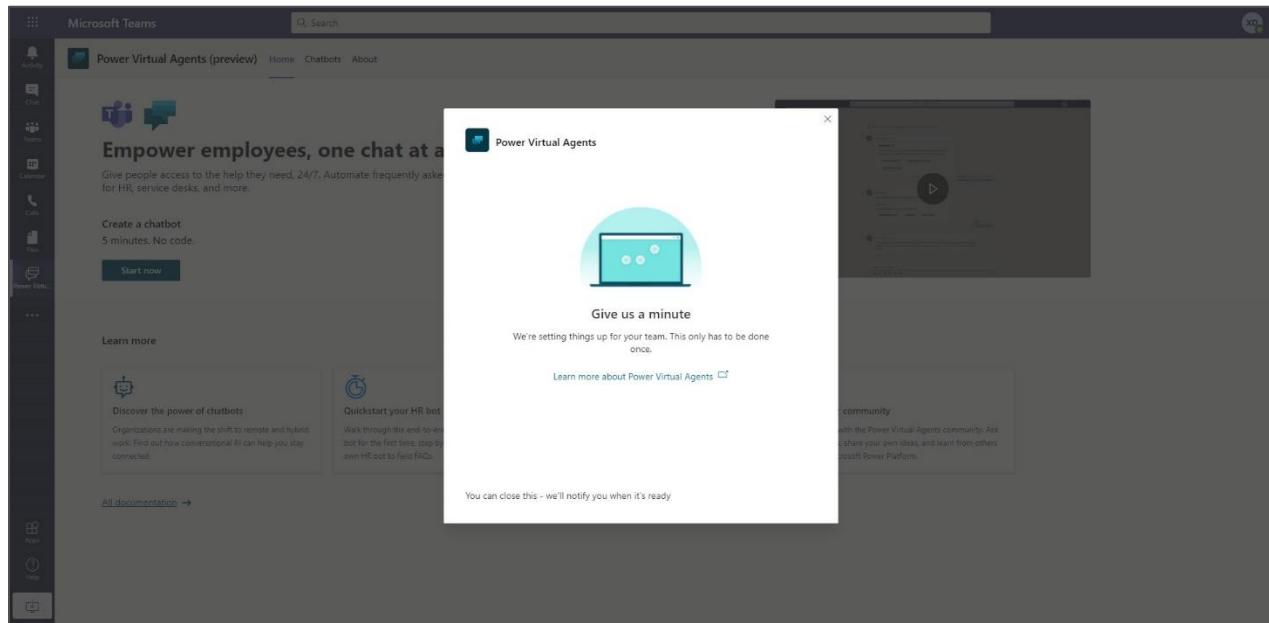


You will be asked which team you want your bot to join. When you are adding a bot as an app in Teams, you can think of it as adding a new virtual member of your team. Click on the dropdown and select the team you have used in the previous labs.

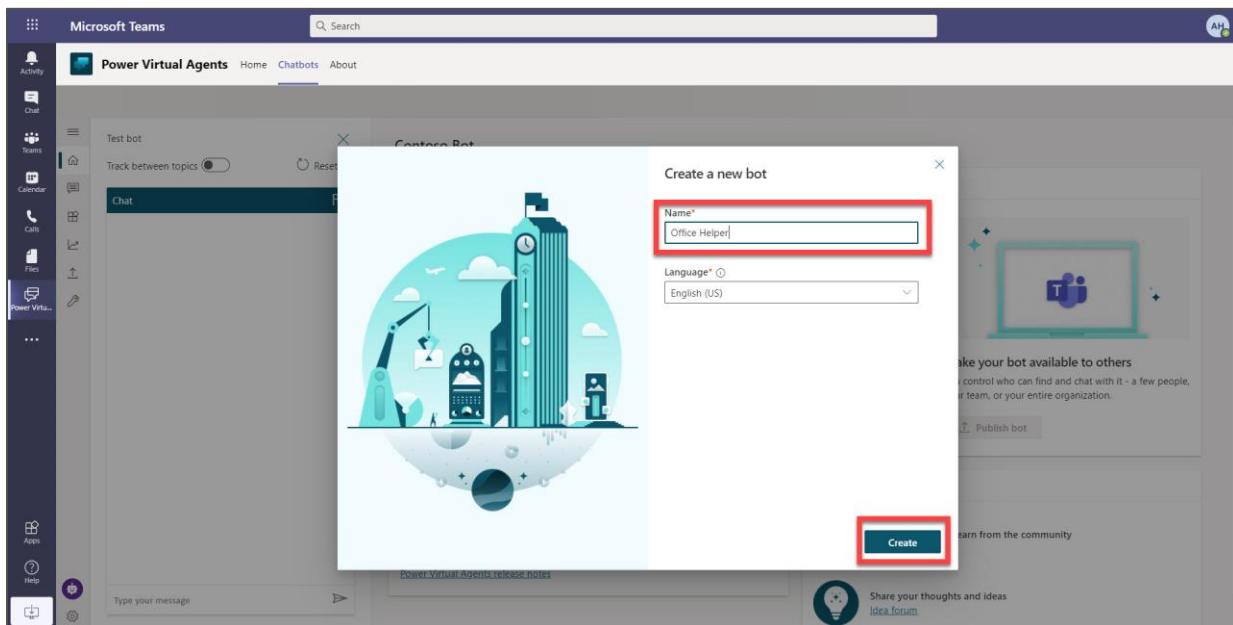
2. Once you have selected your team, click **continue**.



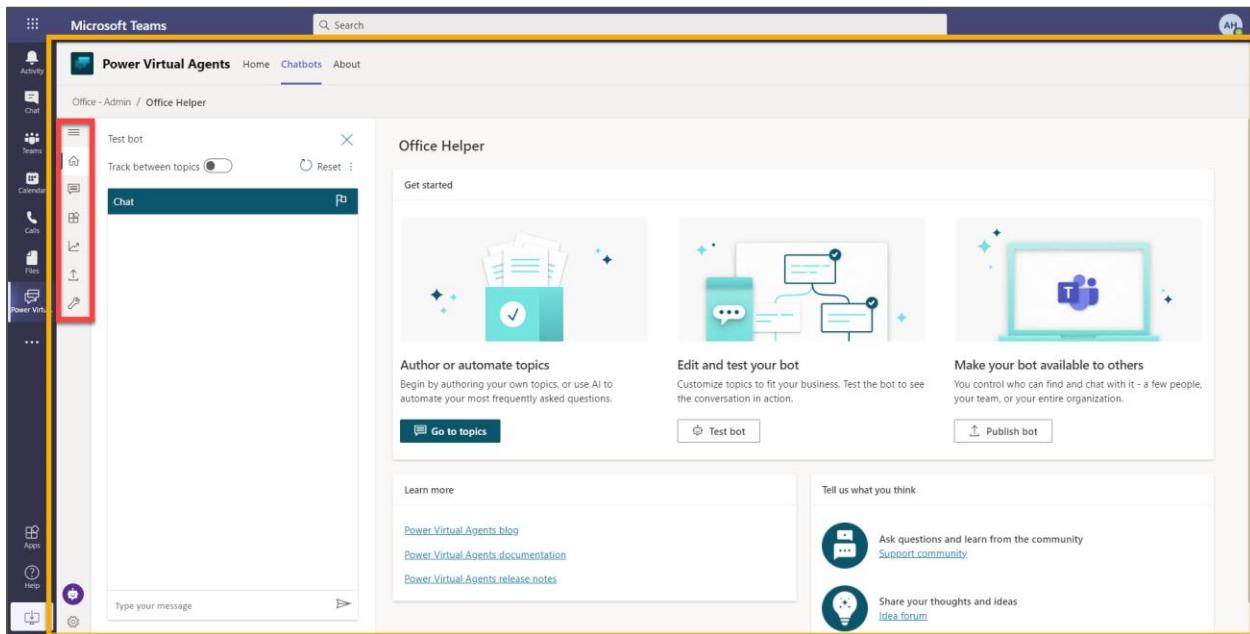
3. You will get a message advising that it takes some time to set this up for the first time. Wait for this to finish.



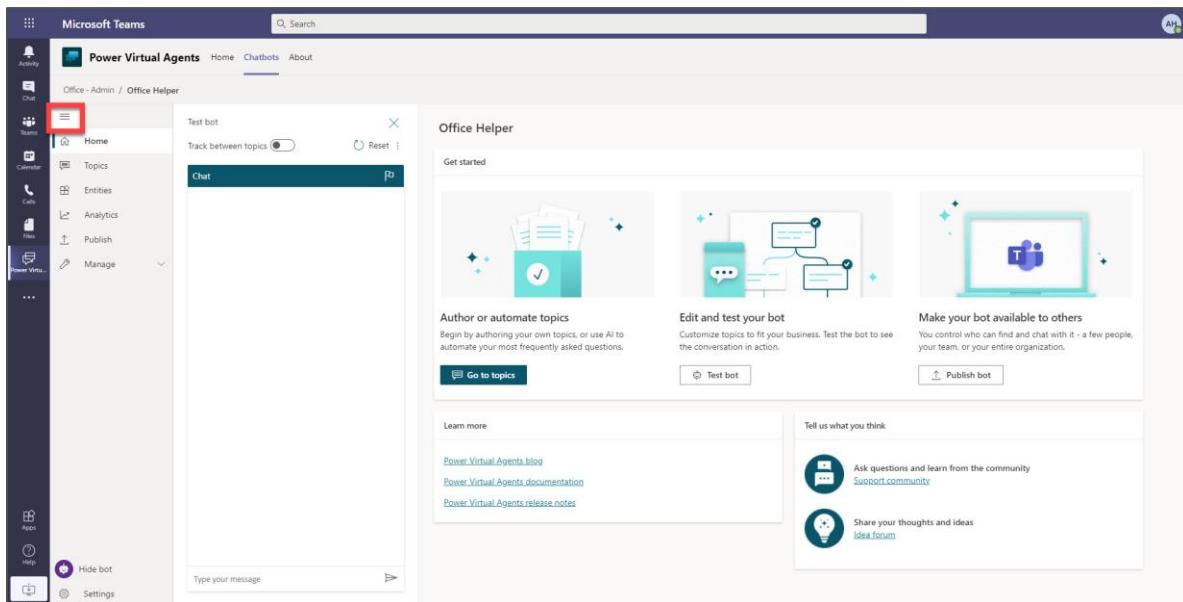
4. You will now be prompted to create a new bot. Give your bot a name (here we have used Office Helper). For this lab, you can make up your own bot name (or add your name to the end of the bot name – e.g. Office Helper – Jane) so that when we come to the collaboration exercise in Lab 6 the bots will have different names).
5. Click **create**. Wait a minute or so until the bot has been provisioned. (Note the capability here to create a bot in other languages).



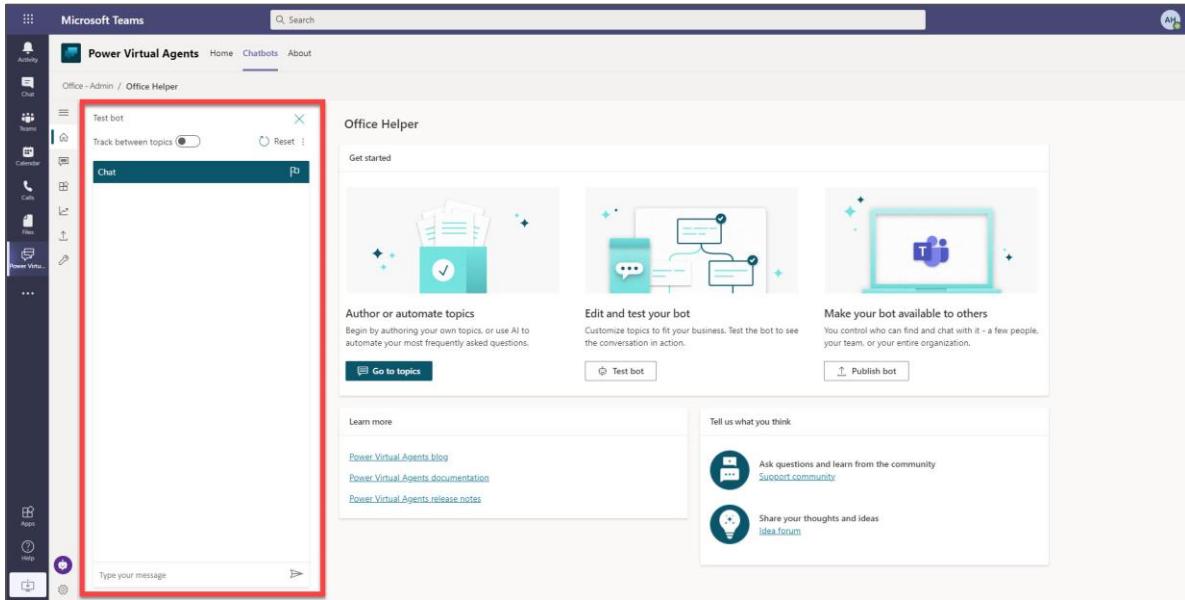
6. You are now seeing the familiar Power Virtual Agents builder inside Microsoft Teams. Your main navigation menu is on the left. By default this is collapsed, to give you more screen space to work with.



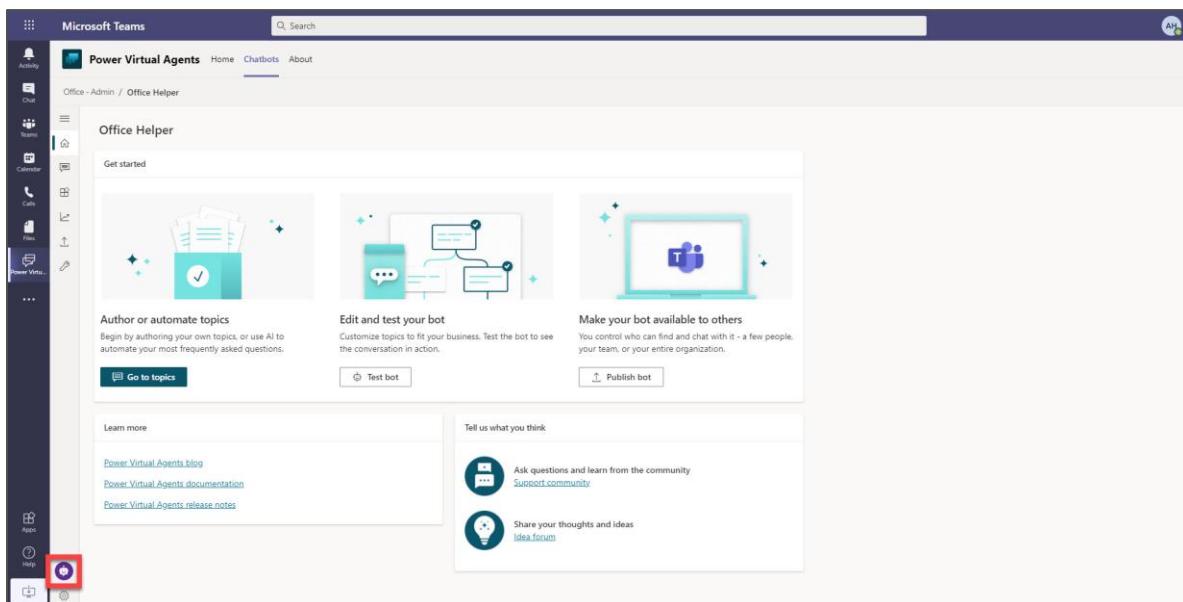
7. Click on the hamburger icon  to expand (or collapse) the navigation menu.



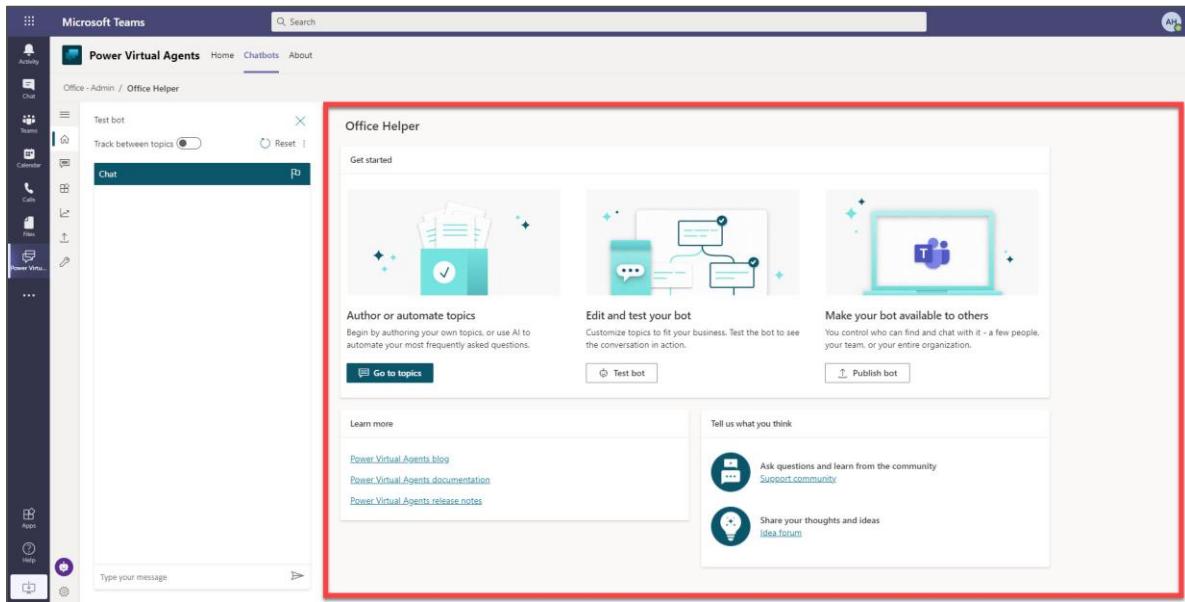
8. To the right of the Power Virtual Agents navigation menu you will find the test pane. You will use this for testing your bot as you build it.



9. You can hide or show this test pane clicking on the chatbot icon  near the bottom left of the screen.



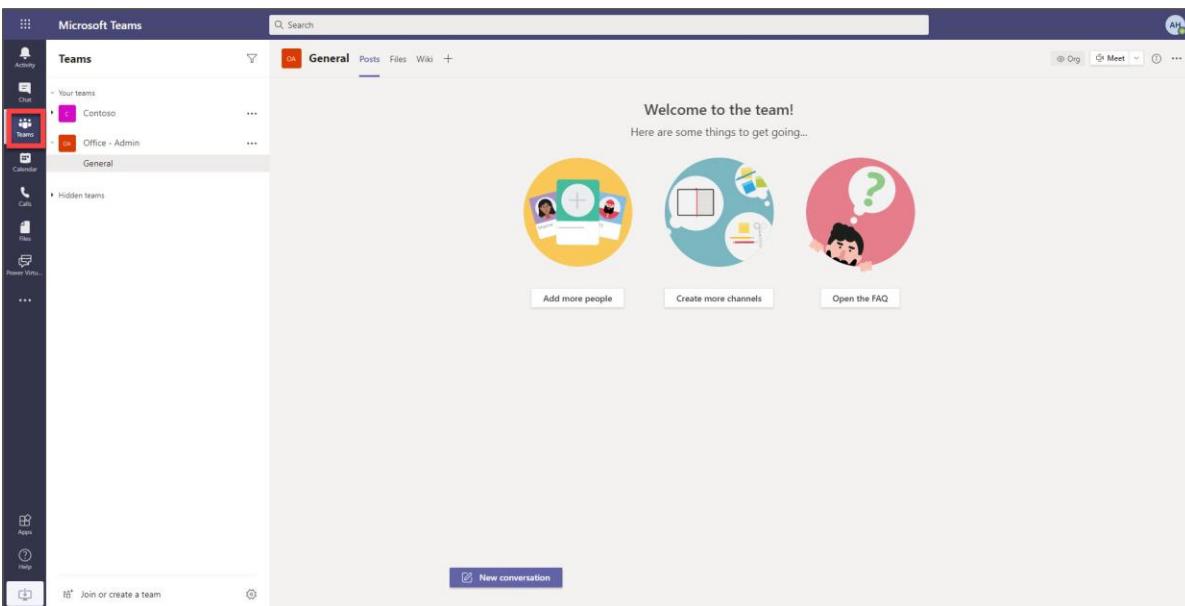
- The rest of the screen is the main canvas where you will be building your bot. As you work on topics, the authoring canvas will be displayed here. When you first create your bot (or when you navigate to the home screen), you'll find guided help steps here to assist you with navigation and getting started.



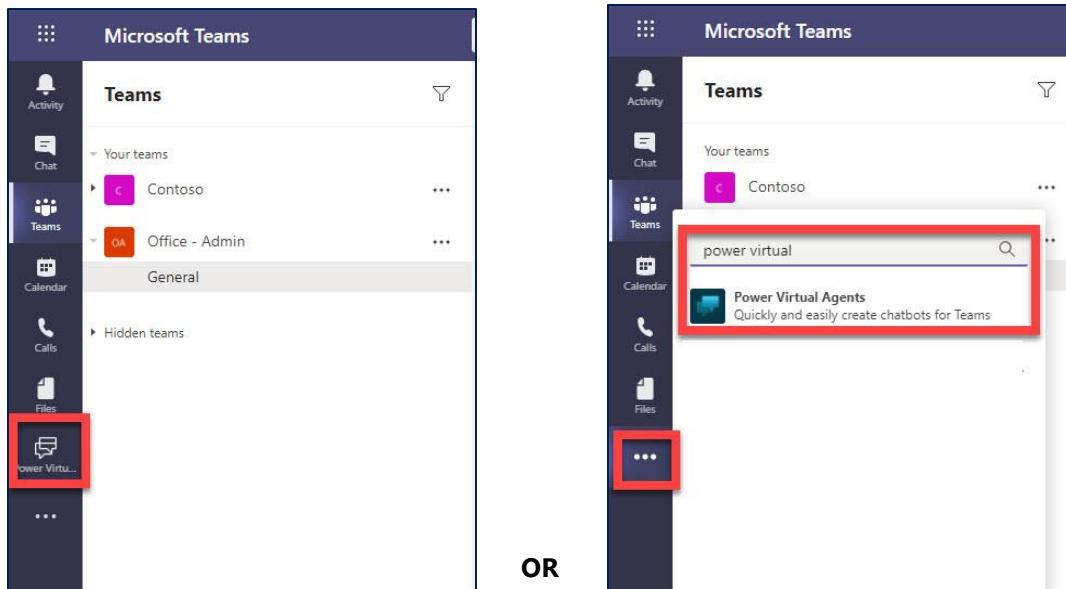
Task 3: Navigating between other parts of Teams and your existing chatbot

You can navigate away from building your chatbot to continue collaborating in Teams and come back to continue working on it at any time. Take note of this step – you will need to navigate back and forth later in the lab.

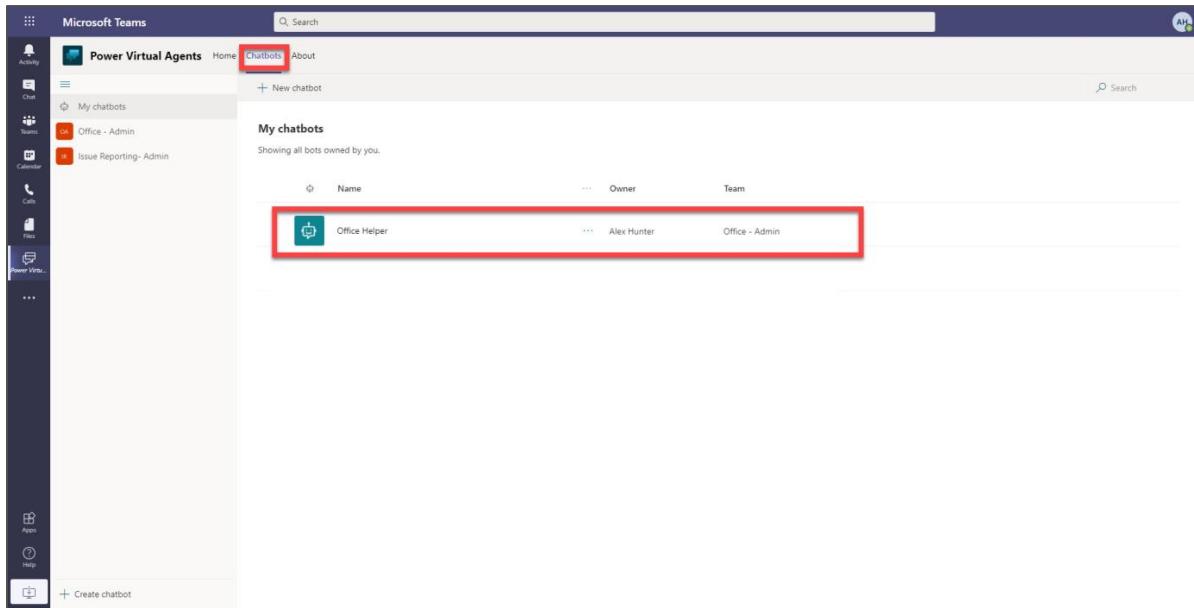
- Click back on the main Teams icon in the left-hand navigation bar to return to your teams.



2. To return to building your chatbot, you need to return to the Power Virtual Agents app in Teams. If you pinned the icon to the navigation menu earlier, you will find it there and you can click on it. If not, you can click on the **...** icon at the bottom of that menu, and search for Power Virtual Agents and select it.



3. Click on **Chatbots** in the menu at the top of your screen to see a list of all existing chatbots. Click on the name of your chatbot to open it for editing.



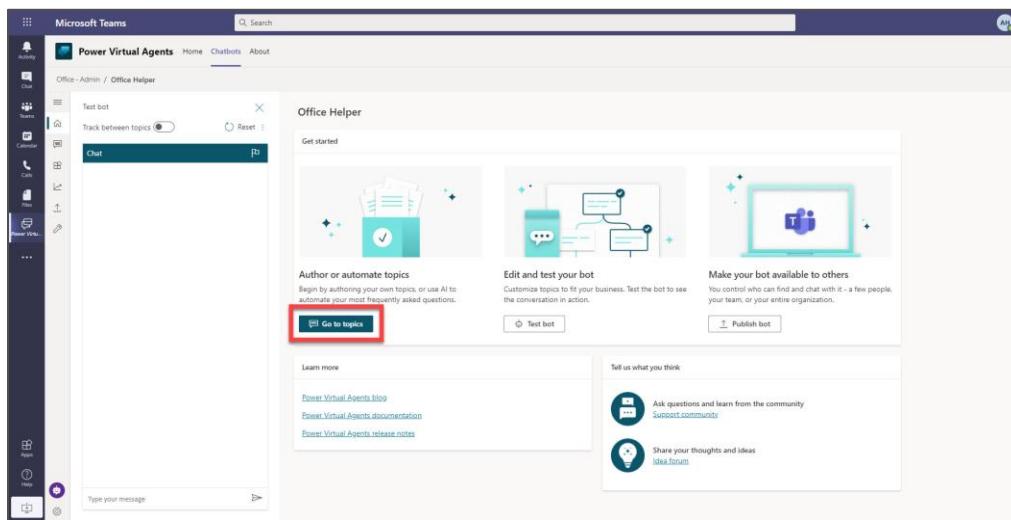
Exercise 2: Personalize the greeting

There are two special user related variables available to help personalize your chatbot conversations, known as claim variables. These variables allow you to show the display name and user ID of the logged in user in your chatbot.

