

Secure Collaboration in Microsoft Teams

Module 3: Power Automate and Microsoft Graph

Hands-on Lab Step-by-Step

Contents

[Power Automate and Microsoft Graph 4](#_Toc38830953)

[Introduction: Teams Request and Automated Provisioning 4](#_Toc38830954)

[Solution overview 4](#_Toc38830955)

[Exercise 1: Building the Provisioning and Configuration Process 5](#_Toc38830956)

[1: Navigate to Flow 5](#_Toc38830957)

[2: Create your Flow 5](#_Toc38830958)

[1. Create the first step 5](#_Toc38830959)

[2. Notify Requestor 6](#_Toc38830960)

[3. Create and Initialize Tenant ID Variable 7](#_Toc38830961)

[4. Create and Initialize Client ID Variable 8](#_Toc38830962)

[5. Create and Initialize Original Secret ID Variable 9](#_Toc38830963)

[6. Create and Initialize Secret ID Variable 10](#_Toc38830964)

[7. Create and Initialize Location Variable 11](#_Toc38830965)

[8. Create and Initialize Site Owner ID Array Variable 12](#_Toc38830966)

[9. Fill Site Owners Array Variable 13](#_Toc38830967)

[10. Create and Initialize ApproverEmail Variable 15](#_Toc38830968)

[11. Scoping the Get Manager “Try” Block 16](#_Toc38830969)

[12. Scoping the Get Manager “Catch” Block 17](#_Toc38830970)

[13. Create and Initialize Primary Site Owner ID Array Variable 19](#_Toc38830971)

[14. Create and Initialize Secondary Site Owner ID Array Variable 20](#_Toc38830972)

[15. Create and Initialize the Create Team API Body Variable 21](#_Toc38830973)

[16. Create Approval Action 21](#_Toc38830974)

[17. Send Approval Request to Each Approver 22](#_Toc38830975)

[18. Wait for an approval 24](#_Toc38830976)

[19. Determine if approved 24](#_Toc38830977)

[20. Generate Access Token 25](#_Toc38830978)

[21. Parse Access Token 25](#_Toc38830979)

[22. Call REST to Create Team 26](#_Toc38830980)

[23. Set Location Variable 26](#_Toc38830981)

[24. Get Index of First Quote 27](#_Toc38830982)

[25. Get Team Id 28](#_Toc38830983)

[26. Delay 28](#_Toc38830984)

[27. Get Group Details 29](#_Toc38830985)

[28. Parse Group Details 30](#_Toc38830986)

[29. Add Members to Team 30](#_Toc38830987)

[30. Determine if Guests Allowed 31](#_Toc38830988)

[30a. Determine if Guests Allowed - Yes 32](#_Toc38830989)

[30b. Determine if Guests Allowed - No 33](#_Toc38830990)

[31. Notification of Completion 34](#_Toc38830991)

[Exercise 2: Process Enhancements 36](#_Toc38830992)

[1: Add a more complex Approval Process 36](#_Toc38830993)

[2: Detailed Notifications 36](#_Toc38830994)

[3: Add Secondary Owners 36](#_Toc38830995)

[4: Replace the Delay Operation with a Check 36](#_Toc38830996)

[5: Add an Access Review for Teams that Allow Guests 36](#_Toc38830997)

[Copyright 37](#_Toc38830998)

# Power Automate and Microsoft Graph

## Introduction: Teams Request and Automated Provisioning

Imagine an organization that would like to have a little more control over the provisioning of assets withing Office 365, specifically in this case Microsoft Teams. The organization would like to have a request and provisioning process that follows their rules, allow for the ability to collaborate with people external to their organization (vendors, partners, etc.) as well as be easy to use.

## Solution overview

The Microsoft collaboration and business application platform technologies enable users to build a solution to meet this goal. The application user interface and interaction logic are built in PowerApps, the approval workflow is automated using Microsoft Flow, and the device order data is stored in SharePoint.

