MIRACLE
SOFTWARE SYSTEMS

Getting Hands On with Progressive Web Applications

Hands-On Workshop | Digital Summit '18

Miracle Innovation Labs

Miracle Software Systems, Inc.

MIRACLE
SOFTWARE SYSTEMS

Getting Hands On with Progressive Web Applications

Goal

This document contains a step-by-step process for pushing an Angular application to Firebase and how to convert an Angular application to PWA.

This guide was prepared by **Miracle's Innovation Labs.**

Pre-Requisites

All attendees must have their workstation (with Internet) to participate in the lab (Both PC and MAC are compatible). The following pre-requisites will help you to make the Hands-on Lab experience easier.

- Google Account
- Node JS and npm installed
- Text editor (Visual Studio Code is preffered)

Technology Involved

- Angular 6 Framework
- HTML, CSS, Javascript
- Typescript

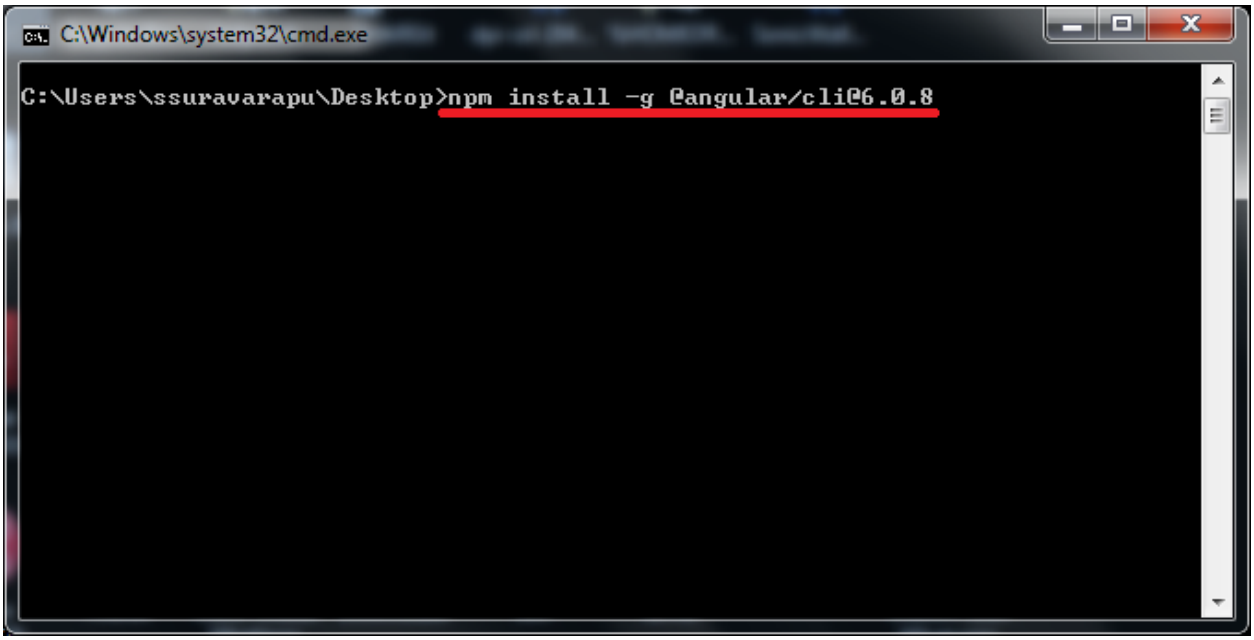
Lab Steps

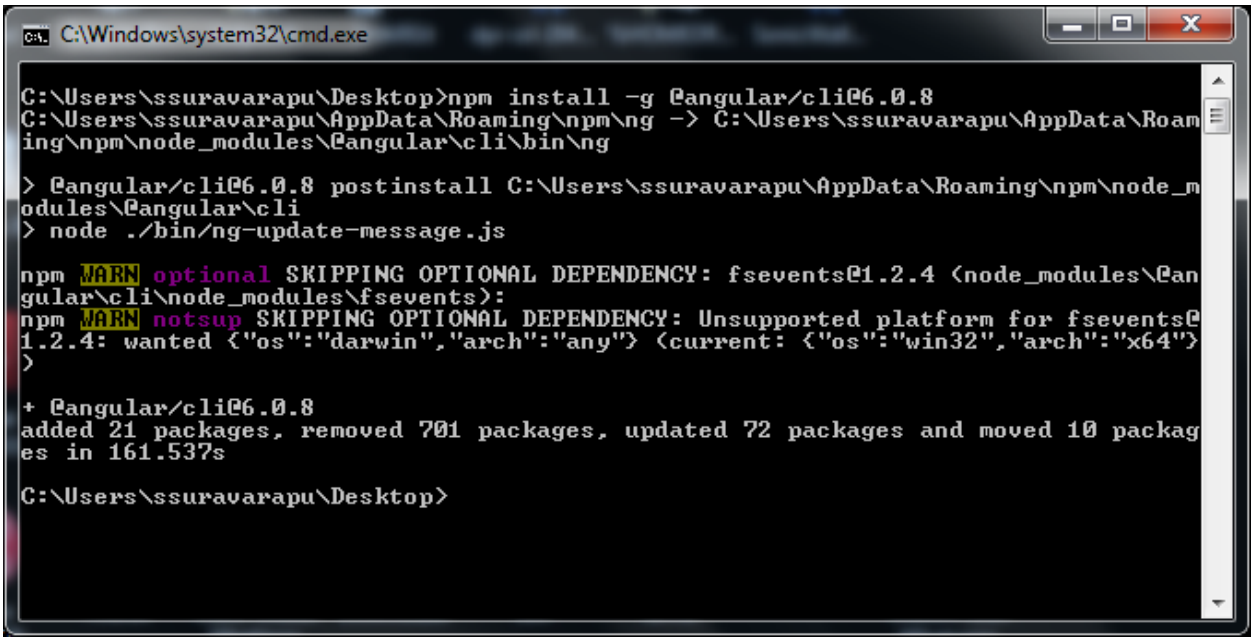
So, let us get started with the application!

Step #1 | Angular 6 Installation

Open the folder which is downloaded from Github account and open command prompt. To install Angular 6, give the following command, (Ignore if already installed)

