# Exercise 7: Deploying SPFx solutions to Microsoft Teams

## Task 1: Create your project

1. From the PowerShell command prompt, change to the C:/LabFiles/SharePoint directory by executing the following command: cd c:/LabFiles/SharePoint
2. Make a new directory for your SharePoint project files by executing the following command: md SPFxTeamsTab
3. Navigate to the newly created SharePoint directory by executing the following command: cd SPFxTeamsTab
4. Run the SharePoint Yeoman generator by executing the following command: yo @microsoft/sharepoint
5. Use the following to complete the prompt that is displayed:
   * **What is your solution name?**: SPFxTeamsTab
   * **Which baseline packages do you want to target for your component(s)?:** SharePoint Online only (latest)
   * **Where do you want to place the files?**: Use the current folder
   * **Do you want to allow the tenant admin the choice of being able to deploy the solution to all sites immediately without running any feature deployment or adding apps in sites?**: Yes
   * **Will the components in the solution require permissions to access web APIs that are unique and not shared with other components in the tenant?**: No
   * **Which type of client-side component to create?**: WebPart
   * **What is your Web Part name?**: SPFx Teams Together
   * **What is your Web Part description?**: SPFx Teams Together description
   * **Which framework would you like to use?**: No JavaScript framework
6. After provisioning the folders required for the project, the generator will install all the dependency packages using NPM.
7. Open the project in an editor such as Visual Studio Code.
8. Enable the web part to be used in Microsoft Teams:
   1. Locate and open the file **./src/webparts/spFxTeamsTogether/SpFxTeamsTogetherWebPart.manifest.json**.
   2. Within the web part manifest file, locate the property **supportedHosts**: "supportedHosts": ["SharePointWebPart"],
   3. Add another option to enable this web part to be used as a tab in a Microsoft Teams team: "supportedHosts": ["SharePointWebPart", "TeamsTab"],

## Task 2: Creating and deploying the Microsoft Teams app package

Notice the project contains a folder **teams** that contains two images. These are used in Microsoft Teams to display the custom tab.

You may notice there is no **manifest.json** file present. The manifest file can be generated automatically by SharePoint from the App Catalog site. However, at this time, this functionality is not fully operational.

Alternatively, you can manually create the manifest file if you want to have more control over the default values that SharePoint will create.

The automatic generation and deployment of the Microsoft Teams manifest is not currently operational; you will manually create the Microsoft Teams manifest and app package:

1. Copy the **manifest.json** file from **Resources/manifest.json** to the **./teams** folder in your project.
2. Open the **manifest.json** file. This file contains multiple strings that need to be updated to match the SPFx component. Use the following table to determine the values that should be replaced. The SPFx component properties are found in the web part manifest file: **./src/webparts/spFxTeamsTogether/SpFxTeamsTogetherWebPart.manifest.json**
3. Create a Microsoft Teams app package by zipping the contents of the **./teams** folder. Make sure to zip just the contents and not the folder itself. This ZIP archive should contain three files at the root: two images and the **manifest.json**.

|  |  |
| --- | --- |
| **manifest.json string** | **Property in SPFx component manifest** |
| {{SPFX\_COMPONENT\_ALIAS}} | alias |
| {{SPFX\_COMPONENT\_NAME}} | preconfiguredEntries[0].title |
| {{SPFX\_COMPONENT\_SHORT\_DESCRIPTION}} | preconfiguredEntries[0].description |
| {{SPFX\_COMPONENT\_LONG\_DESCRIPTION}} | preconfiguredEntries[0].description |
| {{SPFX\_COMPONENT\_ID}} | id |

**Note**: Don’t miss replacing **{{SPFX\_COMPONENT\_ID}}** in **configurableTabs[0].configurationUrl**. You will likely have to scroll your editor to the right to see it. The tokens surrounded by single curly braces (for example, **{teamSiteDomain}**) do not need to be replaced. ## Task 3: Create and deploy the SharePoint package

In order to test the web part in SharePoint and Microsoft Teams, it must first be deployed to an app catalog.

1. Open a browser and navigate to your SharePoint Online Tenant-Scoped App Catalog site.
2. Select the menu item **Apps for SharePoint** from the left navigation menu.
3. Build the project by opening a command prompt and changing to the root folder of the project. Then execute the following command: gulp build
4. Next, create a production bundle of the project by running the following command on the command line from the root of the project: gulp bundle –ship
5. Create a deployment package of the project by running the following command on the command line from the root of the project: gulp package-solution –ship
6. Locate the file created by the gulp task, found in the **./sharepoint/solution** folder with the name \*.sppkg.
7. Drag this file into **the Apps for SharePoint** library in the browser.
8. In the **Do you trust…?** dialog box, select the check box **Make this solution available to all sites in the organization** and then select **Deploy**.
9. This will make the SPFx web part available to all site collections in the tenant, including those that are behind a Microsoft Teams team.

## Task 4: Test the web part in SharePoint and Microsoft Teams

Test the SPFx web part in SharePoint:

1. In the browser, navigate to a modern SharePoint page.
2. Select the **Edit** button in the upper-right portion of the page.
3. Select the **(+)** icon to open the SharePoint web part toolbox and locate the web part SPFx Teams Together:
4. The SharePoint Framework web part will be displayed on the page.

## Task 5: Testing the SPFx web part in Microsoft Teams

1. Create a new Microsoft Teams team.
   1. Using the same browser where you are signed in to SharePoint Online, navigate to [https://teams.microsoft.com](https://teams.microsoft.com/). When prompted, load the web client.
   2. If you do not have any teams in your tenant, you will be presented with the following dialog. Otherwise, select **Join or create a team** at the bottom of the list of teams.
   3. On the **Create your team** dialog box, select **Build a team from scratch**.
   4. On the **What kind of team will this be?** dialog box, select **Public**.
   5. When prompted, use the name **My First Team**.
2. Install the Microsoft Teams application as a new tab that will expose the SharePoint Framework web part in Microsoft Teams:
   1. Select the **My First Team** team previously created.
   2. Select the **General** channel.
3. Add a custom tab to the team using the SPFx web part:
   1. At the top of the page, select the **+** icon in the horizontal navigation.
   2. In the **Add a tab** dialog box, select **More Apps**.
   3. Select the **Upload a custom app** > **Upload for …** from the list of app categories.
   4. Select the Microsoft Teams application ZIP file previously created. This is the file that contains the **manifest.json** and two image files.
   * **Note**: After a moment, the application will appear next to your tenant name. You may need to refresh the page for the app to appear if you are using the browser Microsoft Teams client.
4. Select the **SPFx Teams Together** app.
5. In the **SPFx Teams Together** dialog box, select the **My First Team** in the **Add to a team** drop-down control and select **Install**.
6. In the **SPFx Teams Together is now available for My First Team** dialog box, select the **General** channel and select **Set up**.
7. The next dialog box will confirm the installation of the app. Select **Save**.
8. The application should now load in Microsoft Teams within the **General** channel under the tab **SPFx Teams Together**.

## Review

In this exercise, you learned how to create a SharePoint Framework (SPFx) solution that will work in both SharePoint and as a tab in Microsoft Teams.