## Packaging and Deploying SPFX Solutions

**Lab Time**: 60 minutes

**Lab Folder**: C:\Student\Modules\06\_PackagingAndDeployment\Lab

**Lab Overview**: Learning how to publish and install SharePoint Framework solutions correctly is a core skill that every SharePoint developer should know. In this lab you will create an app catalog and then you will learn how to publish and install SPFx solutions.

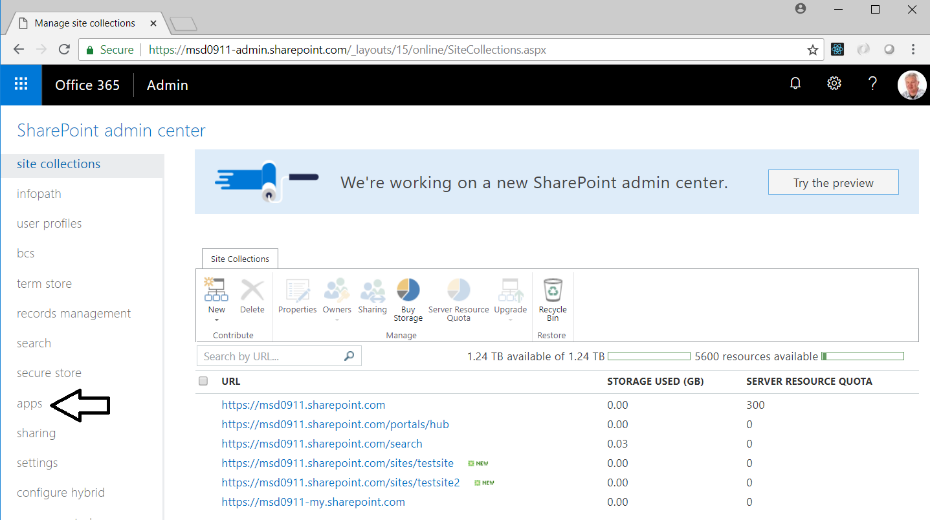
### Exercise 1: Create an App Catalog Site Collection

In this exercise you will create an app catalog to support app deployment and installation.

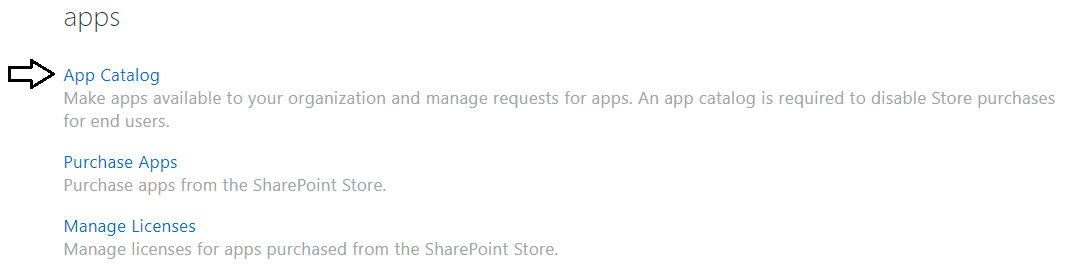
1. Using the browser, navigate to the SharePoint admin center for your Office 365 tenant using the following URL.

https://[TENANT\_NAME]-admin.sharepoint.com

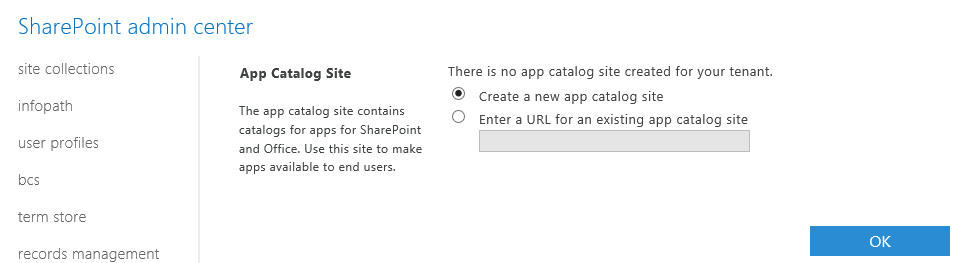
1. Click on the **apps** link to see the **apps** page.