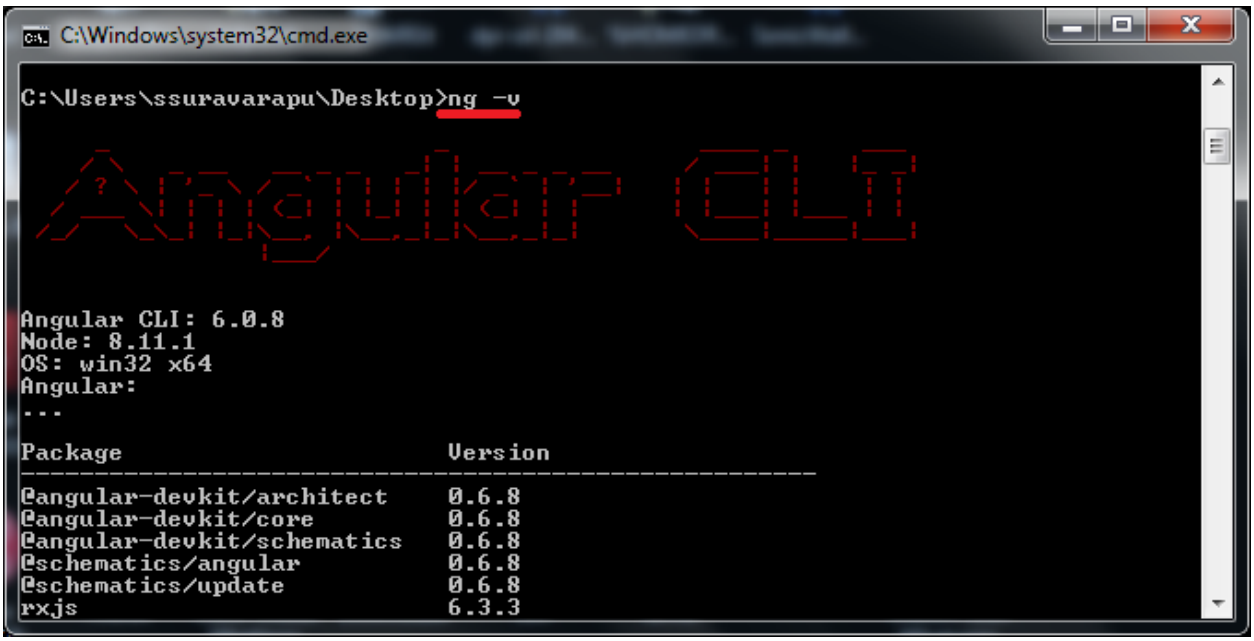
npm install - g @angular/cli@6.0.8





To check whether Angular is installed or not, provide the below command,

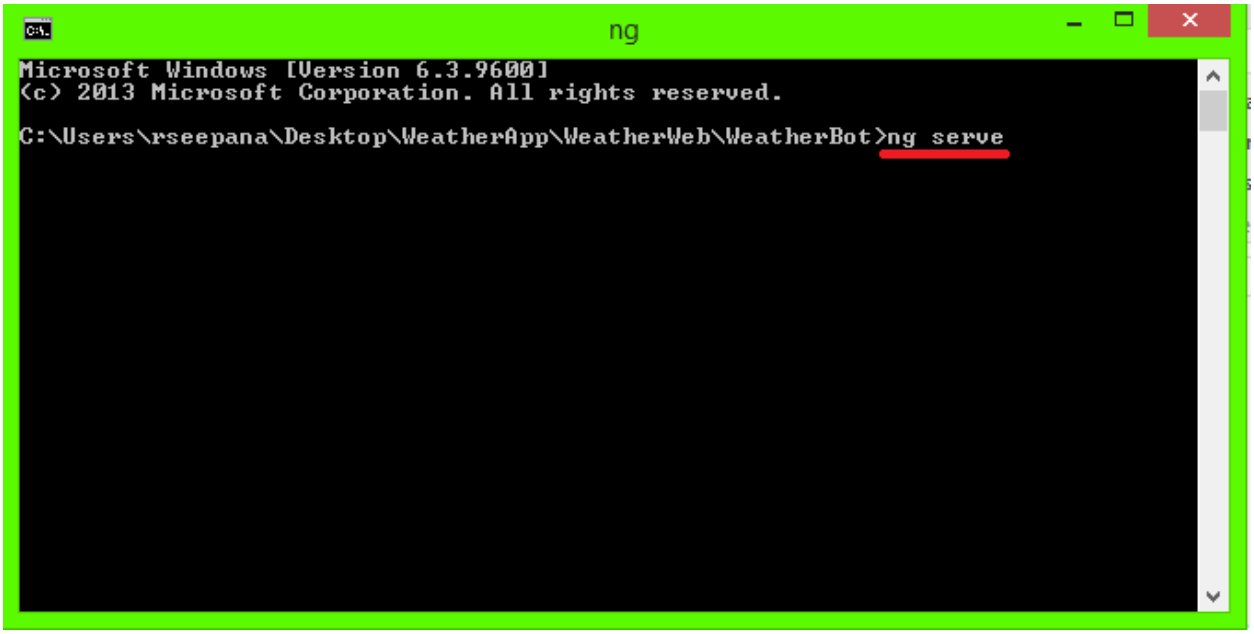
ng - v



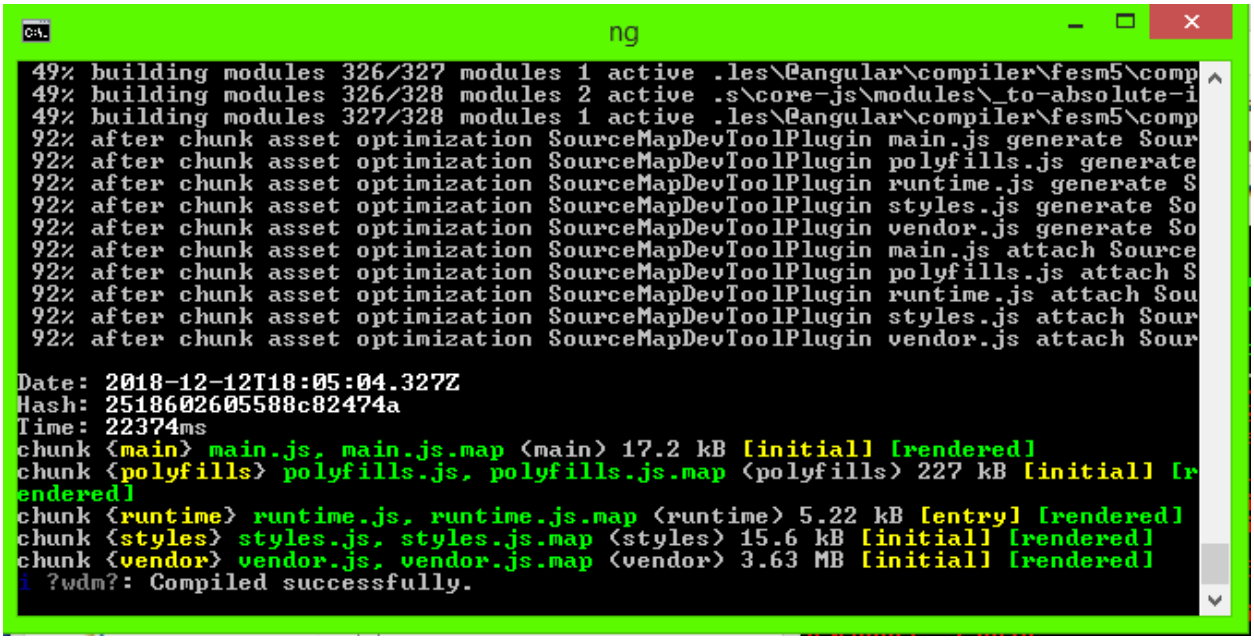
Step #2 | Run the Web App Locally

To run the downloaded application locally give the below command,

ng serve



You will be displayed as below if your app is running successfully

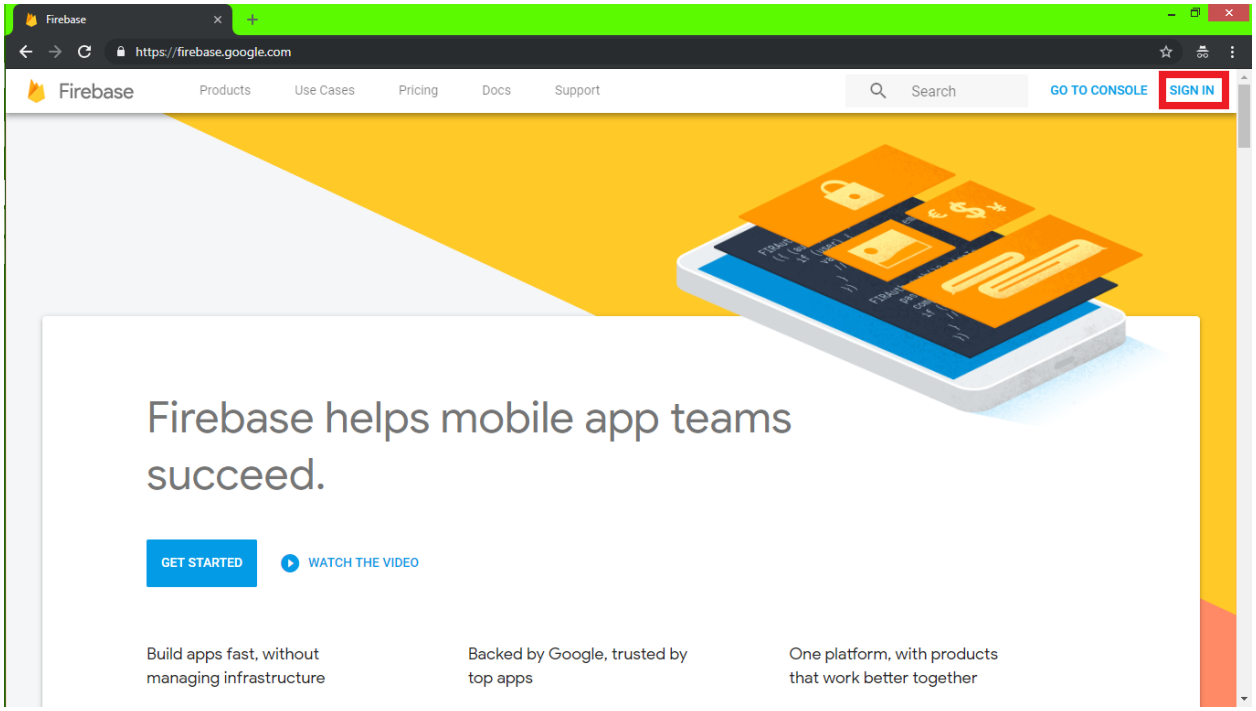


Open the browser and type **localhost:4200**. Now the weather application will be opened.

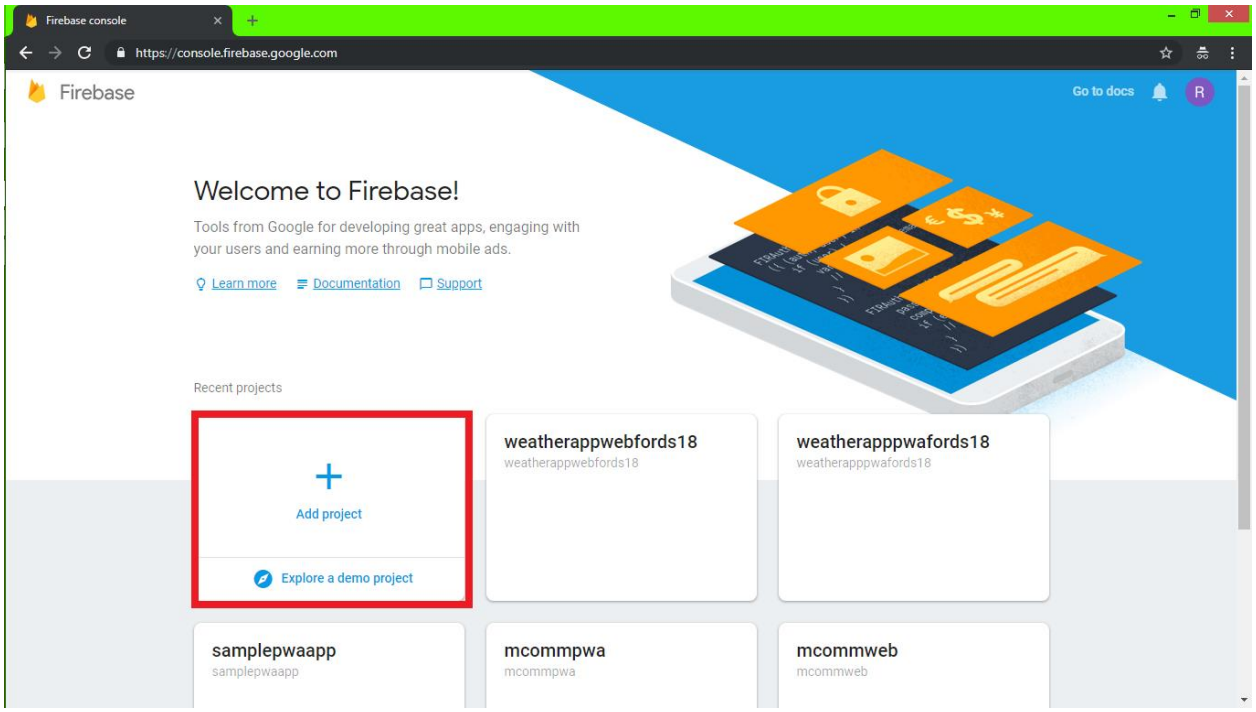
Step #3 | Create Firebase Account and Create a Project

To push the application into Firebase, we need to create a Firebase account. For

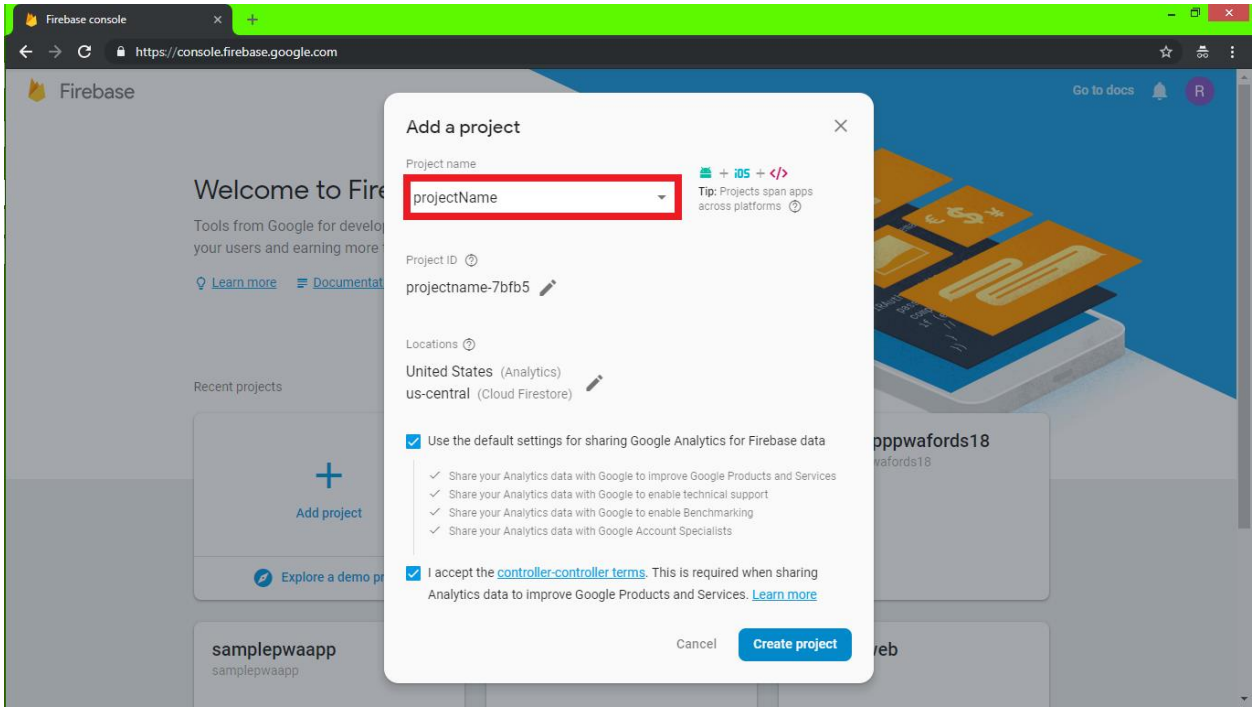
that open the browser, and type **https://firebase.google.com/** and click on **SIGN IN**