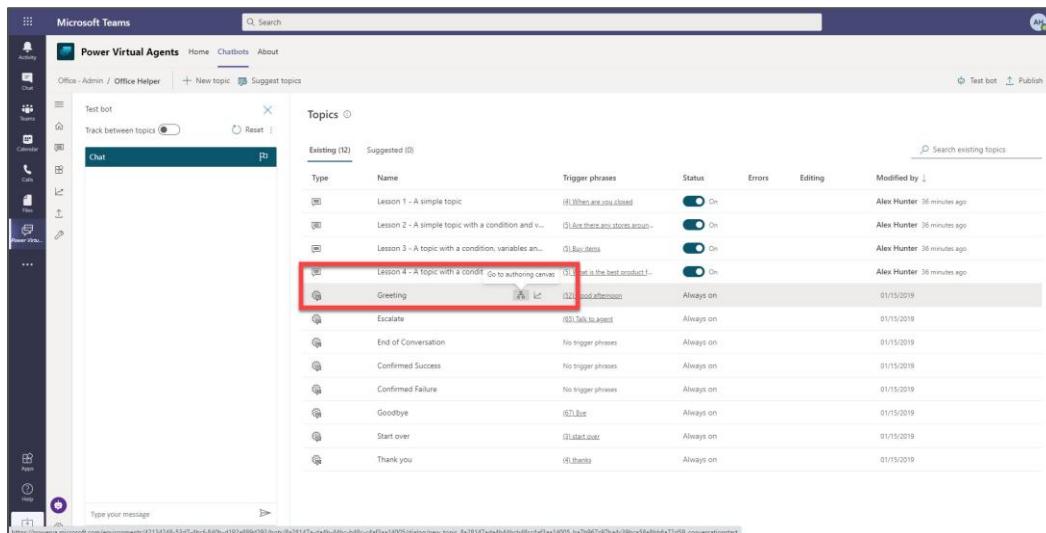
Task 1: Edit and personalize the greeting topic

In this task we will edit the greeting topic so that the chatbot greets the user in Teams by their name. We will also edit the standard greeting topic so that it explains to the user what it is designed to help with.

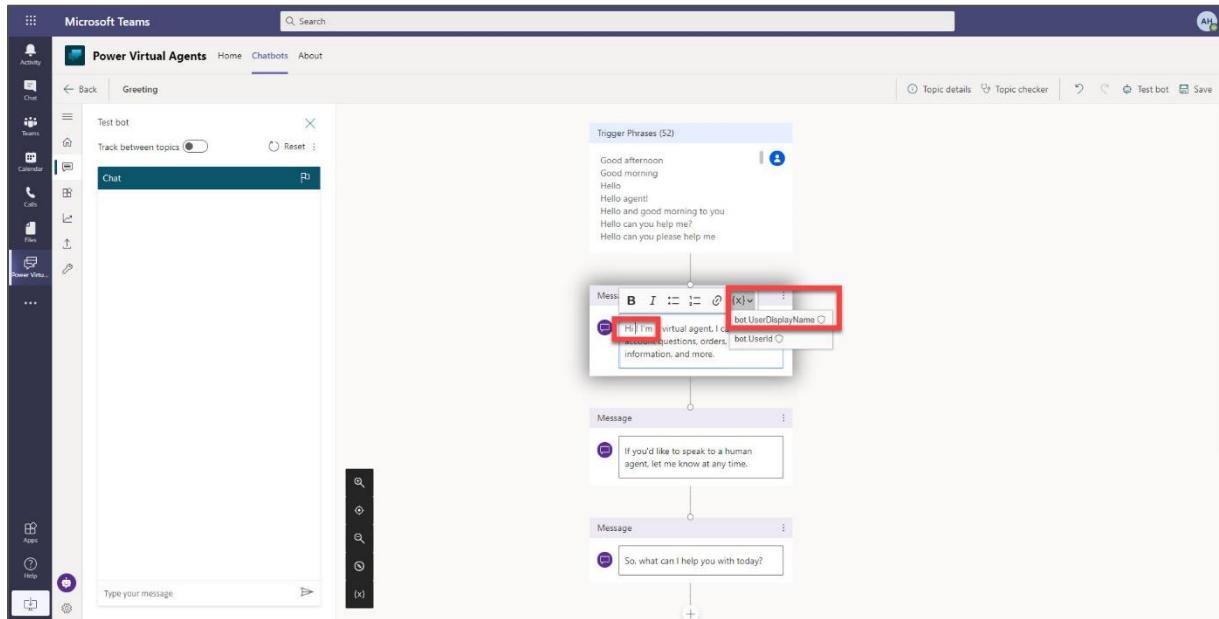
1. Click on **Go to Topics**.



2. Find the Greeting topic and hover your mouse over it. You will see icons appear to the right of the topic name. Click on the icon to open the topic in the authoring canvas.

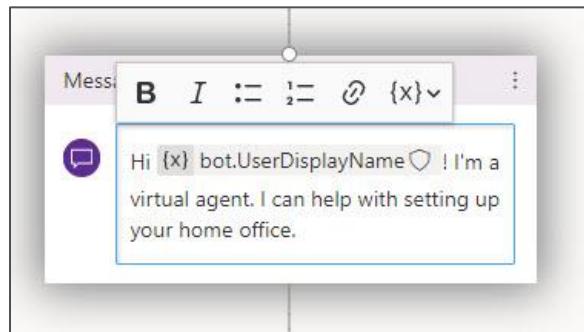


3. Let's personalize the greeting by adding a claim variable that will display the name of the user. Put your cursor between the word "Hi" and "!" and add a space. Click on the variable icon and select **bot.UserDisplayName**.

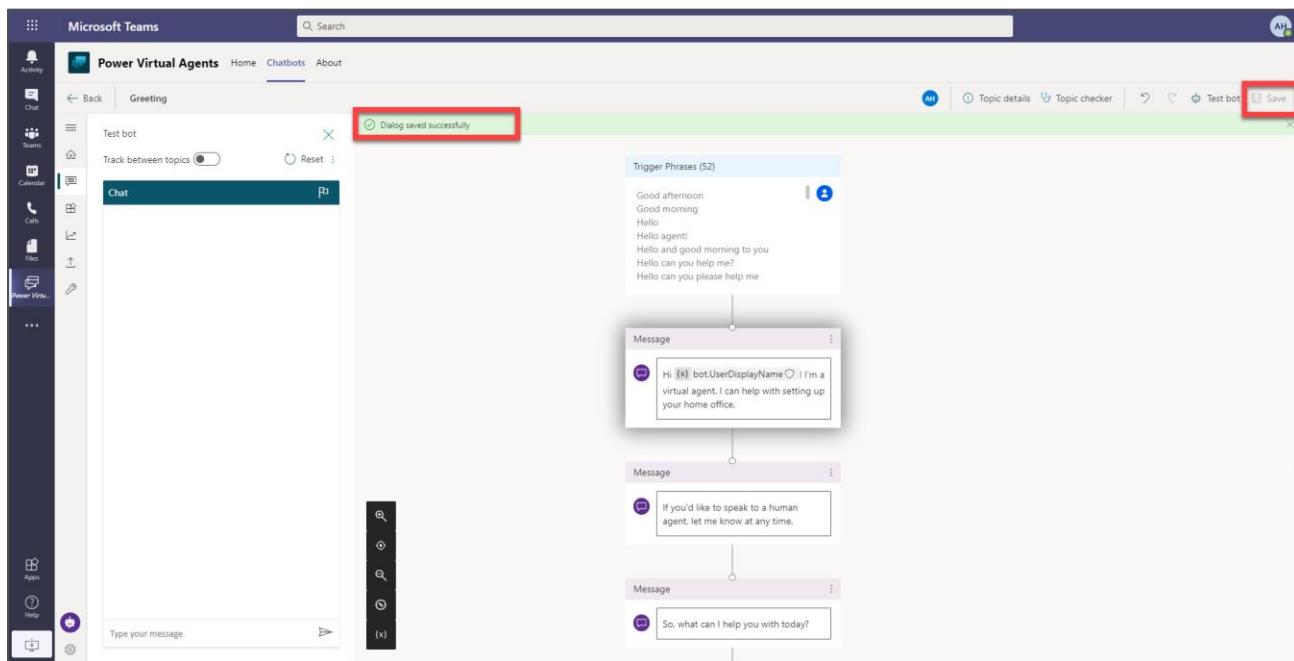


4. Now edit the rest of the message with the following text:

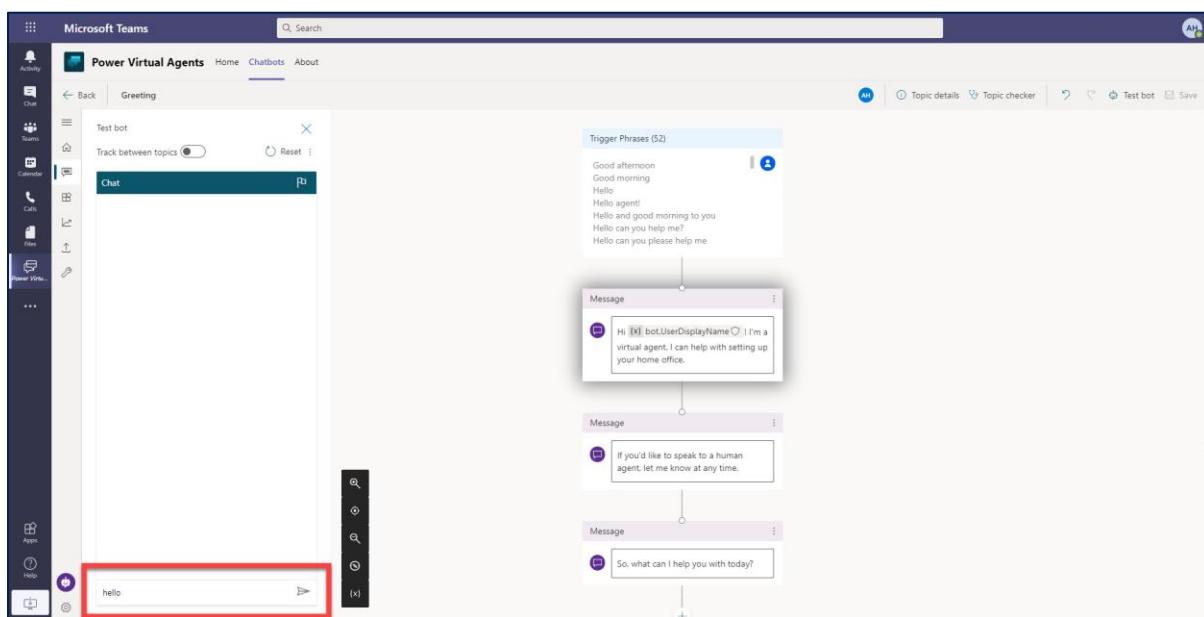
I'm a virtual agent. I can help you with setting up your home office.



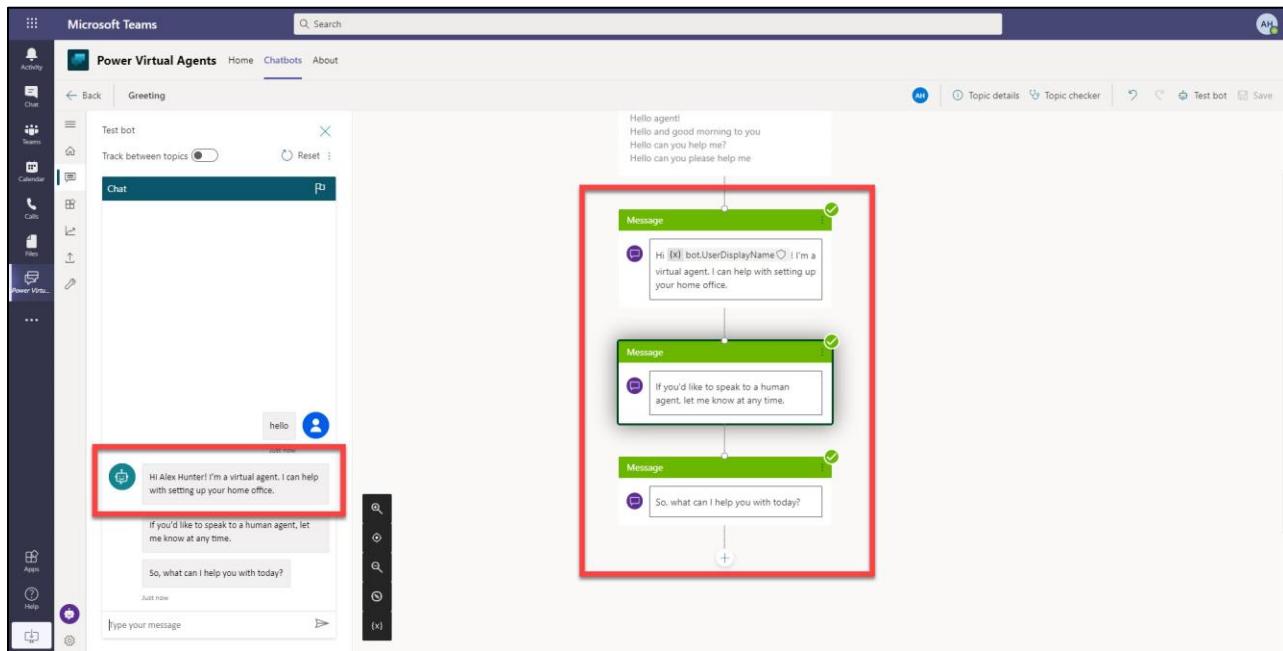
5. Click **Save** and wait until you see the confirmation message.



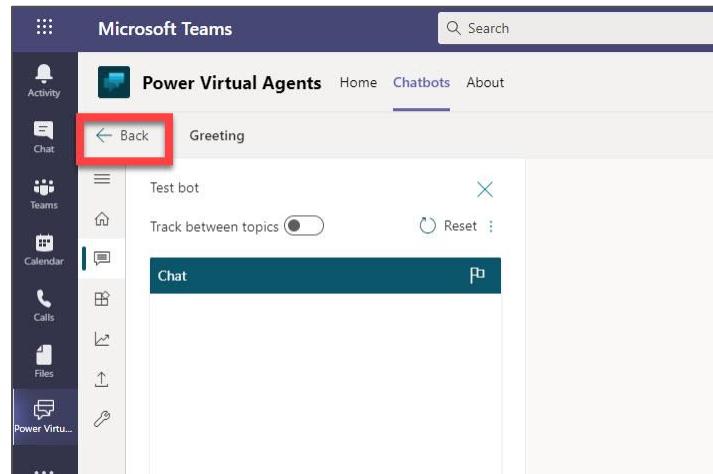
6. Now test your bot by typing **hello** in the test bot pane.



7. You should see your chatbot use your new greeting, including your name in the test pane. You will also see green ticks in the authoring canvas showing the successful path the chatbot has followed during your test.



8. Click **Back** to return to your list of topics.



Exercise 3: Create a topic

In this task we will create a topic for the bot to answer common questions from employees about setting up their home office.

Task 1: Switch off the lesson topics

Let's start by switching off the lesson topics, which are not needed as part of this chatbot.

1. In the list of existing topics, you will see topics named Lesson 1, Lesson 2, Lesson 3 and Lesson 4. Switch the **Status** toggle for each of these to the **off** position.

The screenshot shows the Microsoft Teams Power Virtual Agents interface. On the left, there's a sidebar with various icons like Activity, Chat, Teams, Calendar, Calls, and Power Virtual Agents. The main area is titled 'Topics' and shows a list of existing topics. Four specific topics are highlighted with a red box: 'Lesson 1 - A simple topic', 'Lesson 2 - A simple topic with a condition and variable...', 'Lesson 3 - A topic with a condition, variables an...', and 'Lesson 4 - A topic with a condition, variables an...'. Each of these topics has its 'Status' toggle switch set to 'Off'. Other topics listed include Greeting, Escalate, End of Conversation, Confirmed Success, Confirmed Failure, Goodbye, Start over, and Thank you, all of which have their status toggles set to 'On'.

2. You will need to wait a few seconds each time while the topic is turned off. You'll see a progress message and then a confirmation message at the top of the topics list.

This screenshot shows the same Microsoft Teams Power Virtual Agents interface as the previous one, but with a progress bar at the top of the 'Topics' list. The progress bar is labeled 'Turning off 'Lesson 2 - A simple topic with a condition and variable...' - 0% 0% 0%' and is currently at 0%. Below the progress bar, the list of topics remains the same, with the 'Status' column showing that the four lesson topics are now off. The other topics (Greeting, Escalate, etc.) remain on.

The screenshot shows the Microsoft Teams Power Virtual Agents interface. At the top, there's a red box highlighting a message: "Lesson 4 - A topic with a condition, variables and custom entity- has been turned off". Below this, the "Topics" section lists 12 existing topics. The first topic, "Lesson 4 - A topic with a condition, variables and custom entity-", is shown with its status as "Off". Other topics listed include "Lesson 3 - A topic with a condition, variables an...", "Lesson 2 - A simple topic with a condition and v...", "Lesson 1 - A simple topic", "Greeting", "Escalate", "End of Conversation", "Confirmed Success", "Confirmed Failure", "Goodbye", "Start over", and "Thank you".

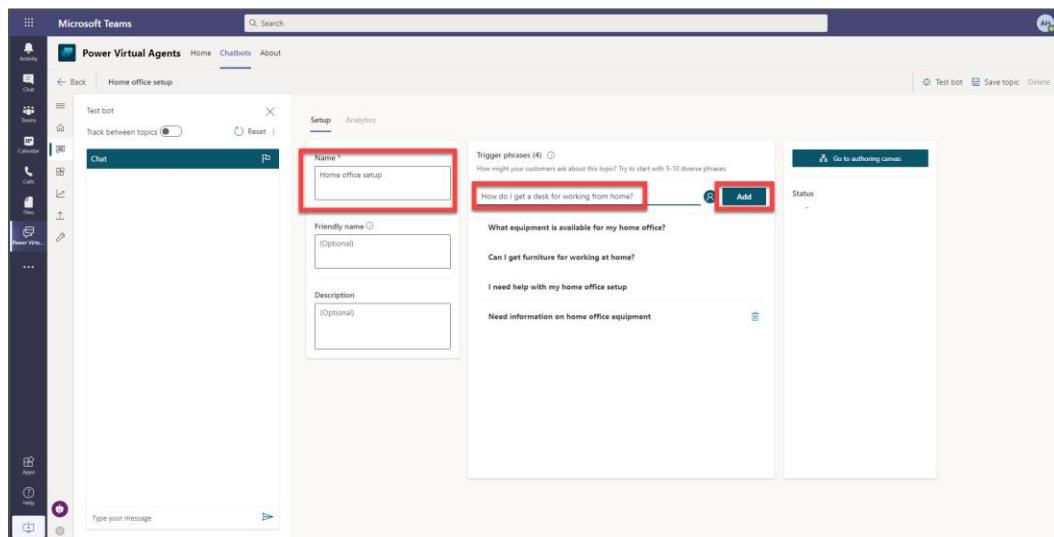
Task 2: Create a new topic for questions about home office setup

1. Create a new topic by clicking on +New topic.

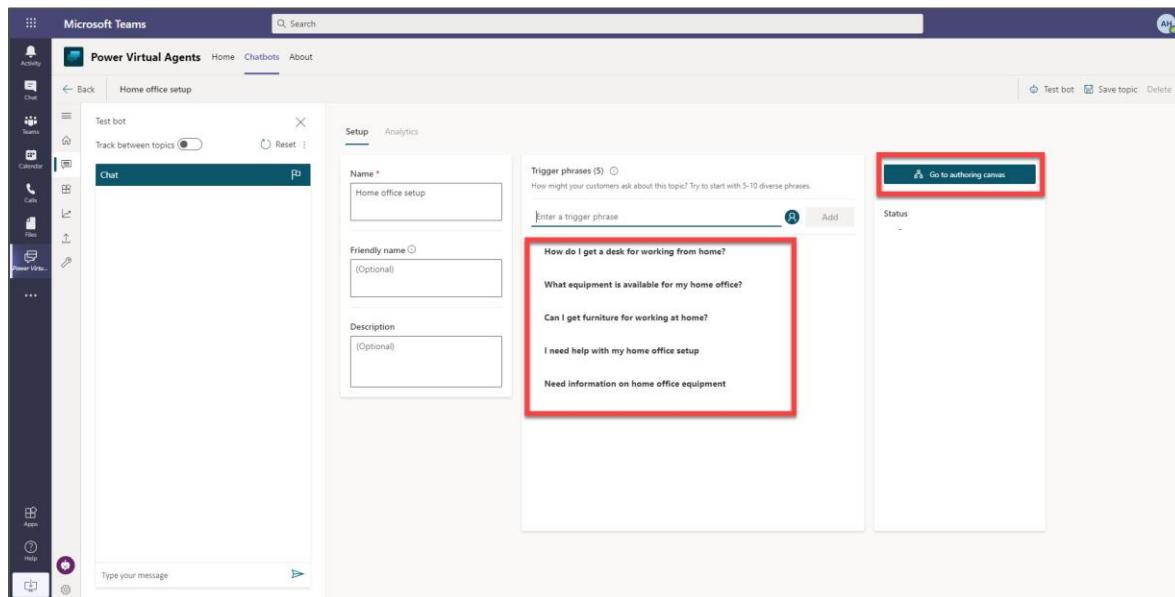
This screenshot is similar to the previous one but focuses on the creation of a new topic. The "+ New topic" button in the top navigation bar is highlighted with a red box. The rest of the interface, including the list of existing topics, is identical to the first screenshot.

2. Give the topic a name by typing: **Home office setup** in the Name box. Then add the following trigger phrases by typing them in and clicking the **Add** button after each one, until they all appear in the list

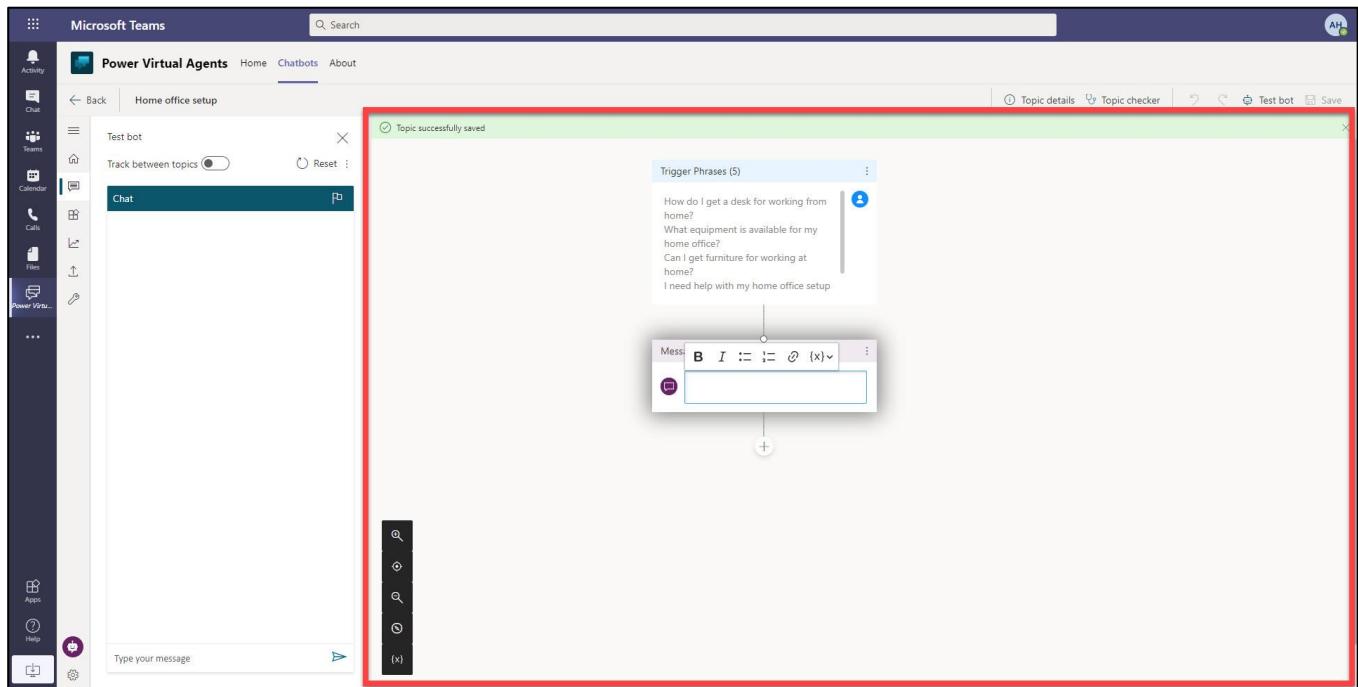
Need information on home office equipment
 I need help with my home office setup
 Can I get furniture for working at home?
 What equipment is available for my home office?
 How do I get a desk for working from home?



3. When you have added all the trigger phrases, click on the **Go to authoring canvas** button. This will save your topic and take you to the authoring canvas.



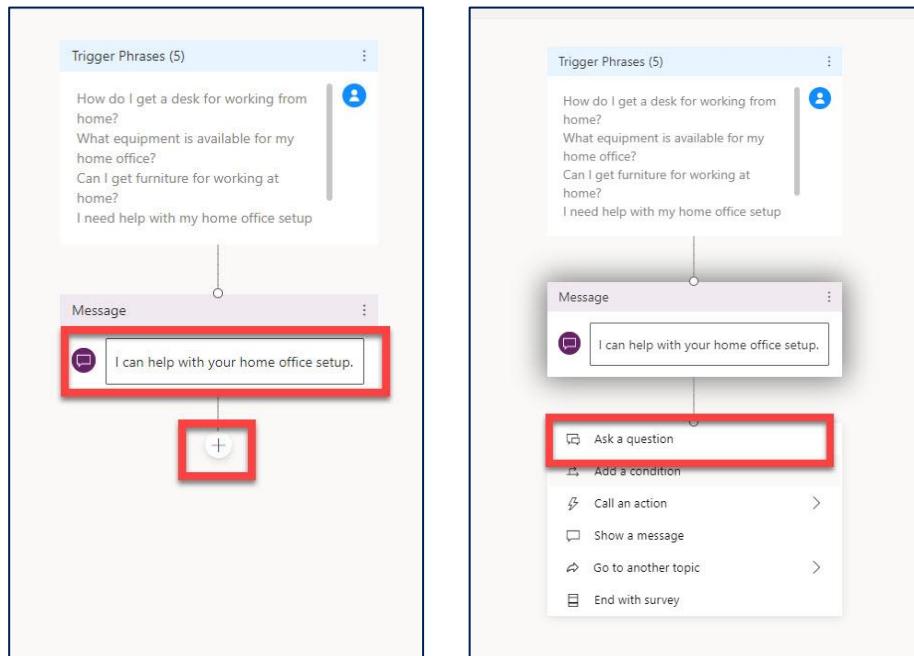
4. You will now see the authoring canvas for Power Virtual Agents, with your trigger phrases at the top. This is a drag and drop interface that you will use to build out your conversation. You will create a conversation tree with a series of messages, questions and conditional branches.



5. Add the following text into the message box:

I can help with your home office setup.

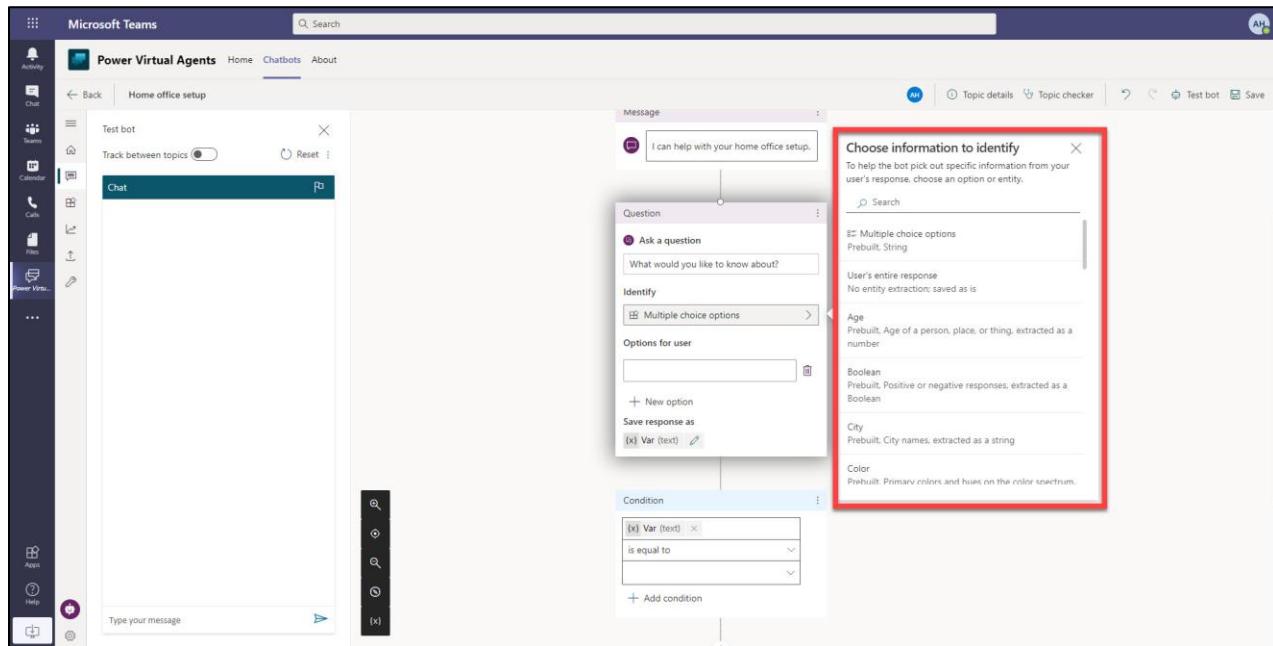
Then click the **Add node**  button underneath the message box and select: **Ask a question**.