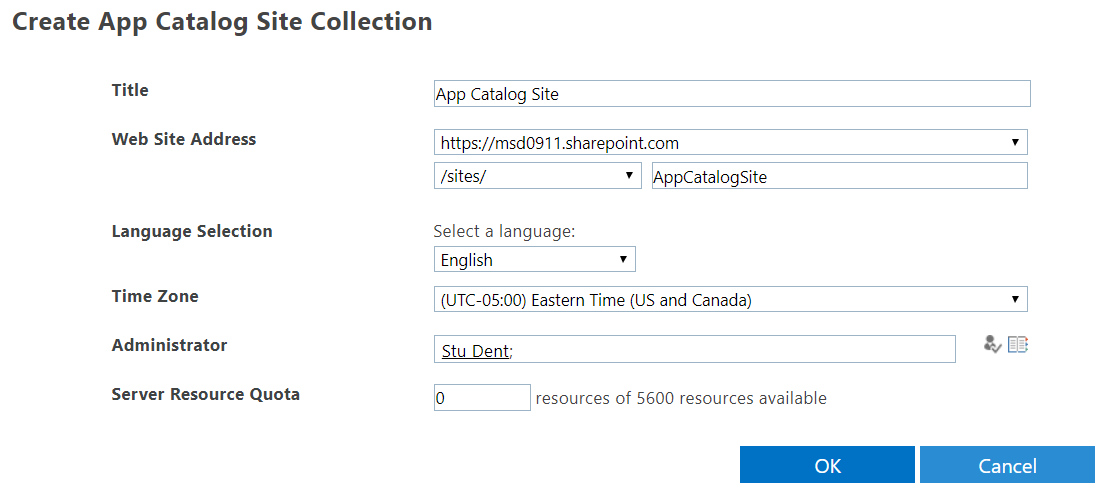
1. Click on the first link in the **apps** page with the caption of **App Catalog**.



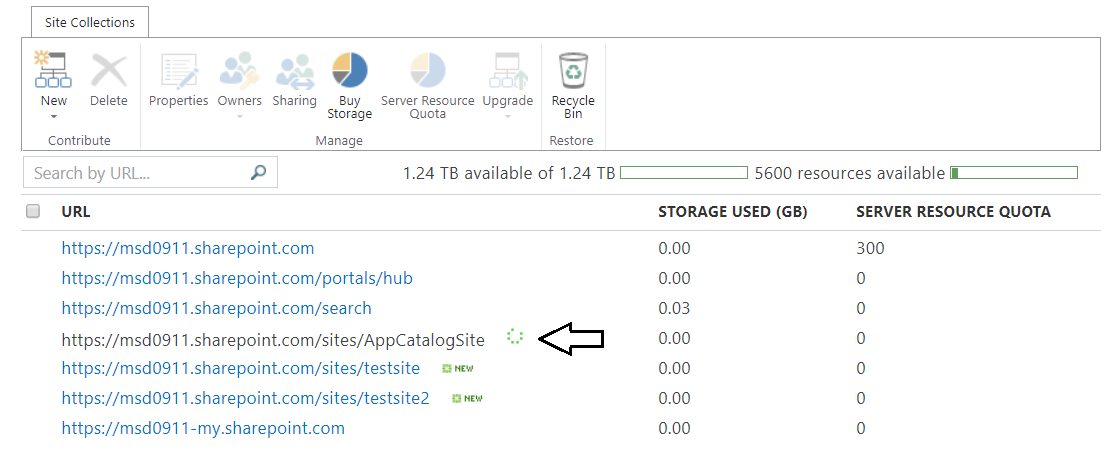
1. On the **App Catalog Site** page, leave the default selection of **Create a new app catalog site** and click **OK**.



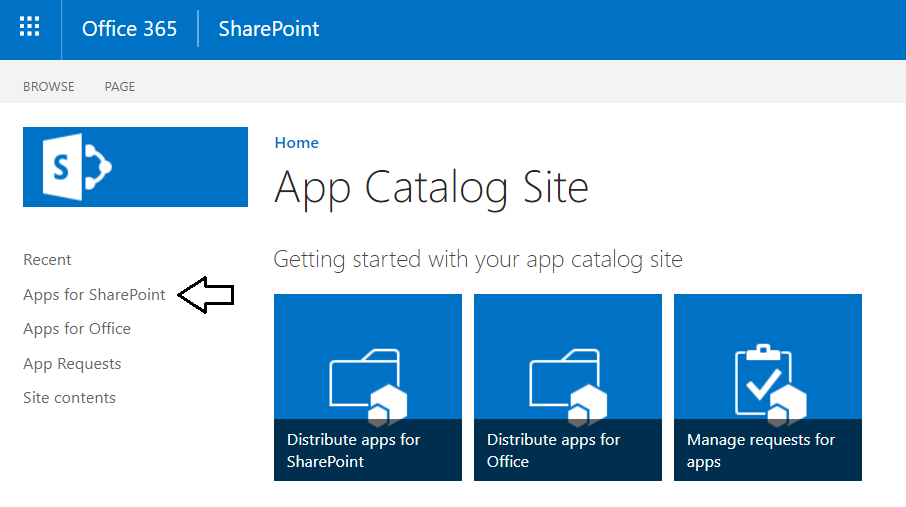
1. On the **Create App Catalog Site Collection** page…
   1. Enter a Title of **App Catalog Site**.
   2. Enter a **Web Site Address** of **https://[YOUR\_TENANT\_NAME].sharepoint.com/sites/AppCatalogSite**.
   3. Fill in the rest of the page using data shown in the following screenshot and click **OK** to create the App Catalog site collection.