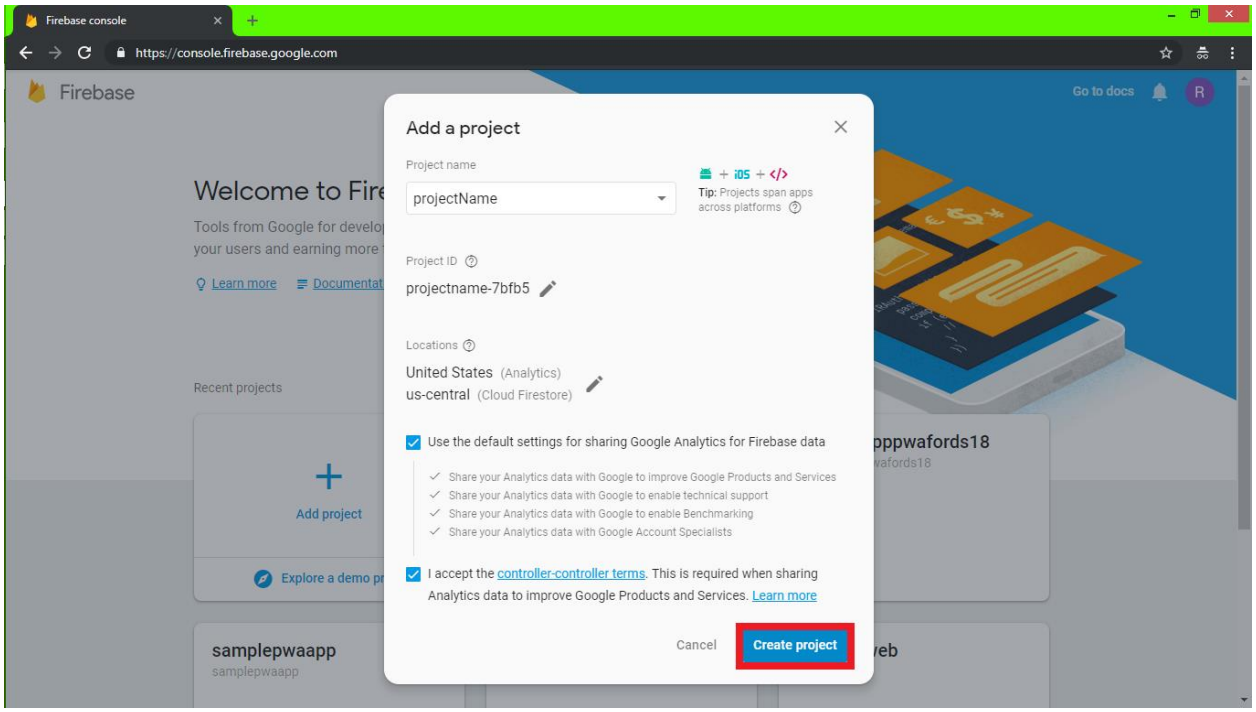
Provide your Gmail credentials to login. After successful login, you will be seeing the home page. Click on **Add project**



Now, enter your project name in the given field.



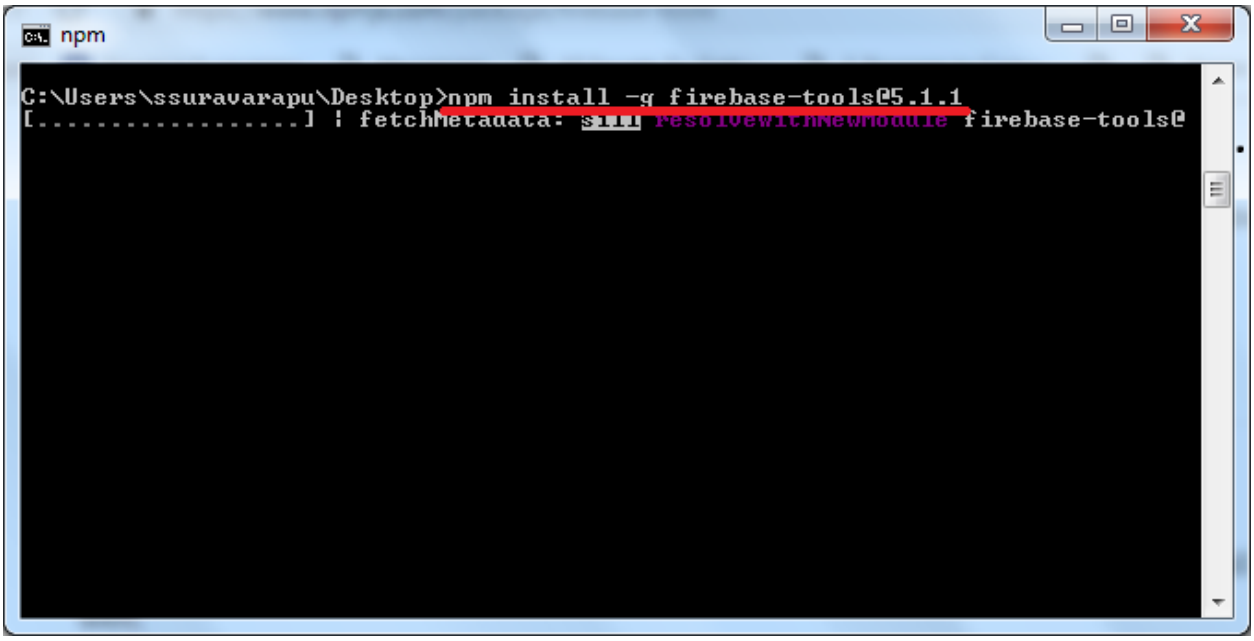
Click on **Create project**.