6. Now type the following in the question box

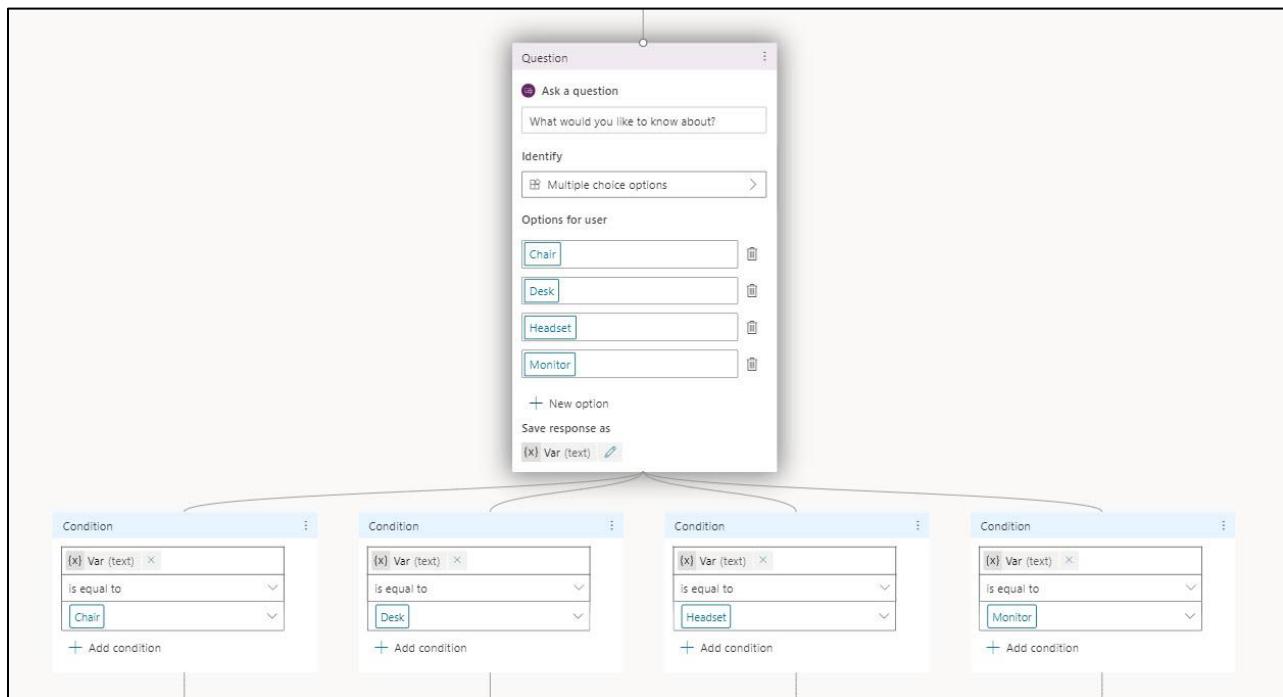
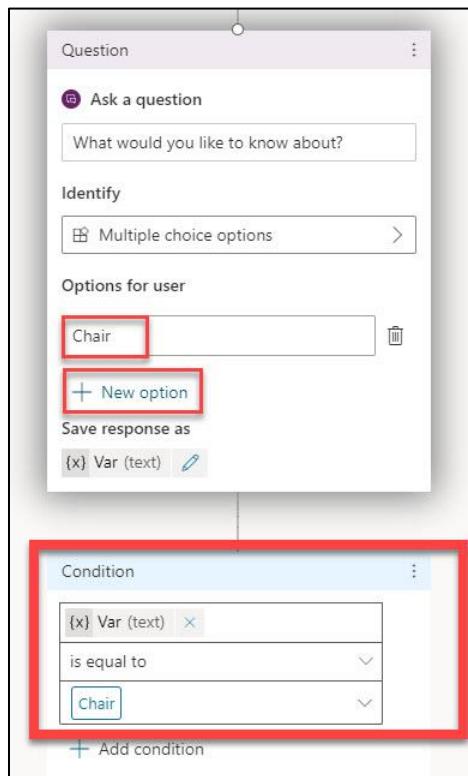
What would you like to know about?

7. Power Virtual Agents can identify various types of information (known as entities) from unstructured text in the response from the user. When you add a question to your conversation tree, you choose what kind of information the bot should look for and extract from the response. Set the Identify field to **Multiple choice options**.



8. Add the following as **Multiple choice options**. You will see that the conditional branching is automatically created as you add each new option.

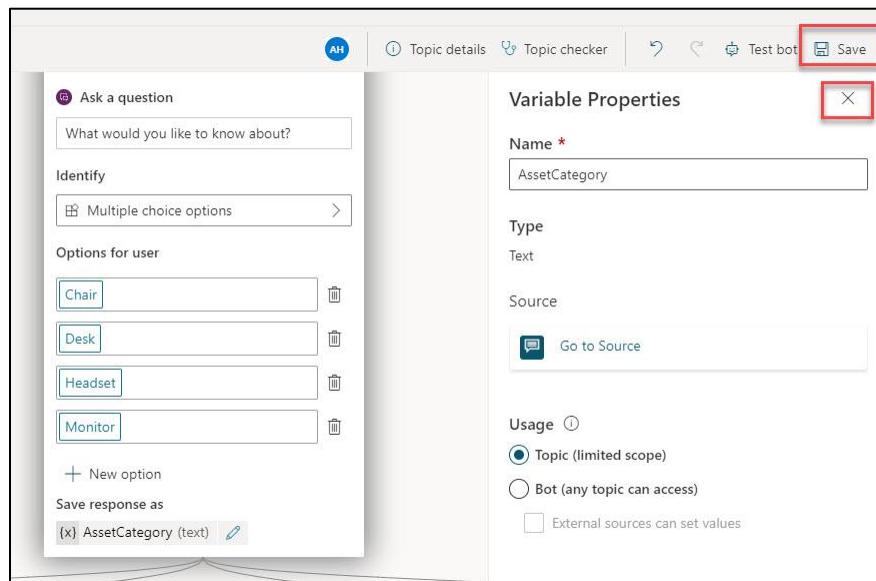
Chair
Desk
Headset
Monitor



9. The response the user chooses here will be stored as a variable. Edit the variable name by clicking on the pencil icon, and then entering the name **AssetCategory** in the Variable Properties box.

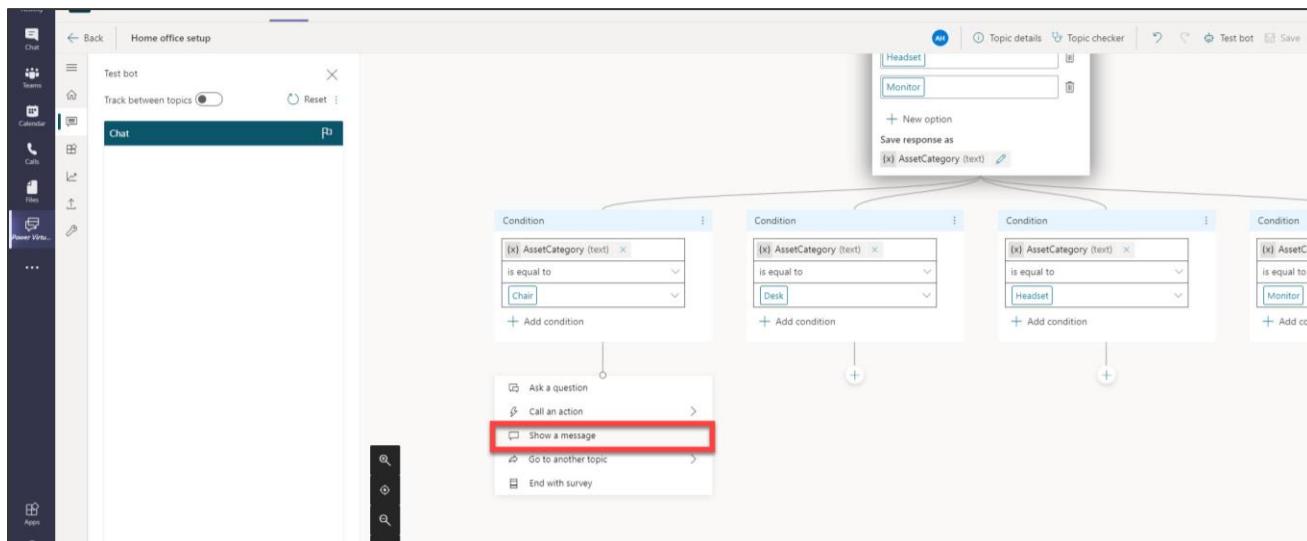
The image consists of two vertically stacked screenshots from a Microsoft Power Automate or similar low-code platform. The top screenshot shows a card configuration screen with a title 'Ask a question' and a sub-section 'Identify' containing 'Multiple choice options'. Below this are four options: 'Chair', 'Desk', 'Headset', and 'Monitor', each with a small trash bin icon. At the bottom of this section is a button '+ New option'. Below the card is a 'Save response as' section with a dropdown menu '(x) Var (text)' followed by a pencil icon, which is highlighted with a red box. The bottom screenshot shows a 'Variable Properties' dialog box. It has a red border around the 'Name' field. The 'Name' field contains the value 'AssetCategory'. Below it are sections for 'Type' (set to 'Text'), 'Source' (with a 'Go to Source' button), 'Usage' (radio buttons for 'Topic (limited scope)' and 'Bot (any topic can access)', with 'Topic' selected), and a checkbox for 'External sources can set values' (which is checked). The background of the bottom screenshot shows the same card configuration screen as the top one.

10. Close the Variable Properties pane by clicking on the X and then save your chatbot by clicking the **Save** button.



11. Now we will add a message under each Asset Category, giving the user information about the category they choose.

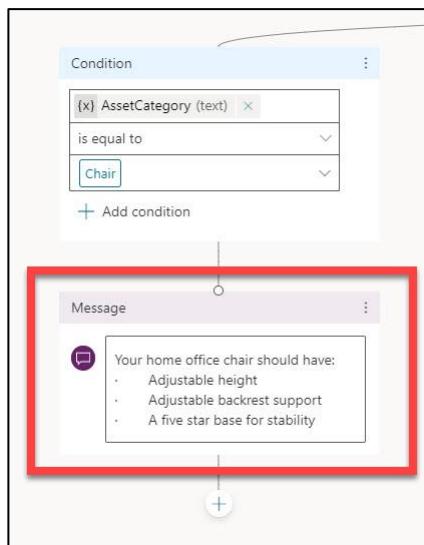
Under the first conditional node (Chair) click on the **Add node**  button and select **Show a message**:



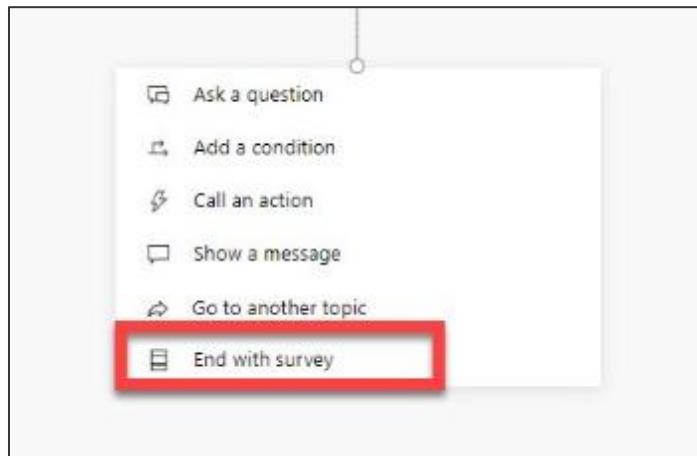
12. Add the following text to the message box:

Your home office chair should have:

- Adjustable height
- Adjustable backrest support
- A five star base for stability



13. Then add a node underneath the message to end the conversation with a survey. This will trigger the pre-built "End of Conversation" topic in Power Virtual Agents which asks the user whether or not their question was answered, and then finish with a satisfaction survey question.



The screenshot shows the Microsoft Teams Power Virtual Agents builder interface. On the left, there's a sidebar with various icons for Activity, Chat, Teams, Calendar, Calls, Files, and Power Virtual Agents. The main area has tabs for Home, Chatbots, and About, with 'Home office setup' selected. A 'Test bot' button is at the top. Below it, a 'Track between topics' toggle is turned off. The main workspace shows a conversation flow starting with a 'Chat' node. This leads to three parallel 'Condition' nodes: 'AssetCategory (text) is equal to Chair', 'AssetCategory (text) is equal to Desk', and 'AssetCategory (text) is equal to Headset'. Each condition leads to a 'Message' node containing a list of requirements. Finally, each message node leads to an 'End' node, which then leads to an 'End of Conversation' node. The 'End of Conversation' node is highlighted with a red box.

14. Under the **Desk** node, add this **message**.

Make sure you have plenty of desk space, and put the things you use a lot within comfortable arm's reach. Standard desks are between 28 and 32 inches in height, but we can also provide adjustable height and standing desks.

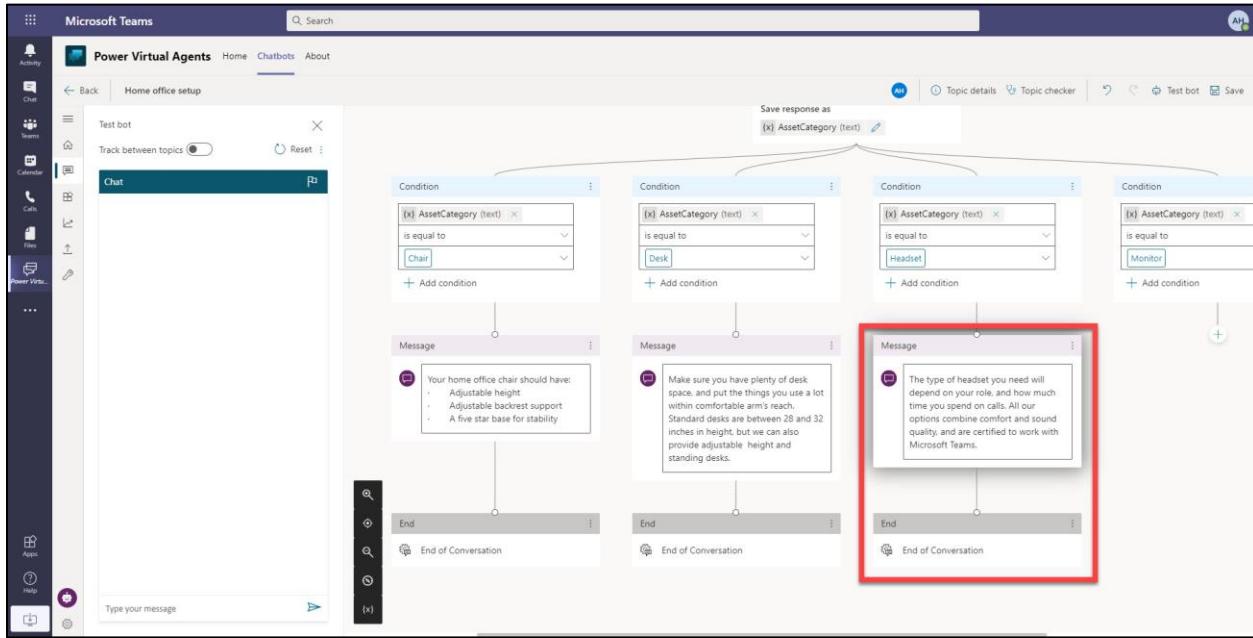
15. Then add a node to **End with survey**.

This screenshot shows the same Microsoft Teams Power Virtual Agents builder interface as the previous one, but with an additional node added. The 'End of Conversation' node from the previous step has been replaced by a new 'End with survey' node. This new node is highlighted with a red box. The rest of the flow remains the same, with conditions for Chair, Desk, Headset, and Monitor, each leading to a message node about desk requirements and an 'End with survey' node.

16. For the **Headset** node, add a **message** with the following text:

The type of headset you need will depend on your role, and how much time you spend on calls. All our options combine comfort and sound quality, and are certified to work with Microsoft Teams.

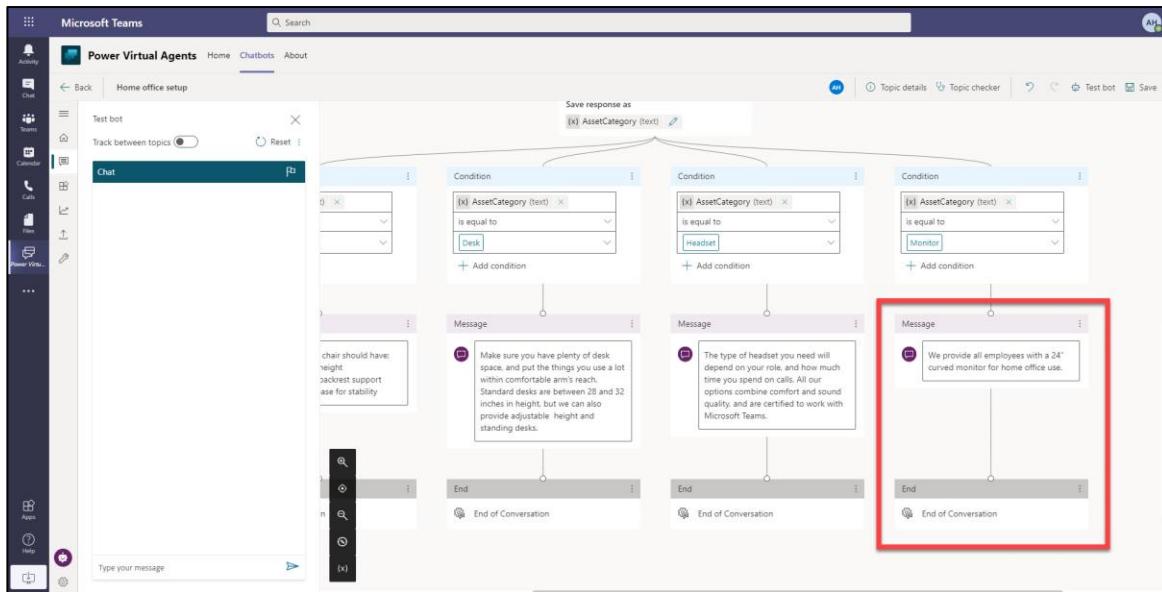
17. Then add a node to **End with survey**.



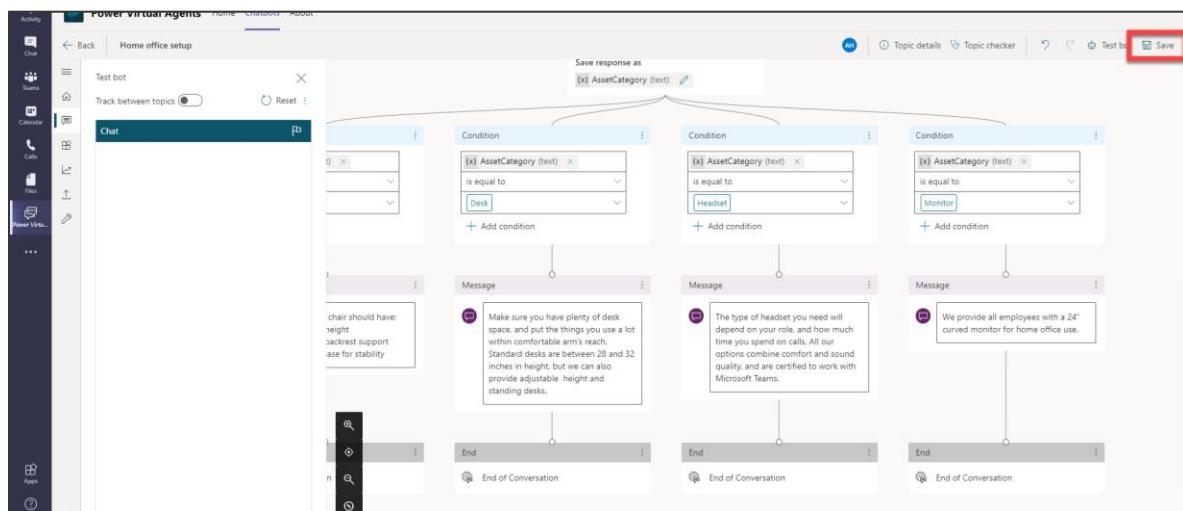
18. For the **Monitor** node, add a **message** with the following text:

We provide all employees with a 24" curved monitor for home office use.

19. Then add a node to **End with survey**.

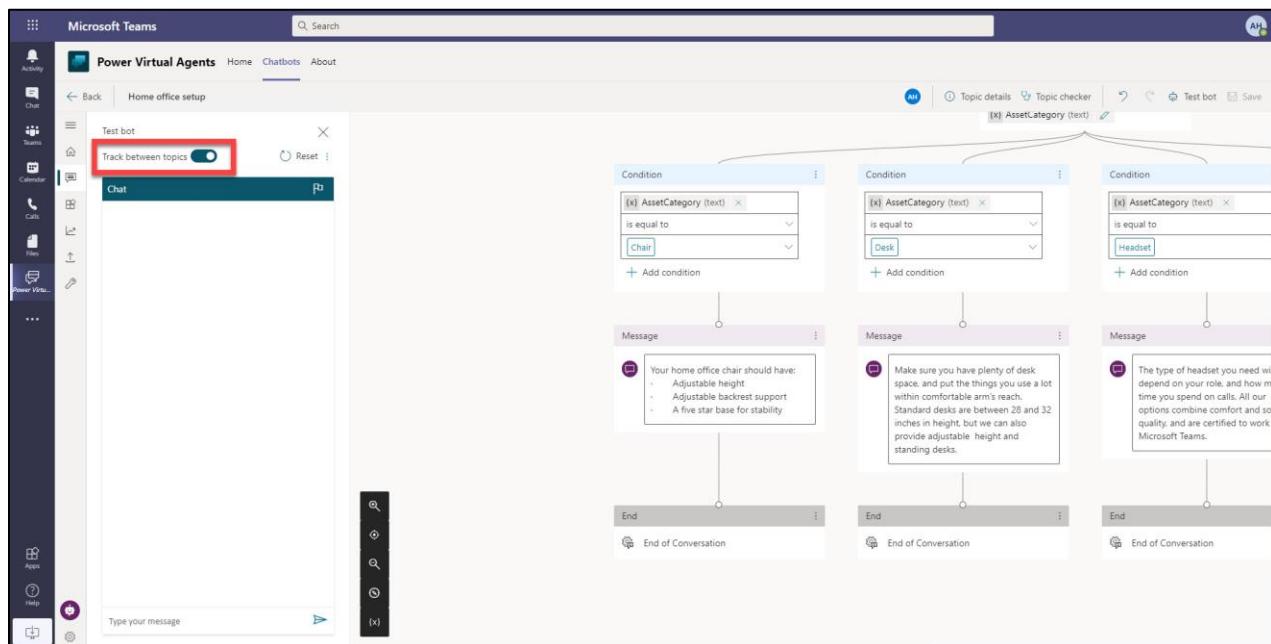


20. Click **Save** to save your topic.

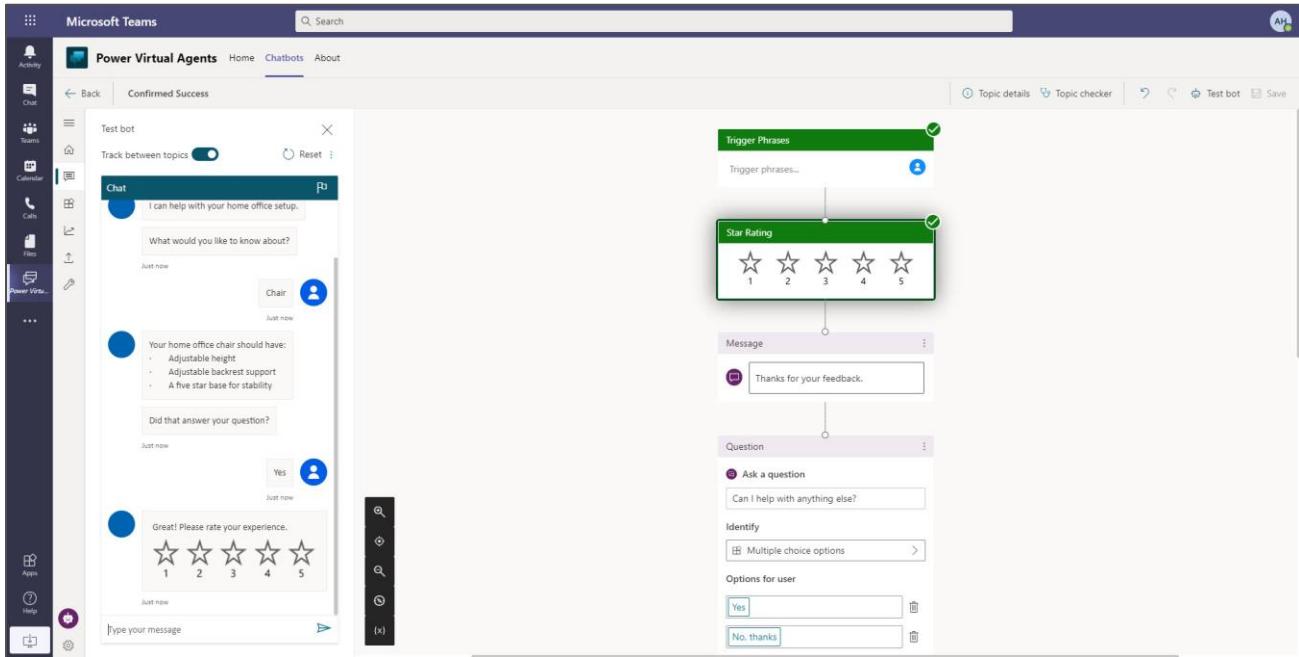


Task 3: Test your chatbot

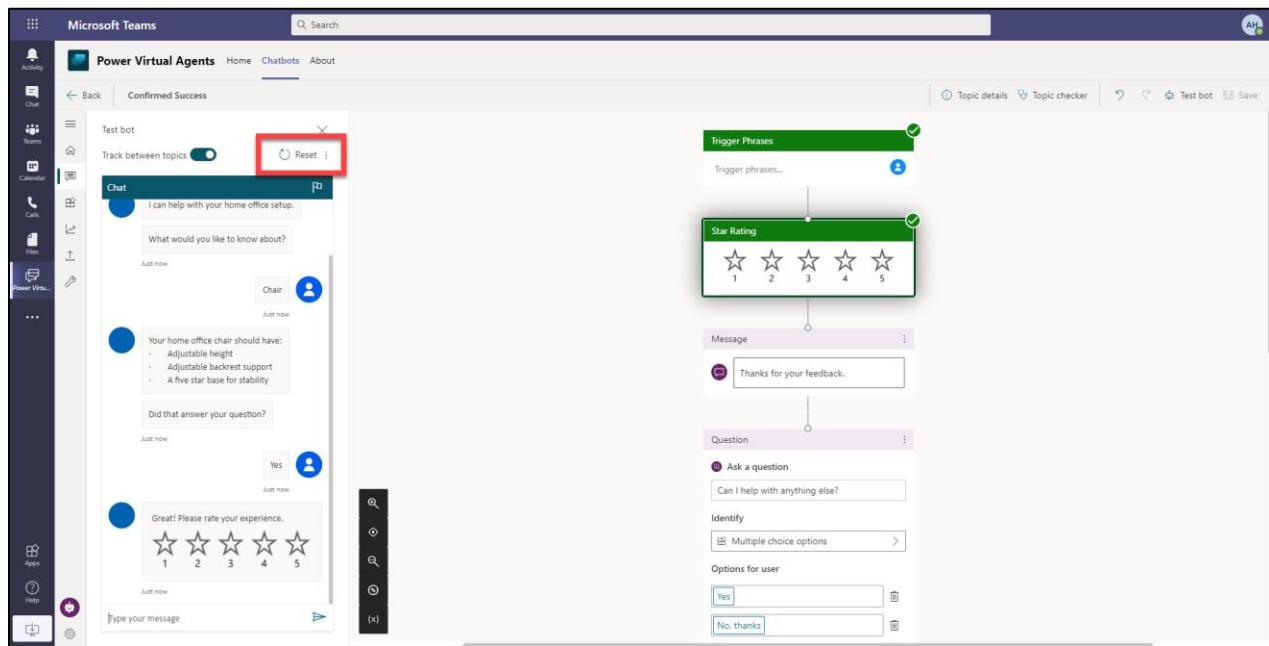
1. Toggle the “Track between topics” setting in the Test bot pane to **on**. This means that as you test your bot, you will see the tracking of the conversation flow across different topics.