Key features of the solution:

1. Ability to submit a request for a new Microsoft Team
2. Follow the provisioning process for the request
3. Provide notifications in Teams to the relevant parties involved
4. Allow for individual Teams to have Guests allowed, or not

## Exercise 1: Building the Provisioning and Configuration Process

As requests are submitted, they are saved into SharePoint (previous exercise). To fulfill the actual request and provision and configure the Microsoft Team we’ll use a Flow in Power Automate that interfaces with the Microsoft Graph. We will leverage the connection we created via the Azure Active Directory Application Registration and the relevant calls to the Graph to accomplish this.

### 1: Navigate to Flow

1. Open a new browser table and go to <https://flow.microsoft.com>
2. Sign in if you are prompted

### 2: Create your Flow

A Flow should be created to contain this entire process. We will start from an existing “template” and add to it from there. **Note that flows do not auto save.** Save your work after each time you complete adding an action.

#### 1. Create the first step

|  |  |
| --- | --- |
| 1. Select **My Flows** |  |
| 1. Select **New 🡪 Automated from blank** |  |
| 1. Enter a name for your Flow    1. Use something unique, for example: **User2CreateTeam** |  |
| 1. Select as the **trigger**: *When an item is created (SharePoint)* |  |
| 1. Click **<Create>** |  |
| 1. In the **When an item is created** action:    1. In the **Site Address** enter the URL of **your site** that holds your request list       1. If your site **doesn’t** appear select **Enter custom value** in the dropdown and enter it in yourself. **Note**: Remember your personal site URL in the format of **https://teamswsiqvia.sharepoint.com/sites/teamsprovisioning#**    2. In the **List Name** select: **Team Requests** |  |

#### 2. Notify Requestor

|  |  |
| --- | --- |
| 1. Select **New Step** |  |
| 1. Type *Teams* in the **Search** field |  |
| 1. Select the **Microsoft Teams** connector |  |
| 1. Select the **Post a message as the Flow bot to a user (preview)** action |  |
| 1. Rename the action | See text file |
| 1. For the **Headline** in the action | See text file |
| 1. For the **Recipient** of the action, click in the field and in the dynamic content window select **Created By Email** |  |
| 1. For the **Message** in the action | See text file |
| 1. For the **Summary** in the action in the advanced options | See text file |

#### 3. Create and Initialize Tenant ID Variable

|  |  |
| --- | --- |
| 1. Select **New Step** |  |
| 1. Type *init* in the **Search** field |  |
| 1. Select the **Variable** connector |  |
| 1. Select the **Initialize variable** action |  |
| 1. Rename the action | See text file |
| 1. For the **Name** in the action | See text file |
| 1. For the **Type** in the action, select **String** |  |
| 1. For the **Value** in the action | Paste the tenant id from AAD App Reg that was previously created |

#### 4. Create and Initialize Client ID Variable

|  |  |
| --- | --- |
| 1. Select **New Step** |  |
| 1. Type *init* in the **Search** field |  |
| 1. Select the **Variable** connector |  |
| 1. Select the **Initialize variable** action |  |
| 1. Rename the action | See text file |
| 1. For the **Name** in the action | See text file |
| 1. For the **Type** in the action, select **String** |  |
| 1. For the **Value** in the action | Paste the client id from AAD App Reg that was previously created |

#### 5. Create and Initialize Original Secret ID Variable

|  |  |
| --- | --- |
| 1. Select **New Step** |  |
| 1. Type *init* in the **Search** field |  |
| 1. Select the **Variable** connector |  |
| 1. Select the **Initialize variable** action |  |
| 1. Rename the action | See text file |
| 1. For the **Name** in the action | See text file |
| 1. For the **Type** in the action, select **String** |  |
| 1. For the **Value** in the action | Paste the secret key from AAD App Reg that was previously created |

#### 6. Create and Initialize Secret ID Variable

|  |  |
| --- | --- |
| 1. Select **New Step** |  |
| 1. Type *init* in the **Search** field |  |
| 1. Select the **Variable** connector |  |
| 1. Select the **Initialize variable** action |  |
| 1. Rename the action | See text file |
| 1. For the **Name** in the action | See text file |
| 1. For the **Type** in the action, select **String** |  |
| 1. For the **Value** in the action | See text file |

