Department of Information Technology, PICT Pune

Third Year Information Technology (2019 Course)

314458: Laboratory Practice-II (Cloud Computing)

Assignment No. : 08
Design and develop "Money management" custom application using Firebase Authentication and the Google App Engine.

Name and roll number of group members:

Deepak Abande	**33101**
Sumit Chavan	**33113**
Sakshi Dalvi	**33117**
Divya Dhomase	**33123**

Guided by Prof. Manish Khodaskar

Academic Year 2021- 2022 Sem - II

Application:

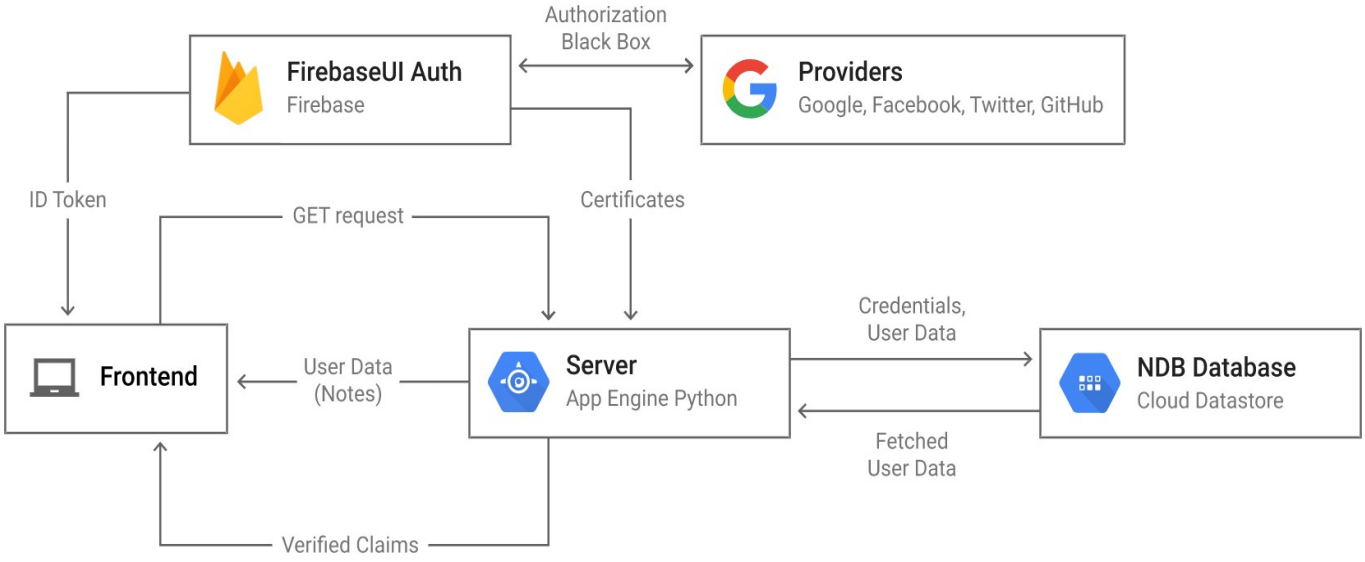
1. **Problem Statement:**
   - The custom application to be designed is a type of suggestion system or feedback system that can be used by an organization, community, institute oreven by an ecommerce site for taking customer views.
   - The application has a login interface and then an input area for accepting thesuggestions or complaints from the members of the organization.
   - The Firebase Authentication has been used for security purpose and the Google App Engine standard environment is used for deploying the application on Firebase. Also, the data is stored in Google Cloud Data Store.

2. **Working:**
   - The Suggestion Box application is a simple application to accept suggestionsor complaints. The frontend has been designed using simple web development technologies like HMTL, CSS and also framework like Bootstrap and also backend framework Flask. It has two webpages which are explained as follows:

   A. Login page:
   The login page provides the users to login and then they are directed to the main page. This page has heading at the top.
      - This page provides this option for login:
        a. Mail.
      - The users can login either by their Google account or any other valid emailaddress.

   B. Add (income & expenses page)
   This page contains four input forms. After submitting these forms the data will be added to firebase cloud.