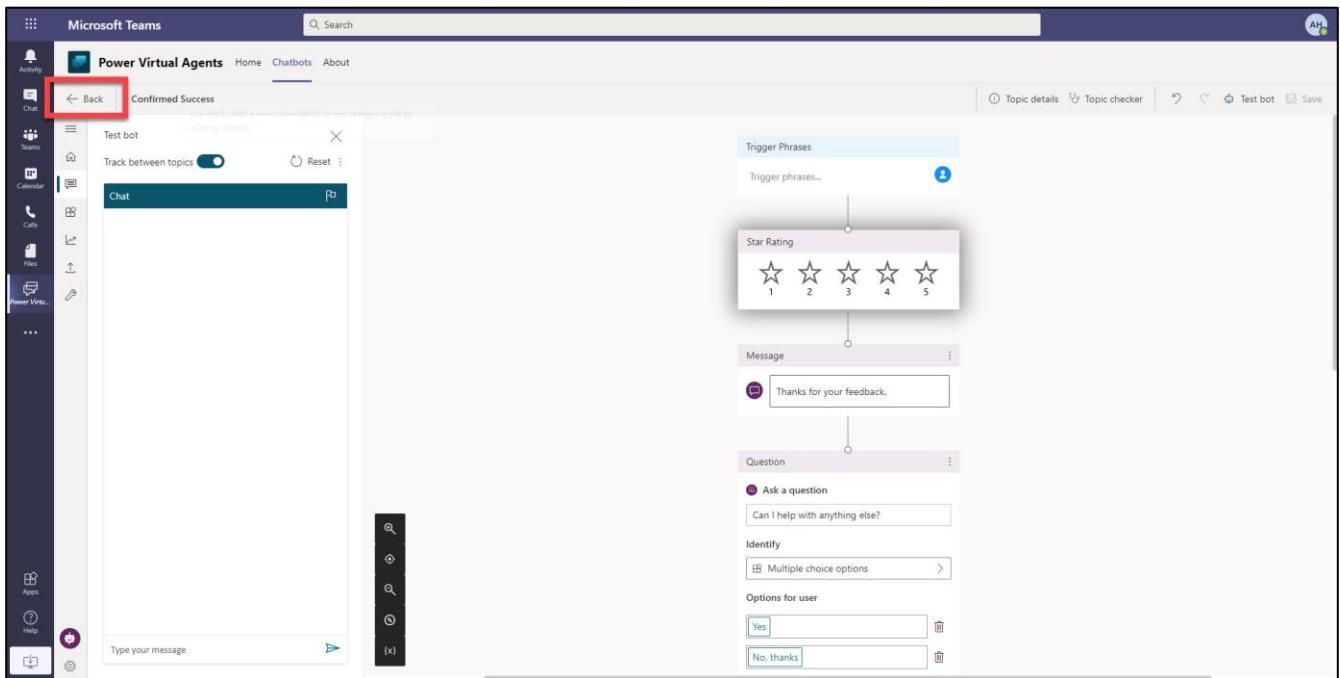
2. Test your chatbot in the Test bot pane by typing in one of the trigger phrases and going through the conversation. You will see the path the bot has followed indicated with green ticks in the authoring canvas, and you will see it move from the topic you created to the built in **End of Conversation** topic.



3. You can reset your Test bot pane by using the reset button at any stage.



4. Click on the **Back** button to return to the Topics page.

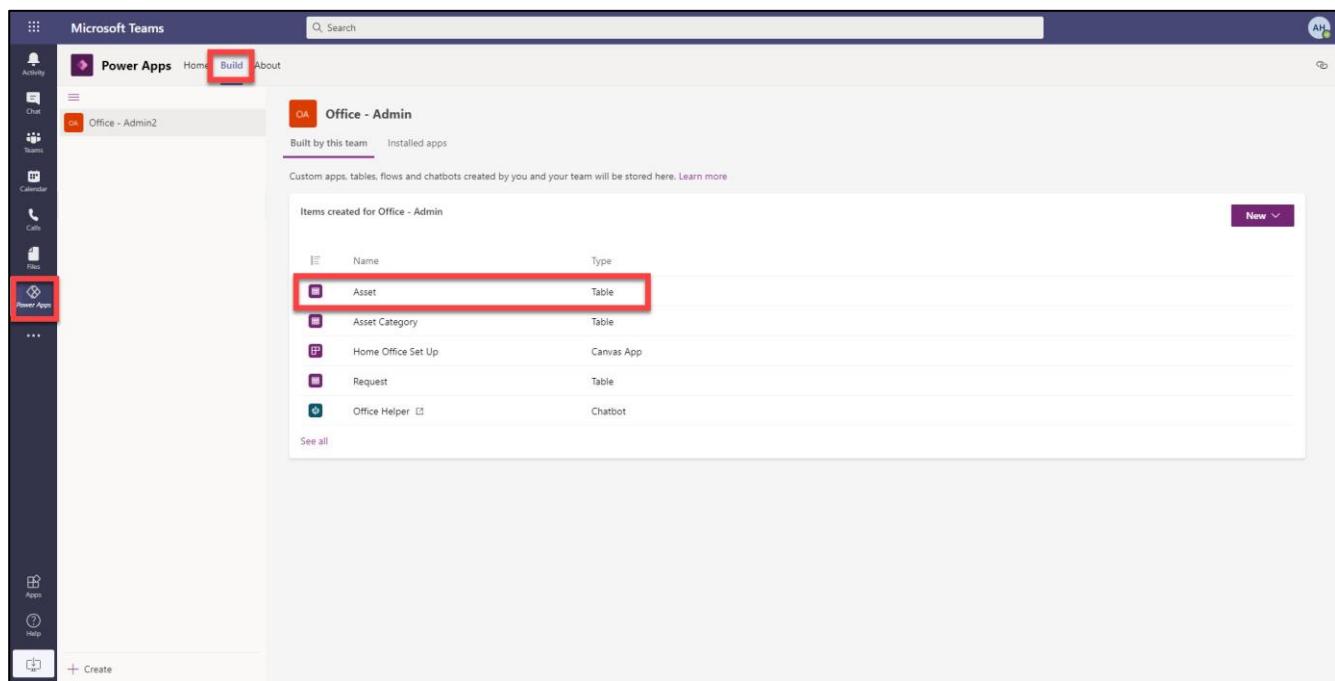


Exercise 4: Create a flow using the embedded flow experience

In this part of the scenario, we will set up the chatbot to interact with the Dataverse for Teams tables we created in Lab 01 so that the user can submit a request for home office equipment using the chatbot. To do this we will be using the embedded flow experience inside the Power Virtual Agents app in Teams.

Task 1: Get the schema name for your Dataverse for Teams table

1. When you created your Dataverse for Teams tables and columns, each component is given a technical or schema name behind the scenes. We will need to find that in order to use it in our flow. Navigate back to Power Apps using the icon on the Teams left hand navigation bar. Select the **Build** tab, and select the Team you have been working in. Then click on the **Asset table** in the list of items created.



2. You will now see a behind the scenes view of your table and columns. Under the heading Name you will see schema names, in a format like **cr26c_assetid**. Your schema names will be slightly different – you will have your own unique prefix which is not the same as this example.

Display name	Name	Data type	Type	Customizable	Required	Searchable
Asset	cr26c_assetid	Unique Identifier	Standard	✓	Required	✓
Asset Category	cr26c_assetcategory	Lookup	Custom	✓	Optional	✓
Asset Name	cr26c_name	Text	Custom	✓	Required	✓
Created By	createdby	Lookup	Standard	✓	Optional	✓
Created By (Delegate)	createdonbehalfby	Lookup	Standard	✓	Optional	✓
Created On	createdon	Date and Time	Standard	✓	Optional	✓
Import Sequence Number	importsequencenumber	Whole Number	Standard	✓	Optional	✓
Modified By	modifiedby	Lookup	Standard	✓	Optional	✓
Modified By (Delegate)	modifiedonbehalfby	Lookup	Standard	✓	Optional	✓
Modified On	modifiedon	Date and Time	Standard	✓	Optional	✓
Owner	ownerid	Owner	Standard	✓	Required	✓
Owning Business Unit	owningbusinessunit	Lookup	Standard	✓	Optional	
Owning Team	owningteam	Lookup	Standard	✓	Optional	
Owning User	owninguser	Lookup	Standard	✓	Optional	
Price	cr26c_price	Decimal Number	Custom	✓	Optional	✓
Product ID	cr26c_productid	Text	Custom	✓	Optional	✓
Record Created On	overriddencreatedon	Date Only	Standard	✓	Optional	✓
Status	statecode	Choice	Standard	✓	Required	✓
Status Reason	statuscode	Choice	Standard	✓	Optional	✓

3. Make a note of your schema names for the following columns. You will need these in this and the following tasks.

Asset

Asset Name

Display name	Name	Data type	Type	Customizable	Required	Searchable
Asset	cr26c_assetid	Unique Identifier	Standard	✓	Required	✓
Asset Category	cr26c_assetcategory	Lookup	Custom	✓	Optional	✓
Asset Name	cr26c_name	Text	Custom	✓	Required	✓
Created By	createdby	Lookup	Standard	✓	Optional	✓
Created By (Delegate)	createdonbehalfby	Lookup	Standard	✓	Optional	✓
Created On	createdon	Date and Time	Standard	✓	Optional	✓
Import Sequence Number	importsequencenumber	Whole Number	Standard	✓	Optional	✓
Modified By	modifiedby	Lookup	Standard	✓	Optional	✓
Modified By (Delegate)	modifiedonbehalfby	Lookup	Standard	✓	Optional	✓
Modified On	modifiedon	Date and Time	Standard	✓	Optional	✓
Owner	ownerid	Owner	Standard	✓	Required	✓
Owning Business Unit	owningbusinessunit	Lookup	Standard	✓	Optional	
Owning Team	owningteam	Lookup	Standard	✓	Optional	
Owning User	owninguser	Lookup	Standard	✓	Optional	
Price	cr26c_price	Decimal Number	Custom	✓	Optional	✓
Product ID	cr26c_productid	Text	Custom	✓	Optional	✓
Record Created On	overriddencreatedon	Date Only	Standard	✓	Optional	✓
Status	statecode	Choice	Standard	✓	Required	✓
Status Reason	statuscode	Choice	Standard	✓	Optional	✓

4. Navigate back to Power Virtual Agents using the icon on the Teams left hand navigation bar. Select the **Chatbots** tab, and click on the chatbot you have been working on to open it.

The screenshot shows the Microsoft Teams interface with the Power Virtual Agents app open. The 'Chatbots' tab is selected. In the 'My chatbots' section, there is one entry: 'Office Helper' (Owner: Alex Hunter, Team: Office - Admin). A red box highlights this entry.

Task 2: Create a new Topic

1. Navigate to Topics and click on **+ New Topic**.

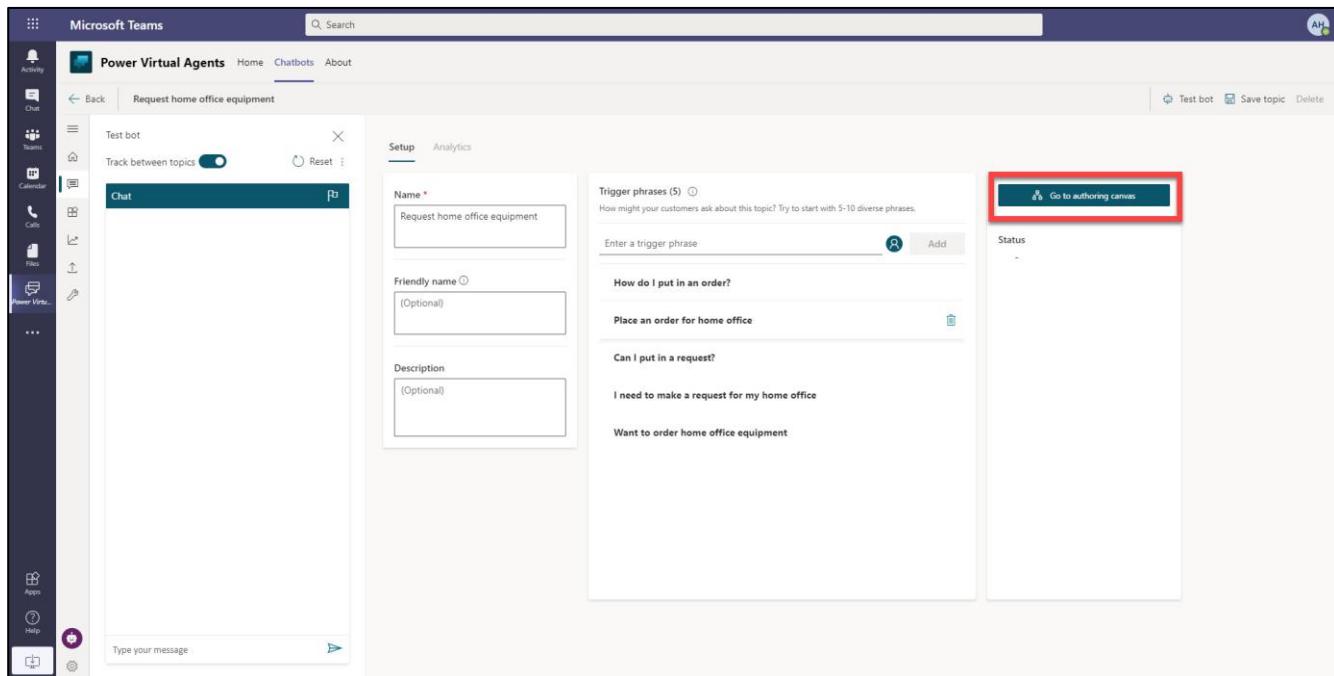
The screenshot shows the Microsoft Teams interface with the Power Virtual Agents app open. The 'Topics' tab is selected. On the left, there is a 'Test bot' sidebar with a 'Track between topics' toggle switch. A red box highlights the '+ New topic' button. The main area shows a table of existing topics:

Type	Name	Trigger phrases	Status	Errors	Editing	Modified by
Chat	Home office setup	(S) How do I get a desk for w...	On			Alex Hunter 20 minutes ago
Chat	Lesson 4 - A topic with a condition, variables an...	(S) What is the best product I -	Off			Alex Hunter 2 hours ago
Chat	Lesson 3 - A topic with a condition, variables an...	(S) Buy items	Off			Alex Hunter 2 hours ago
Chat	Lesson 2 - A simple topic with a condition and v...	(S) Are there any stores aroun...	Off			Alex Hunter 2 hours ago
Chat	Lesson 1 - A simple topic	(I) When are you closed	Off			Alex Hunter 2 hours ago
Greeting	Greeting	(S2) Good afternoon	Always on			Alex Hunter 2 hours ago
Escalate	Escalate	(I5) Talk to agent	Always on			01/15/2019
End of Conversation	End of Conversation	No trigger phrases	Always on			01/15/2019
Confirmed Success	Confirmed Success	No trigger phrases	Always on			01/15/2019
Confirmed Failure	Confirmed Failure	No trigger phrases	Always on			01/15/2019
Goodbye	Goodbye	(I6) Bye	Always on			01/15/2019
Start over	Start over	(I) start_over	Always on			01/15/2019
Thank you	Thank you	(I) thanks	Always on			01/15/2019

2. Give the topic a name by typing: **Request home office equipment** in the Name box. Then add the following trigger phrases by typing them in and clicking the Add button after each one, until they all appear in the list.

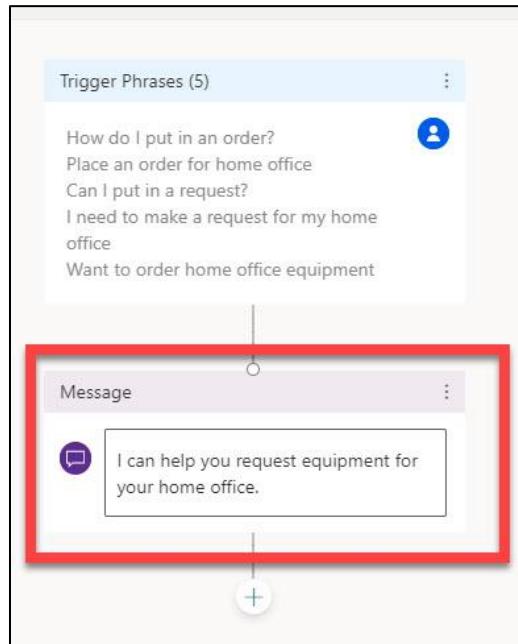
Want to order home office equipment
I need to make a request for my home office
Can I put in a request?
Place an order for home office
How do I put in an order?

3. Click on **Go to authoring canvas** to save your topic and go to the authoring canvas.



4. In the Message box, add the following text:

I can help you request equipment for your home office.

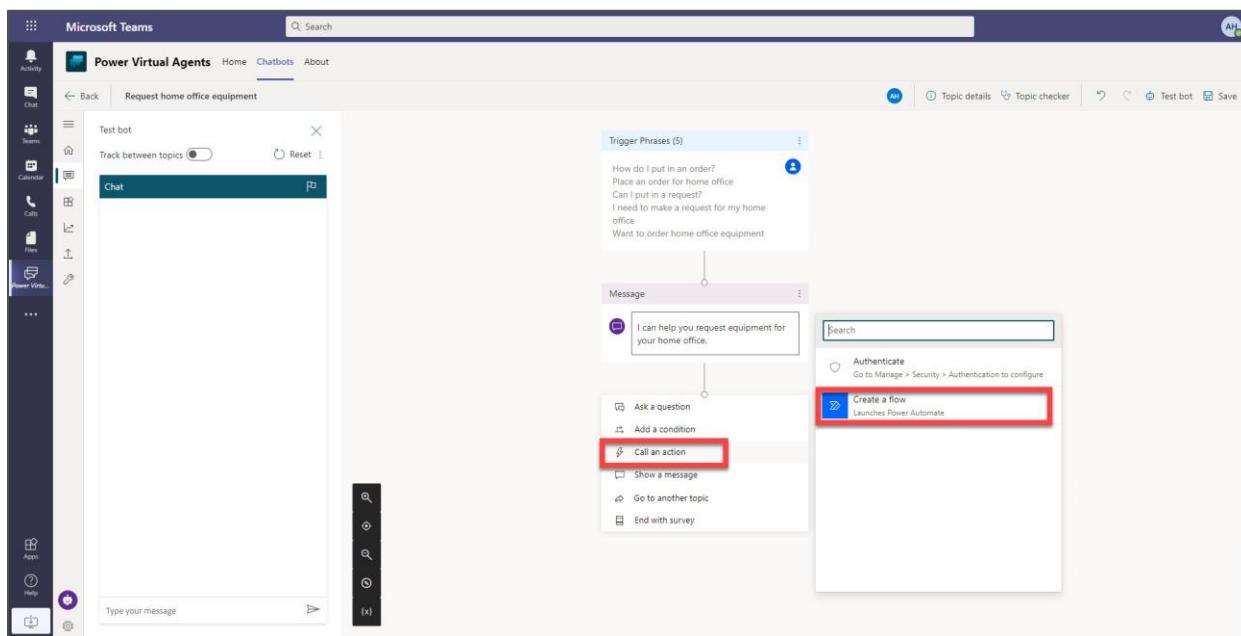


5. Save your bot before continuing to the next task, by clicking on the **Save** icon in the top right corner.

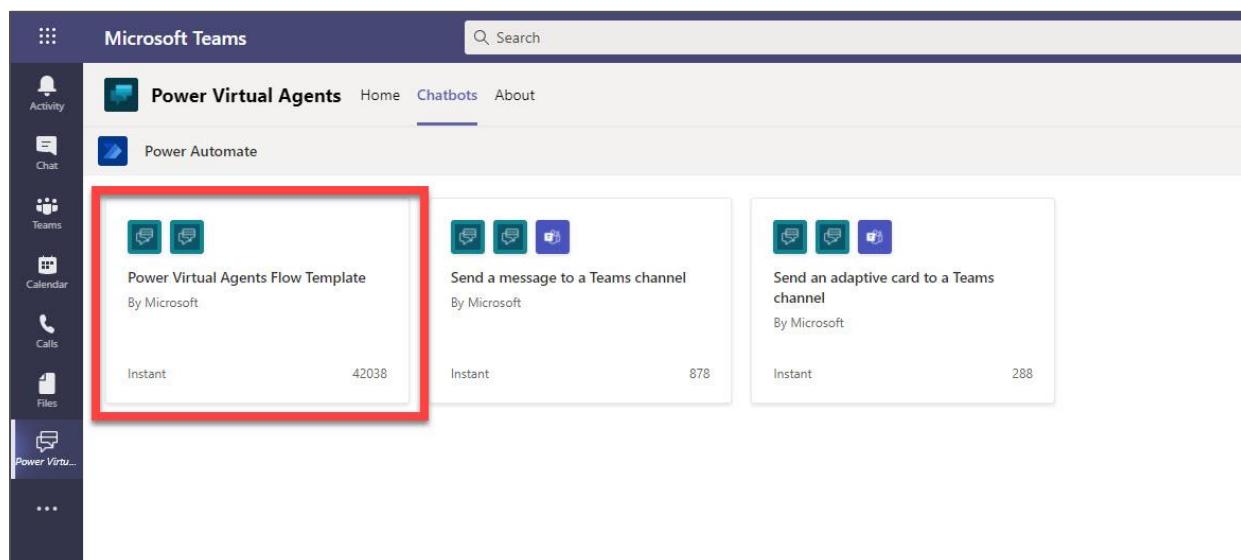
Task 3: Create a flow to get a row from a Dataverse for Teams table

Power Virtual Agents in Teams comes with a built-in experience for Power Automate, with templates to make it easy for you to get your chatbot to call an action. In this task we will use Power Automate to look up the first item in our Assets table in Dataverse for Teams, and return that as a suggestion for the user to request that item.

1. Add a node and select **Call an action**. Click on **Create a flow**. This will launch Power Automate inside Microsoft Teams.

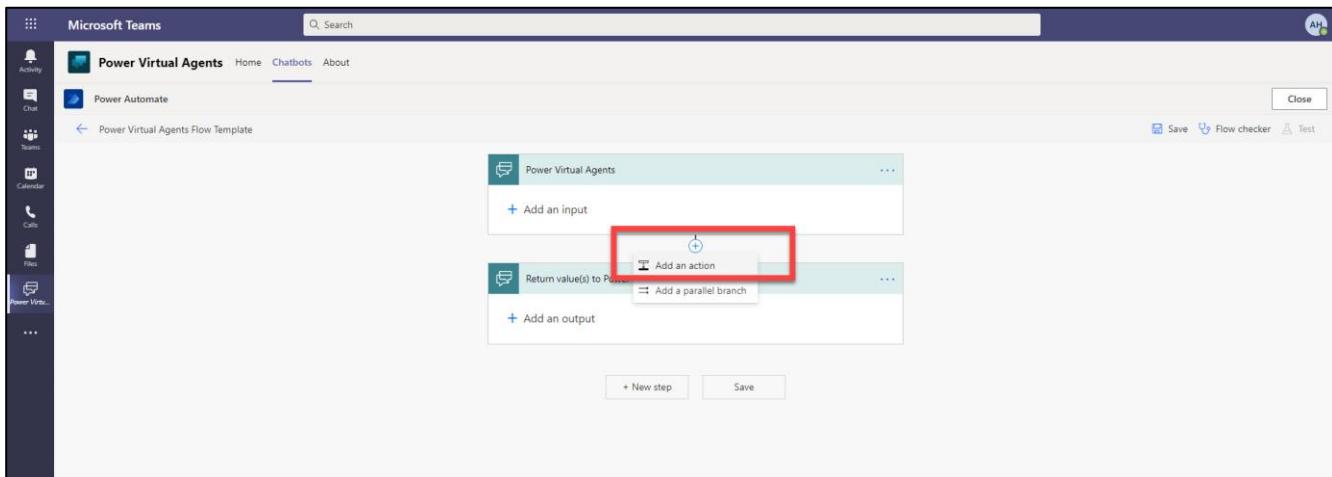


2. Select the **Power Virtual Agents Flow Template**. This template is designed to pass inputs from the chat to a flow, and then to provide outputs from the flow back to Power Virtual Agents.



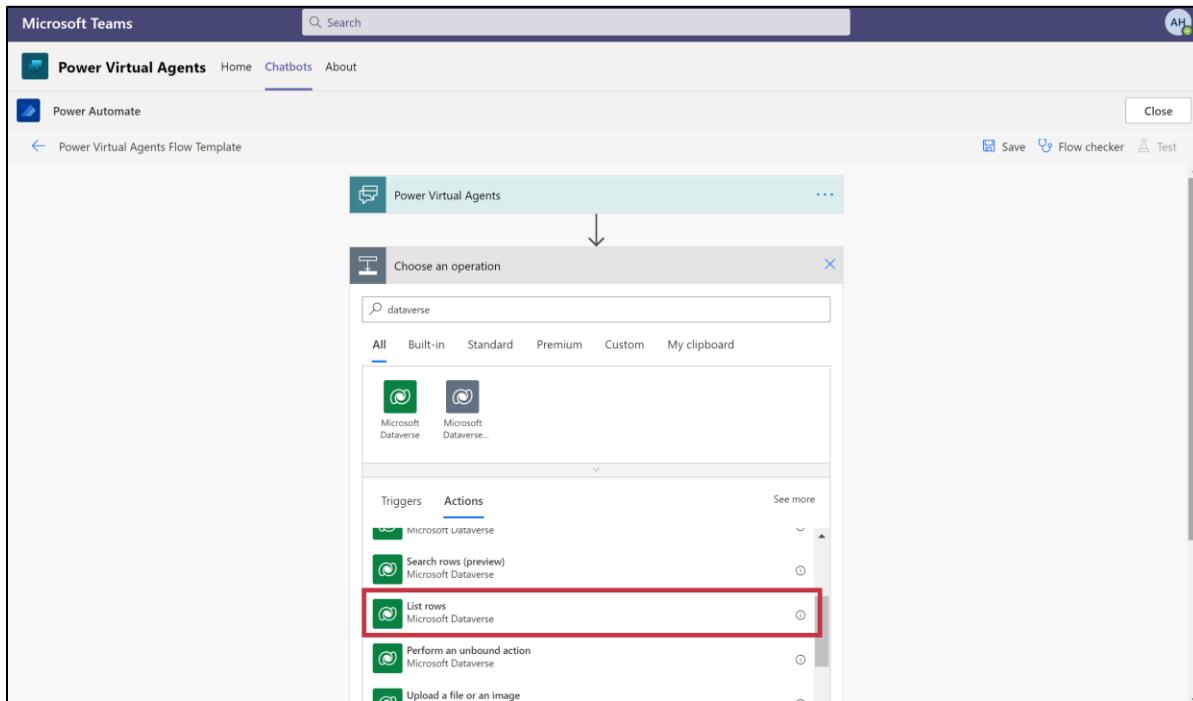
3. You may see a screen here confirming your flow connections. Click **Continue**.