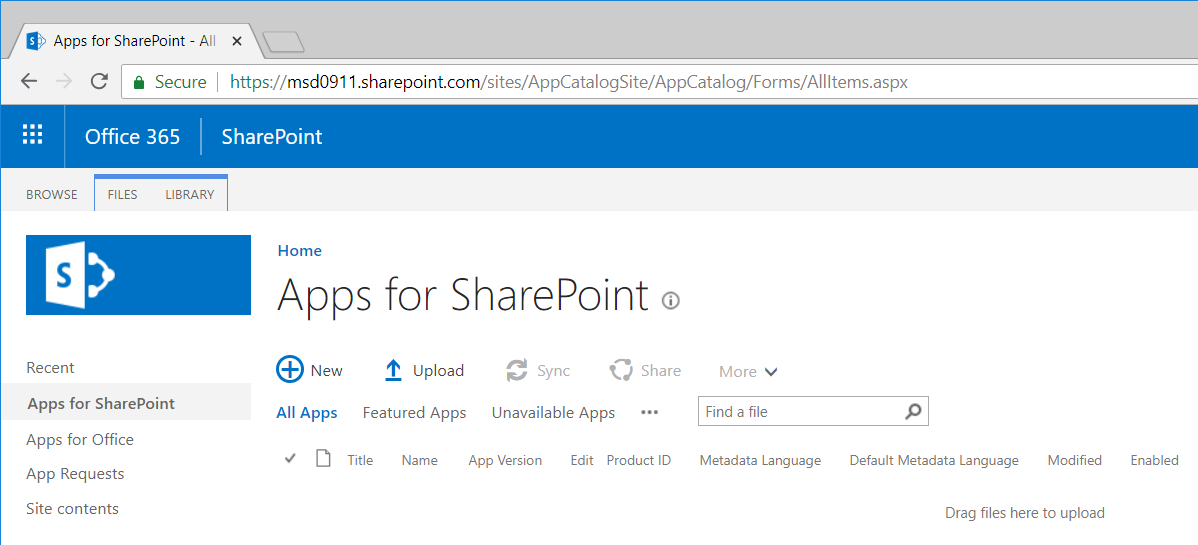
* 1. You should be able to see that the new site collection is being created.



* 1. Once the new site collection has been fully created, navigate to it to see the App Catalog site.



* 1. You should now see the default view of the document library named **Apps for SharePoint**. This is the library where you will upload your solution packages to publish them.



Now you are done configuring the your SharePoint Online tenant for publishing and deploying SharePoint Framework. Now, you will work to publish and install several different SharePoint Framework solution packages so you can get first-hand experience deploying SharePoint solutions in an Office 365 tenant.

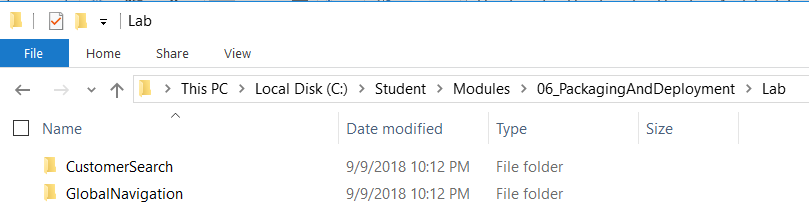
### Exercise 2: Build SharePoint Framework Solution Packages

In this exercise you will go through the steps to build solution packages for two SharePoint Framework projects for distribution.

