**Exercise 1 - Create a custom Microsoft Teams personal tab**

The Microsoft Teams developer platform makes it easy for you to extend Teams and integrate your own applications and services seamlessly into the Teams workspace. These apps can then be distributed to your enterprise or for teams around the world.

Before starting the exercise, verify your environment. In this exercise, I am using the tools mentioned below installed in my environment:

* Node.js - v10.16.0
* NPM - 6.9.0
* Gulp
  + CLI version: 2.3.0
  + Local version: 4.0.2
* Yeomen Generator of MS Teams - 2.14.0
* Visual Studio Code

and

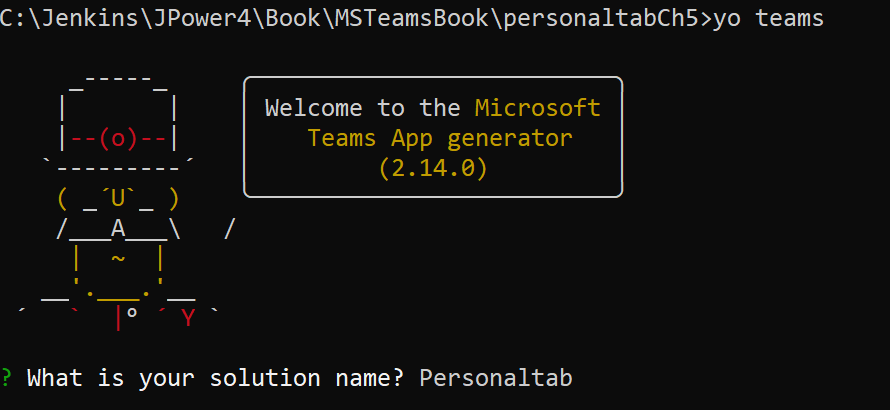
* Microsoft Azure Subscription
* Office 365 Subscription

In this exercise, you will create a new Microsoft Teams personal tab by using the Microsoft Teams Yeoman generator, Visual Studio Code, and App Studio.

**Create personal tab using yeoman generator**

In this section, you will create a new personal app Teams app using Yeoman generator (yo teams). This exercise will guide you on how to create a custom personal tab.

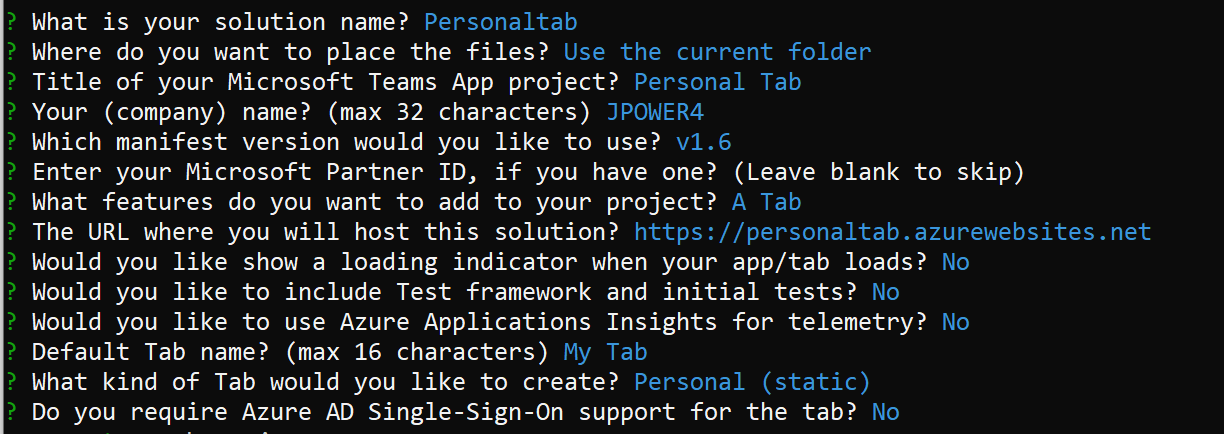
* To create a new personal tab app project
* Create a new project directory in your favorite location.
* Open command prompt
* create a new folder “personaltabCh5”
* Navigate to a newly created directory
* Run the Yeoman Generator for Microsoft Teams by running the following command: **yo teams**



yo teams generator

Yeoman will launch and ask you a series of questions. Answer the questions with the following values:

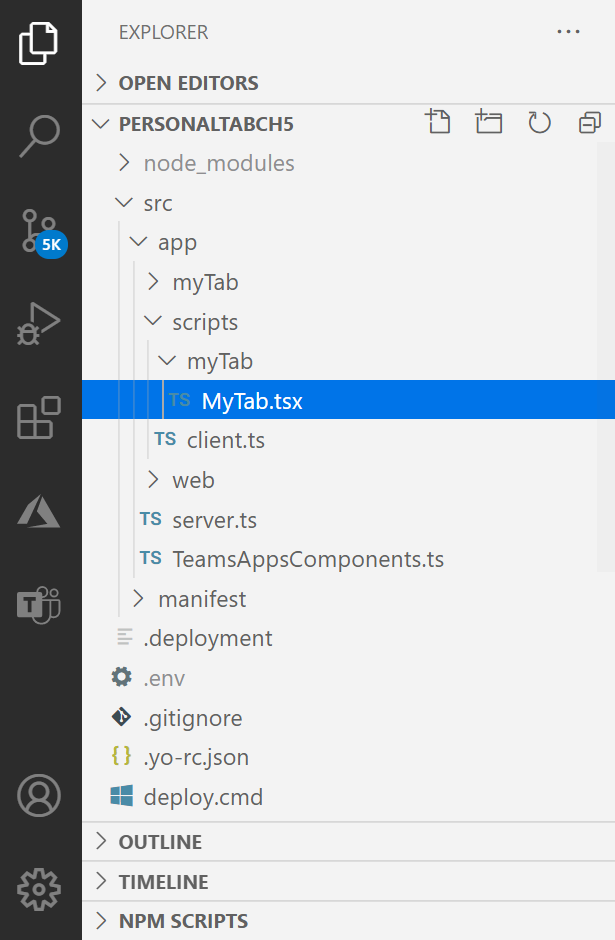
* What is your solution name? Personaltab
* Where do you want to place the files? Use the current folder
* Title of your Microsoft Teams App project? Personal Tab
* Your (company) name? (max 32 characters) JPOWER4
* Which manifest version would you like to use? v1.6
* Enter your Microsoft Partner ID, if you have one? (Leave blank to skip)
* What features do you want to add to your project? A Tab
* The URL where you will host this solution? https://personaltab.azurewebsites.net
* Would you like show a loading indicator when your app/tab loads? No
* Would you like to include Test framework and initial tests? No
* Would you like to use Azure Applications Insights for telemetry? No
* Default Tab name? (max 16 characters) My Tab
* What kind of Tab would you like to create? Personal (static)
* Do you require Azure AD Single-Sign-On support for the tab? No



***answer yeoman questionnaire***

 **Note** Most of the answers to these questions can be changed after creating the project. For example, the URL where the project will be hosted is not important at the time of creating or testing the project

Open Visual Studio Code using **code .** in command prompt.



***visual studio code***

**Step 1:** Manifest

* Navigate to ./src/manifest/manifest.json file and replace name and description section as per your tab Line of Business.
* Find "staticTabs": section and replace entityid - default-data to ‘personaldata’

**Step 2:** Update react component MyTab.tsx (response to user)

* Navigate to ./src/app/scripts/MyTab/MyTab.tsx
* Find export interface IMyTabState and define userEmailID typed variable

export interface IMyTabState extends ITeamsBaseComponentState {

    entityId?: string;

    userEmailID?: string;

}

* Go to public async componentWillMount() method and get user email Id from tab context.

this.setState({

                    entityId: context.entityId,

                    userEmailID: context.upn

                });

* Then go to render method to render user information.

replace <Text content={this.state.entityId} /> to

User Email ID: <Text content={this.state.userEmailID} />

This component will display user email id and sample button. This is the TypeScript React based class for your Tab. Locate the render() method and add user line of business.