4. You will now see the flow template, with the Power Virtual Agents input and output steps. We are not bringing any inputs from the chat into this flow, so we don't need to do anything here in that first stage. Add a new action between the input and output steps.

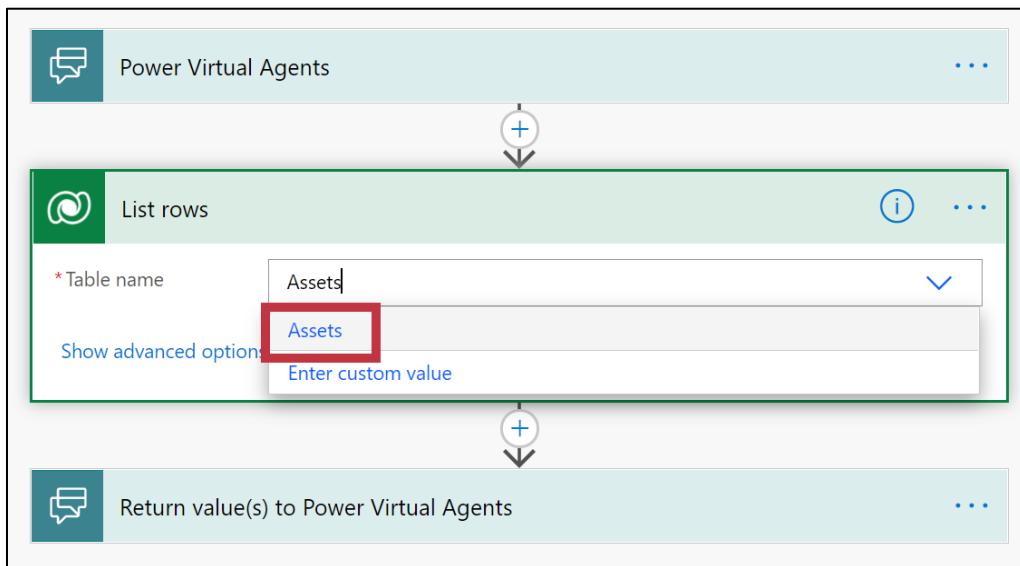


5. Find and select **Dataverse** and select the **List Rows** action.

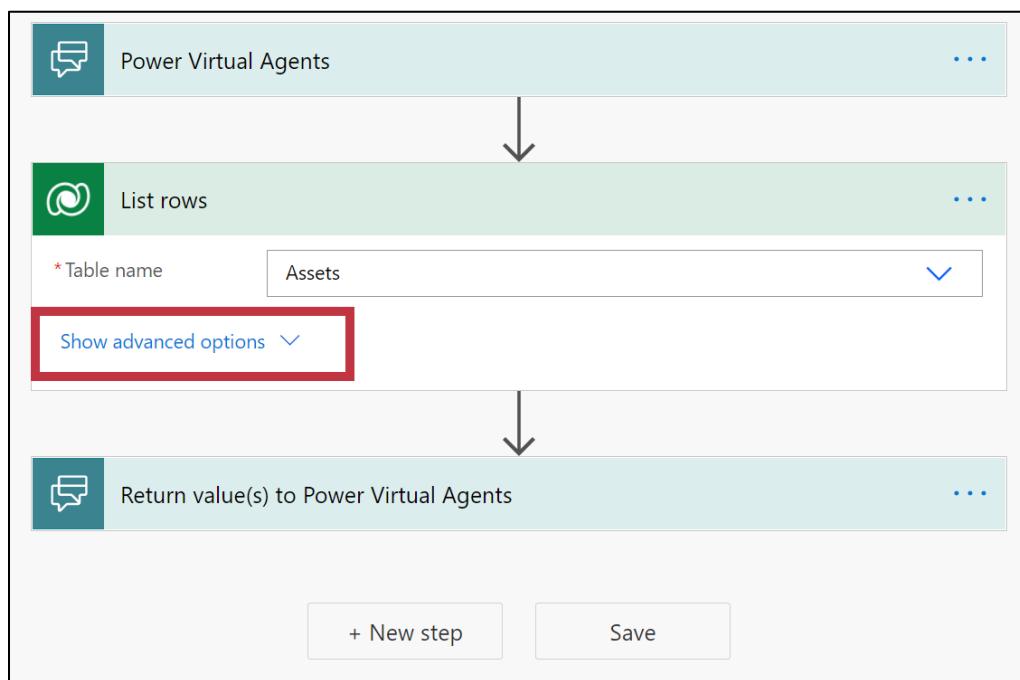
This connector allows your chatbot to connect to your Dataverse for Teams tables, with various actions, including creating, updating, deleting or retrieving records.



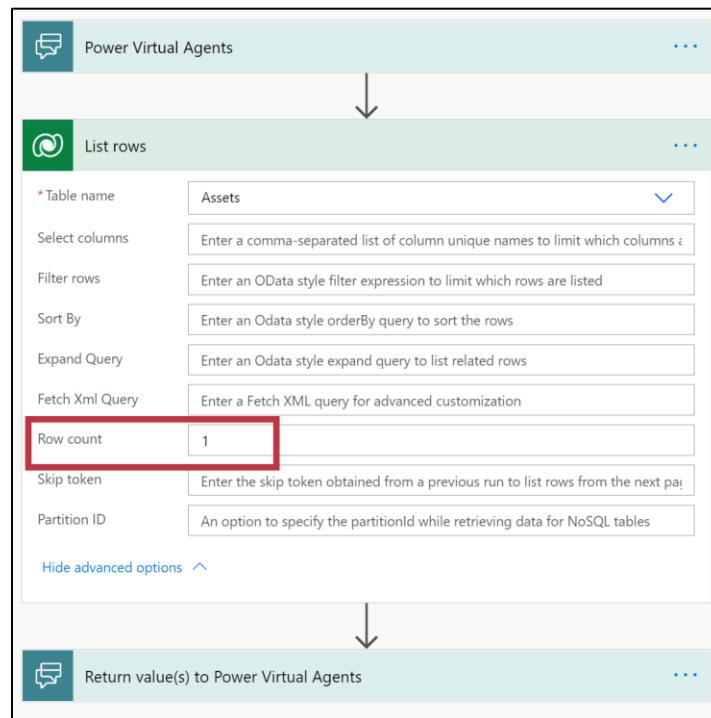
6. Wait while the connection is established. Now choose the Assets Table name from the dropdown list. You can type the word **Assets** in to the box to make it easy to find your table.



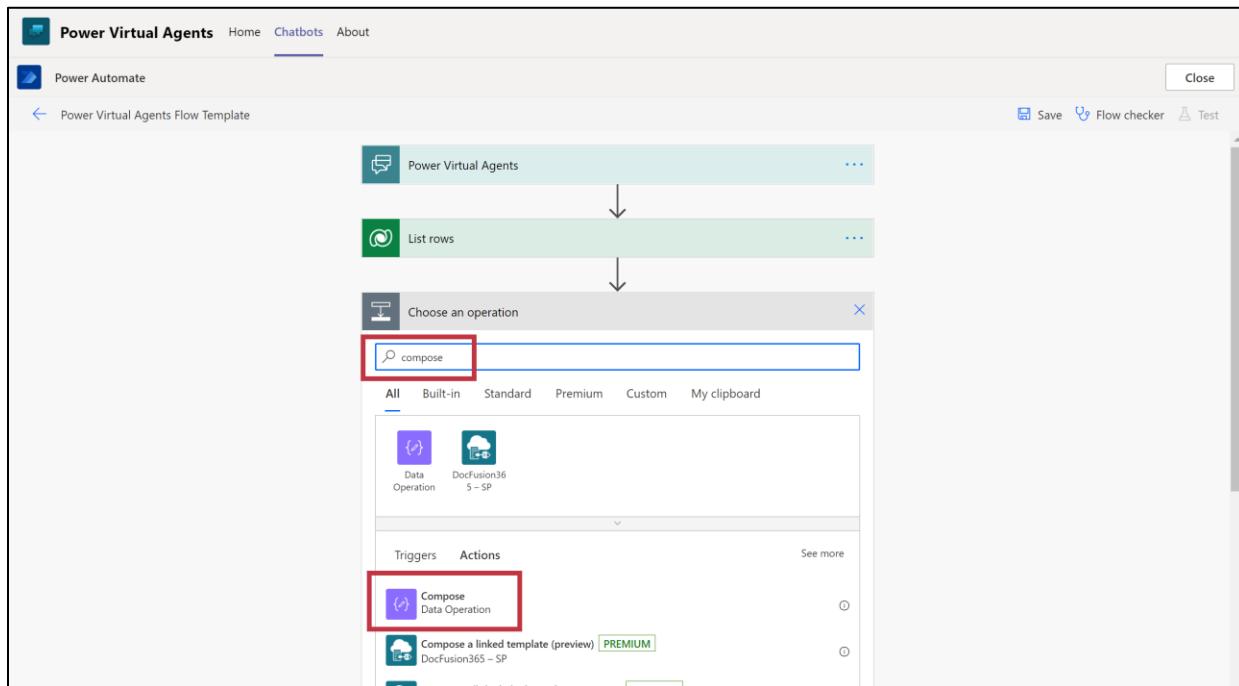
7. Now click **Show advanced options** to see the rest of this action step.



8. We just want to retrieve the first row from the table, so enter the number **1** in the **Row count** box.

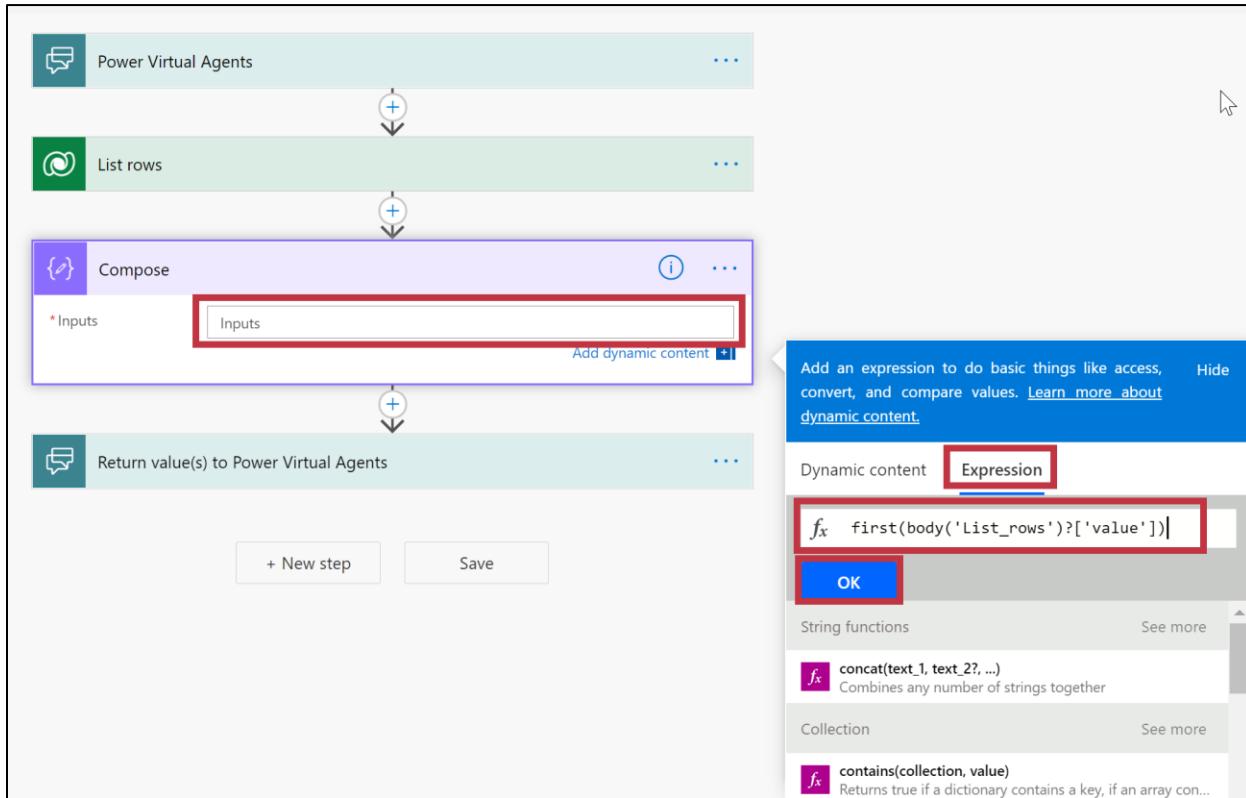


9. We are now going to compose a variable to store the information from the item we retrieved in the previous step, to make it easier to pass it back to the chatbot. Add an action underneath this one, search for **compose** and choose the **Compose (Data Operation)**.

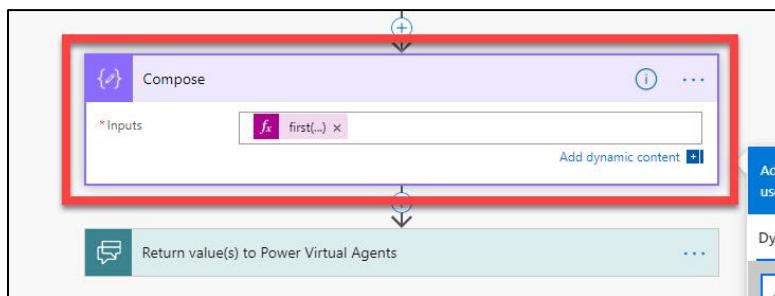


10. Click on the **Inputs** box, in the action step and then select the **Expression** tab in the Dynamic content box that pops up. Type the following expression into the function box and click **OK**. This expression gets the value of the item:

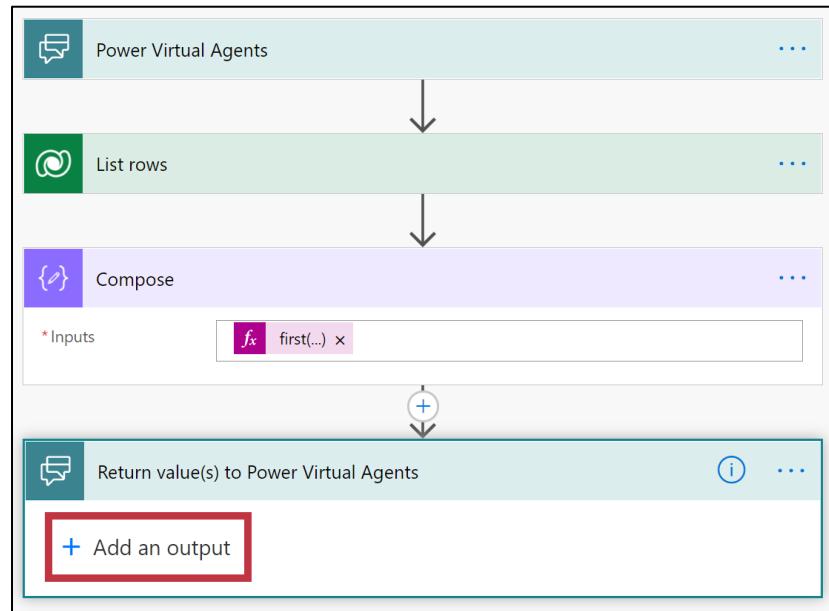
```
first(body('List_rows')?['value'])
```



11. You will now see a formula in the Inputs box.

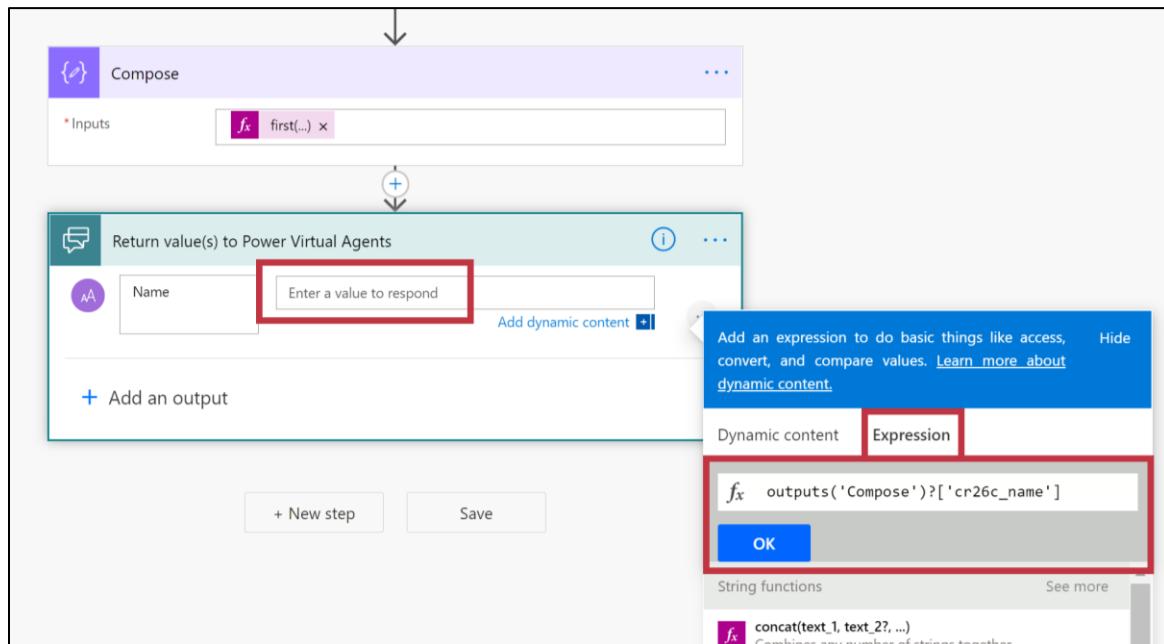


12. Now we will return the asset name and ID to the chatbot. Click on the **Return value(s) to Power Virtual Agents** step, and select **+Add an output**.



13. Choose **Text** and enter the title: **Name**. In the **Enter a value to respond** box go to the Expression tab on the Dynamic content box, and enter the following formula – substitute the highlighted schema name here for your own schema name for the **Asset Name** column in your table that you noted earlier. Yours will have a different prefix. Then click **OK**.

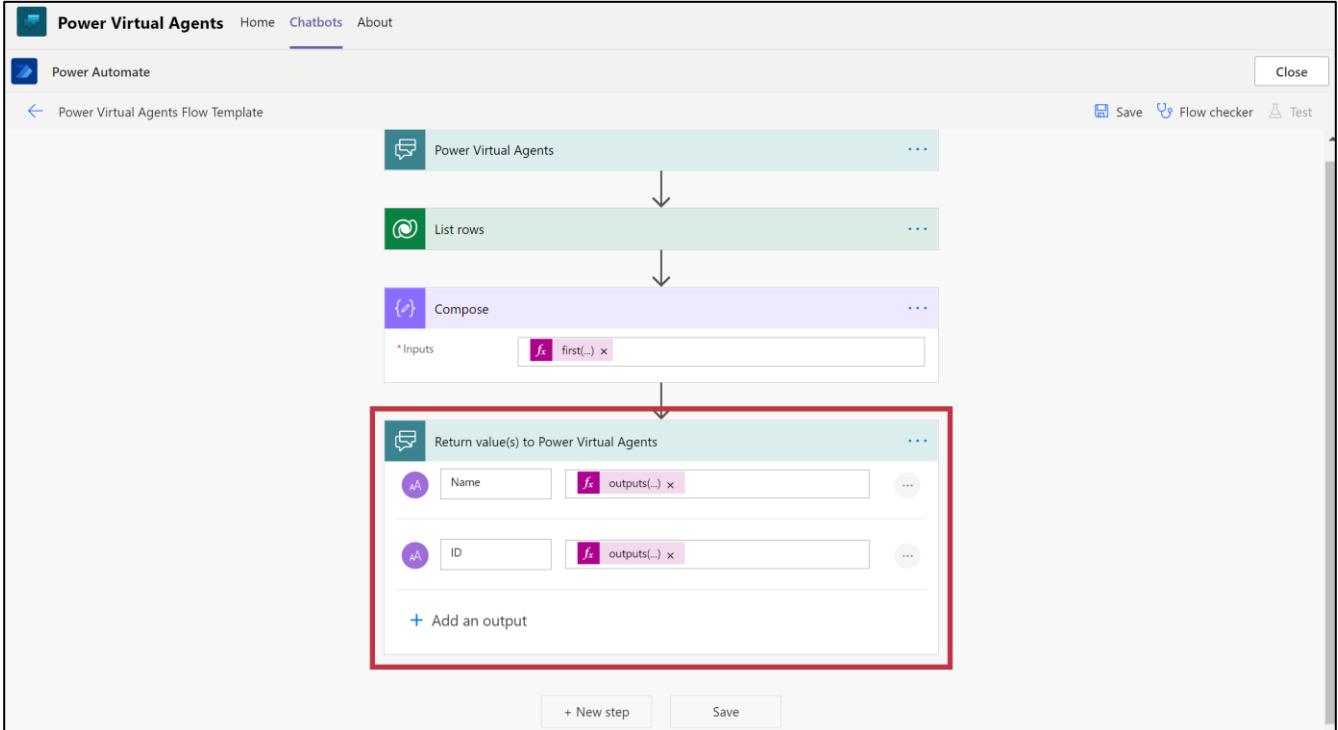
```
outputs('Compose')?['cr26c_name']
```



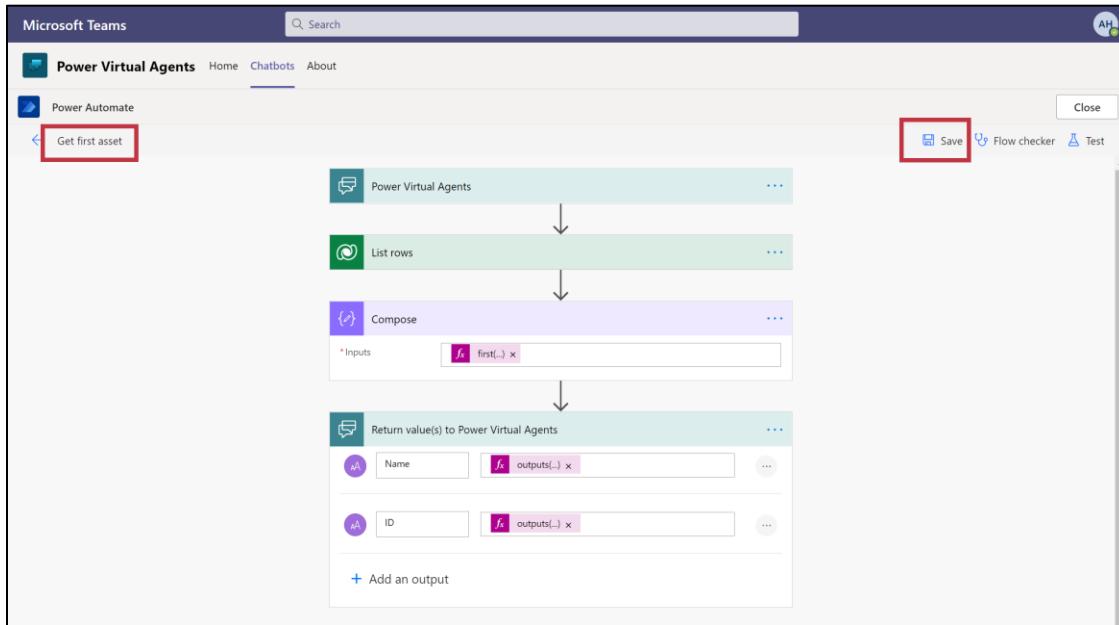
14. Repeat this step to add the ID to the outputs. Click **+Add an output** and choose **Text**. Enter the Title: **ID**. In the **Enter a value to respond** box go to the Expression tab on the Dynamic content box, and enter the following formula – substitute the highlighted schema name here for your own schema name for the **Asset** column in your table that you noted earlier. Yours will have a different prefix. Then click **OK**.

```
outputs('Compose')?['cr26c_assetid']
```

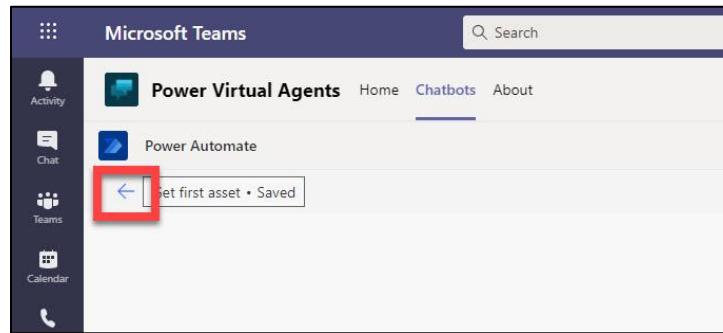
15. You will now see formulas in both output boxes.



16. Change the name of your flow at the top left of the screen to: **Get first asset** and then and save by clicking on the **save** button.

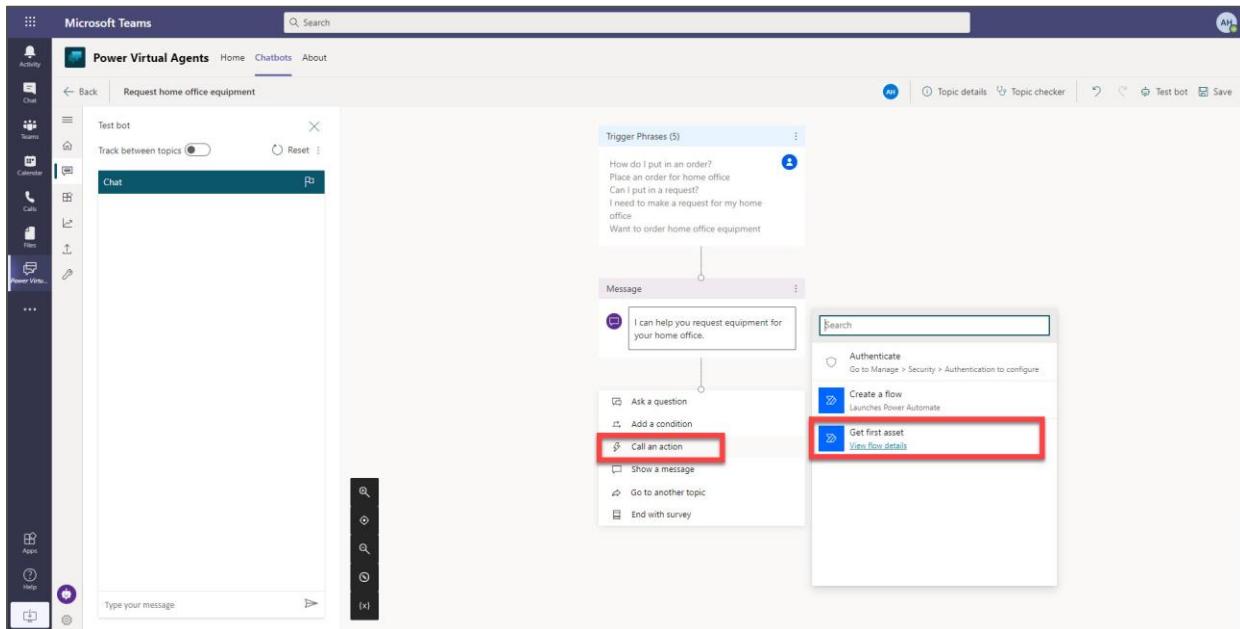


17. Once your flow is saved, click on the blue arrow next to the name of your flow to return to Power Virtual Agents. In the next task you will connect this flow to your chatbot.