#### 7. Create and Initialize Location Variable

|  |  |
| --- | --- |
| 1. Select **New Step** |  |
| 1. Type *init* in the **Search** field |  |
| 1. Select the **Variable** connector |  |
| 1. Select the **Initialize variable** action |  |
| 1. Rename the action | See text file |
| 1. For the **Name** in the action | See text file |
| 1. For the **Type** in the action, select **String** |  |

#### 8. Create and Initialize Site Owner ID Array Variable

|  |  |
| --- | --- |
| 1. Select **New Step** |  |
| 1. Type *init* in the **Search** field |  |
| 1. Select the **Variable** connector |  |
| 1. Select the **Initialize variable** action |  |
| 1. Rename the action | See text file |
| 1. For the **Name** in the action | See text file |
| 1. For the **Type** in the action, select Array |  |

#### 9. Fill Site Owners Array Variable

|  |  |
| --- | --- |
| 1. Select **New Step** |  |
| 1. Select the **Control** connector |  |
| 1. Select the **Apply to each** control |  |
| 1. For the **Title** of the control | See text file |
| 1. For the **Select an output from previous steps** of the control, click in the field and in the dynamic content window select **Asset Owners** |  |
| 1. Click **Add an action** inside the control |  |
| 1. Type *Users* in the **Search** field |  |
| 1. Select the **Office 365 Users** connector |  |
| 1. Select the **Get user profile (V2)** action |  |
| 1. Rename this action | See text file |
| 1. For the **User (UPN)** of the control, click in the field and in the dynamic content window select **Asset Owners Email** |  |
| 1. Click **Add an action** inside the **Apply to each** control |  |
| 1. Type *Append* in the **Search** field |  |
| 1. Select **Variable** |  |
| 1. Select the **Append to Array Variable** action |  |
| 1. Rename this action | See text file |
| 1. For the **Name** in this action, set it to | **SOIDArray** |
| 1. For the **Value** in this action | See text file |

#### 10. Create and Initialize ApproverEmail Variable

|  |  |
| --- | --- |
| 1. Select **New Step** |  |
| 1. Type *init* in the **Search** field |  |
| 1. Select the **Variable** connector |  |
| 1. Select the **Initialize variable** action |  |
| 1. Rename the action | See text file |
| 1. For the **Name** in the action | See text file |
| 1. For the **Type** in the action, select **String** |  |

#### 11. Scoping the Get Manager “Try” Block

In this section we’ll create an action to handle the steps of obtaining the requested Team’s owner’s manager.

|  |  |
| --- | --- |
| 1. Select **New Step** |  |
| 1. Select the **Control** connector |  |
| 1. Select the **Scope** control |  |
| 1. Rename the action | See text file |
| 1. Click **Add an action** inside the **Scope** control |  |
| 1. Type *Users* in the **Search** field |  |
| 1. Select the **Office 365 Users** connector |  |
| 1. Select the **Get manager (v2)** action |  |
| 1. Rename the action | See text file |
| 1. For the **User (UPN)** of the action, click in the field and in the dynamic content window select **Created By Email** |  |
| 1. Click **Add an action** inside the **Scope** control |  |
| 1. Type *Set* in the **Search** field |  |
| 1. Select the **Variable** connector |  |
| 1. Select the **Set variable** action |  |
| 1. Rename the action | See text file |
| 1. For the **Name** of the action select: | **ApproverEmail** |
| 1. For the value of the action, click in the field and in the dynamic content window select **Mail** |  |

#### 12. Scoping the Get Manager “Catch” Block

In this section we’ll create an action to handle the steps of handling an exception when obtaining the requested Team’s owner’s manager.