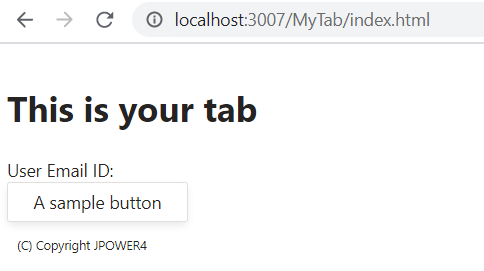
**Run your Personal tab locally**

Open command Prompt navigate to the project folder and execute the following command, to run your tab, you use the **gulp serve** command.

**gulp serve**

This will build and start a local web server for you to test your app. The command will also rebuild the application whenever you save a file in your project.

You should now be able to browse *http://localhost:3007/MyTab/index.html* to ensure that your tab is rendering.



Personal tab Localhost rendering

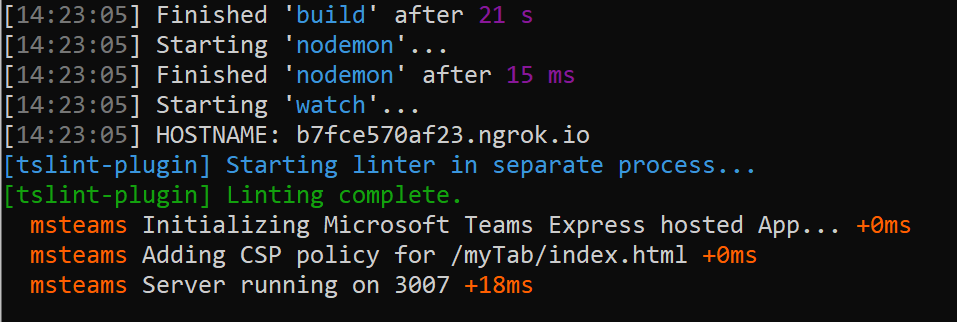
**Test the Personal Tab in Teams**

Open command Prompt, navigate to the project folder and execute the following command:

**gulp ngrok-serve**

This gulp task will run many other tasks that are displayed within the command-line console. The ngrok-serve task builds your project and starts a local web server (http://localhost:3007). It then starts ngrok with a random subdomain that creates a secure URL to your local webserver.

In development, testing can be done using the tool ngrok that creates a secure rotatable URL to your local HTTP webserver. ngrok is included as a dependency within the project so there is nothing to setup or configure.



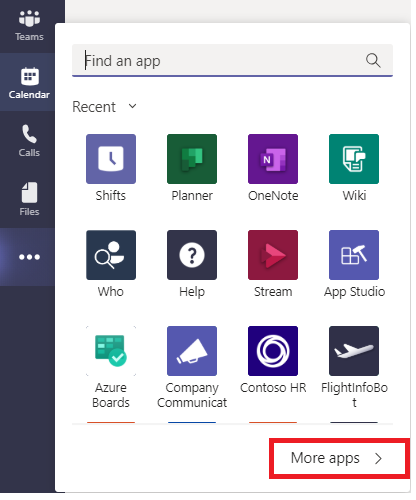
gulp ngrok serve execution

ngrok has created the temporary URL b7fce570af23.ngrok.io that will map to our locally running web server.

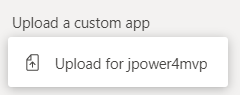
**Install the personal tab in Microsoft Teams**

Now let's install the app in Microsoft Teams. In the browser, navigate to <https://teams.microsoft.com> and sign in with the credentials of a Work and School account.

Microsoft Teams is available for use as a web client, desktop client and a mobile client. Using the app bar navigation menu, select the More added apps button. Then select More apps followed by Upload a custom app and then Upload for me or my teams.

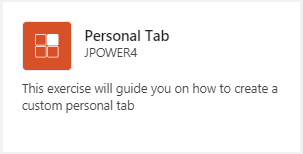


***Add apps to MS Teams***

In the file dialog that appears, select the Microsoft Teams package in your project. This app package is a ZIP file that can be found in the projects **./package** folder.