- The application involves the use of a number of platforms like Firebase, GAE and Google Cloud Storage. The workflow or the architecture of the system isexplained as follows:



- The application stores users' notes or suggestions in their own personal notebooks. Notebooks are stored per user, and identified by each user's unique Firebase Authentication ID. The application has following important components:
  a. The frontend configures the sign-in user interface and retrieves the Firebase Authentication ID. It also handles authentication state changes and lets users see their notes.
  b. FirebaseUI is an open-source, drop-in solution that simplifies authentication and UI tasks. The SDK handles user login, linking multiple providers to one account, recovering passwords, and more. It implements authentication best practices for a very smooth and secure sign-in experience.
  c. The backend verifies the user's authentication state and returns user profileinformation as well as the user's notes.
  d. The application stores user credentials and details in Datastore by using the client library from Google Cloud, but you can store the credentials ina database of your choice.

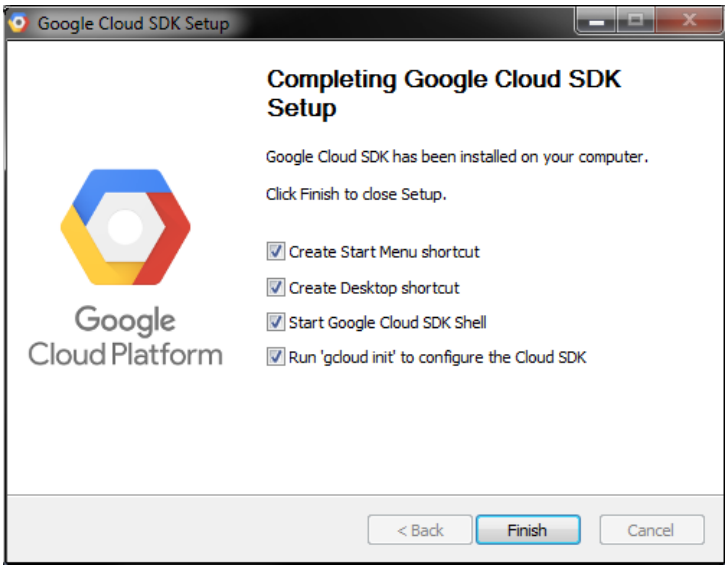
3. **Features:**
   The income page application is very useful and its features are mentioned as follows:

- The application using some of the well-known technologies for its functioning. The Firebase Authentication has been used for security purpose and the Google App Engine standard environment is used for deploying the application on Firebase. Also, the data is stored in Google Cloud Data Store.
- The mobile application developed in this project will help to manage money and learn how to save money. This mobile application will allow students to manage money well and to save with fun. The implementation of the application includes a money manager which the students able to use it for their money-saving and help the students to manage money.
- This made it easier for students to trace money monthly or daily. This Money Management helps the user track the user financial activity efficiently. It allows the user to set up the income and expense from various options such as salary, awards, grants, sale, food, bills, and others. The application will calculate the income and expense. Then, it will display the balance of the income for the current month.