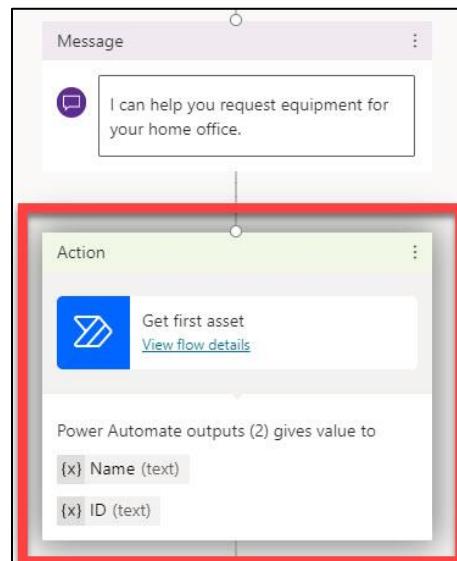


Task 4: Connect your flow with outputs to your chatbot

- Under your message node select **Call an action**, and select the flow you just created in the previous task. Make sure you click on the Name of the flow, not the blue hyperlink to show flow details (that will take you back to editing the flow).



- Your flow is now connected to your chatbot.

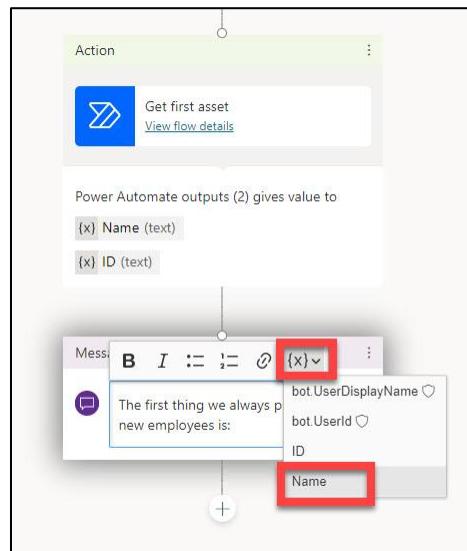


Task 5: Create a flow to submit a request for an asset

1. Add a new **message node** underneath the flow action node, and enter the following:

The first thing we always provide for new employees is:

2. Then click on the variable button and select the **Name** variable. This brings in the name of the first item from your Asset table that the flow retrieved in the previous step.



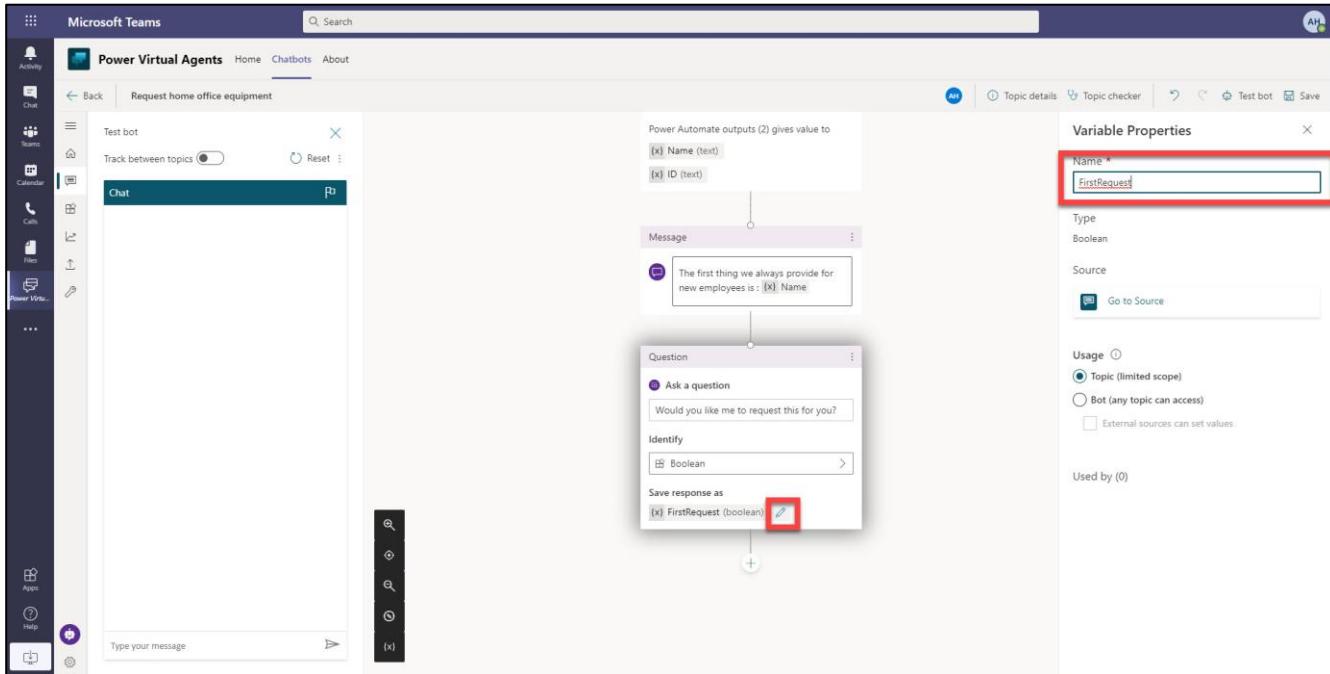
3. Add a node to **Ask a question**. Enter this text into the question box:

Would you like me to request this for you?

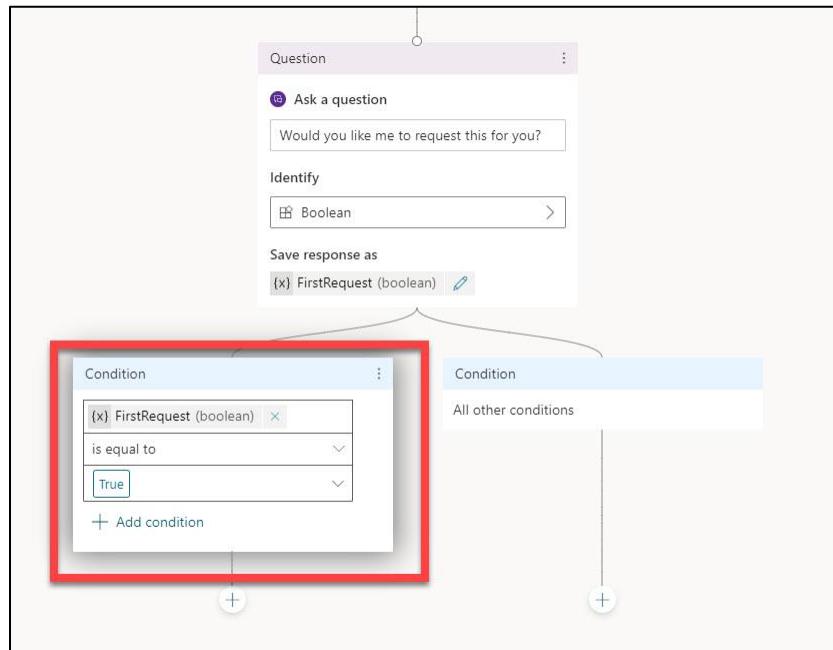
4. Then in the Identify box, open the options and select **Boolean**.

A screenshot of the Microsoft Teams Power Virtual Agents interface. On the left, there is a sidebar with various icons and a main chat window titled 'Test bot' with the message 'Request home office equipment'. In the center, there is a 'Chat' card with the text 'The first thing we always provide for new employees is: : (x) Name'. Below this is a 'Question' node with the text 'Would you like me to request this for you?'. Underneath the question node is an 'Identify' section with a dropdown menu. The 'Multiple choice options' option is selected and highlighted with a red box. In the dropdown menu, the 'Boolean' option is also highlighted with a red box. Other options listed in the menu include 'Multiple choice options', 'User's entire response', 'Age', 'Color', and 'City'.

5. Edit the name of the variable to **FirstRequest**. Close the Variable Properties pane by clicking on the X in the top right corner.

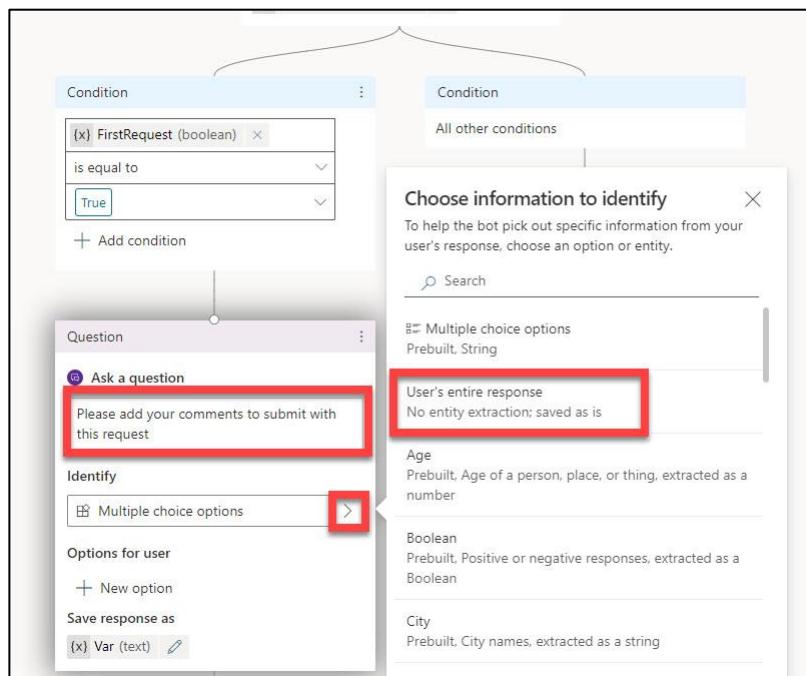


6. Add a new node and select **Add a Condition**. In the left branch of the Condition, select the **FirstRequest** variable from the dropdown list, and select **True** from the second dropdown list.

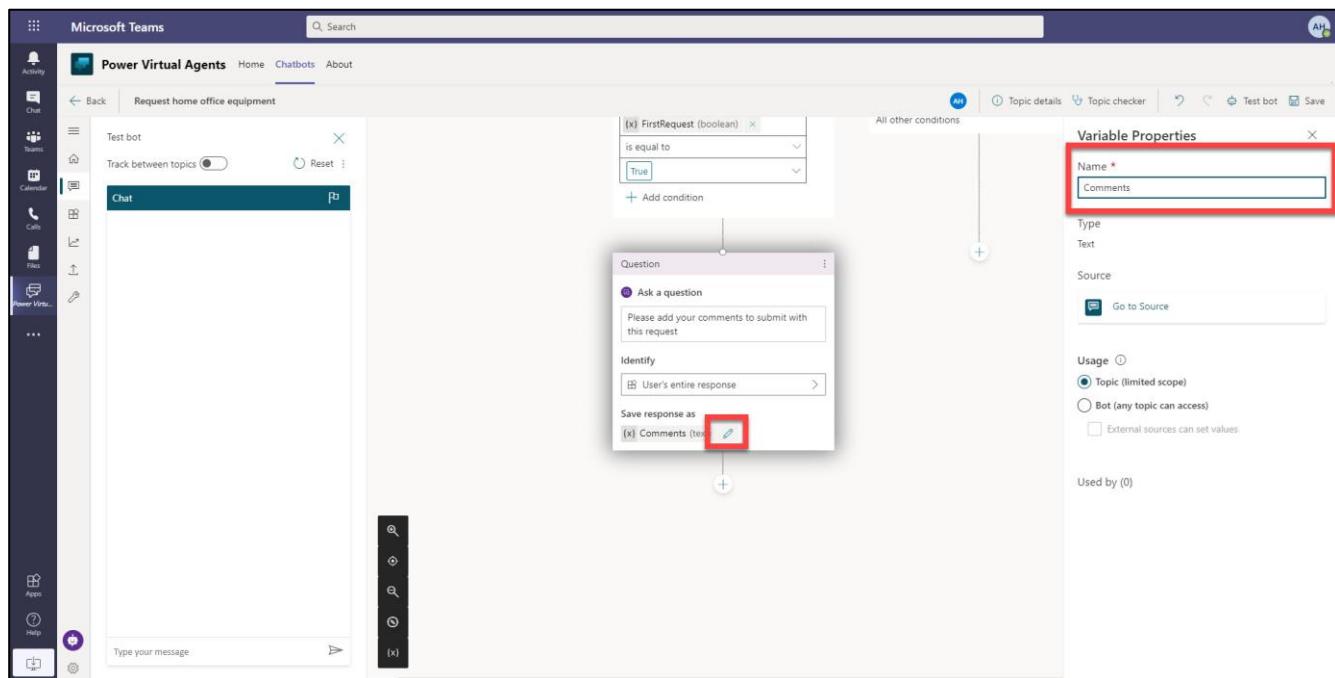


7. Add another node underneath and select **Ask a question**. Enter the question:
Please add your comments to submit with this request

8. From the Identify options, select **User's Entire Response**.

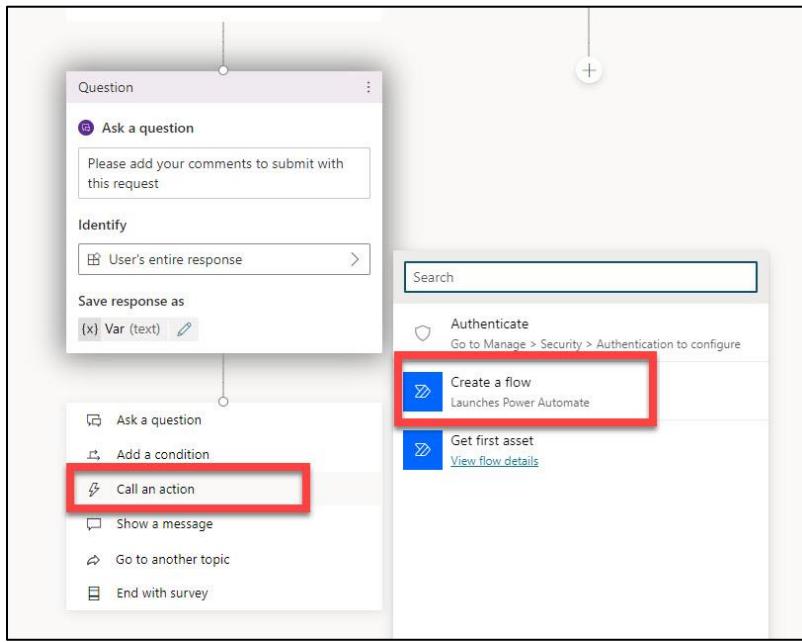


9. Change the name of the variable to **Comments**, and close the **Variable Properties** pane.

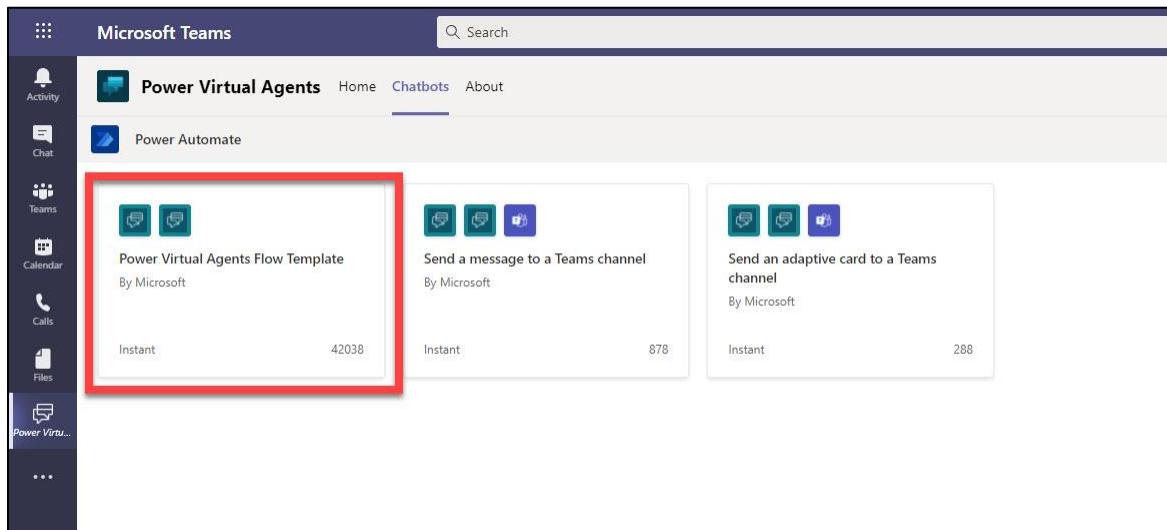


10. **Save** your bot using the save button in the top right corner.

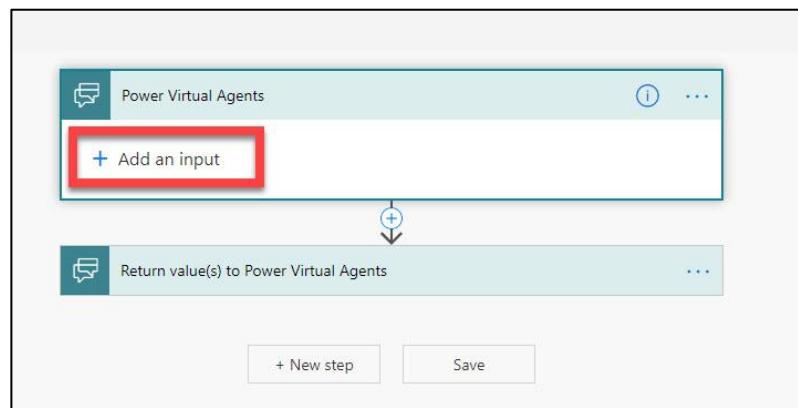
11. Now create another flow, this time to request the asset on behalf of the user. Add a new node, select **Call an Action**, and then **Create a Flow**.



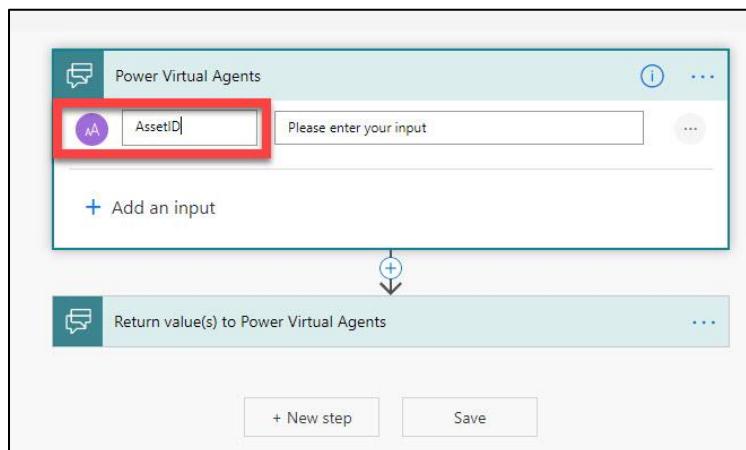
12. Select the **Power Virtual Agents Flow Template**.



13. In the previous flow, we got the ID of the asset the person is now requesting. We need to pass that into this flow as an input from the chatbot. Open the first Power Virtual Agents step in the flow by clicking on it, and select **+ Add an input**.



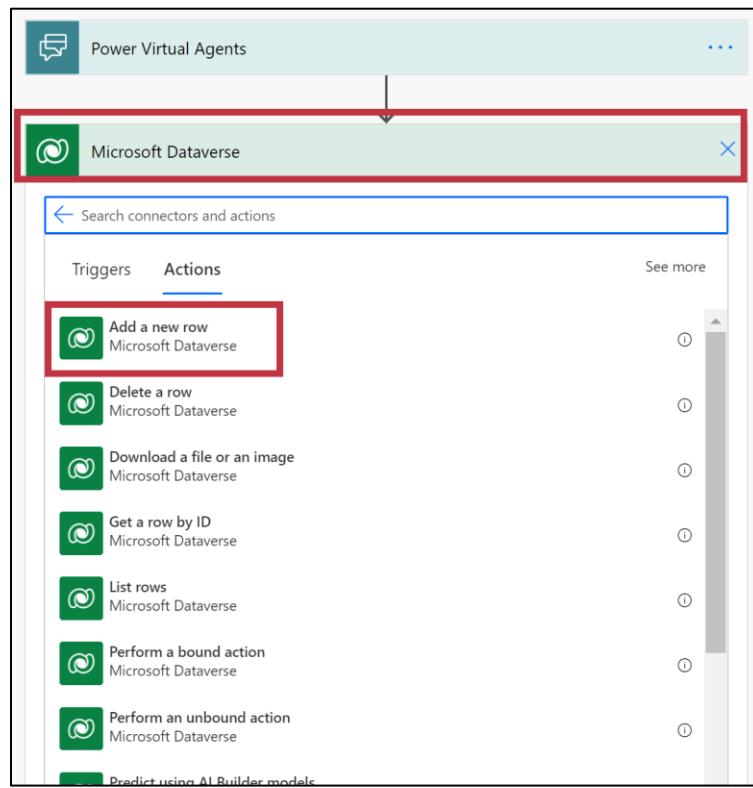
14. Select Text and then in the Input name, enter **AssetID**.



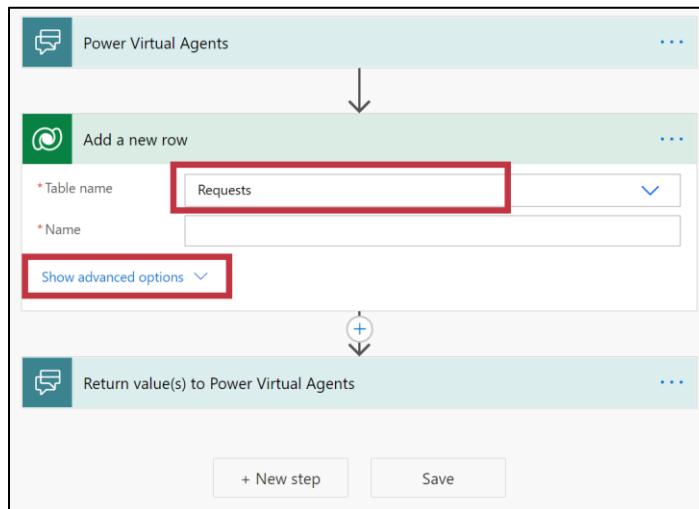
15. Repeat this step to add another input for the comments provided by the user. Click **+Add an input**, select Text and in the Input name enter: **Comments**.



16. Add an action under this input step. Find and select the **Dataverse** connector, and select the **Add a new row** action.



17. This flow is going to submit a request for the user. Select **Requests** as the Table name from the dropdown list and click on **Show advanced options**.



18. In the **Name** box, enter **New employee home office request**. Click in the Comment box, and select **Comments** from the Dynamic content box.

The screenshot shows the 'Add a new row' interface for the 'Requests' table. The 'Asset (Assets)' field is selected and highlighted with a red box. A dynamic content modal is open, listing categories such as Power Virtual Agents, AssetID, and Comments. The 'Comments' category is also highlighted with a red box.

19. Click in the the Asset (Asset) box, and enter the following expression (replace the highlighted section with the same prefix you found in your own schema names earlier). *When we create the request and fill in the asset, we are connecting to the asset table as well, with the lookup column. The flow needs us to define the name of that connected table when creating this record.*

`cr26c_assets()`

The screenshot shows the 'Add a new row' interface for the 'Requests' table. The 'Asset (Assets)' field contains the expression 'cr26c_assets()' and is highlighted with a red box.

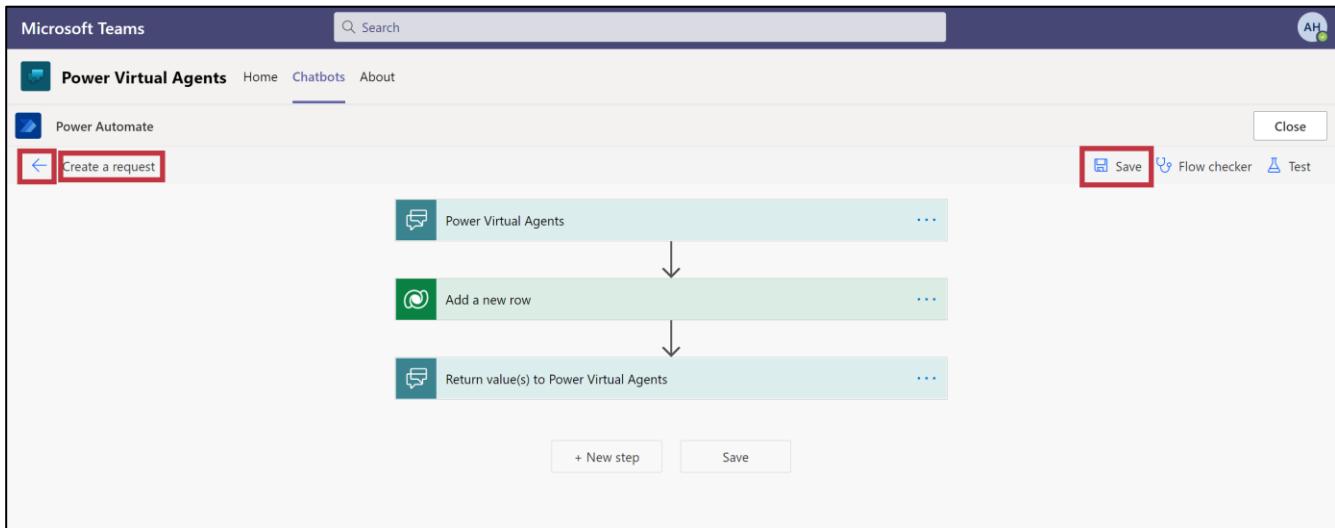
20. Click in between the brackets in this formula and then select the **AssetID** from the Dynamic content box.

The screenshot shows the 'Add a new row' step in Power Automate. The 'Asset (Assets)' field has the value 'cr26c_assets()' highlighted with a red box. To the right, a dynamic content box is open, showing the 'Power Virtual Agents' section with 'AssetID' selected, also highlighted with a red box.

21. Your Add a new row step should now look like this. Make sure the AssetID Dynamic content is between the brackets in your expression.

The screenshot shows the 'Add a new row' step in Power Automate. The 'Asset (Asset)' field has the value 'cr26c_assets(AssetID)' with a red box around the 'AssetID' placeholder.

22. Change the name of your flow to: **Create a request**. Save your flow and then use the back arrow next to the name to return to Power Virtual Agents.



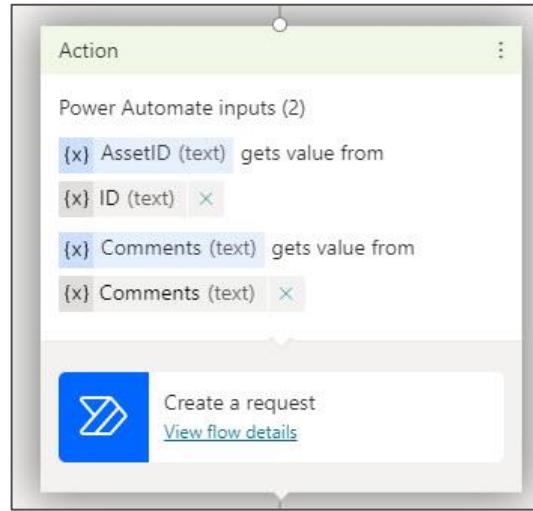
Task 6: Connect your flow with inputs to your chatbot

1. Scroll down to the question where you asked the user to add comments. Add a new node, select **Call an Action**, and select the flow you just created.