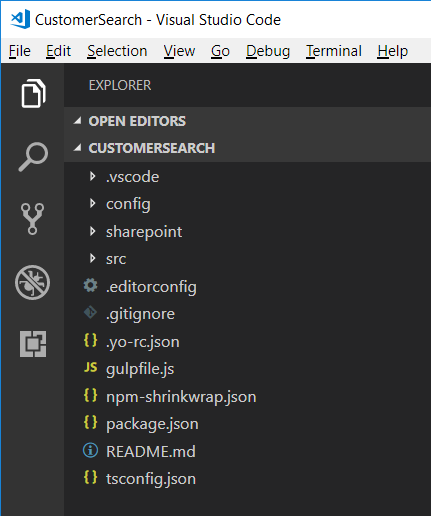
1. Open the SharePoint Framework project named **CustomerSearch**.
   1. Using Windows Explorer, navigate to the lab folder for this module at the following path.

C:\Student\Modules\06\_PackagingAndDeployment\Lab

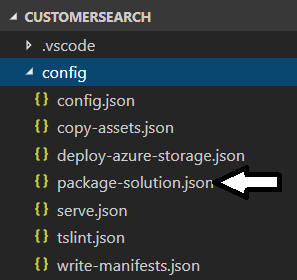
* 1. You should see two folders with SharePoint Framework projects named **CustomerSearch** and **GlobalNavigation**.



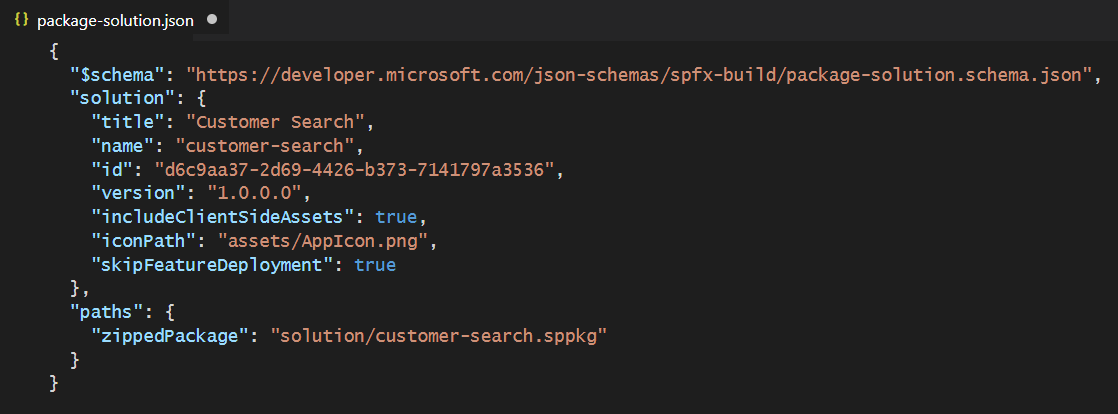
* 1. Open the folder named **CustomerSearch** with Visual Studio Code.
  2. Take a moment to inspect the files and folder inside the project.



* 1. Inside the config folder, locate and open the configuration file named **package-solution.json**.



* 1. Review the child properties of the **solution** property in **package-solution.json**. There is no need to make any edits to this file.



1. Run npm install to restore the require npm packages to the project.
   1. Use the **View > Integrated Terminal** menu command to display the Integrated Terminal.
   2. Locate the console of the **Integrated Terminal** where you can type in and execute **npm** commands.
   3. Run the npm install command.

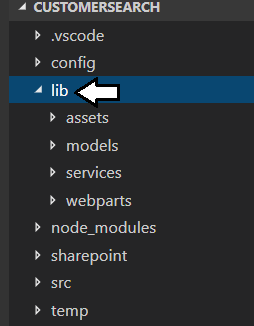
npm install

Once you have restored the npm packages, you can now build the solution into a distribution package.

1. Build the solution into a package for distribution.
   1. From the Terminal console, execute the gulp build command.

gulp build

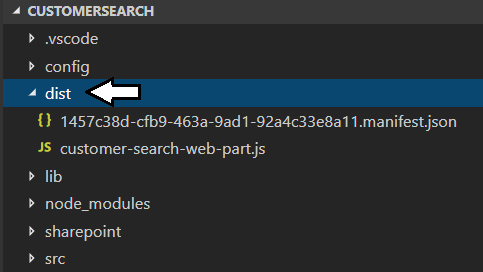
* 1. When you execute the **gulp build** command, you will notice that it adds a new folder to the project named **lib**.



