NAME: OLAIYA BISOYE OLORUNFEMI

TRACK: SOFTWARE DEVELOPMENT (WEB DEVELOPMENT)

TASK: WEEK 6

DATE OF SUBMISSION: 16th December, 2019

GITHUB USERNAME: bisoyefemi

Project Link: https://gitlab.com/bisoyefemi/week-six-task.git

Week 6 Task:

1. Explore google firebase authentication and firebase cloud storage
2. Explore highcharts liabrary for plotting barcharts and linecharts
3. Create a project that uses firebase authentication for login, or a project that uses cloud storage to store inputs from a form
4. Create a project to plot any random set of values using highcharts on a barchart or line chart

Solution:

Project 1: Create a project that uses firebase authentication for login

What is Firebase: Firebase is a complete backend as a service with many different features we can plug into software applications. To get started, I created an account on firebase.google.com and created a project I called “The Guide Project”. I also downloaded an extension called “Live Server” by Ritwick Dey. This extension allows you launch a development local server with live reload feature for static and dynamic sites. For this project I used HTML, CSS, JS and Materialize Design to develop the project.

I created a project for a hairline store that allows users log in to create new guides to be displayed on the website for visitors to read. This project can be found in src file.

After creating a project in my firebase account, I navigated to the Authentication tab to enable the Email/Password option for the sign-in method of authentication. Then, I enabled the database setup security rules in “Start in test mode”

Next thing for me was to connect my front-end application to the firebase backend sever. To do this, I copied the code snippet and I pasted it at the bottom of my HTML file before the script tags.