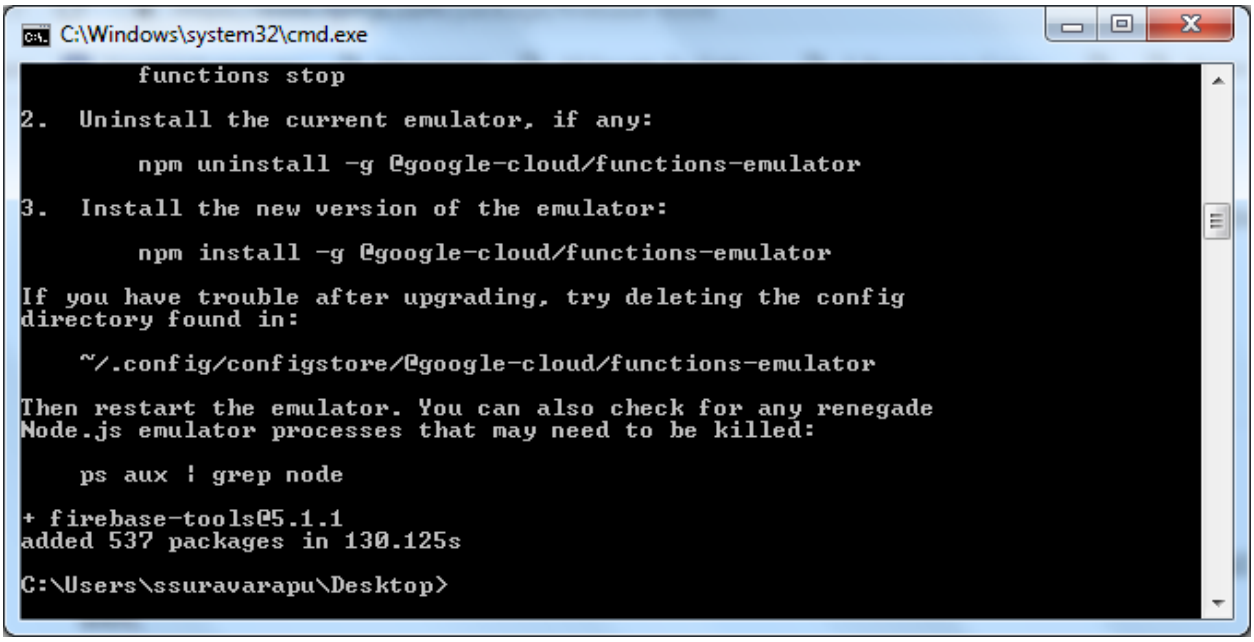


Step #4 | Install Firebase and Authentication in CLI

Open the command prompt and type

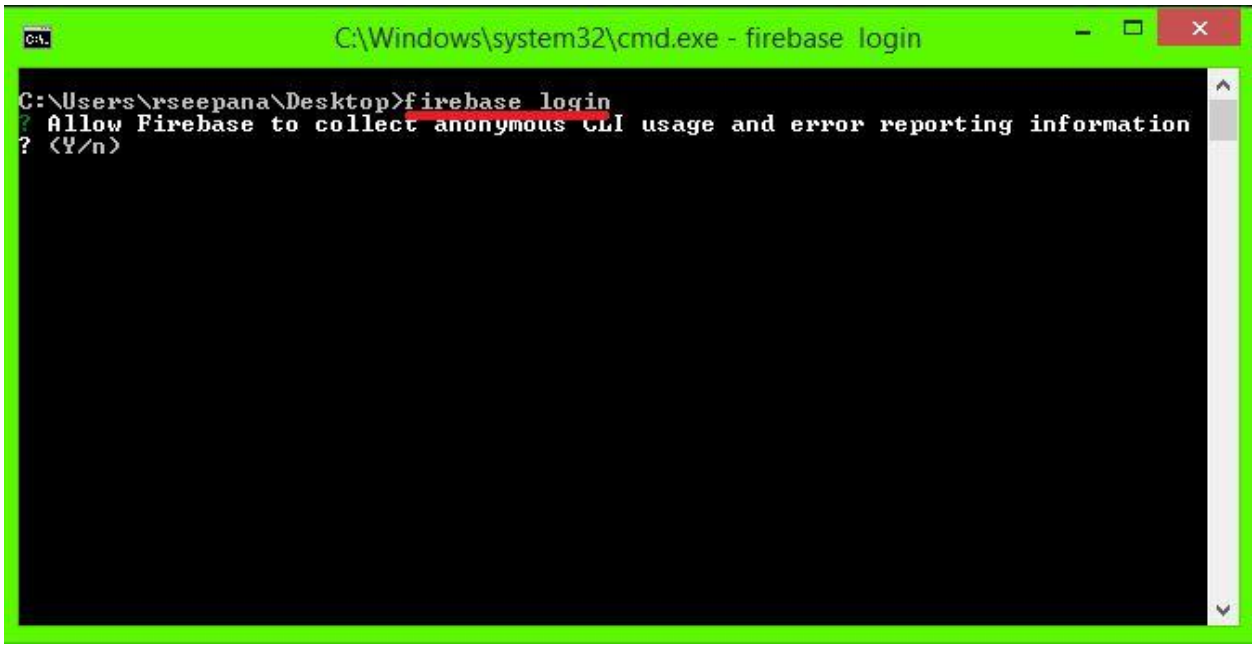
npm install -g firebase-tools@5.1.1



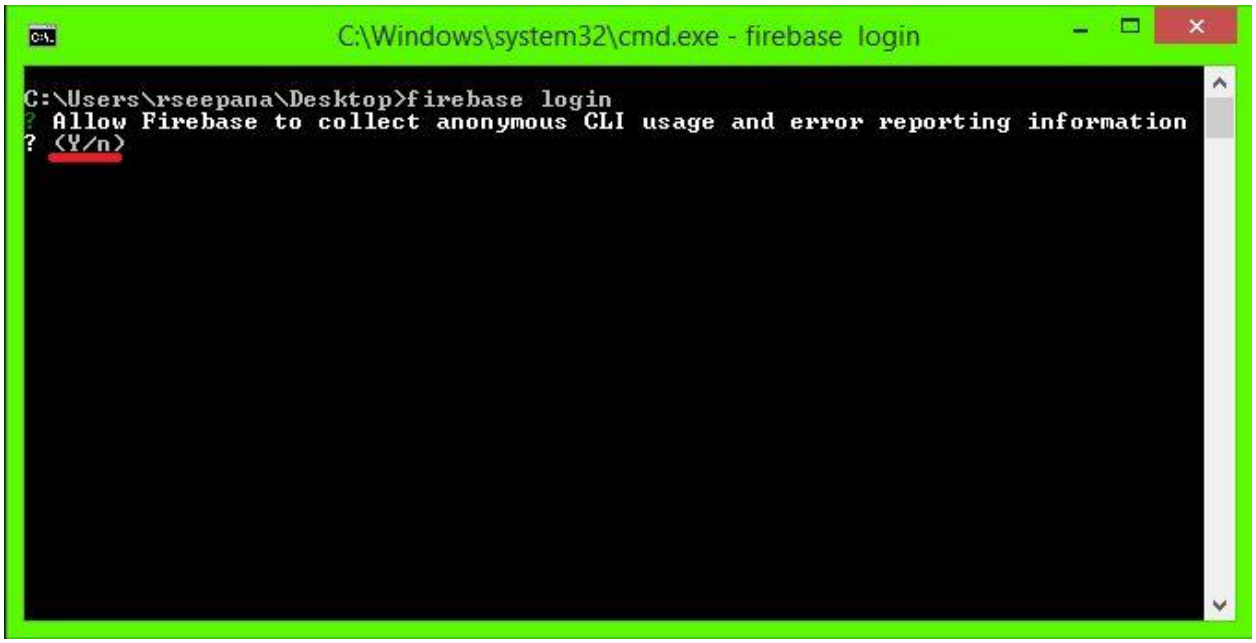


After successful firebase tools installation, we need to login with firebase credentials in command prompt. For that provide the following command

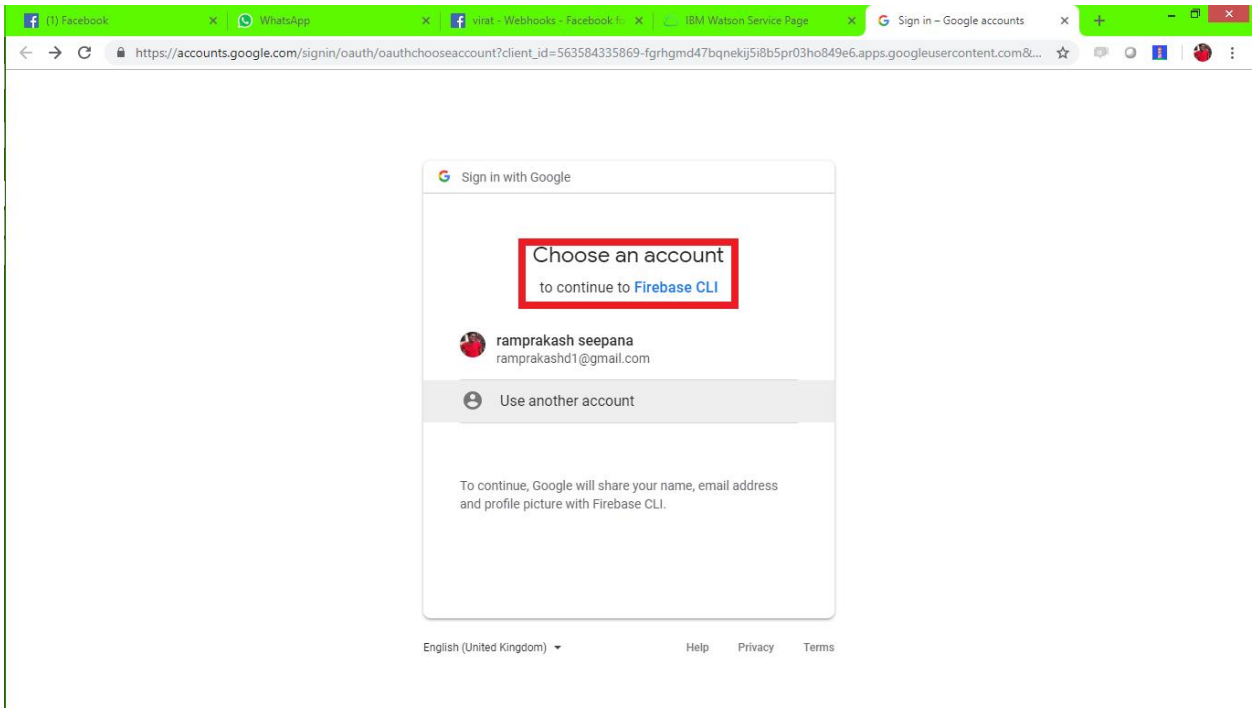
firebase login



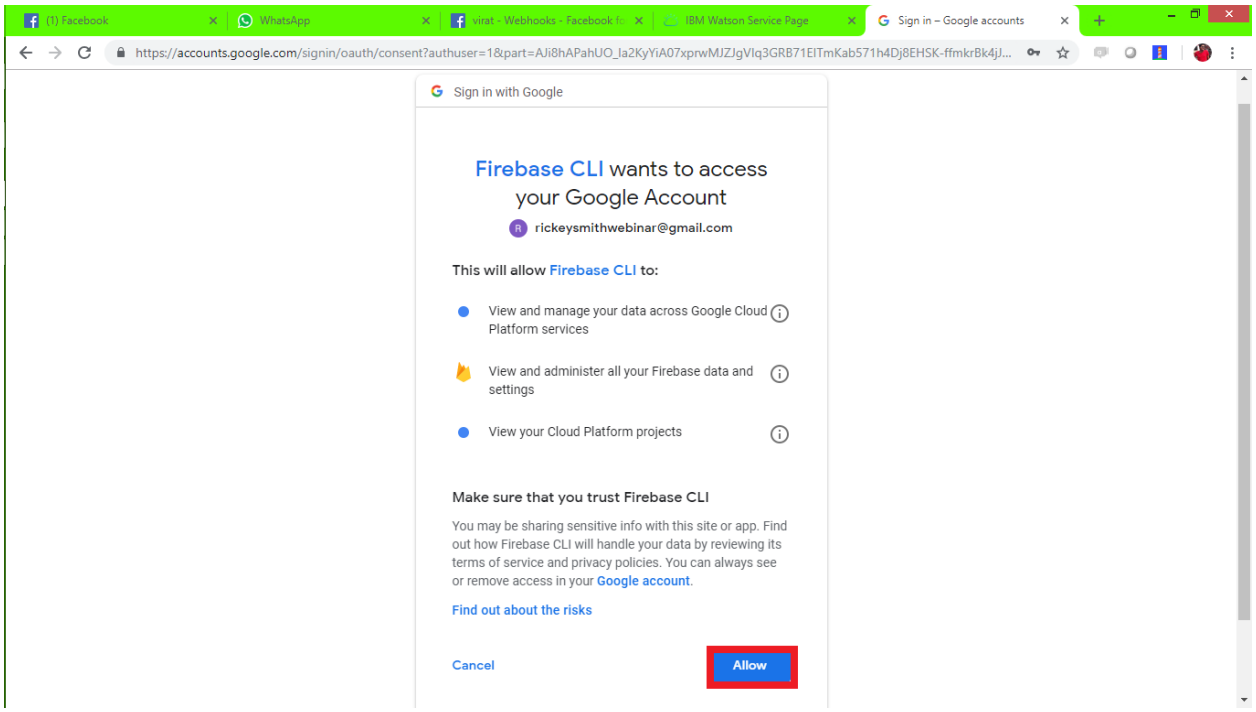
Now it asks "Allow Firebase to collect anonymous CLI usage and error reporting information?". Give **Y** and enter



You will be routed to a web browser and asks to login for Firebase authentication. Enter your Firebase account credentials.



After successful login, you will be asked to allow the access to your Google account. Click on **Allow**.