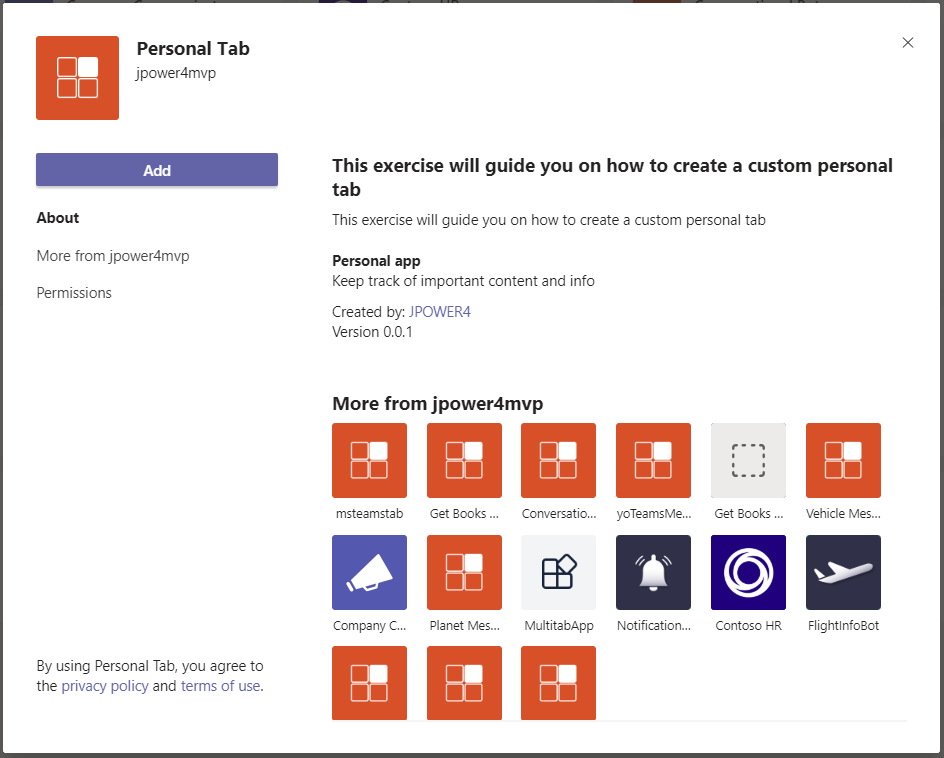
***upload apps***

After installation, you can be able to see the app in the apps list.



***personal apps installed***

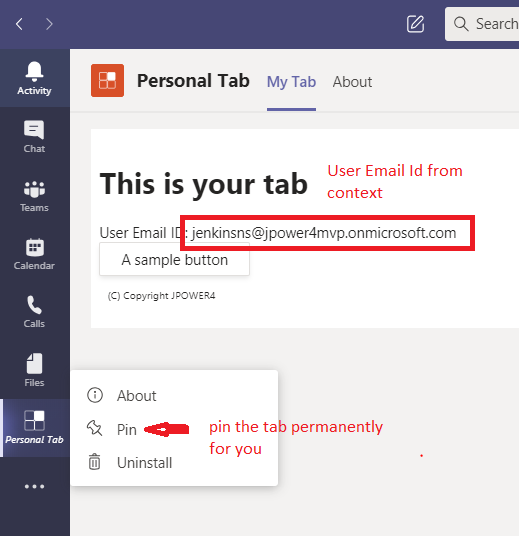
Click the app, Microsoft Teams will display a summary of the app.



***Add the personal tab***

**Testing Personal tab**

* **Click Add button to navigate to the personal tab to test.**
* **Also you can pin the personal tab in far-left navigation bar permanently.**
* **Or select the ... menu and choose your app from the list.**

****

***personal tab output***

**In this exercise, you have created a personal tab for a custom Microsoft Teams app.**