2. Now you need to map the variables you created in Power Virtual Agents to the inputs from the flow. Select your variables from the dropdowns as follows

AssetID (text) gets value from: **ID**

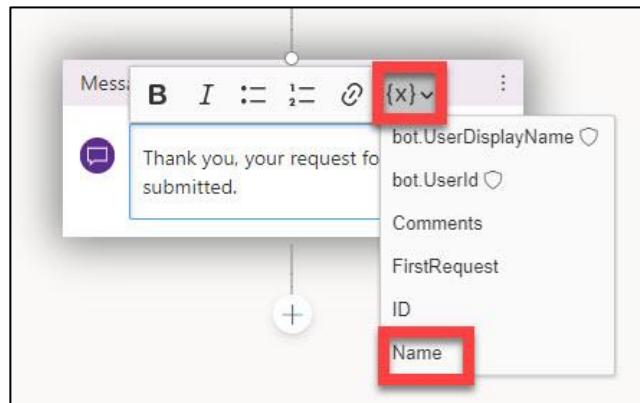
Comments (text) gets value from: **Comments**



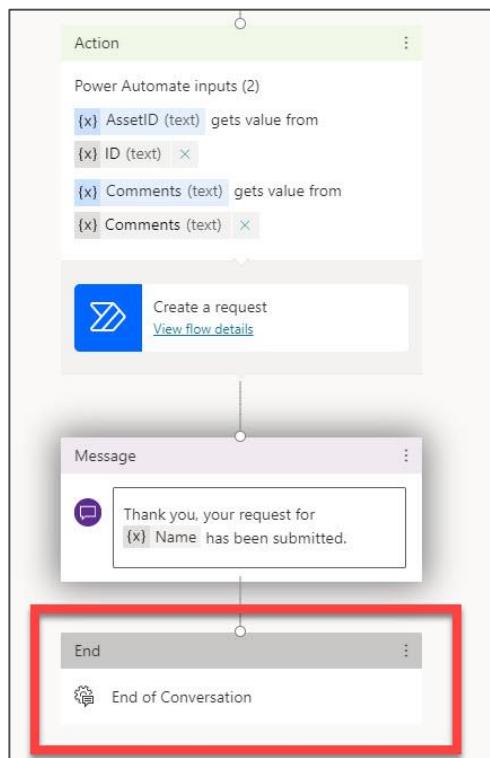
3. Add a confirmation message so that the user can see that their request has been sent. Add a node underneath and select **Show a message**. Type the following in the message box.

Thank you, your request for has been submitted.

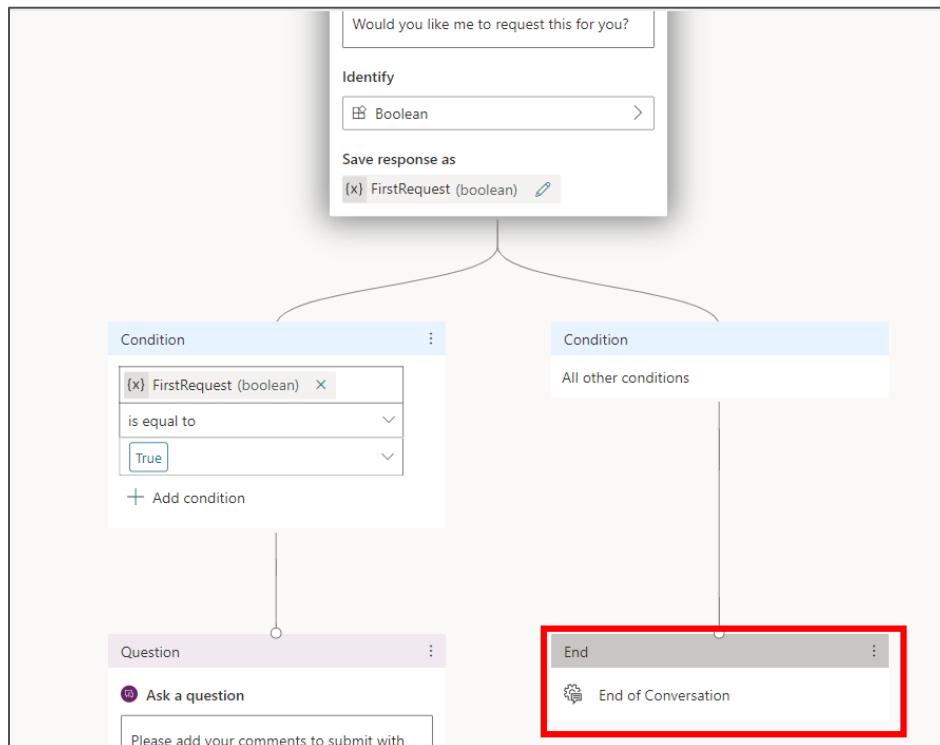
4. Click after the word "for" and select the Name variable.



5. Add another node underneath to end the conversation with a survey.



6. Scroll up to the question node for **Would you like me to request this for you?**. Under the condition branch for All other conditions, add an **End with survey** node.



7. Save your bot by clicking on the **Save** icon.

Task 7: Test your bot

1. Switch on the **Track between topics** toggle at the top of the Test bot pane. Type one of your trigger phrases (e.g. I need to make a request for my home office) in the test bot pane. The bot should return a suggested item which is the first item in your asset table.

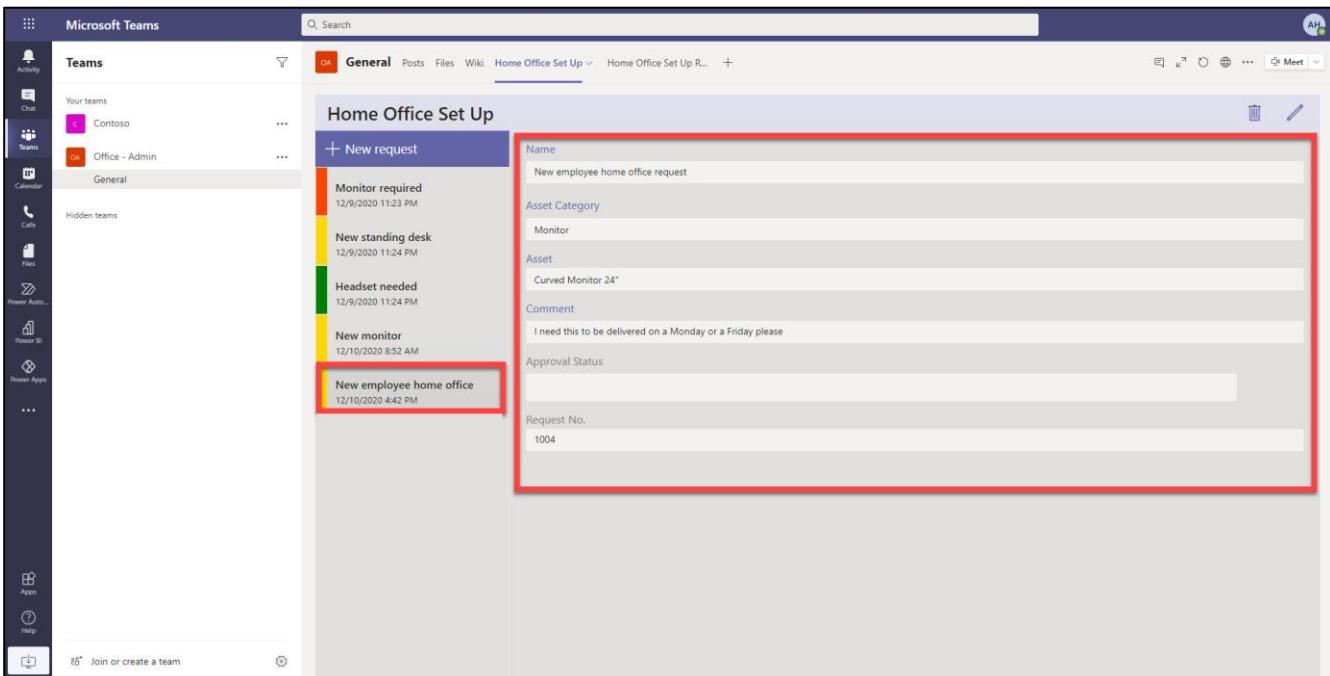
The screenshot shows a Microsoft Teams window with the 'Power Virtual Agents' app open. In the chat pane, a user message 'I need to make a request for my home office' is highlighted with a red box. The bot's response 'The first thing we always provide for new employees is : Curved Monitor 24'' is also highlighted with a red box. On the right, the logic editor shows a 'Message' block followed by a 'Question' block. The 'Question' block has an 'Ask a question' field with the text 'Would you like me to request this for you?'. The 'Identify' field is set to 'Boolean'. The 'Save response as' field is '(x) FirstRequest (boolean)'. Below the question, a condition is defined: '(x) FirstRequest (boolean)' is equal to 'True'. There are two 'Condition' blocks: 'All other conditions' and 'Condition' (which is currently empty).

2. Continue the conversation with the chatbot by selecting **Yes** and then entering your comments:
I need this to be delivered on a Monday or a Friday please.

3. You should then see a confirmation message with the name of the asset, and the End of Conversation topic.

The screenshot shows the continuation of the conversation. The user message 'I need this to be delivered on a Monday or a Friday please' is highlighted with a red box. The bot's response 'Thank you, your request for Curved Monitor 24'' has been submitted.' is also highlighted with a red box. On the right, the logic editor shows a 'Trigger-Phrases' block followed by a 'Question' block. The 'Question' block asks 'Did that answer your question?' and identifies the response as a Boolean value. There are three conditions: one for 'True' (labeled 'Confirmed Success') and one for 'False' (labeled 'Confirmed Failure'). An 'All other conditions' path is also present.

4. Now let's check that the request has been submitted. Click on the Teams icon in the left-hand navigation bar, and go to the team and channel where you created your app in Lab 1. You should see the request you just submitted in the app.



The screenshot shows the Microsoft Teams interface. On the left, the navigation bar includes icons for Activity, Chat, Teams, Calendar, Calls, Files, Power Automate, Power BI, and Power Apps. The Teams section shows 'Your teams' with 'Contoso' and 'Office - Admin' listed, and 'Hidden teams'. The main area shows the 'General' tab selected in the 'Home Office Set Up' channel. A red box highlights the 'New request' list, which contains four items:

- Monitor required (12/9/2020 11:23 PM)
- New standing desk (12/9/2020 11:24 PM)
- Headset needed (12/9/2020 11:24 PM)
- New monitor (12/10/2020 8:52 AM)

A fifth item, 'New employee home office', is also listed and highlighted with a red box. The right side of the screen shows the details for this request, including fields for Name, Asset Category, Asset, Comment, Approval Status, and Request No., all of which are currently empty or have placeholder text.

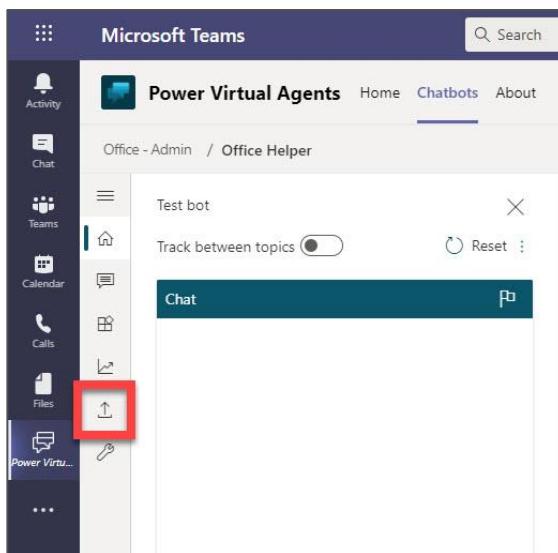
Congratulations! You have just built a bot which can answer questions, retrieve an asset from your Dataverse for Teams database, and submit a request on behalf of the user.

In the next exercise we will publish and share it and use it in Teams chat.

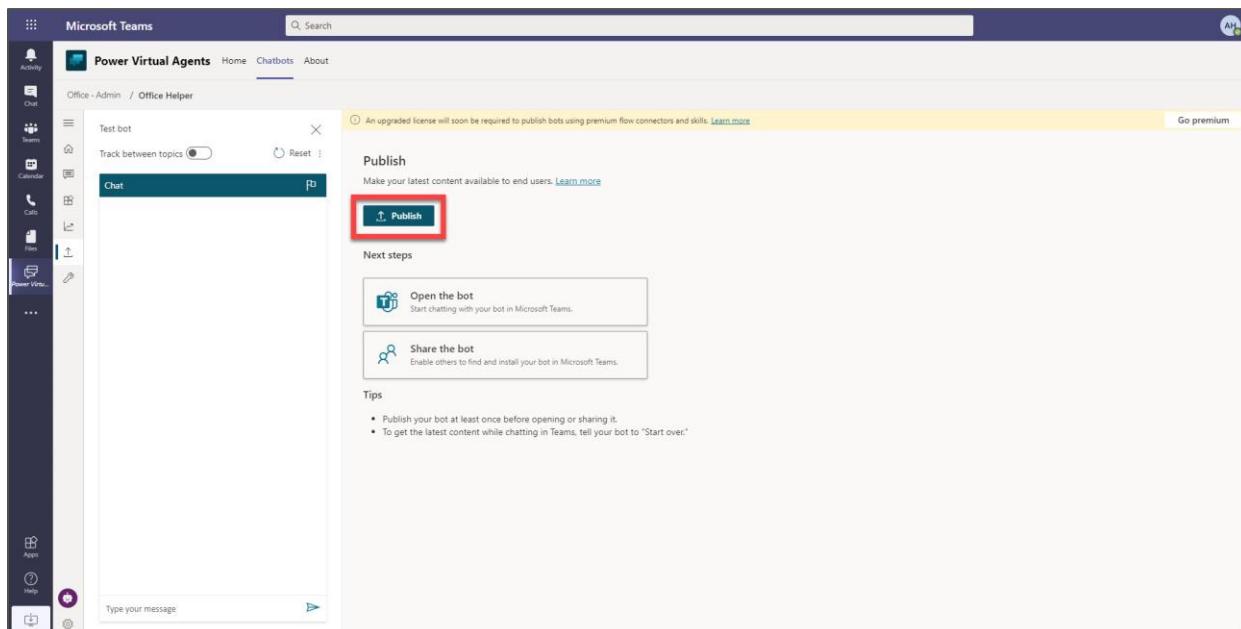
Exercise 5: Publishing and sharing your chatbot

Task 1: Publish your bot and open it in Teams

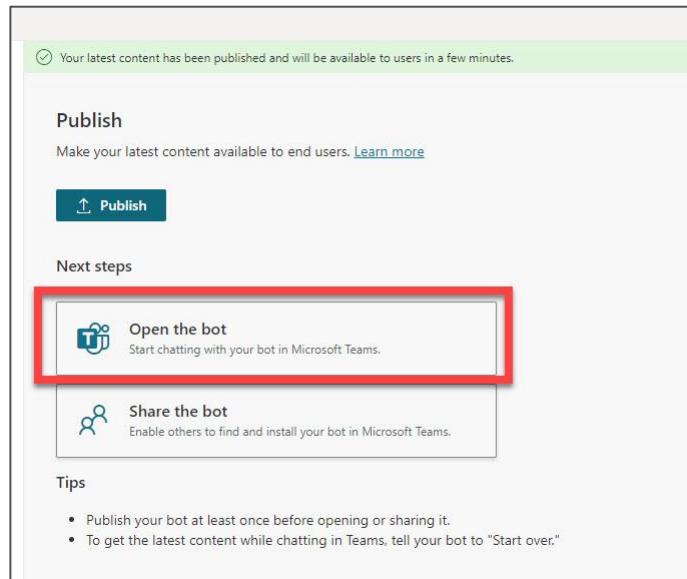
- Now that you have finished editing and testing your bot, you are ready to publish it and test or use it yourself in Teams. Return to your Power Virtual Agents app via the left-hand navigation bar and open your chatbot for editing. Click on the **Publish** icon  in the left-hand Power Virtual Agents navigation menu.



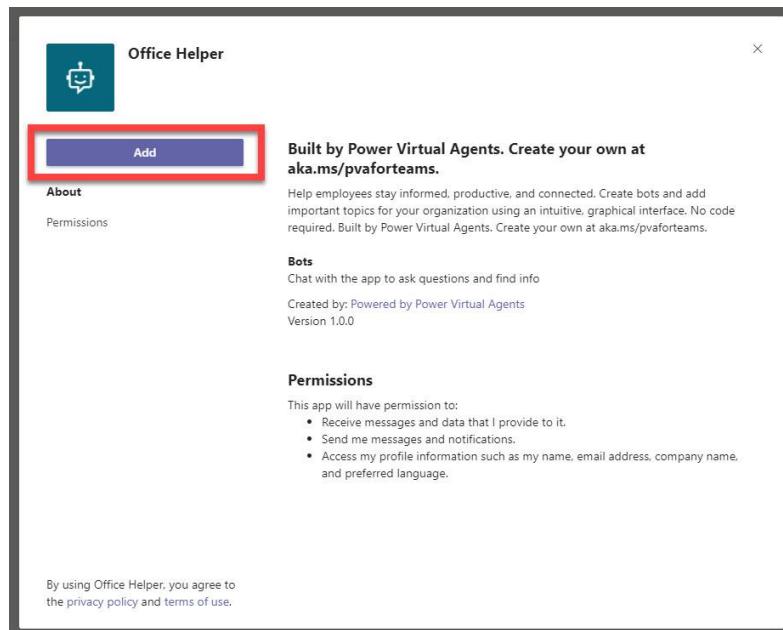
- Click on the **Publish** button (and then confirm by clicking on the Publish button in the pop up prompt asking if you want to Publish latest content).



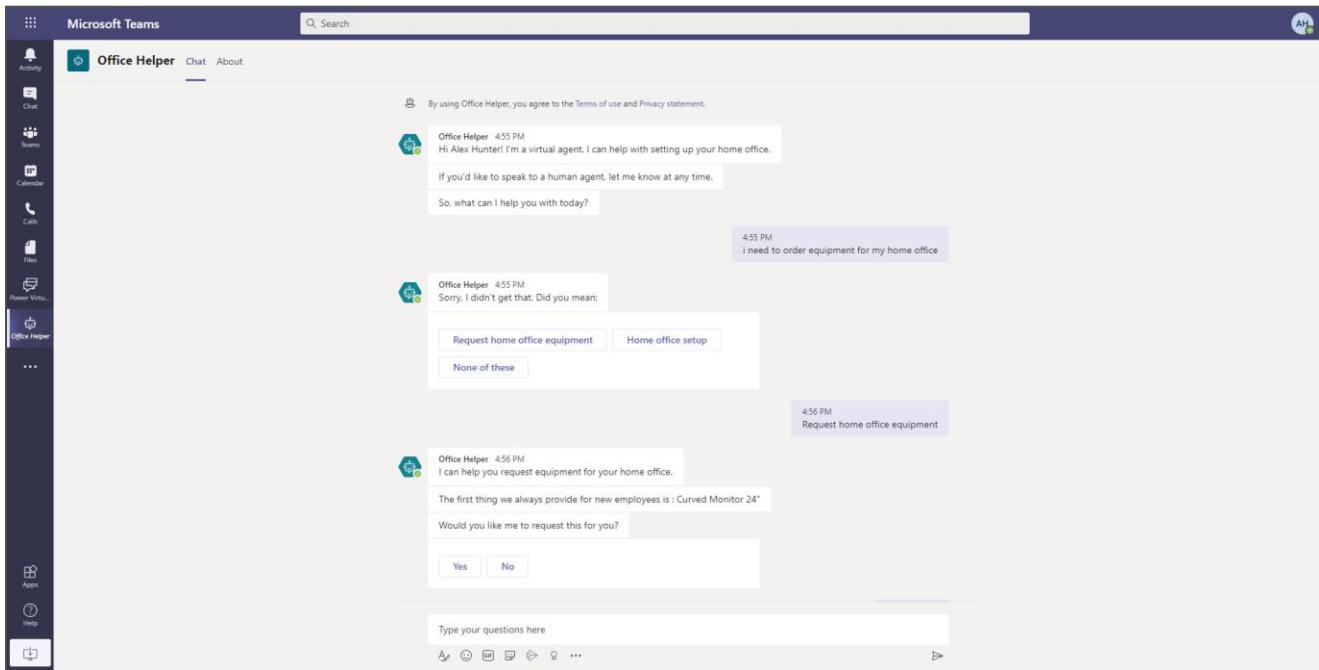
3. When your bot is published, you will see a confirmation message, and the **Open the bot option** will be active. Click on this option to open your bot in the Teams chat.



4. Click on **Add** to add the bot you just made to the Teams chat. At this stage you will be the only one who can see and use it. This is a useful step to do a final test before sharing with others.



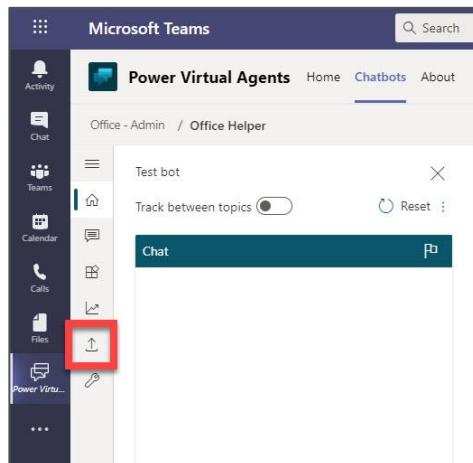
- Your bot will start the conversation (note it will address you by your logged in username). Test the conversation with the bot in the Teams chat.



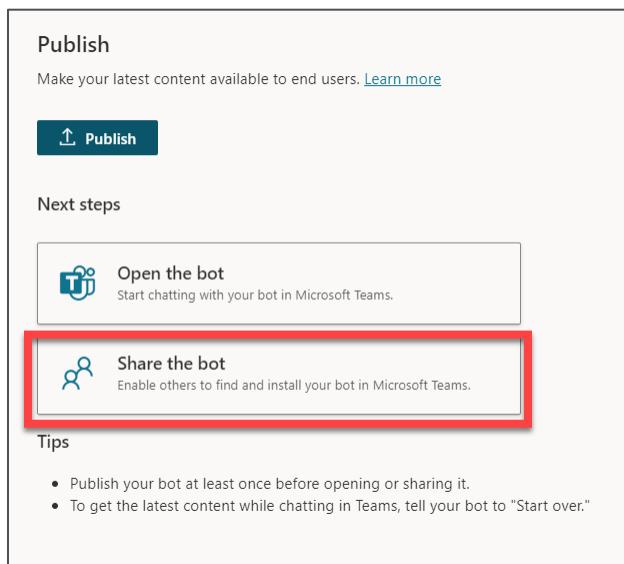
- If you run another test to submit a request, you can go and see again that your request is showing in your app in your Team.
- Now let's share the bot with others. Return to your chatbot in Power Virtual Agents app via the left-hand navigation menu.

Task 2: Share your chatbot with other members of your team

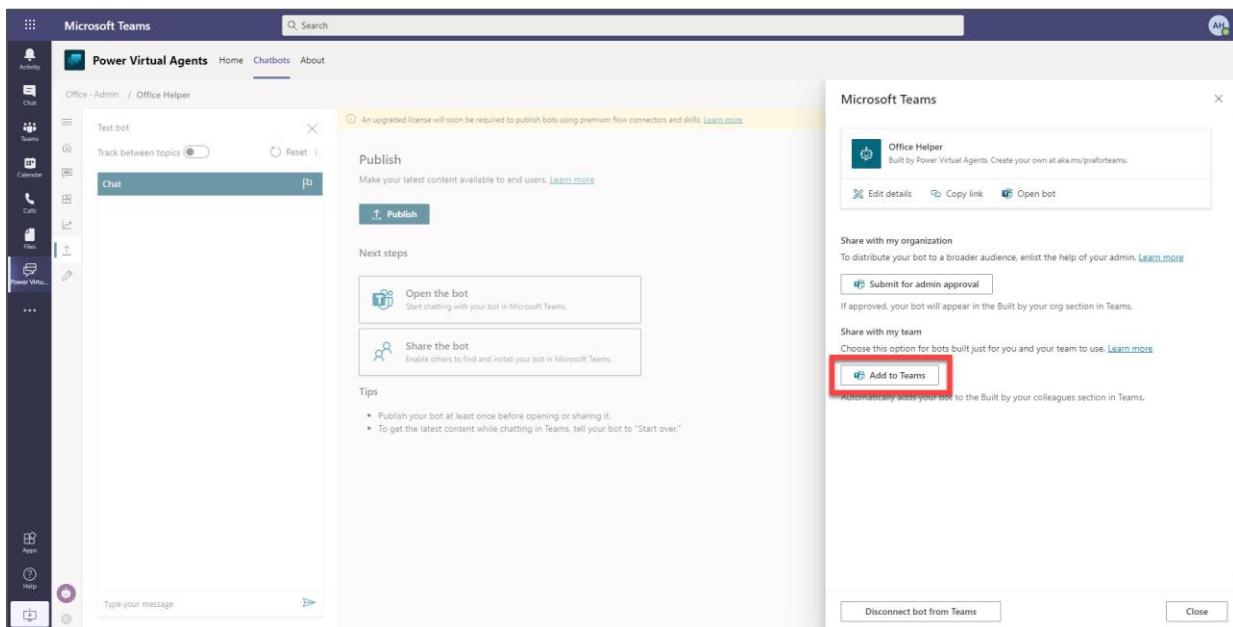
- In this task you will share your bot with the other members of your team, so that they can discover and use it in Teams. Click on the **Publish** icon on the Power Virtual Agents navigation bar.



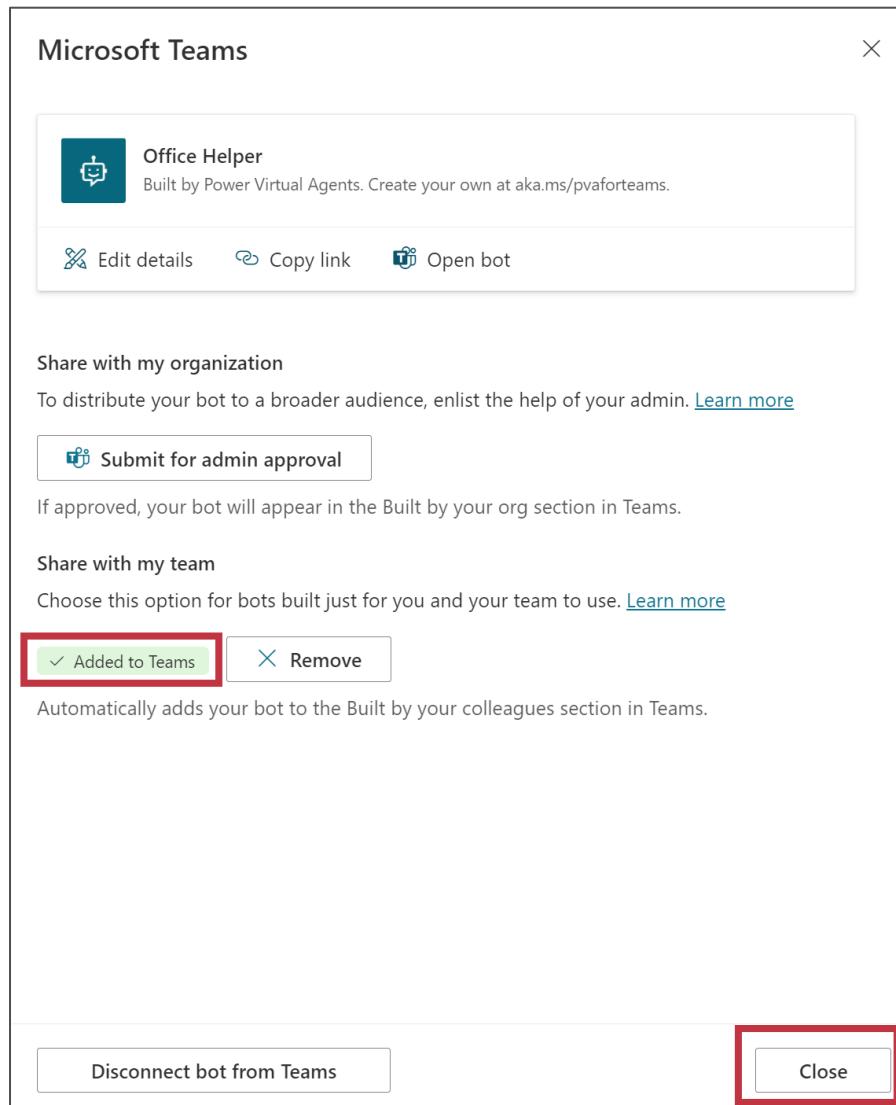
2. You don't need to publish again (unless you have made changes). Click on **Share the bot**.