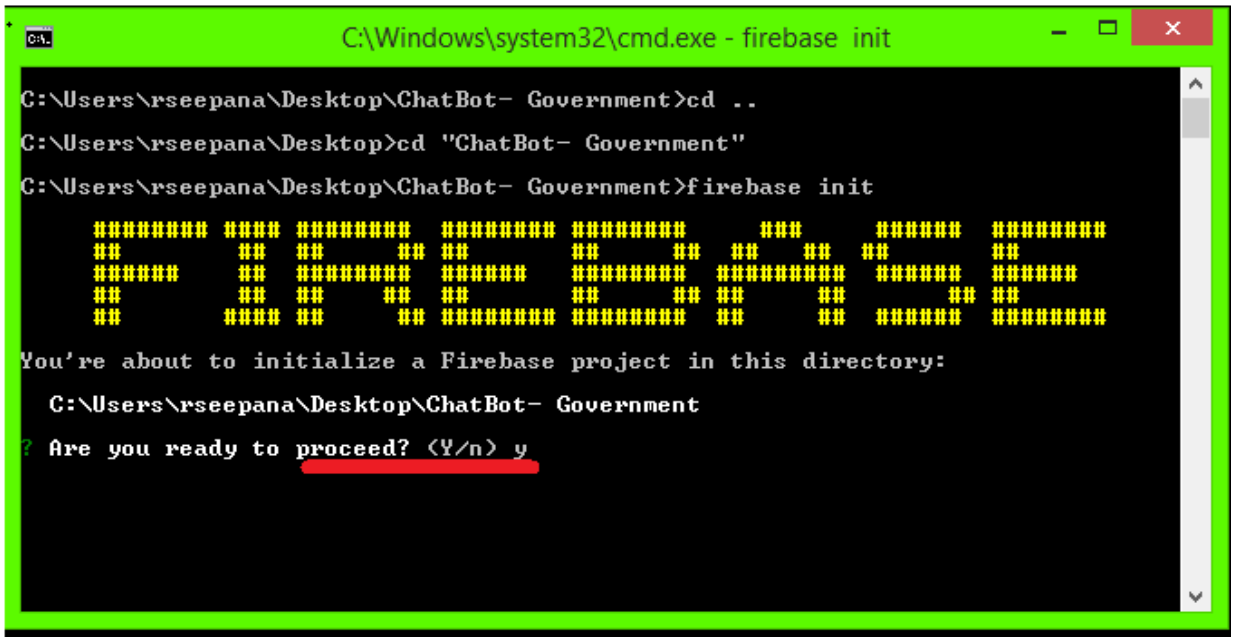


Step #5 | Configuring Firebase in the App

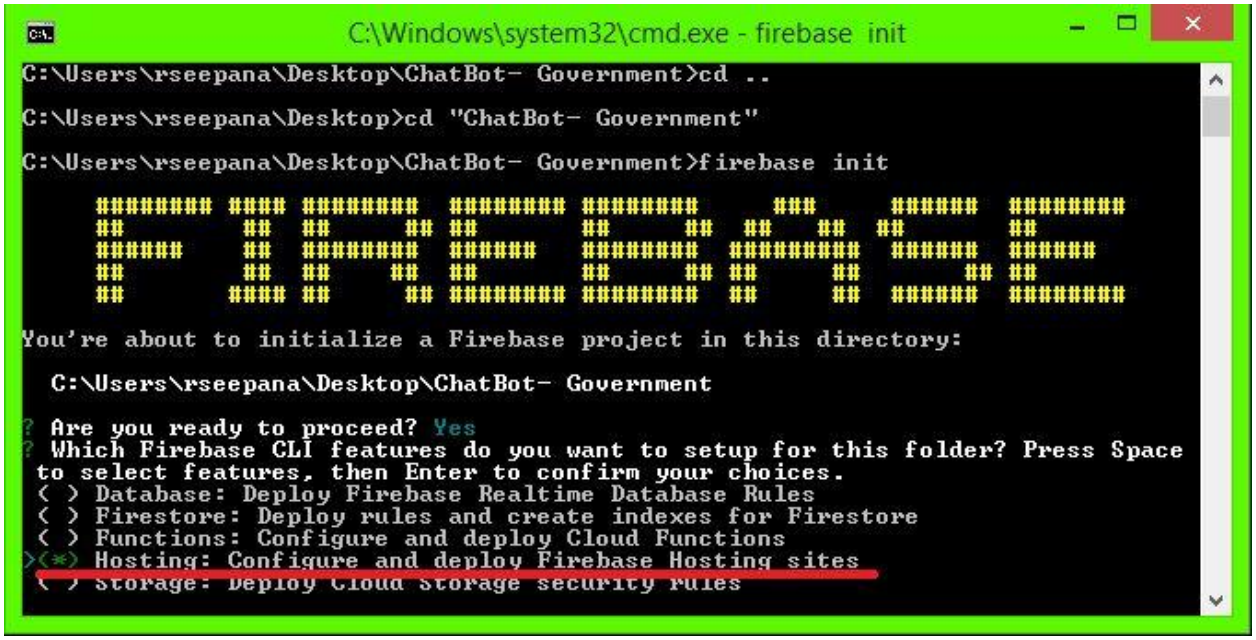
As of now, you are done with authentication for Firebase account through command prompt. Now we need to initilize Firebase configuration.

For that we need to give the following command: **firebase init**

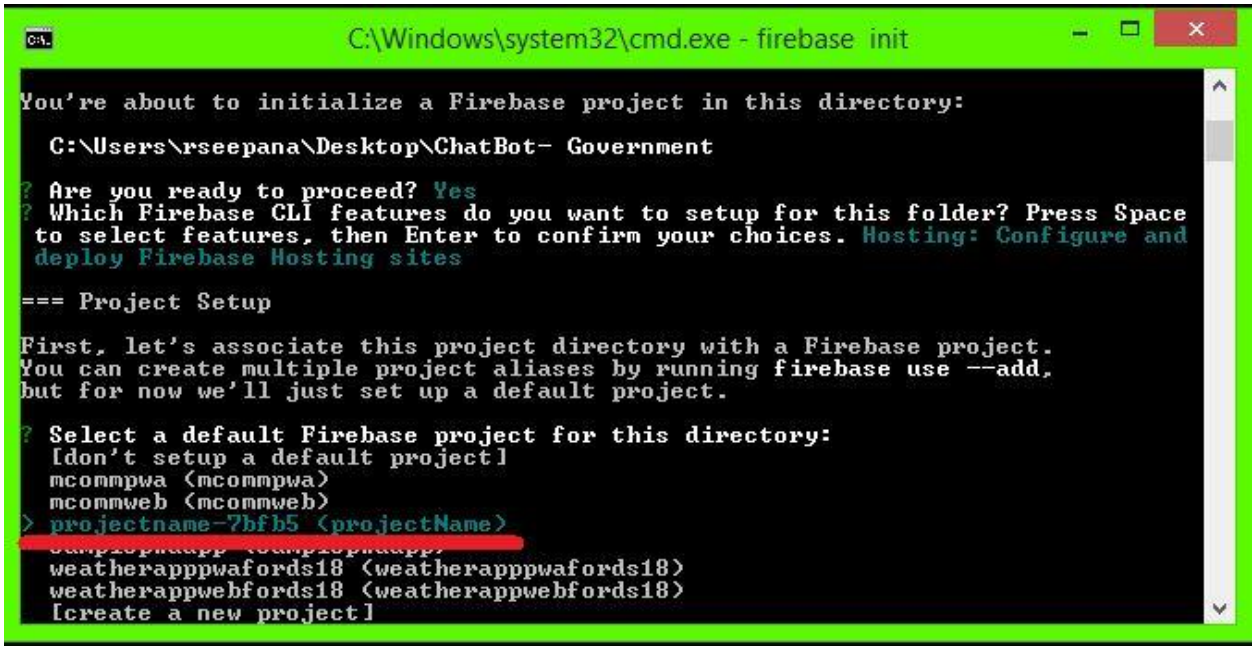
It asks "Are you ready to proceed?" enter **Y**



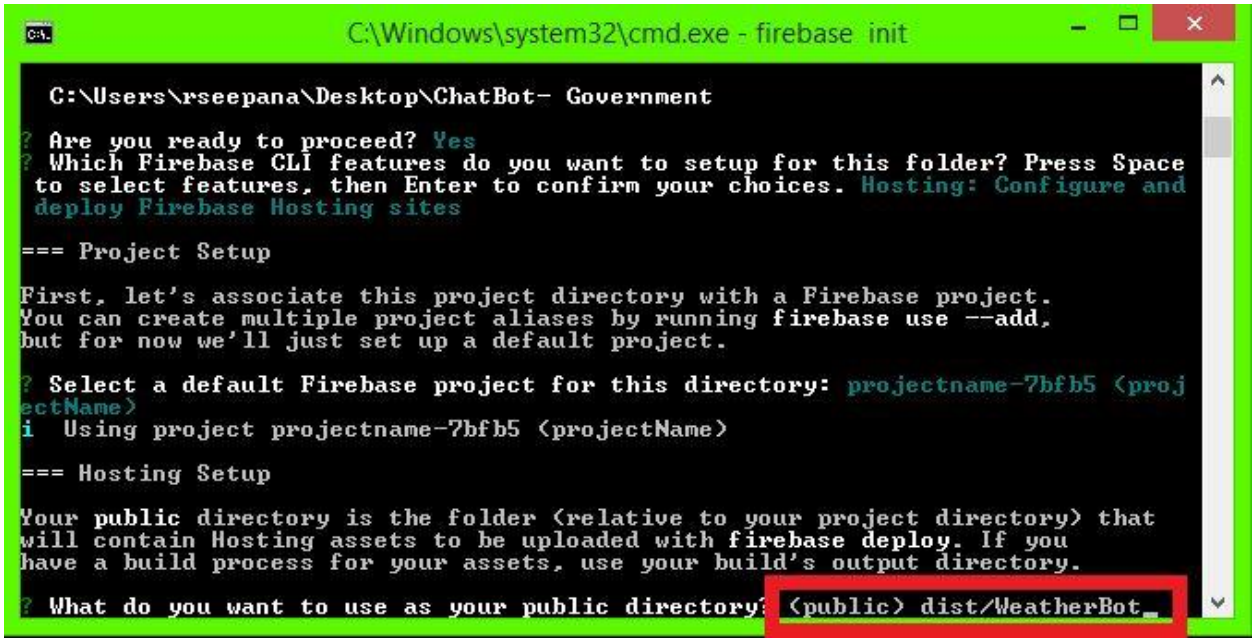
It asks to select feature. Select **Hosting** and enter