|  |  |
| --- | --- |
| 1. Select **New Step** |  |
| 1. Select the **Control** connector |  |
| 1. Select the **Scope** control |  |
| 1. Rename the action | See text file |
| 1. Click **Add an action** inside the **Scope** control |  |
| 1. Type *Set* in the **Search** field |  |
| 1. Select the **Variable** connector |  |
| 1. Select the **Set variable** action |  |
| 1. Rename the action | See text file |
| 1. For the **Name** of the action select: | **ApproverEmail** |
| 1. For the value of the action, click in the field and in the dynamic content window select **Created by Email** |  |
| 1. Select the ellipsis (…) of this Scope and select **Configure run after** |  |
| 1. Uncheck **is successful** and check **has failed** checkboxes then click <Done> |  |

#### 13. Create and Initialize Primary Site Owner ID Array Variable

|  |  |
| --- | --- |
| 1. Select **New Step** |  |
| 1. Type *init* in the **Search** field |  |
| 1. Select the **Variable** connector |  |
| 1. Select the **Initialize variable** action |  |
| 1. Rename the action | See text file |
| 1. For the **Name** in the action | See text file |
| 1. For the **Type** in the action, select Array |  |
| 1. For the **Value** in the action | See text file |
| 1. Select the ellipsis (…) of this action and select **Configure run after** |  |
| 1. Make sure only **is successful** and **is skipped** is checked then click <Done> |  |

#### 14. Create and Initialize Secondary Site Owner ID Array Variable

|  |  |
| --- | --- |
| 1. Select **New Step** |  |
| 1. Type *init* in the **Search** field |  |
| 1. Select the **Variable** connector |  |
| 1. Select the **Initialize variable** action |  |
| 1. Rename the action | See text file |
| 1. For the **Name** in the action | See text file |
| 1. For the **Type** in the action, select Array |  |
| 1. For the **Value** in the action | See text file |

#### 15. Create and Initialize the Create Team API Body Variable

|  |  |
| --- | --- |
| 1. Select **New Step** |  |
| 1. Type *init* in the **Search** field |  |
| 1. Select the **Variable** connector |  |
| 1. Select the **Initialize variable** action |  |
| 1. Rename the action | See text file |
| 1. For the **Name** in the action | See text file |
| 1. For the **Type** in the action, select **String** |  |
| 1. For the **Value** in the action | See text file |

#### 16. Create Approval Action

|  |  |
| --- | --- |
| 1. Select **New Step** |  |
| 1. Type *approval* in the **Search** field |  |
| 1. Select the **Approvals** connector |  |
| 1. Select the **Create an approval** action |  |
| 1. For the **Approval type** of the action, select **Approve/Reject – First to respond** |  |
| 1. For the **Title** in the action | See text file |
| 1. For the **Assigned to** in the action, click in the field and in the dynamic content window select **ApproverEmail** |  |
| 1. For the **Details** in the action | See text file |
| 1. For the **Item link** in the action, click in the field and in the dynamic content window select **Link to item** |  |
| 1. For the **Item link description** in the action, click in the field and in the dynamic content window select **Title** |  |

#### 17. Send Approval Request to Each Approver

|  |  |
| --- | --- |
| 1. Select **New Step** |  |
| 1. Select **Control** |  |
| 1. Select the **Apply to each** control |  |
| 1. Rename the action | See text file |
| 1. For the **Select an output from previous steps** in the action, click in the field and in the dynamic content window select **Approvers** |  |
| 1. Click **Add an action** inside the **Apply to each** control |  |
| 1. Type *Teams* in the search field |  |
| 1. Select the **Microsoft Teams** connector |  |
| 1. Select the **Post your own adaptive card as the Flow bot to a user (preview)** action |  |
| 1. Rename this action | See text file |
| 1. For the **Recipient** in the action, click in the field and in the dynamic content window select **Approvers Approver email** |  |
| 1. For the **Message** in the action, click in the field and in the dynamic content window select **Adaptive Card** |  |
| 1. For the **Summary** in the action | See text file |

#### 18. Wait for an approval

|  |  |
| --- | --- |
| 1. Select **New Step** |  |
| 1. Type *approval* in the **Search** field |  |
| 1. Select the **Approvals** connector |  |
| 1. Select the **Wait for an approval** action |  |
| 1. For the **Approval ID** in the action, click in the field and in the dynamic content window select **Approval ID** |  |