* 1. Bundle the solution by executing the following on the command line:

gulp bundle --ship

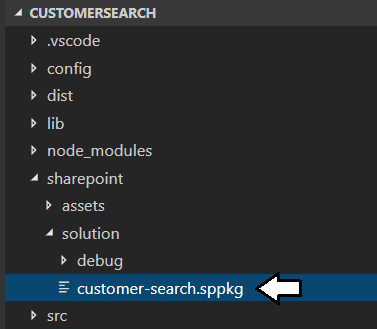
* 1. When you execute the **gulp bundle** command, you will notice that it adds a new folder to the project named **dist**.



* 1. Package the solution by executing the following on the command line:

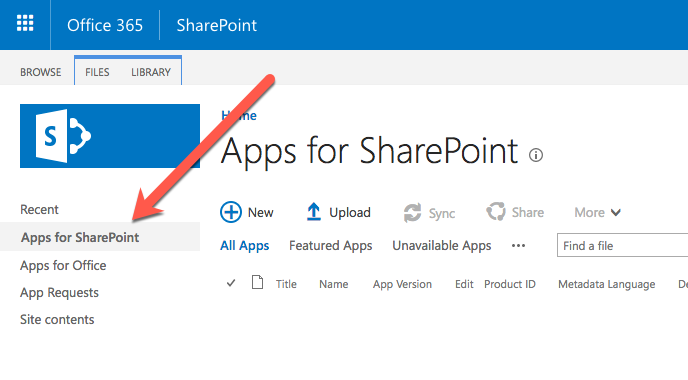
gulp package-solution --ship

* 1. When you execute the **gulp package-solution**, the command generates a solution package named **customer-search.sppkg** in the **sharepoint/solution/debug** folder.

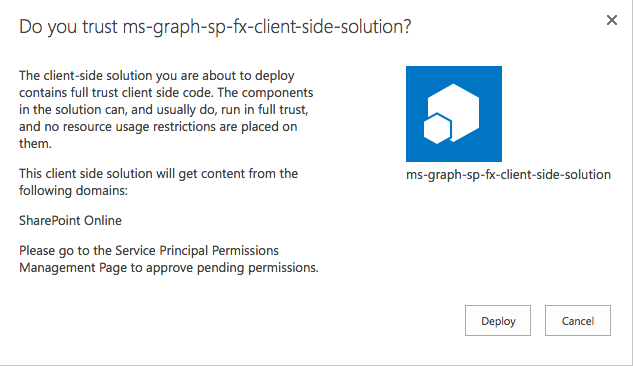


Now you have created the solution package you need to publish to the app catalog.

1. Deploy and trust the SharePoint package:
   1. In the browser, navigate to your SharePoint Online Tenant App Catalog.
   2. Select the **Apps for SharePoint** link in the navigation:

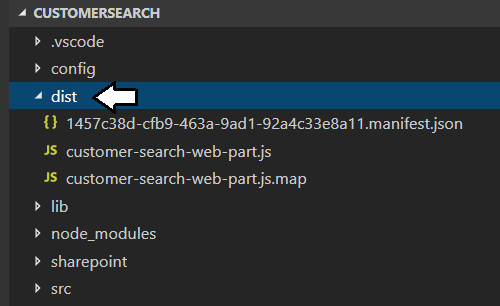
[](https://github.com/microsoftgraph/msgraph-training-spfx/blob/master/Images/tenant-app-catalog-01.png)

* 1. Drag the generated SharePoint package from **\sharepoint\solution\ms-graph-sp-fx.sppkg** into the **Apps for SharePoint** library.
  2. In the **Do you trust ms-graph-sp-fx-client-side-solution?** dialog, select **Deploy**.

[](https://github.com/microsoftgraph/msgraph-training-spfx/blob/master/Images/tenant-app-catalog-02.png)

gulp bundle

* 1. s



* 1. x

### Exercise 3: Publish an SharePoint Framework Solution to the App Catalog

In this exercise you will package an existing SharePoint-Hosted app and publish it to the Wingtip App Catalog site.

1. Launch **Visual Studio 2015** as administrator:
   1. appears just as before in that you can use the calculator.

### Exercise 4: Install a SharePoint Framework Solution in a SharePoint Site.

In this exercise you will package an existing SharePoint-Hosted app and publish it to the Wingtip App Catalog site.

1. Launch **Visual Studio 2015** as administrator:
   1. appears just as before in that you can use the calculator.

### Exercise 5: Update an SPFX Solution that's Already in Use

In this exercise, you will learn what is involved with updating an existing App by making a small update to the start page of an app and deploying the new version.

1. Return to Visual Studio 2015.
   1. home page is visible.