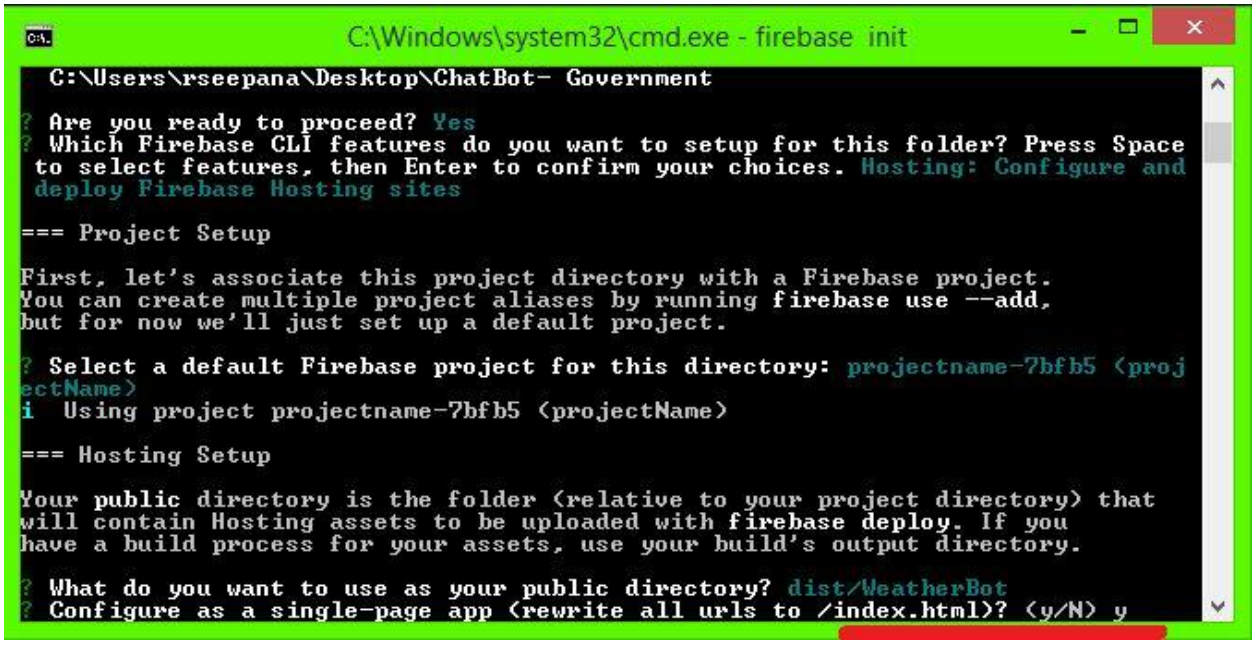
It asks to select Firebase project to deploy the app. Select the project name that we have created in the account.



It asks to enter the directory path. Enter **dist/WeatherBot** and hit enter



Then, it asks to **Configure as a single page app?** Type **Y** and enter



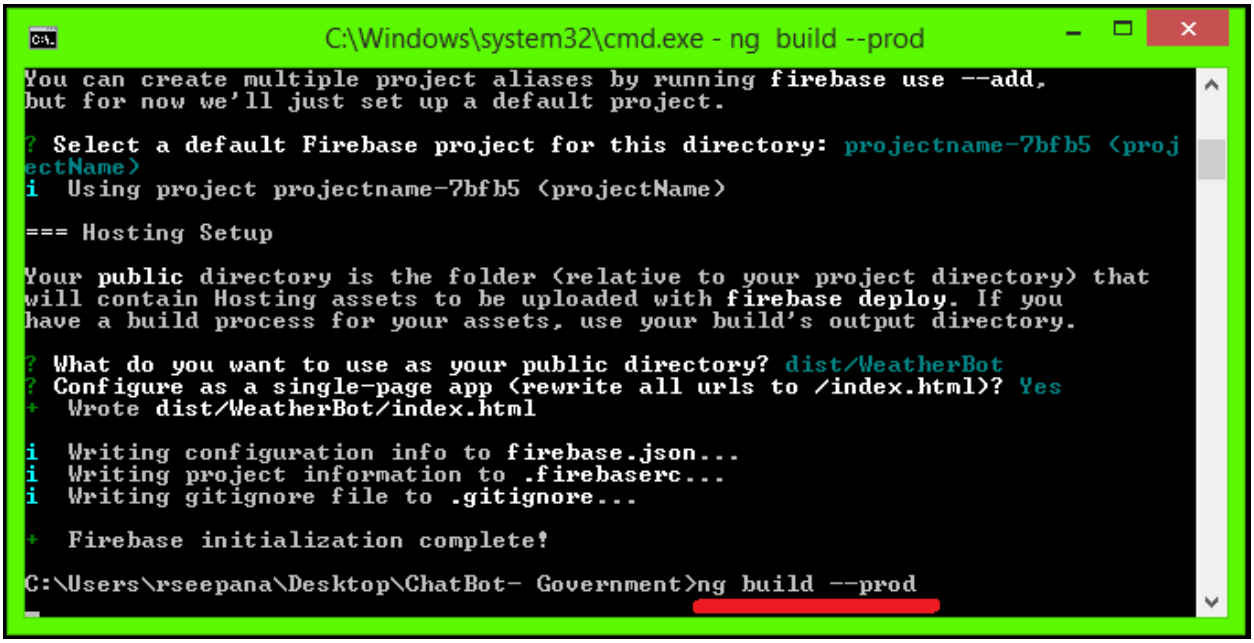
Now if it asks to overwrite the index.html file, type **N** and enter.

Step #6 | Build and Deploy the Application

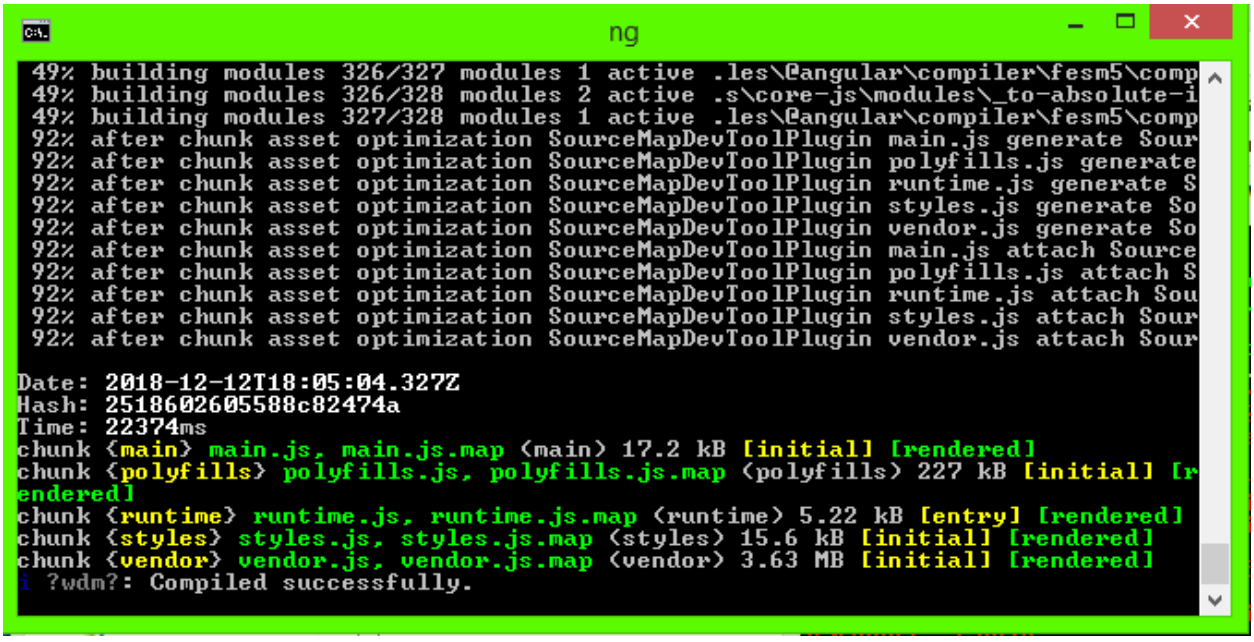
As of now, we are done with configuring the Firebase initialization. We need to deploy the application into Fireabase. For that we need to build the application and deploy it.