#### 19. Determine if approved

|  |  |
| --- | --- |
| 1. Select **New Step** |  |
| 1. Select Control |  |
| 1. Select the **Condition** action |  |
| 1. Rename the action | See text file |
| 1. For the **first value** of the condition in the action, click in the field and in the dynamic content window select **Outcome** |  |
| 1. For the **second value** of the condition in the action | See text file |

#### 20. Generate Access Token

|  |  |
| --- | --- |
| 1. Click **Add an action** inside the **If yes** path of the **Determine if Allowed action** |  |
| 1. Type *http* in the Search field |  |
| 1. Select the **HTTP [Premium]** connector |  |
| 1. Rename the action | See text file |
| 1. Select the **Method** | **POST** |
| 1. Paste the URI into the **URI** field | See text file |
| 1. For the first **Header** enter its **key** | See text file |
| 1. For the first **Header** enter its **value** | See text file |
| 1. For the **Body** field | See text file |

#### 21. Parse Access Token

|  |  |
| --- | --- |
| 1. Click **Add an action** inside the **If yes** path of the **Determine if Allowed action** |  |
| 1. Type *Parse* in the Search field |  |
| 1. Select the **Parse JSON** data operation |  |
| 1. Rename the action | See text file |
| 1. For the value of the **Content** in the action, click in the field and in the dynamic content window select **Body** |  |
| 1. For the **Schema** | See text file |

#### 22. Call REST to Create Team

|  |  |
| --- | --- |
| 1. Click **Add an action** inside the **If yes** path of the **Determine if Allowed action** |  |
| 1. Type *http* in the Search field |  |
| 1. Select the **HTTP [Premium]** connector |  |
| 1. Rename the action | See text file |
| 1. Select the **Method** | **POST** |
| 1. Paste the URI into the **URI** field | See text file |
| 1. For the first **Header** enter its **key** | See text file |
| 1. For the first **Header** enter its **value** | See text file |
| 1. For the second **Header** enter its **key** | See text file |
| 1. For the second **Header’s value**    1. Enter the word *Bearer*    2. Hit the <Space> bar once    3. Select from the dynamic content flyout **access\_token** |  |
| 1. For the **Body** | See text file |

#### 23. Set Location Variable

|  |  |
| --- | --- |
| 1. Click **Add an action** inside the **If yes** path of the **Determine if Allowed action** |  |
| 1. Type *Set* in the **Search** field |  |
| 1. Select the **Variable** connector |  |
| 1. Select the **Set variable** action |  |
| 1. Rename the action | See text file |
| 1. For the **Name** in the action select the **Location** variable from the dropdown list |  |
| 1. For the **Value** in the action | See text file |

#### 24. Get Index of First Quote

|  |  |
| --- | --- |
| 1. Click **Add an action** inside the **If yes** path of the **Determine if Allowed action** |  |
| 1. Type *Compose* in the **Search** field |  |
| 1. Select the **Data Operation** connector |  |
| 1. Select the **Compose** action |  |
| 1. Rename the action | See text file |
| 1. For the **Inputs** in the action | See text file |

#### 25. Get Team Id

|  |  |
| --- | --- |
| 1. Click **Add an action** inside the **If yes** path of the **Determine if Allowed action** |  |
| 1. Type *Compose* in the **Search** field |  |
| 1. Select the **Data Operation** connector |  |
| 1. Select the **Compose** action |  |
| 1. Rename the action | See text file |
| 1. For the **Inputs** in the action | See text file |

#### 26. Delay

|  |  |
| --- | --- |
| 1. Click **Add an action** inside the **If yes** path of the **Determine if Allowed action** |  |
| 1. Type *Delay* in the **Search** field |  |
| 1. Select the **Schedule** connector |  |
| 1. Select the **Delay** action |  |
| 1. Rename the action | See text file |
| 1. For the **Counts** in the action | See text file |
| 1. For the **Unit** select **Second** from the dropdown |  |