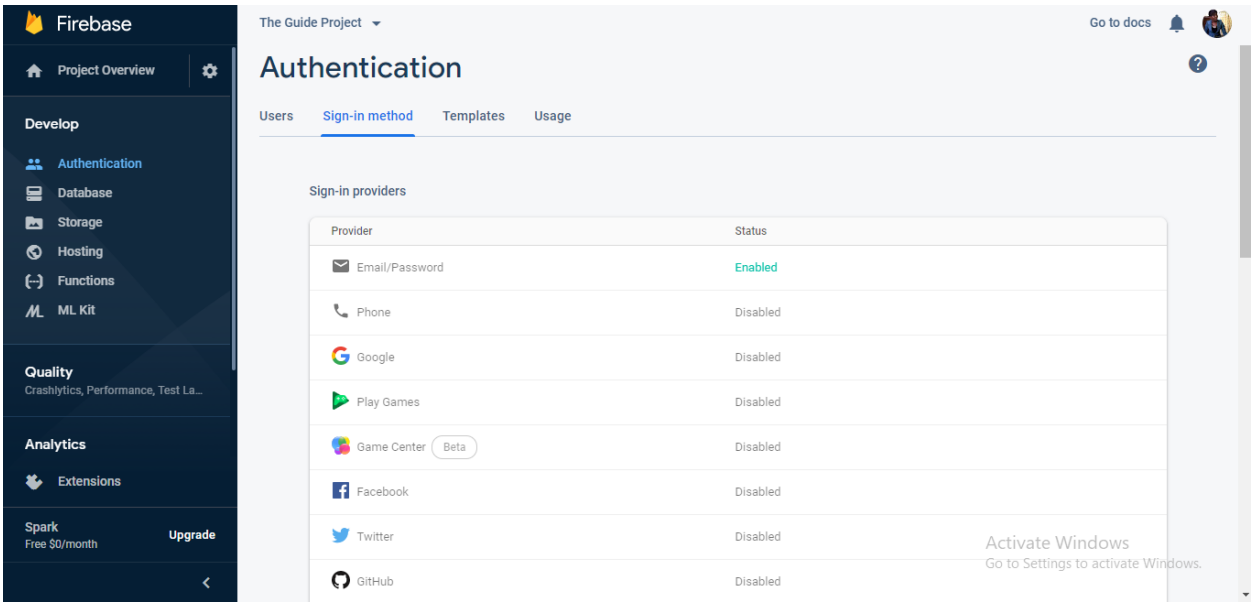


Figure 1.1: Screenshot showing Email/Password authentication enabled

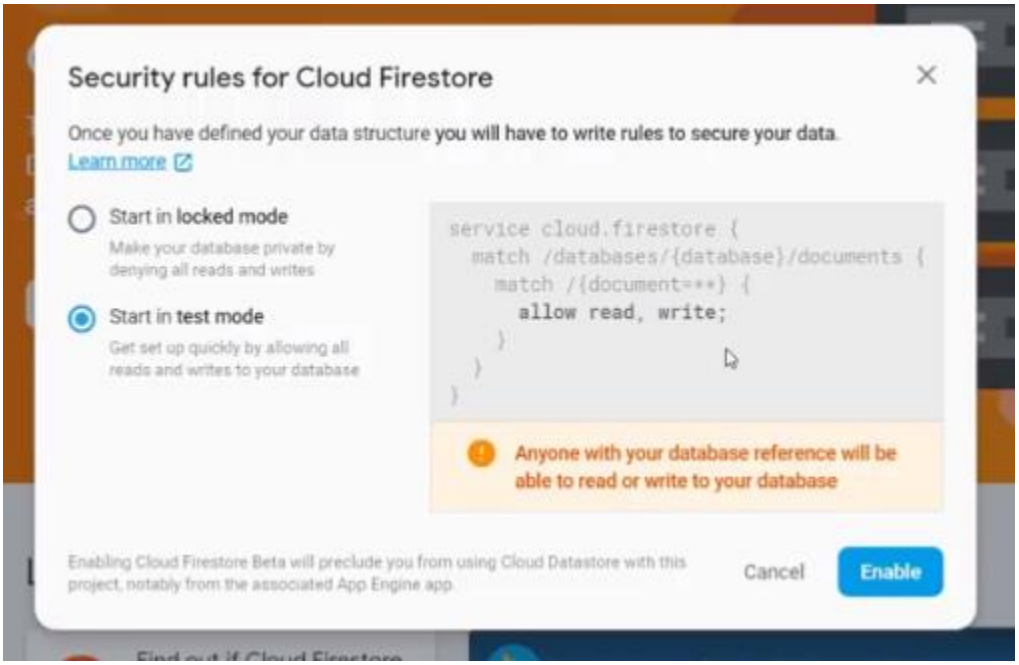


Figure 1.2: Screenshot showing Database Setup configured in test mode

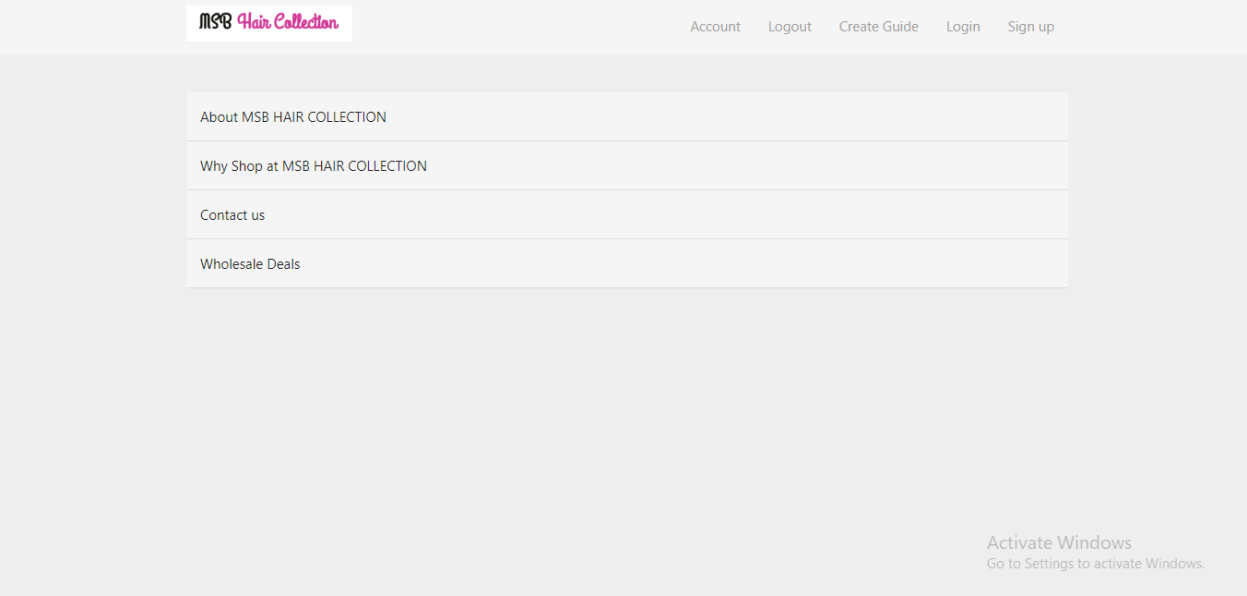


Figure 1.3: Screenshot showing the template interface design for the project

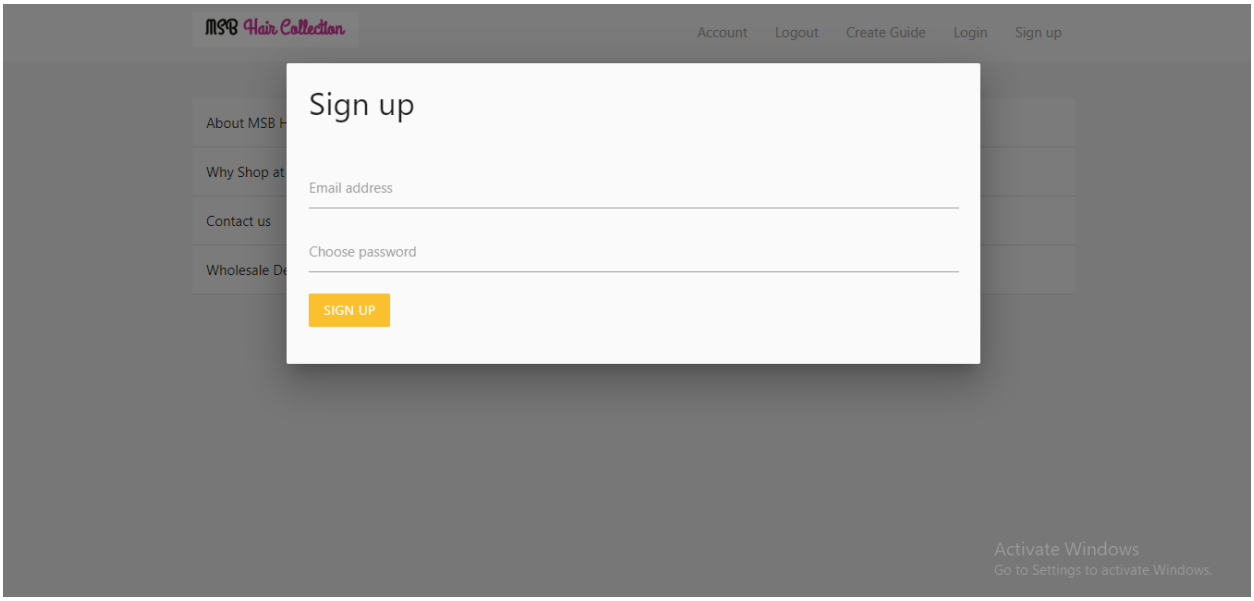


Figure 1.4: Screenshot showing the Sign-up interface

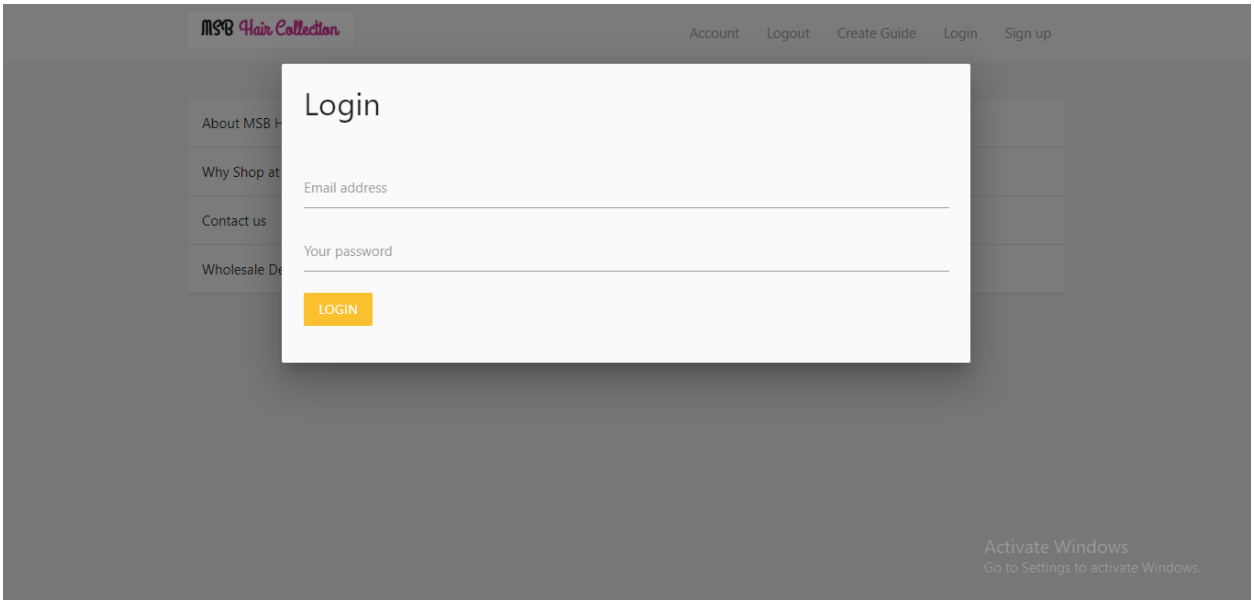


Figure 1.5: Screenshot showing the Login Interface

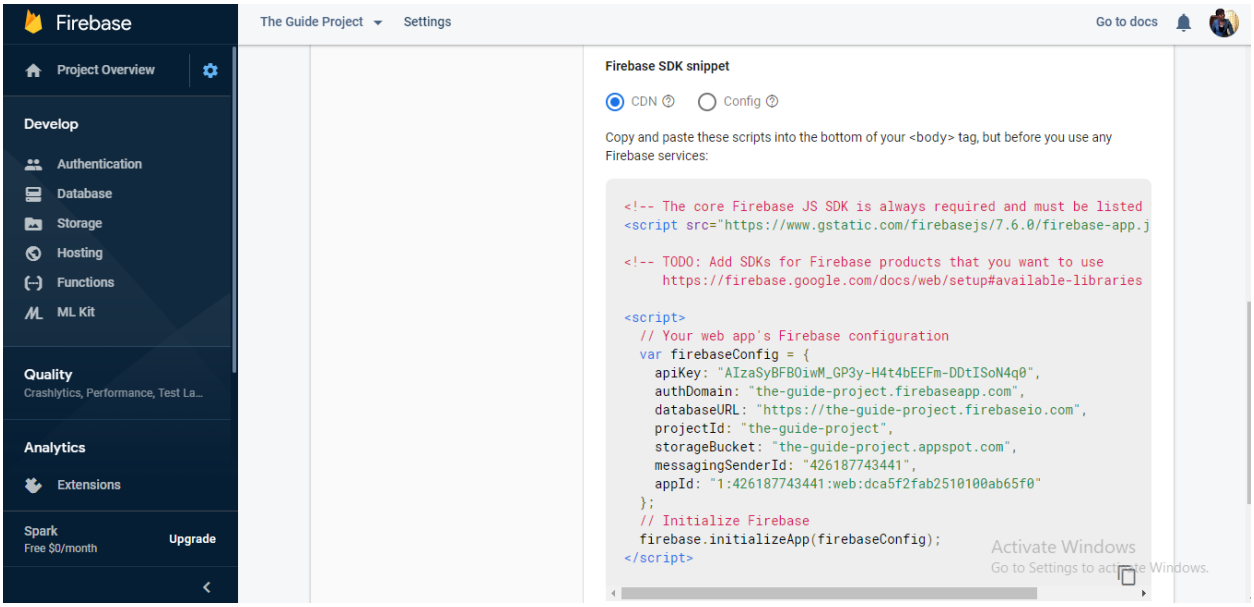


