

Embedding Reports and Dashboards using SharePoint Framework

Setup Time: 60 minutes

Lab Folder: C:\Student\Modules\08_PBIEmbedding\Lab

Overview: In this lab you will work through the steps to develop a custom web application that embeds Power BI reports and dashboards. You will begin by creating an app workspace and populating it embeddable Power BI resources including a dataset, a report and a dashboard. Next, you will register a new application with Azure AD in the Azure portal and configure this Azure AD application with the proper permissions to program using the Power BI Service API. After that, you will create a new web application using Visual Studio 2017 and ASP.NET MVC. As you move through the exercises of this lab, you will program against both the Power BI Service API and the Power BI JavaScript API to implement the required steps to embed Power BI reports and dashboards into your custom web application.

Exercise 1: Create a New SPFx Project

In this exercise you will embed a dashboard. As you will see, it's not very different from the steps you have already implemented to embed a report.

1. Add a new method to the **PbiEmbeddingManger** class named **GetDashboardEmbeddingData**.

Exercise 2: Call the Power BI Service API using AadHttpClient

In this exercise you will embed a dashboard. As you will see, it's not very different from the steps you have already implemented to embed a report.

2. Add a new method to the **PbiEmbeddingManger** class named **GetDashboardEmbeddingData**.

Exercise 3: Embed Reports and Dashboards in SPFx Web Parts

In this exercise you will embed a dashboard. As you will see, it's not very different from the steps you have already implemented to embed a report.

3. Add a new method to the **PbiEmbeddingManger** class named **GetDashboardEmbeddingData**.

Exercise 4: Add Web Part Properties to Customize the Embedding Experience

In this exercise you will embed a dashboard. As you will see, it's not very different from the steps you have already implemented to embed a report.

4. Add a new method to the **PbiEmbeddingManger** class named **GetDashboardEmbeddingData**.