#### 27. Get Group Details

|  |  |
| --- | --- |
| 1. Click **Add an action** inside the **If yes** path of the **Determine if Allowed action** |  |
| 1. Type *http* in the Search field |  |
| 1. Select the **HTTP [Premium]** connector |  |
| 1. Rename the action | See text file |
| 1. Select the **Method** | **GET** |
| 1. Paste the URI into the **URI** field | See text file |
| 1. For the first **Header** enter its **key** | See text file |
| 1. For the first **Header’s value**    1. Enter the word *Bearer*    2. Hit the <Space> bar once    3. Select from the dynamic content flyout **access\_token** |  |

#### 28. Parse Group Details

|  |  |
| --- | --- |
| 1. Click **Add an action** inside the **If yes** path of the **Determine if Allowed action** |  |
| 1. Type *Parse* in the Search field |  |
| 1. Select the **Parse JSON** data operation |  |
| 1. Rename the action | See text file |
| 1. For the value of the **Content** in the action, click in the field and in the dynamic content window select **Body** |  |
| 1. For the Schema | See text file |

#### 29. Add Members to Team

|  |  |
| --- | --- |
| 1. Click **Add an action** inside the **If yes** path of the **Determine if Allowed action** |  |
| 1. Select the **Control** connector |  |
| 1. Select the **Apply to each** operation |  |
| 1. Rename this operation | See text file |
| 1. For the **Select an output from previous steps** field: *Asset Members* |  |
| 1. Click **Add an action** inside the **Loop** action |  |
| 1. Type *Groups* in the search field |  |
| 1. Select the **Office 365 Groups** connector |  |
| 1. Select the **Add member to group** action |  |
| 1. For the **Group ID** select **Enter custom value** from the dropdown |  |
| 1. For the value of the **Group Id** in the action, click in the field and in the dynamic content window select **Outputs** from the **Get Team Id** section |  |
| 1. For the value of the **User Principal Name** in the action, click in the field and in the dynamic content window select **Asset Members Email** |  |

#### 30. Determine if Guests Allowed

|  |  |
| --- | --- |
| 1. Click **Add an action** inside the **If yes** path of the **Determine if Allowed action (not inside the previous Members loop)** |  |
| 1. Select Control |  |
| 1. Select the **Condition** action |  |
| 1. Rename the action | See text file |
| 1. For the **first value** of the condition in the action, click in the field and in the dynamic content window select **Allow Guests** |  |
| 1. Make sure the condition is ‘*is equal to*’ |  |
| 1. For the **second value** of the condition in the action, click in the field and in the dynamic content window, click Expression and then enter *true* then select <OK> |  |

#### 30a. Determine if Guests Allowed - Yes

|  |  |
| --- | --- |
| 1. Click **Add an action** inside the **If yes** path of the **Determine Allow Guests** action |  |
| 1. Type *http* in the Search field |  |
| 1. Select the **HTTP [Premium]** connector |  |
| 1. Rename the action | See text file |
| 1. Select the **Method** | **POST** |
| 1. Paste the URI into the **URI** field | See text file |
| 1. For the first **Header** enter its **key** | See text file |
| 1. For the first **Header’s value**    1. Enter the word *Bearer*    2. Hit the <Space> bar once    3. Select from the dynamic content flyout **access\_token** |  |
| 1. For the second **Header** enter its **key** | See text file |
| 1. For the Second **Header** enter its **value** | See text file |
| 1. For the **Body** | See text file |

#### 30b. Determine if Guests Allowed - No

|  |  |
| --- | --- |
| 1. Click **Add an action** inside the **If no** path of the **Determine Allow Guests** action |  |
| 1. Type *http* in the Search field |  |
| 1. Select the **HTTP [Premium]** connector |  |
| 1. Rename the action | See text file |
| 1. Select the **Method** | **POST** |
| 1. Paste the URI into the **URI** field | See text file |
| 1. For the first **Header** enter its **key** | See text file |
| 1. For the first **Header’s value**    1. Enter the word *Bearer*    2. Hit the <Space> bar once    3. Select from the dynamic content flyout **access\_token** |  |
| 1. For the second **Header** enter its **key** | See text file |
| 1. For the Second **Header** enter its **value** | See text file |
| 1. For the **Body** | See text file |
| 1. Click **Add an action** inside the **If no** path of the **Determine Allow Guests** action |  |
| 1. Type *SharePoint* in the Search field |  |
| 1. Select the **SharePoint** connector |  |
| 1. Select the **Send an HTTP request to SharePoint** action |  |
| 1. Rename the action | See text file |
| 1. For the **Site Address**, select the **Communication Site** in the dropdown. If the site does not appear in the list enter the following custom value: *https://teamswsiqvia.sharepoint.com* |  |
| 1. For the **Method** | **POST** |
| 1. Paste the **URI** into the **URI** field | See text file |
| 1. For the **Body** | See text file |