Figure 1.6: Screenshot showing the Firebase SDK snippet code for the project

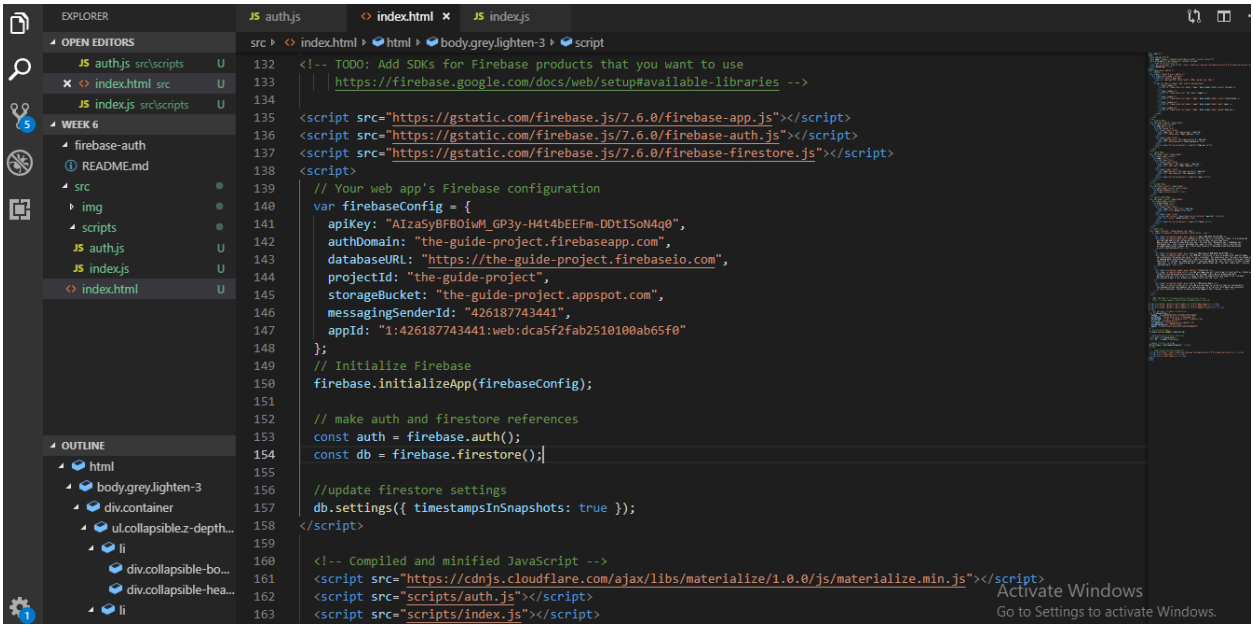


Figure 1.7: Screenshot showing the Firebase code snippet connection set up on the HTML file

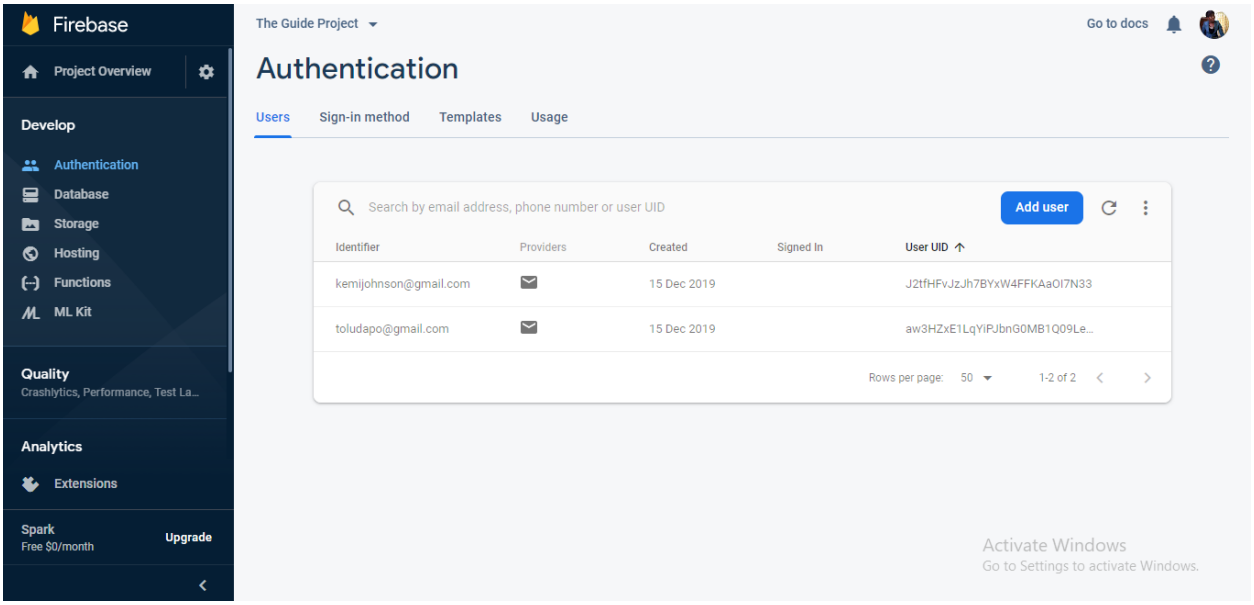


Figure 1.8: Screenshot showing users at the Firebase backend console

Project 2: Create a project to plot any random set of values using highcharts on a barchart or line chart

I carried out this project using React JavaScript Framework. First, I entered the command “npx create-react-app chartapp” and called the project chartapp. I also installed the react-router-dom “npm install -save react-router-dom” to configure my routing. React Material Design was used for the project template “npm install –save react-mdl”. For this project, I created two components called chartone.js and chart_two.js that displays a profit and loss line chart respectively. I created a main.js component to handle my routes, and modified app.js and index.js to capture react-router-dom setup. To display the charts, I installed two packages called highcharts “npm install highcharts” and highcharts-react-official “npm install highcharts-react-official” This project can be found in src_two file.