To build the application, give the below command,

ng build --prod

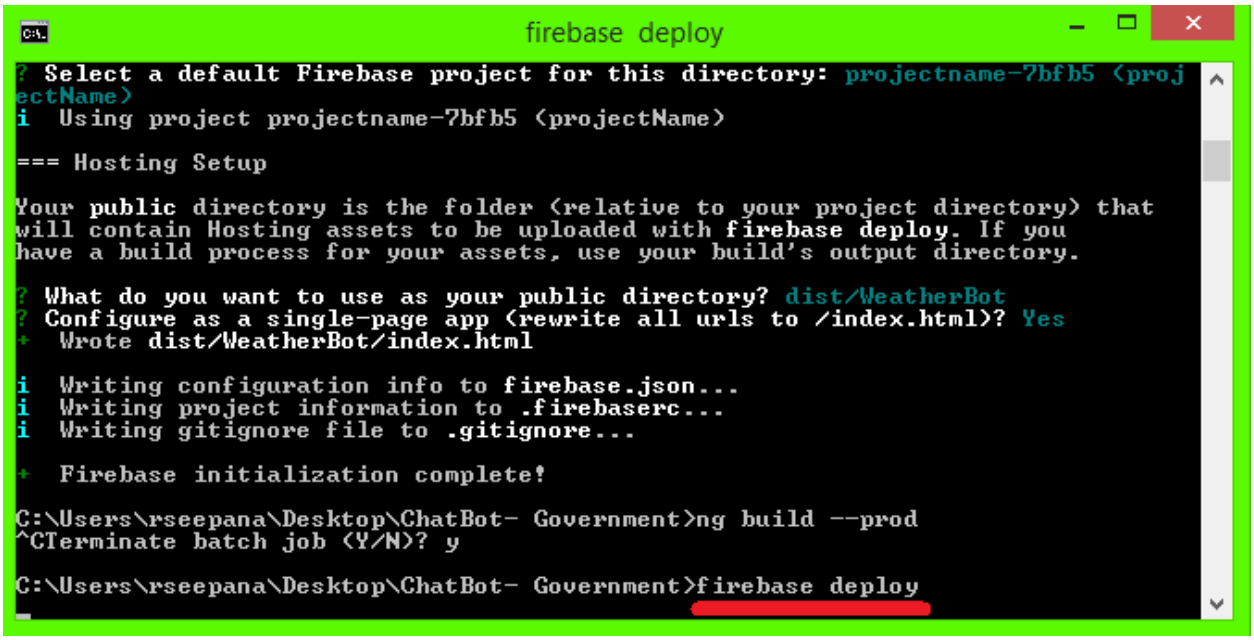


After successful build of the app you will be displayed like this,

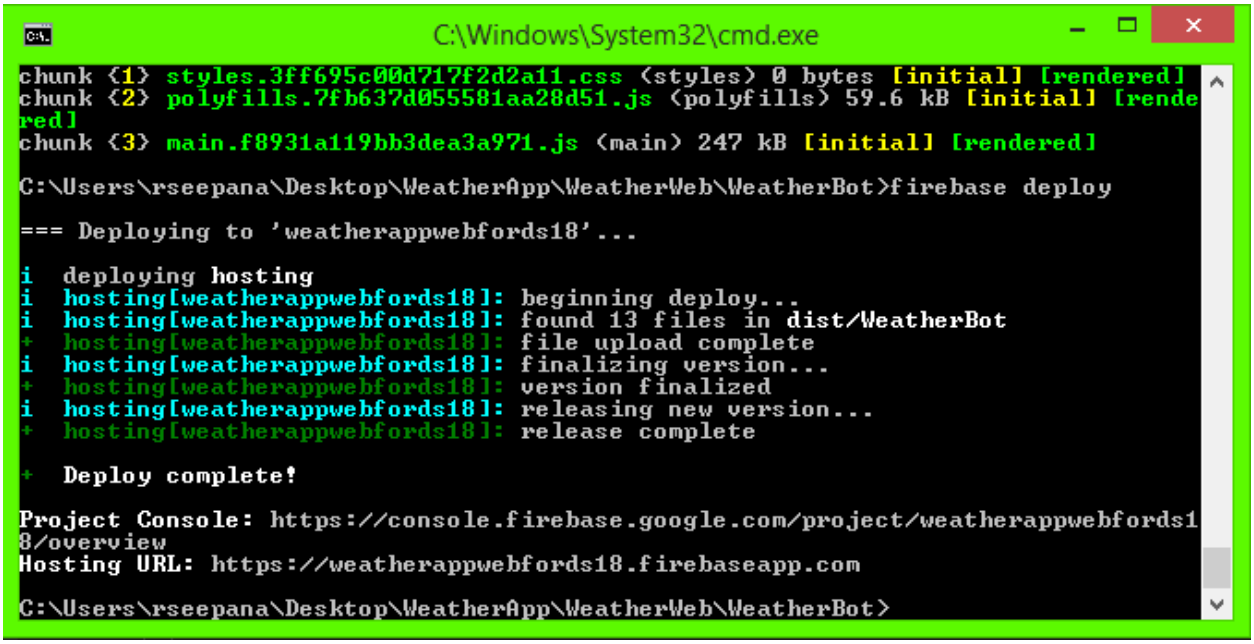


Now to deploy our application into firebase, give the below command,

firebase deploy



If the app is deployed in Firebase then you will be given with the hosted URL as shown below:



We are done with deploying a web application in the Firebase. Open the URL and check if the application is working properly or not.

Step #7 | Converting Web App to PWA

Let us see how to convert an Angular Application to PWA. Give the below command,

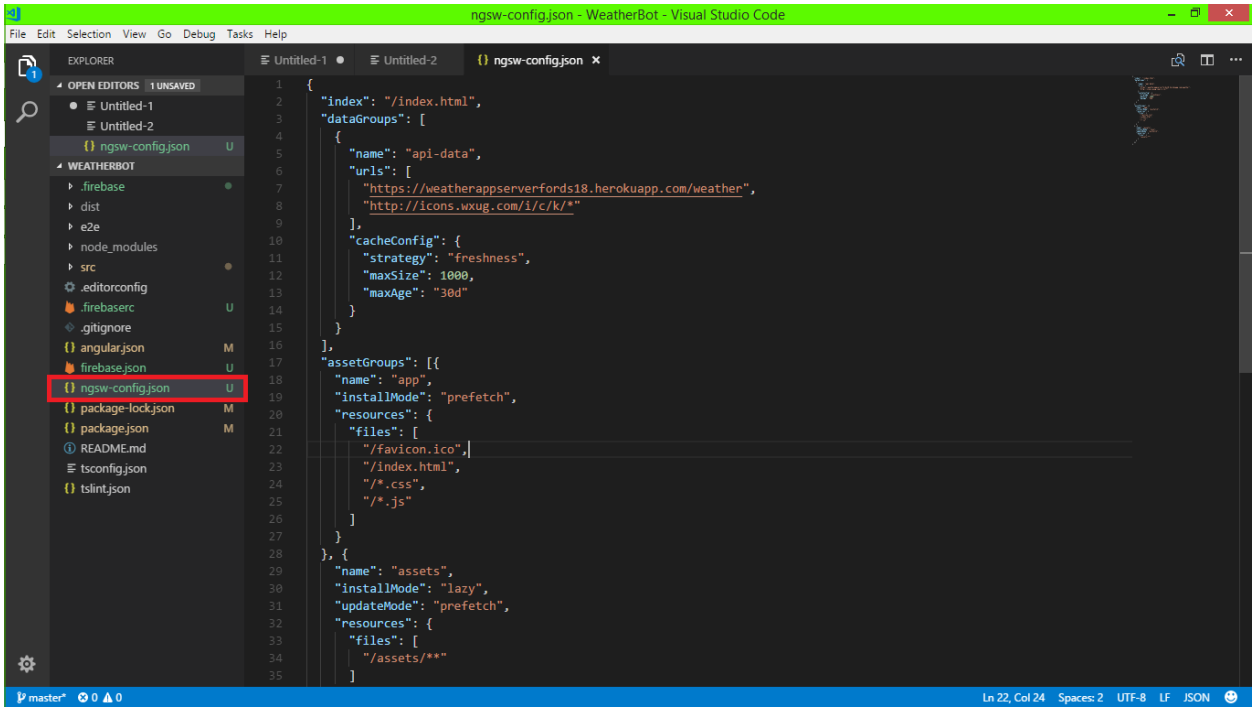
ng add @angular/pwa

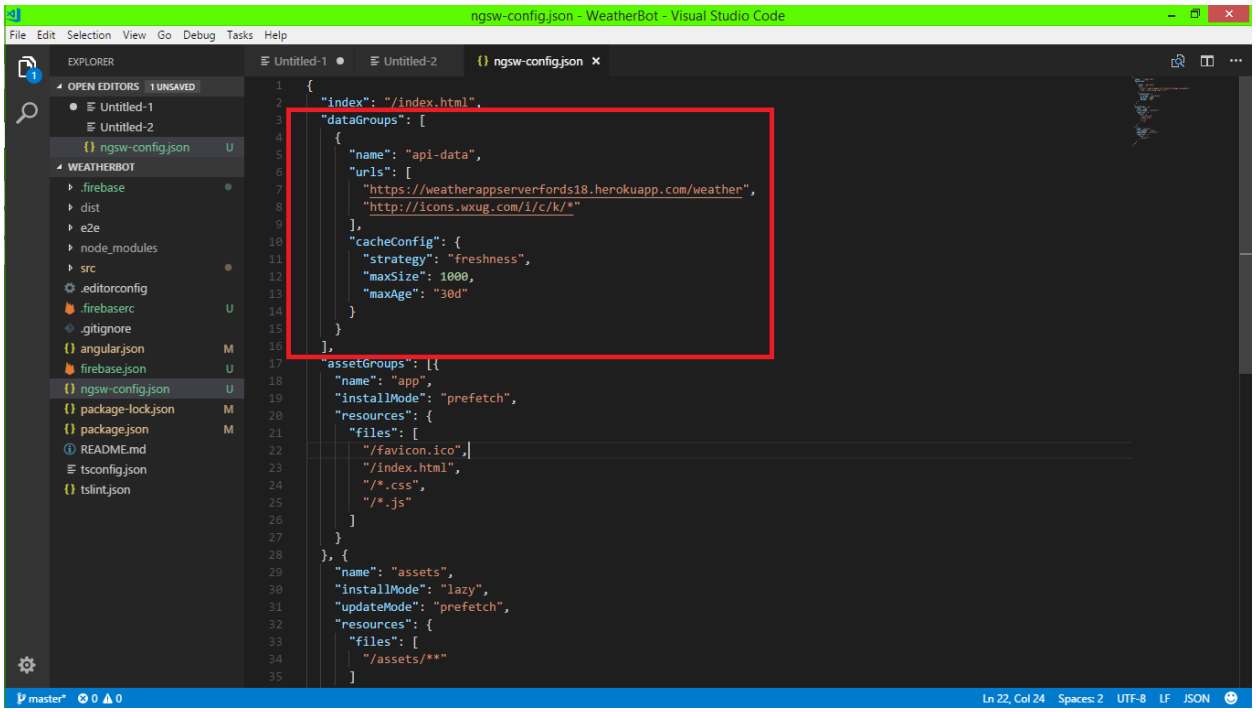
Now, you will be added with required files to support the service worker. So that we will be able to use the features of Progressive enhancements. To enhance the application, follow the below step.

Step #8 | PWA Features Enhancements

Open **app.component.ts** file and replace the code available in **#app.component.ts** in **cheat sheet** file. This updated code is to enhance the PWA by adding code for Updating App and Subscribe/Unsubscribe notifications.

To enhance the offline capability, we need to add the urls which we are going to use in the application in **ngsw-config.json** file. For this open **cheet sheet** file and copy the code in **#ngsw-config.json** and replace in this file. With the above code, we have enhanced the offline capability for **GET** APIs.