#### 31. Notification of Completion

|  |  |
| --- | --- |
| 1. Click **Add an action** inside the **If yes** path of the **Determine if Approved** |  |
| 1. Type *Teams* in the **Search** field |  |
| 1. Select the **Microsoft Teams** connector |  |
| 1. Select the **Post a message as the Flow bot to a user (preview)** action |  |
| 1. Rename the action | See text file |
| 1. For the **Headline** in the action | See text file |
| 1. For the **Recipient** of the action, click in the field and in the dynamic content window select **Created By Email** |  |
| 1. For the **Message** in the action | See text file |
| 1. For the **Summary** in the action in the advanced options | See text file |

## Exercise 2: Process Enhancements

While the flow constructed above performs the bulk of what needs to be done (create a Team, set up its Guest Access configuration), there are a couple of other additional actions that could be added. Below are a couple additional actions that could be added to the process.

### 1: Add a more complex Approval Process

The approval process in this flow is a very simple one. Come up with a more complex process and using the information you learned in this lab implement it.

### 2: Detailed Notifications

The use of more detailed [Adaptive Cards](https://adaptivecards.io) facilitates more detailed notifications to participants in the workflow. Try enhancing those in this process to create more detailed ones.

### 3: Add Secondary Owners

The existing process just takes the first entered owner and makes them the primary. Using the variable that holds all the secondary owners, enhance the process to add them as well.

### 4: Replace the Delay Operation with a Check

You can get a little more efficient by using calls to determine if the Teams was created, instead of forcing a 60-second delay each time. Enhance that area to do the check instead of a delay.

### 5: Add an Access Review for Teams that Allow Guests

The Microsoft Graph is used to create the team and modify the group’s settings. There are also endpoints ([currently in beta](https://docs.microsoft.com/en-us/graph/api/resources/accessreviews-root?view=graph-rest-beta)) that allow you to create Access Reviews. Create an Access Review package for only those Teams that allow guests.

# Copyright

**© 2019 Microsoft Corporation. All rights reserved.**

By using this demo/lab, you agree to the following terms:

The technology/functionality described in this demo/lab is provided by Microsoft Corporation for purposes of obtaining your feedback and to provide you with a learning experience. You may only use the demo/lab to evaluate such technology features and functionality and provide feedback to Microsoft. You may not use it for any other purpose. You may not modify, copy, distribute, transmit, display, perform, reproduce, publish, license, create derivative works from, transfer, or sell this demo/lab or any portion thereof.

COPYING OR REPRODUCTION OF THE DEMO/LAB (OR ANY PORTION OF IT) TO ANY OTHER SERVER OR LOCATION FOR FURTHER REPRODUCTION OR REDISTRIBUTION IS EXPRESSLY PROHIBITED.

THIS DEMO/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 DEMO/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 FUNCTIONALITY IN A PHYSICAL ENVIRONMENT MAY ALSO BE DIFFERENT.

**FEEDBACK**. If you give feedback about the technology features, functionality and/or concepts described in this demo/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 DEMO/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 DEMO/ LAB, OR SUITABILITY OF THE INFORMATION CONTAINED IN THE DEMO/LAB FOR ANY PURPOSE.

**DISCLAIMER**

This demo/lab contains only a portion of new features and enhancements in Microsoft PowerApps. Some of the features might change in future releases of the product. In this demo/lab, you will learn about some, but not all, new features.