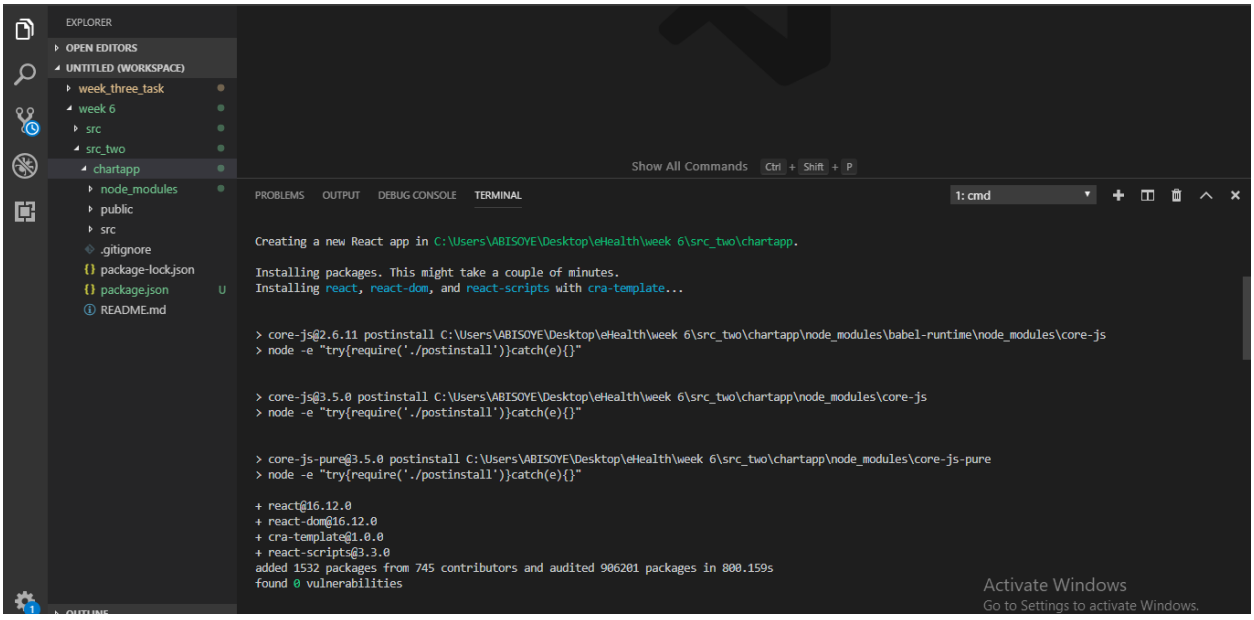


Figure 1.9: Screenshot showing successful create-react-app installation

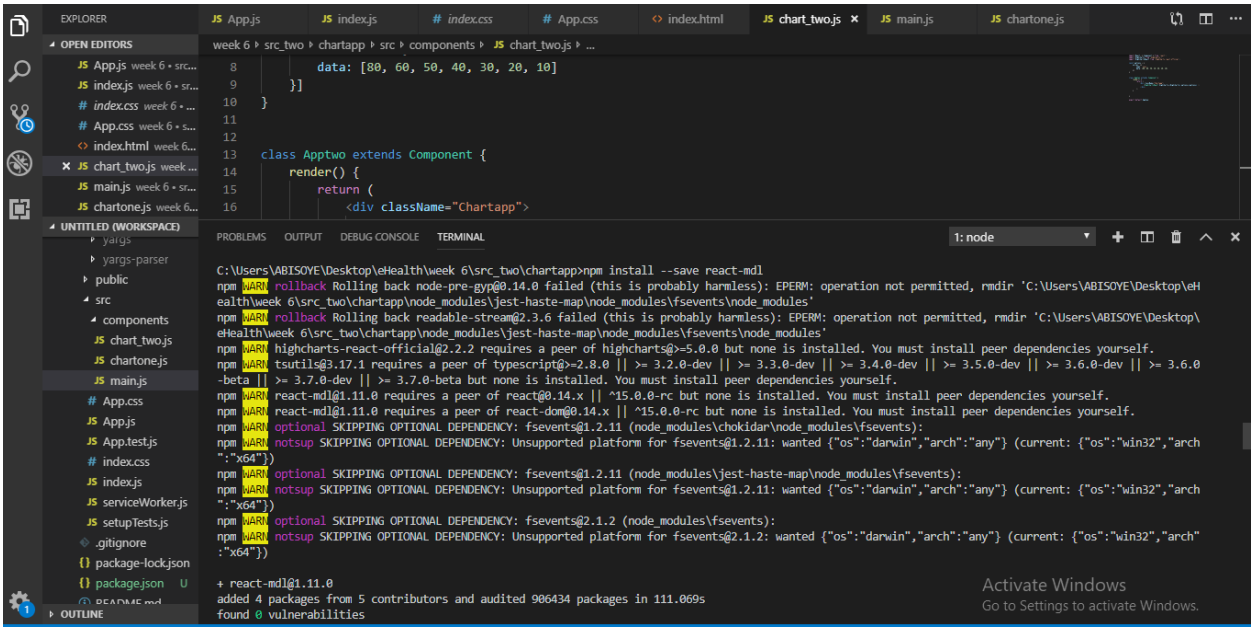


Figure 2.0: Screenshot showing successful installation react material design

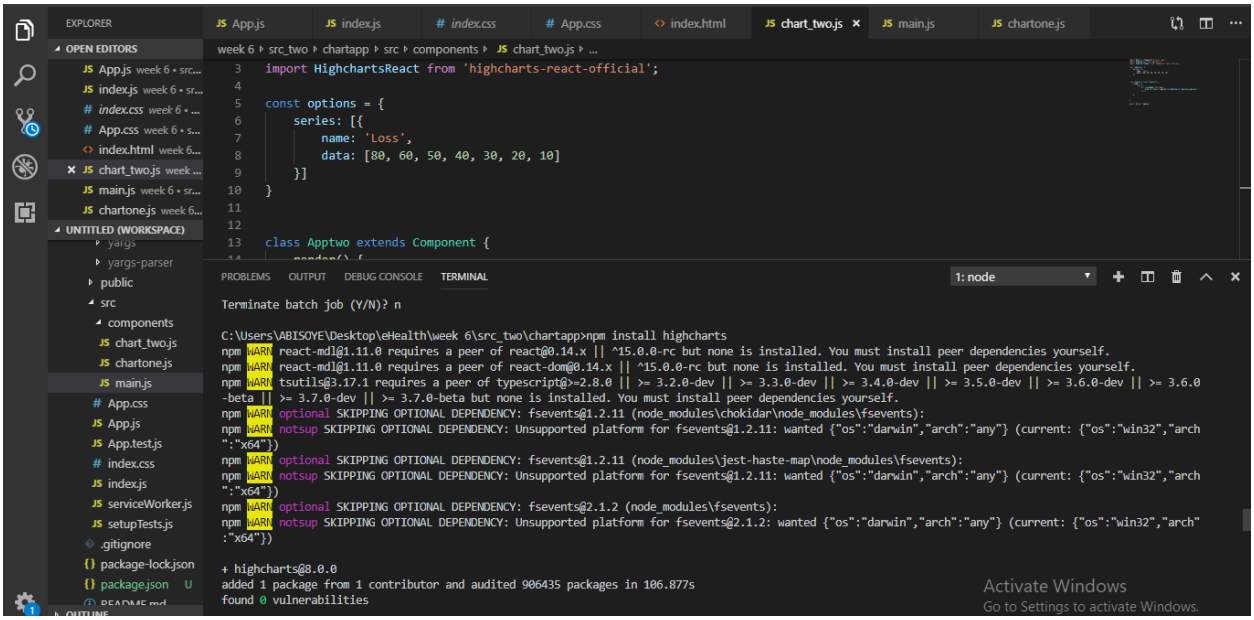


Figure 2.1: Screenshot showing successful installation of highcharts liabrary

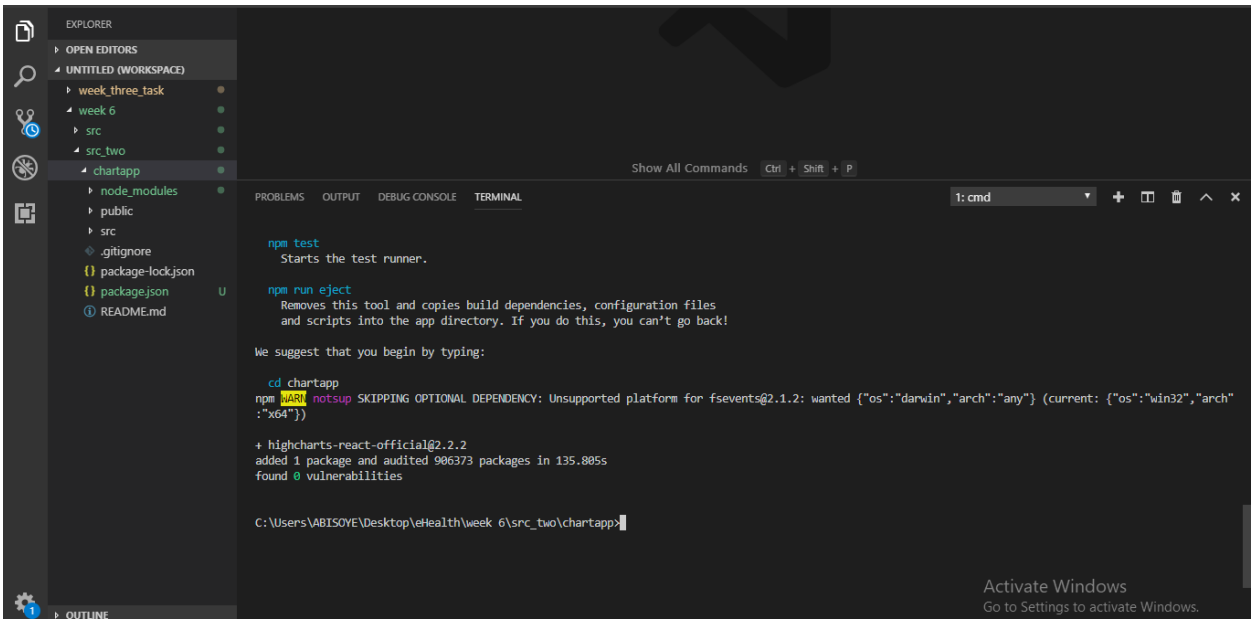


Figure 2.2: Screenshot showing successful install of highcharts-react-official

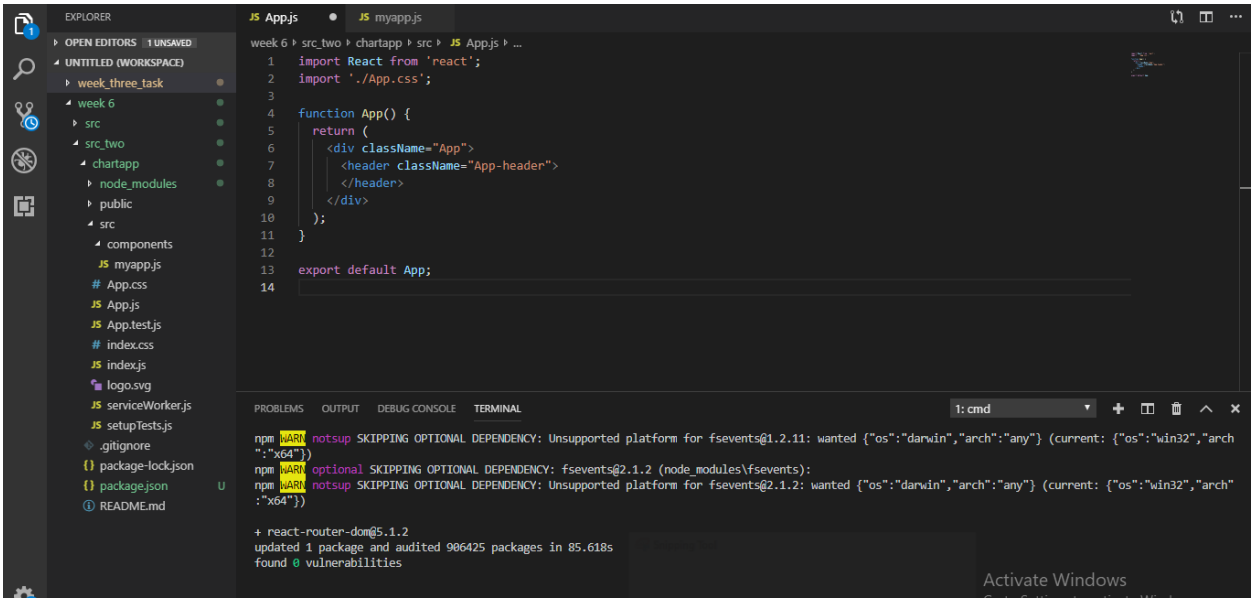