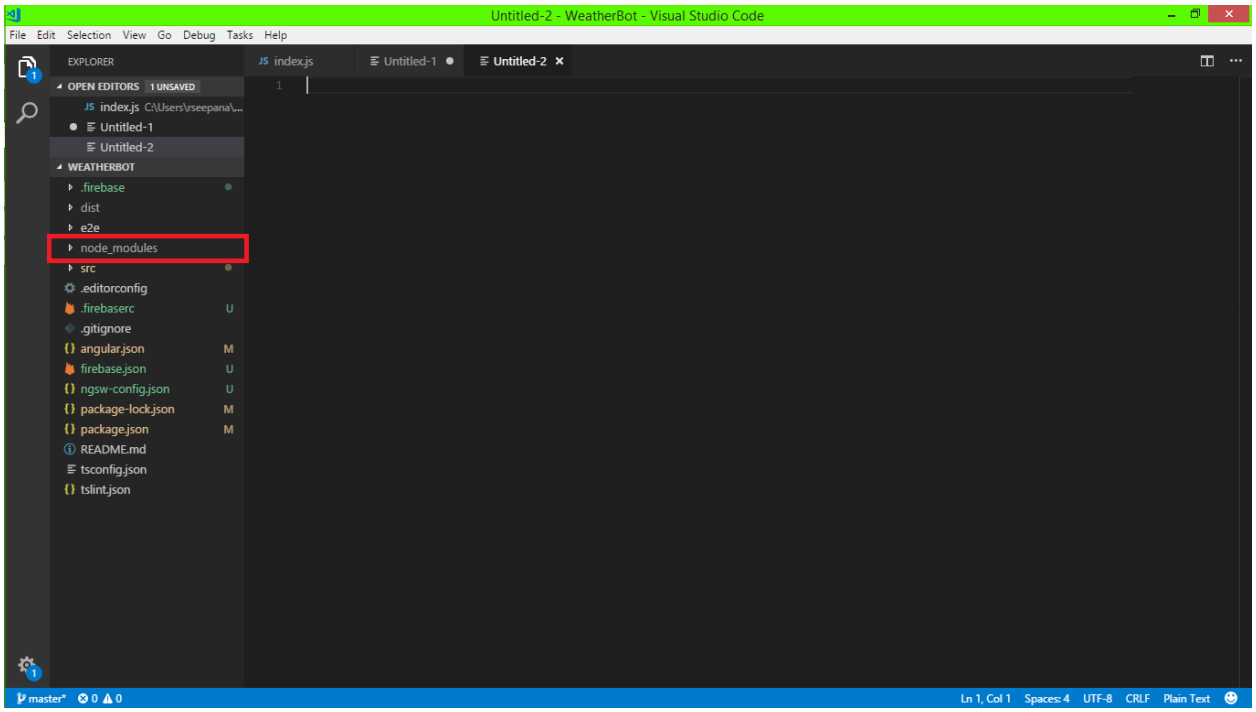


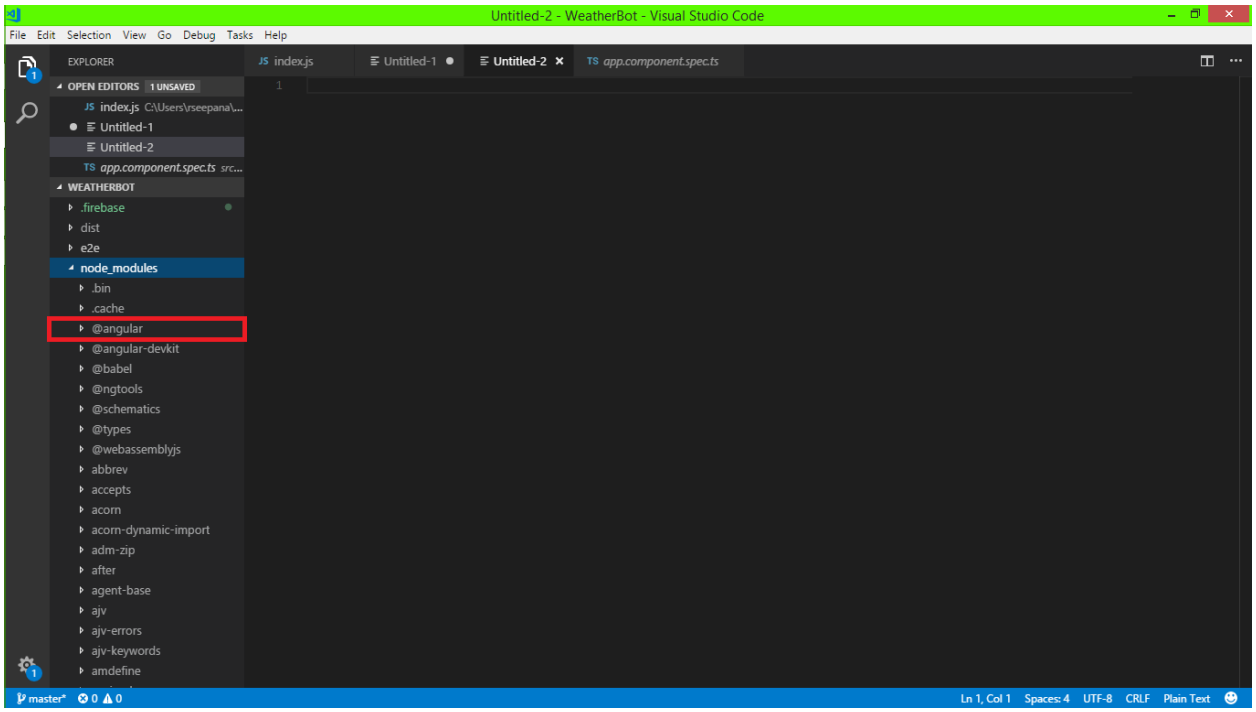
Step #9 | Adding event Listeners in Node Modules

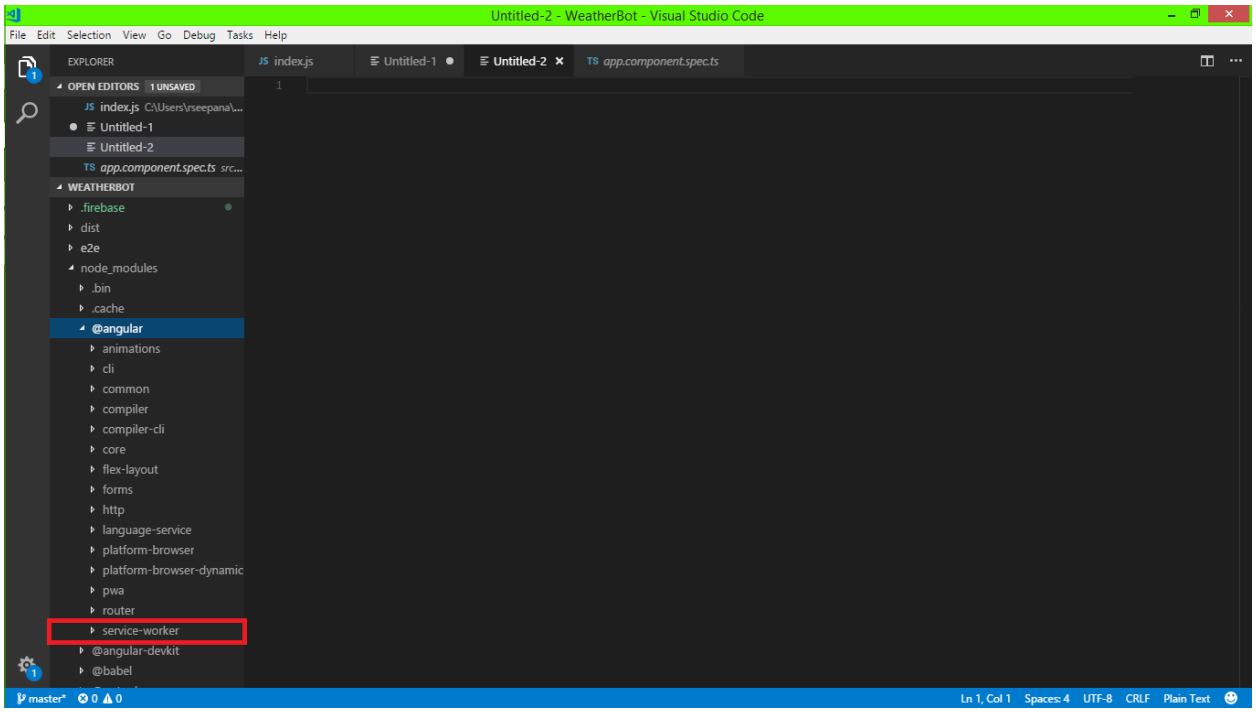
As of now Angular 6 is not fully supported for notifications. When user clicks on the notification, it should redirect to the respective URL. For this, we need to add ***notificationclick*** **eventListener** in **ngsw-worker.ts**.

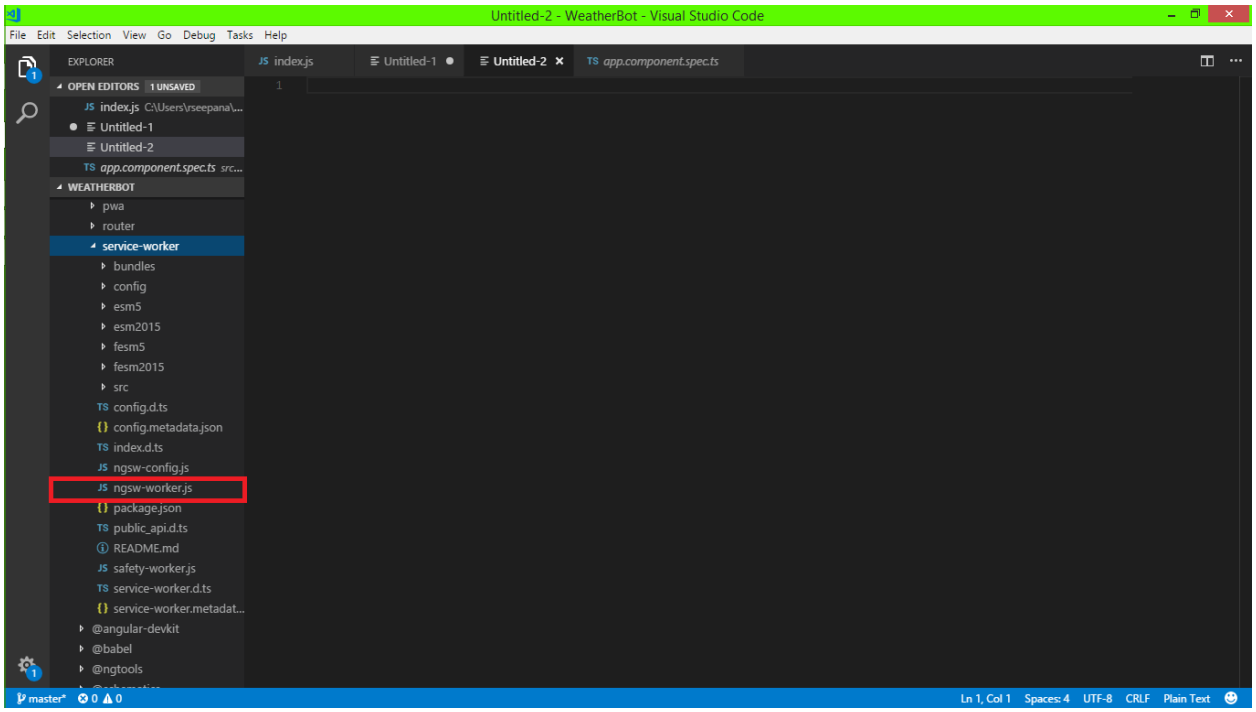
Note - As of now Angular Service Worker does not provide the functionality to handle the incoming push notifications. So, we have to add the following, **[project-name]\node_modules\@angular\service-worker\ngsw-worker.js**

Follow the following to add the event listener,









Now copy the #**ngsw-worker.js** code in **cheet sheet** and paste in **ngsw-worker.js** file. With this we are done with adding code for pwa functionality enhancements.

To deploy the updated code in firebase, run the following command again,
ng build --prod

firebase deploy

After successful app deployment we can open the app with the url. Now, we can see the application with home page like this:

Now, let us see the features of PWA and how the app works in mobile device.

For any questions regarding the lab please feel free to reach out to innovation@miraclesoft.com. **We hope you enjoyed the workshop!**