3. You will be given two options here – to submit for admin approval or to share your bot with your team. Select **Add to Teams**.

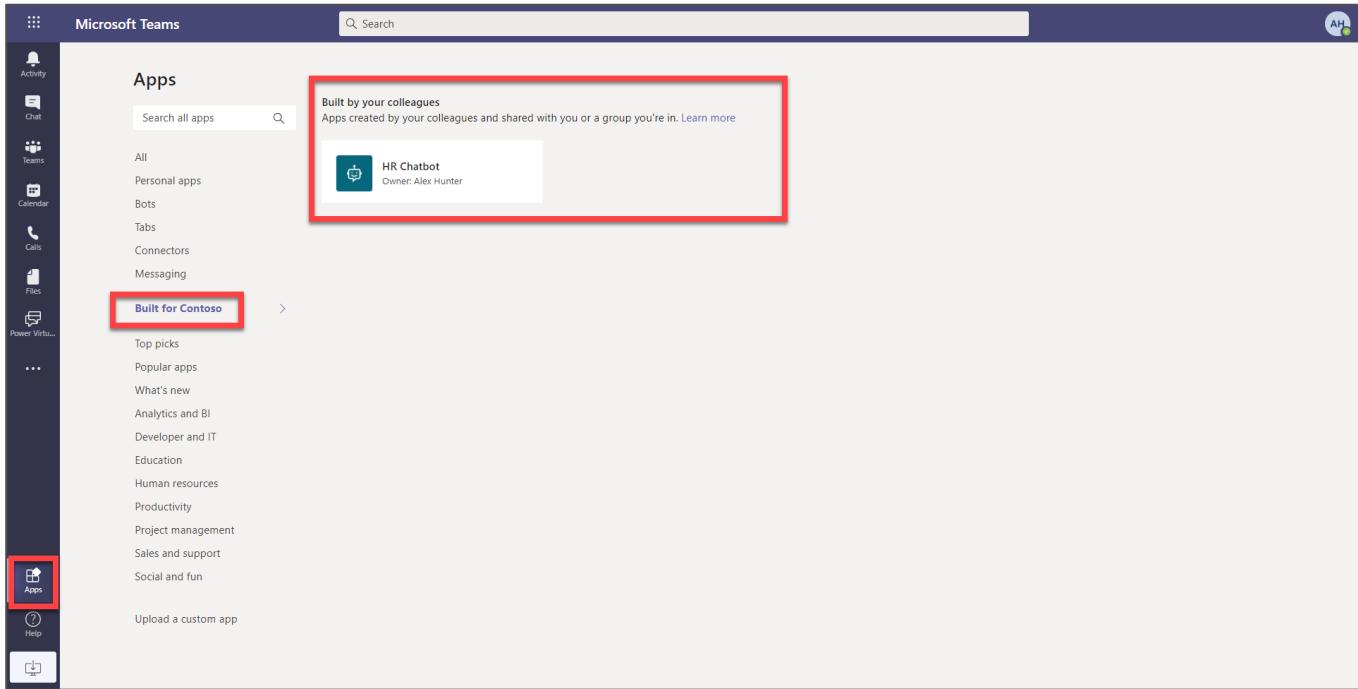


4. Once you see a confirmation message, click **Close**.



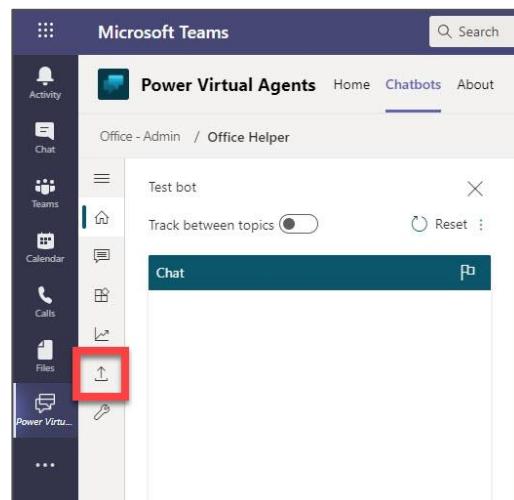
5. Other members of your team will now be able to discover and add your chatbot from the Teams app store, by viewing apps Built for your organization (in the Built by your colleagues section).

- Add someone else from your lab class to your team and ask them to go to the app store and see if they can see your app in the **Built by your colleagues** section. Do the same for someone else in the group. We will be working more with discovering and adding bots built by others in Lab 06: Collaboration Exercise.

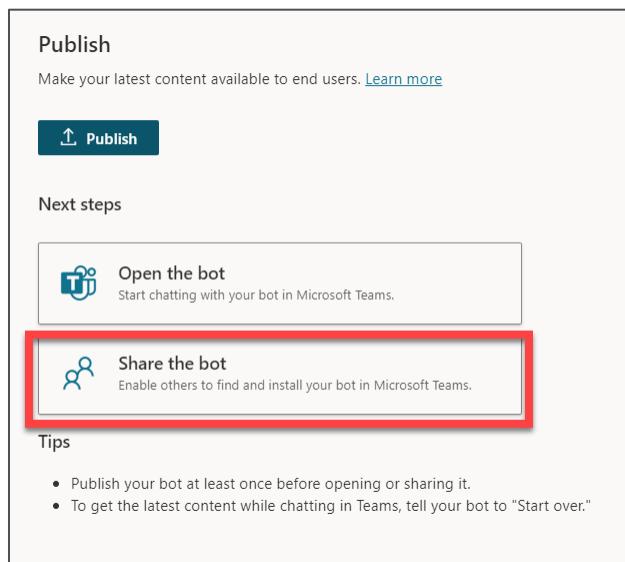


Task 3: Share your chatbot with your organization

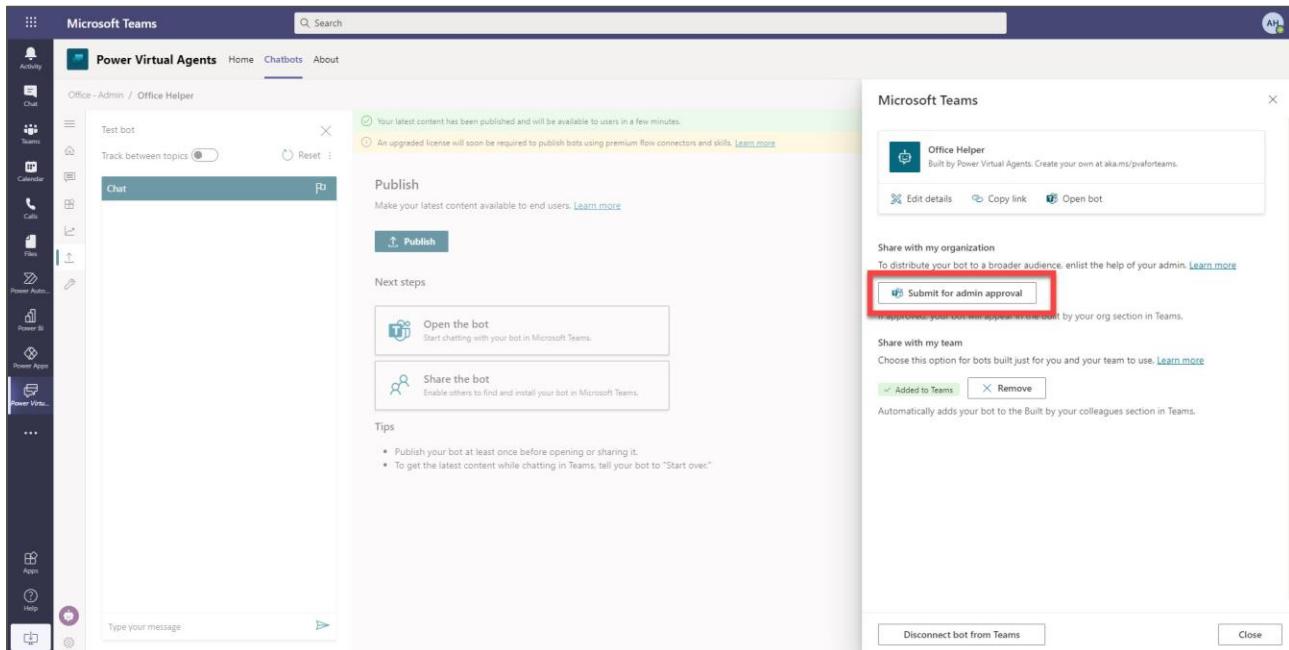
- In this task we will submit the chatbot for admin approval to share with the whole organization. Click on the publish icon on the Power Virtual Agents navigation bar



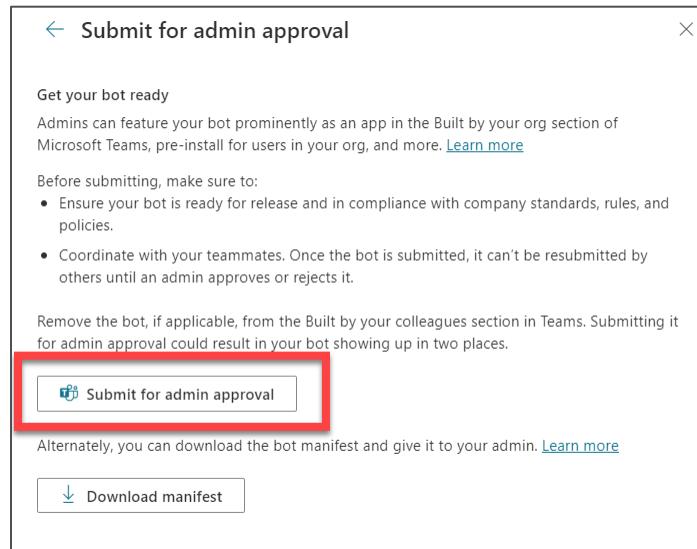
2. You don't need to publish again (unless you have made changes). Click on **Share the bot**.



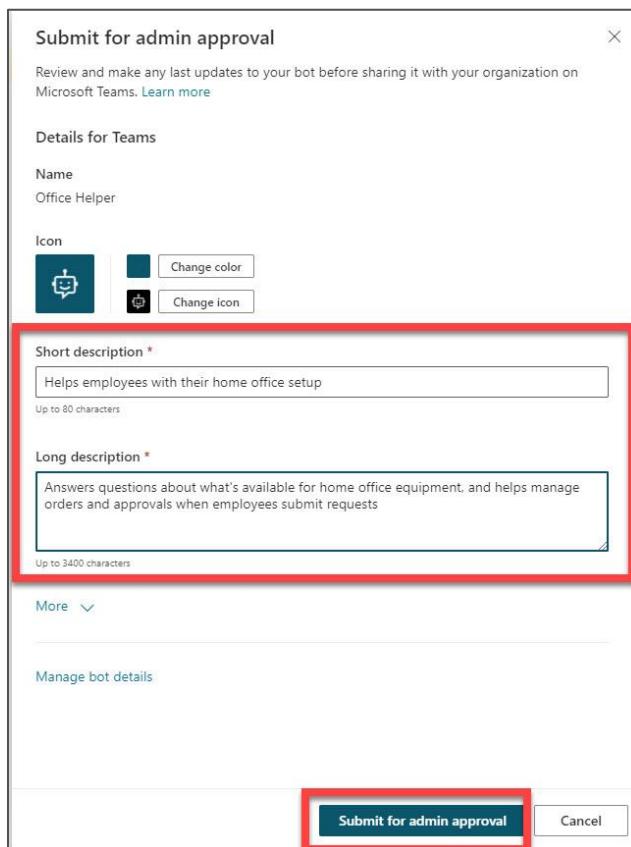
3. Click on **Submit for admin approval**.



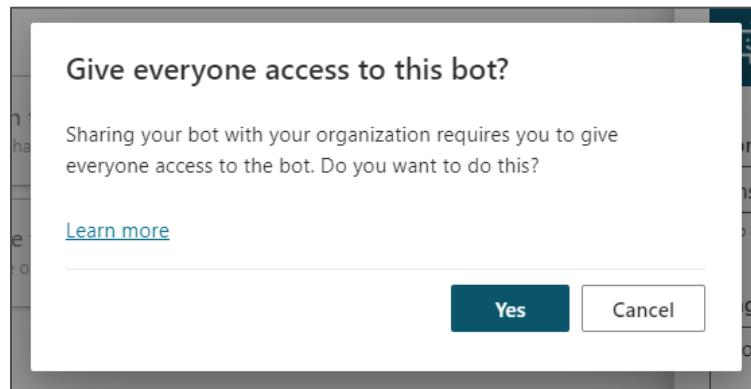
4. Read the message to understand how this works (in a real-world situation this is what you should have done before you submit for approval). Then click on **Submit for admin approval**.



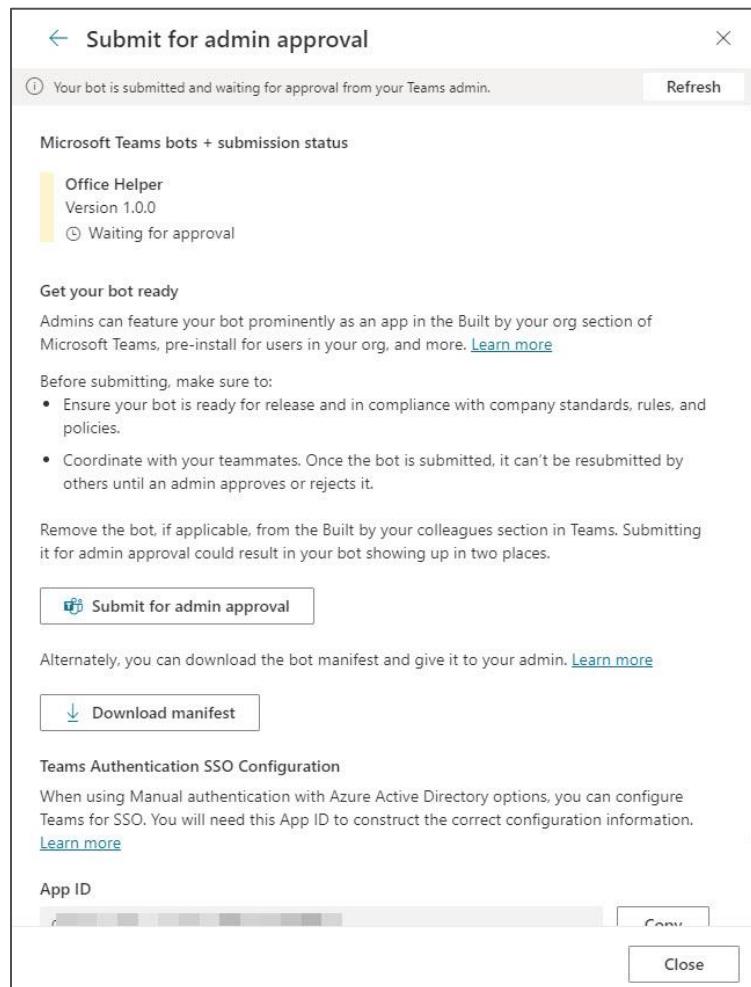
5. Fill in the **Short description** and **Long description** to describe what your chatbot does, and then click **Submit for admin approval**. Note there are also options here to change the color and icon.



6. Confirm that you want to give everyone in your organization access to the bot by clicking **Yes**.



7. When your bot has been submitted, you will get a submission status message.



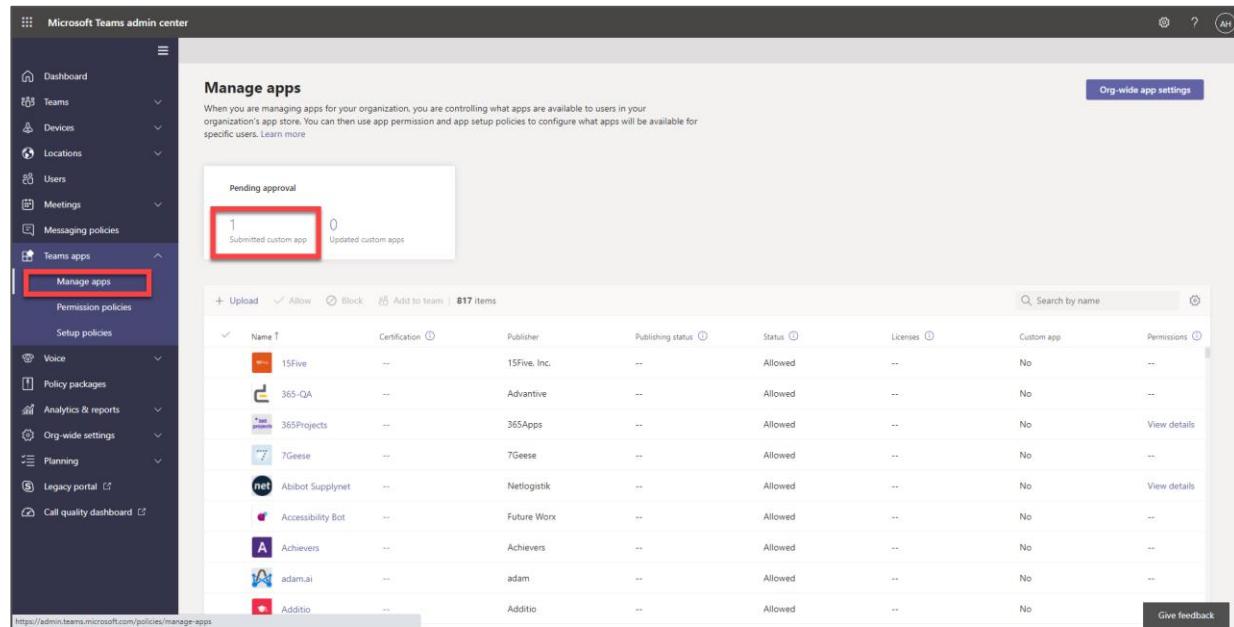
Task 4: Approve the chatbot as admin

If you have the admin login for your tenant, you can play the role of the admin and approve your app. If you are working in a shared tenant provided for training, your trainer will demonstrate this step for you (or you can read through the screenshots and steps below to understand how this works).

1. Open a new browser tab and navigate to the Microsoft Teams admin centre:

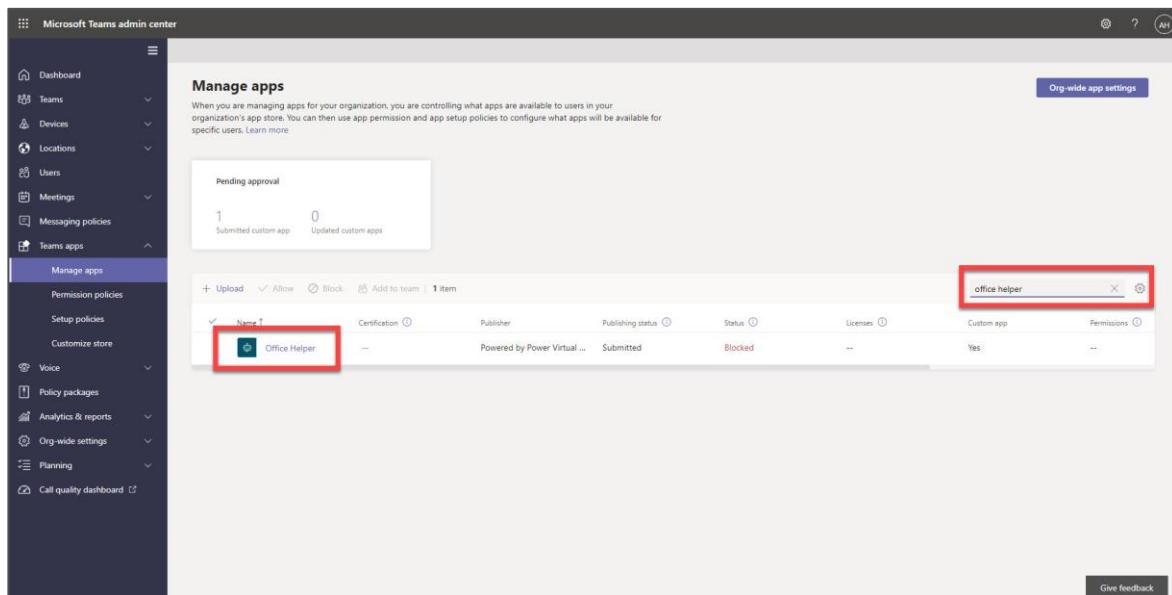
<https://admin.teams.microsoft.com/dashboard>

2. Go to Teams apps – Manage apps in the left hand navigation menu. You will see that there is 1 Submitted custom app pending approval.



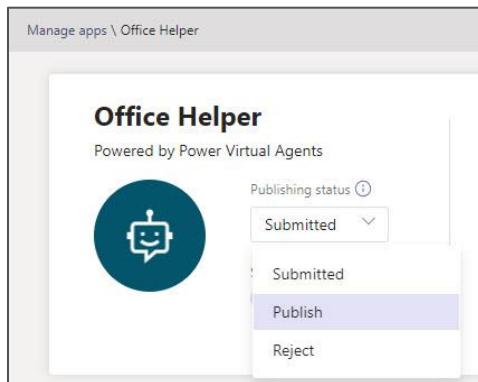
The screenshot shows the Microsoft Teams Admin Center interface. The left sidebar has a dark theme with various navigation options like Dashboard, Teams, Devices, Locations, Users, Meetings, Messaging policies, and Teams apps. Under Teams apps, 'Manage apps' is selected and highlighted with a red box. The main content area is titled 'Manage apps' with a sub-section 'Pending approval'. It displays a count of 1 Submitted custom app and 0 Updated custom apps. Below this, there's a table with columns: Name, Certification, Publisher, Publishing status, Status, Licenses, Custom app, and Permissions. The table lists several apps, including '15Five', '365-QA', '365Projects', '7Geese', 'Abilot Supplynet', 'Accessibility Bot', 'Achievers', 'adam.ai', and 'Additio'. The 'Status' column for all listed apps shows 'Allowed'. At the bottom right of the table, there's a 'Give feedback' button.

3. Search for the name of the chatbot you submitted and then click to open it.



This screenshot shows the same Microsoft Teams Admin Center interface as the previous one, but with a search term 'office Helper' entered into the search bar at the top right. The search results table shows a single item: 'Office Helper'. This result is highlighted with a red box. The rest of the interface is identical to the first screenshot, showing the navigation sidebar and the 'Pending approval' section.

4. Change the Publishing status to **Publish**, and then confirm in the pop up message that appears.



5. You will now see that your chatbot has been published.

Name	Publisher	Publishing status	Status	Licenses	Custom app	Permissions
Office Helper	Powered by Power Virtual Agents	Published	Allowed	--	Yes	--

6. Your chatbot will now be available in the main section of the Teams app store for your organization, for all employees to use.

The screenshot shows the Microsoft Teams App Store interface. On the left, there's a sidebar with various icons for Activity, Chat, Teams, Calendar, Calls, Files, Power Apps, Power BI, and Power Automate. Below these is a red box around the 'App' icon. The main area has a search bar at the top. A callout box highlights the 'Office Helper' app, which is described as 'Powered by Power Virtual Agents' and 'Answers questions about what's available for home office equipment, and helps manage orders and approvals when employees subm...'. Other visible apps include Lucid agreements, Bitbucket, zapier, Zoho Sign, ServiceDesk Plus Cloud, Workstreams.ai, Ment.io, Zoho Projects, Forms, Polly, YouTube, Jira Cloud, Azure Boards, Power BI, Trello, and Power Automate. The 'All' button in the search bar is also highlighted with a red box.

Summary

In this lab you created an internal chatbot embedded in Microsoft Teams, which is able to answer employee questions about home office equipment and place orders on their behalf. You saw that the chatbot in Teams automatically recognizes the logged in user, and learned how to connect your chatbot to your Dataverse for Teams table to retrieve data and to create new requests on behalf of the user. Finally, you learned how to publish and share your bot with other team members, and how to request admin approval to publish your bot for the whole organization.

Lab Survey

We would appreciate your feedback on this hands-on-lab, such as the quality of documentation and the usefulness of the learning experience. Please use the survey at <https://aka.ms/TeamsPPSurvey> to share your feedback. You may provide feedback for each module as you complete it or at the end once you've completed all the modules. Thank you!

Terms of Use

© 2021 Microsoft. All rights reserved.

By using this hands-on lab, you agree to the following terms:

The technology/functionality described in this hands-on lab is provided by Microsoft Corporation in a "sandbox" testing environment for purposes of obtaining your feedback and to provide you with a learning experience. You may only use the hands-on lab to evaluate such technology features and functionality and provide feedback to Microsoft. You may not use it for any other purpose. Without written permission, you may not modify, copy, distribute, transmit, display, perform, reproduce, publish, license, create derivative works from, transfer, or sell this hands-on lab or any portion thereof.

COPYING OR REPRODUCTION OF THE HANDS-ON LAB (OR ANY PORTION OF IT) TO ANY OTHER SERVER OR LOCATION FOR FURTHER REPRODUCTION OR REDISTRIBUTION WITHOUT WRITTEN PERMISSION IS EXPRESSLY PROHIBITED.

THIS HANDS-ON LAB PROVIDES CERTAIN SOFTWARE TECHNOLOGY/PRODUCT FEATURES AND FUNCTIONALITY, INCLUDING POTENTIAL NEW FEATURES AND CONCEPTS, IN A SIMULATED ENVIRONMENT WITHOUT COMPLEX SET-UP OR INSTALLATION FOR THE PURPOSE DESCRIBED ABOVE. THE TECHNOLOGY/CONCEPTS REPRESENTED IN THIS HANDS-ON LAB MAY NOT REPRESENT FULL FEATURE FUNCTIONALITY AND MAY NOT WORK THE WAY A FINAL VERSION MAY WORK. WE ALSO MAY NOT RELEASE A FINAL VERSION OF SUCH FEATURES OR CONCEPTS. YOUR EXPERIENCE WITH USING SUCH FEATURES AND FUNCITONALITY IN A PHYSICAL ENVIRONMENT MAY ALSO BE DIFFERENT.

FEEDBACK If you give feedback about the technology features, functionality and/or concepts described in this hands-on lab to Microsoft, you give to Microsoft, without charge, the right to use, share and commercialize your feedback in any way and for any purpose. You also give to third parties, without charge, any patent rights needed for their products, technologies and services to use or interface with any specific parts of a Microsoft software or service that includes the feedback. You will not give feedback that is subject to a license that requires Microsoft to license its software or documentation to third parties because we include your feedback in them. These rights survive this agreement.

MICROSOFT CORPORATION HEREBY DISCLAIMS ALL WARRANTIES AND CONDITIONS WITH REGARD TO THE HANDS-ON LAB, INCLUDING ALL WARRANTIES AND CONDITIONS OF MERCHANTABILITY, WHETHER EXPRESS, IMPLIED OR STATUTORY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. MICROSOFT DOES NOT MAKE ANY ASSURANCES OR REPRESENTATIONS WITH REGARD TO THE ACCURACY OF THE RESULTS, OUTPUT THAT DERIVES FROM USE OF THE VIRTUAL LAB, OR SUITABILITY OF THE INFORMATION CONTAINED IN THE VIRTUAL LAB FOR ANY PURPOSE.

DISCLAIMER This lab contains only a portion of new features and enhancements in Microsoft Teams and Power Platform. Some of the features might change in future releases of the product.