Figure 2.3: Screenshot showing successful install of react-router-dom

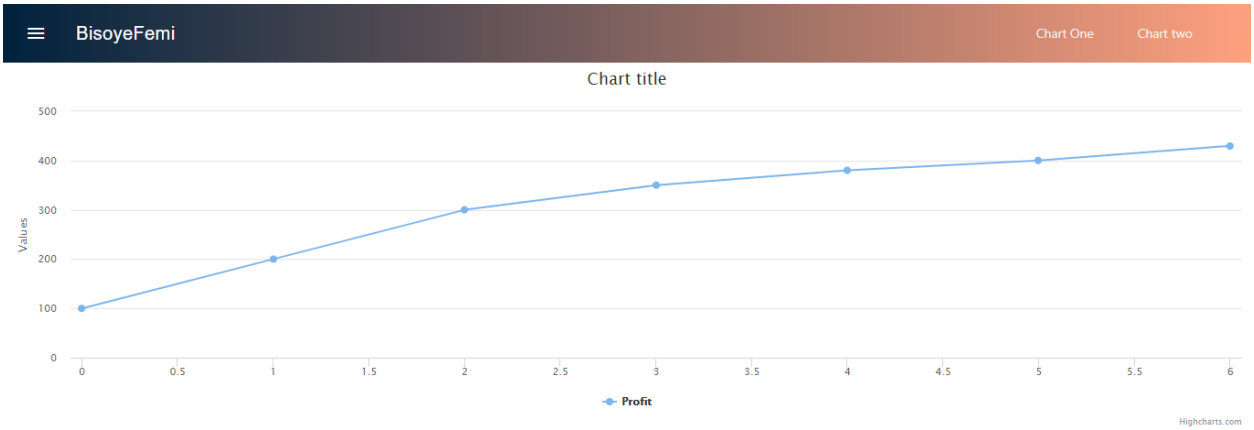


Figure 2.4: Screenshot showing line chart for a Profit statement

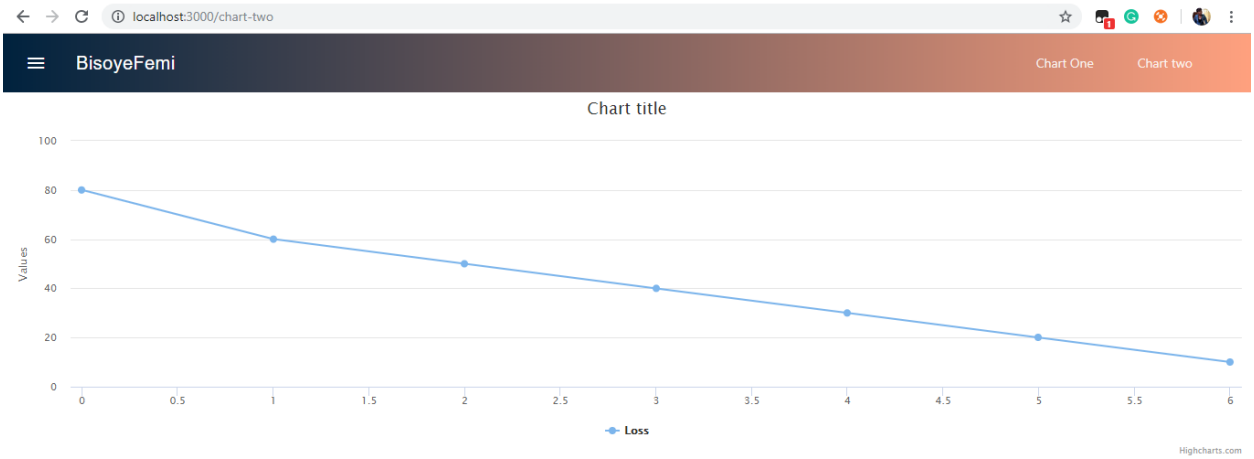
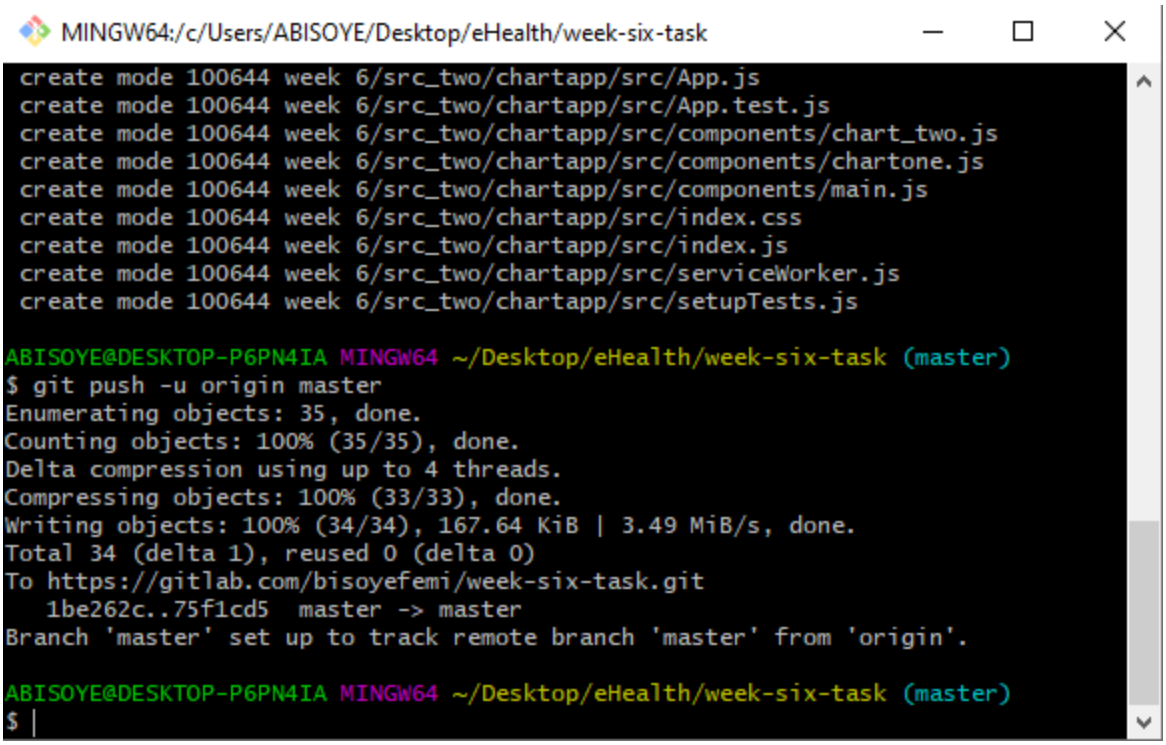


Figure 2.5: Screenshot showing line chart for a Loss statement


MINGW64:/c/Users/ABISOYE/Desktop/eHealth/week-six-task
create mode 100644 week 6/src_two/chartapp/src/App.js
create mode 100644 week 6/src_two/chartapp/src/App.test.js
create mode 100644 week 6/src_two/chartapp/src/components/chart_two.js
create mode 100644 week 6/src_two/chartapp/src/components/chartone.js
create mode 100644 week 6/src_two/chartapp/src/components/main.js
create mode 100644 week 6/src_two/chartapp/src/index.css
create mode 100644 week 6/src_two/chartapp/src/index.js
create mode 100644 week 6/src_two/chartapp/src/serviceWorker.js
create mode 100644 week 6/src_two/chartapp/src/setupTests.js
ABISOYE@DESKTOP-P6PN4IA MINGW64 ~/Desktop/eHealth/week-six-task (master)
$ git push -u origin master
Enumerating objects: 35, done.
Counting objects: 100% (35/35), done.
Delta compression using up to 4 threads.
Compressing objects: 100% (33/33), done.
Writing objects: 100% (34/34), 167.64 KiB | 3.49 MiB/s, done.
Total 34 (delta 1), reused 0 (delta 0)
To https://gitlab.com/bisoyefemi/week-six-task.git
1be262c..75f1cd5 master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
ABISOYE@DESKTOP-P6PN4IA MINGW64 ~/Desktop/eHealth/week-six-task (master)
$