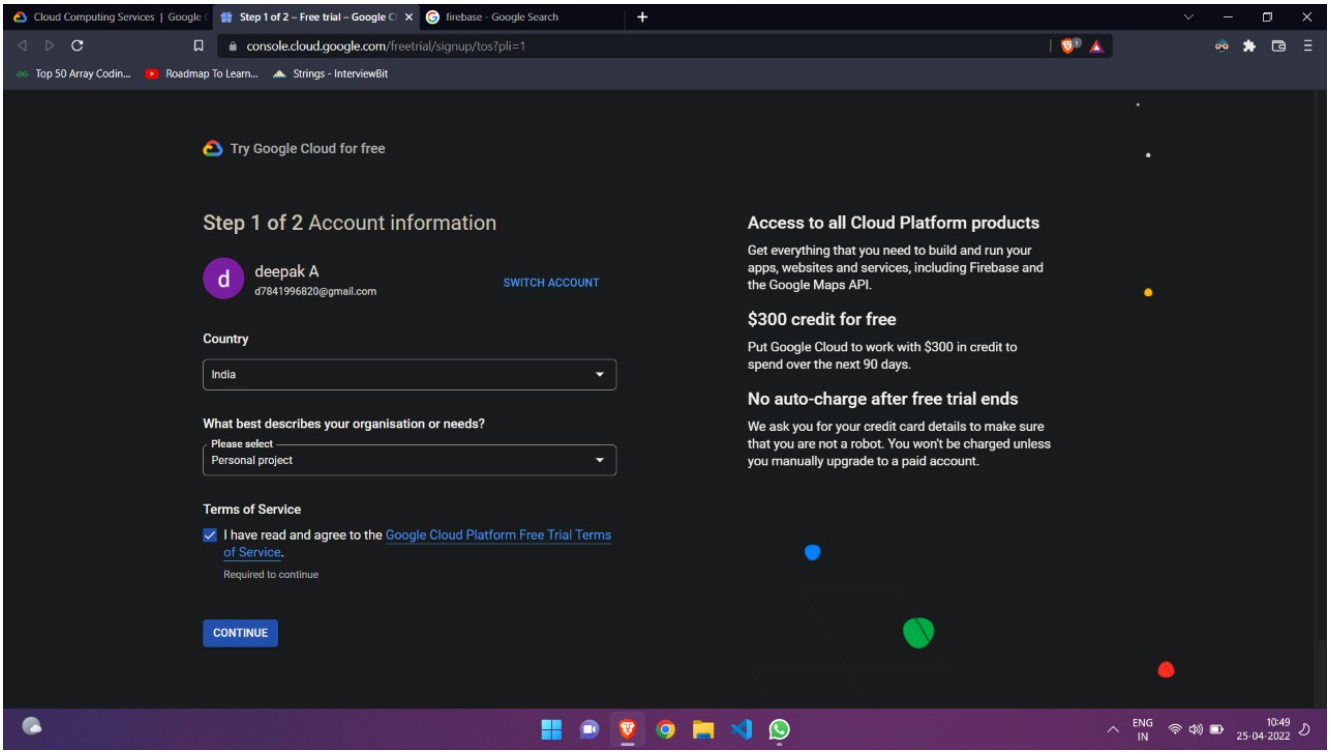
Implementation:

1. Initial Setup:

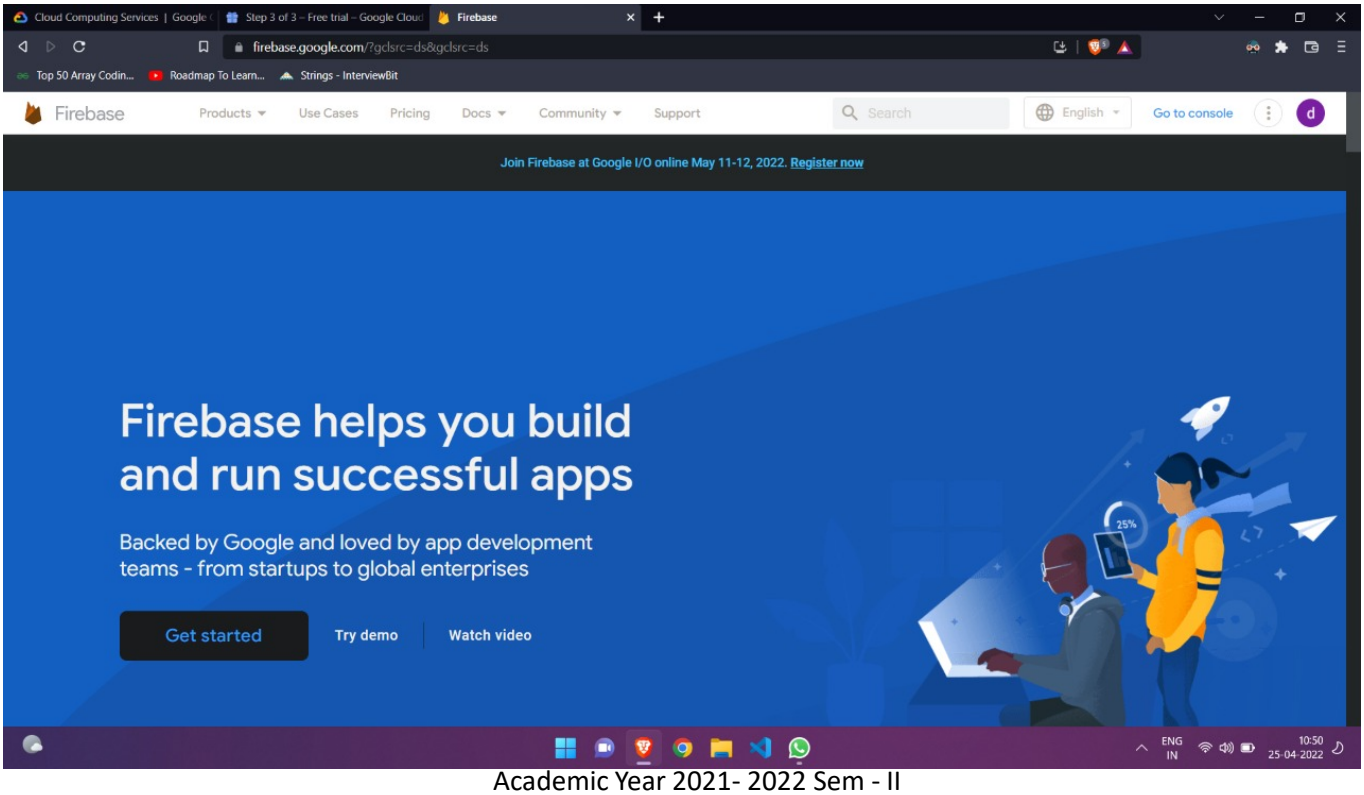
a. Install and configure Google App Engine:



b. Signup and Login to Google Cloud platform:

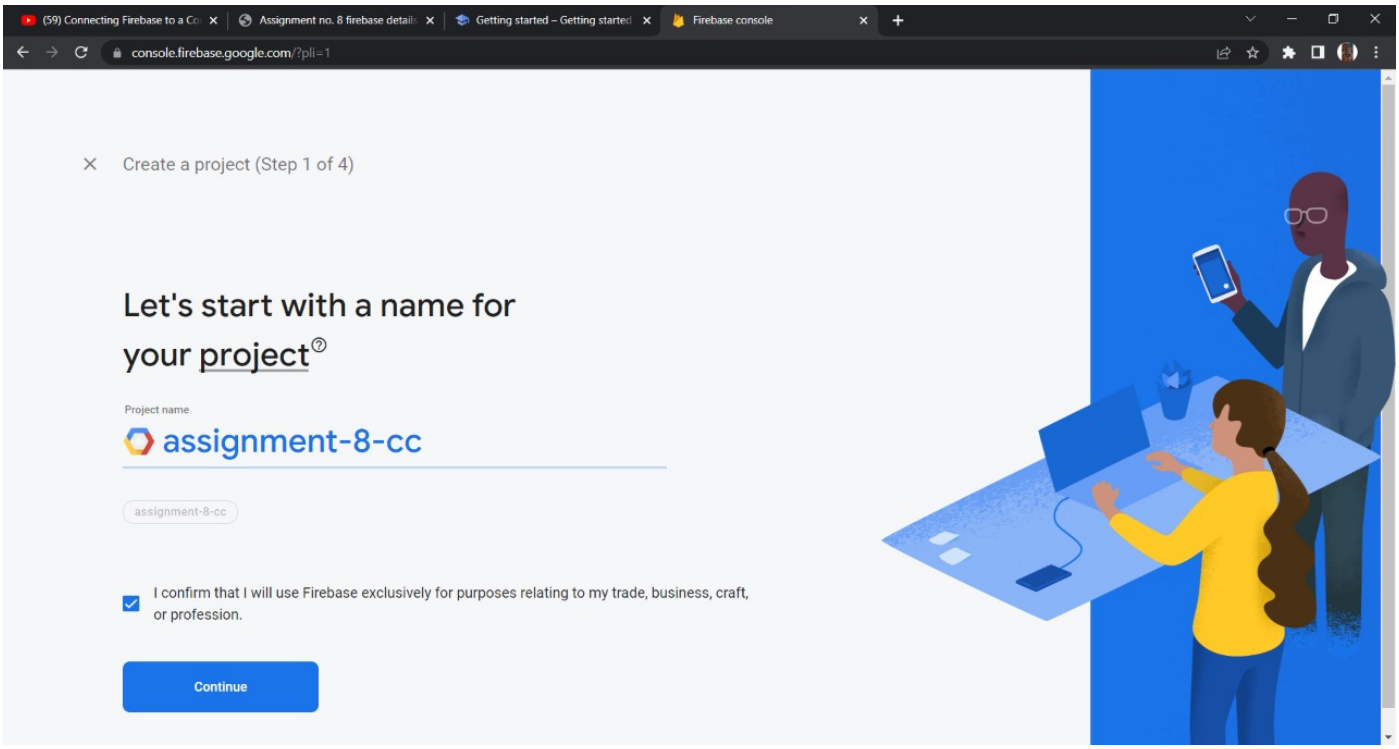


c. Login to Firebase:

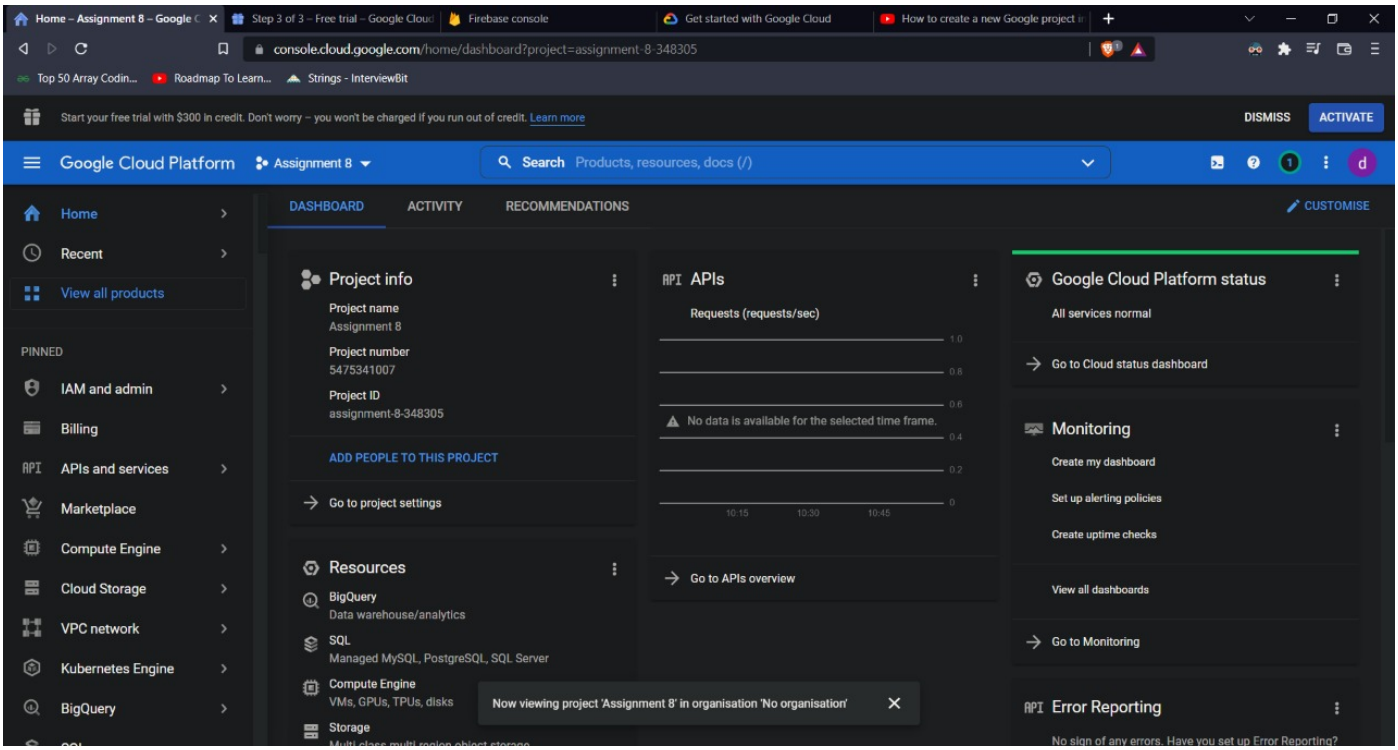


2. Creation of project in Google Cloud:

a. Create a new project in Google Cloud platform:

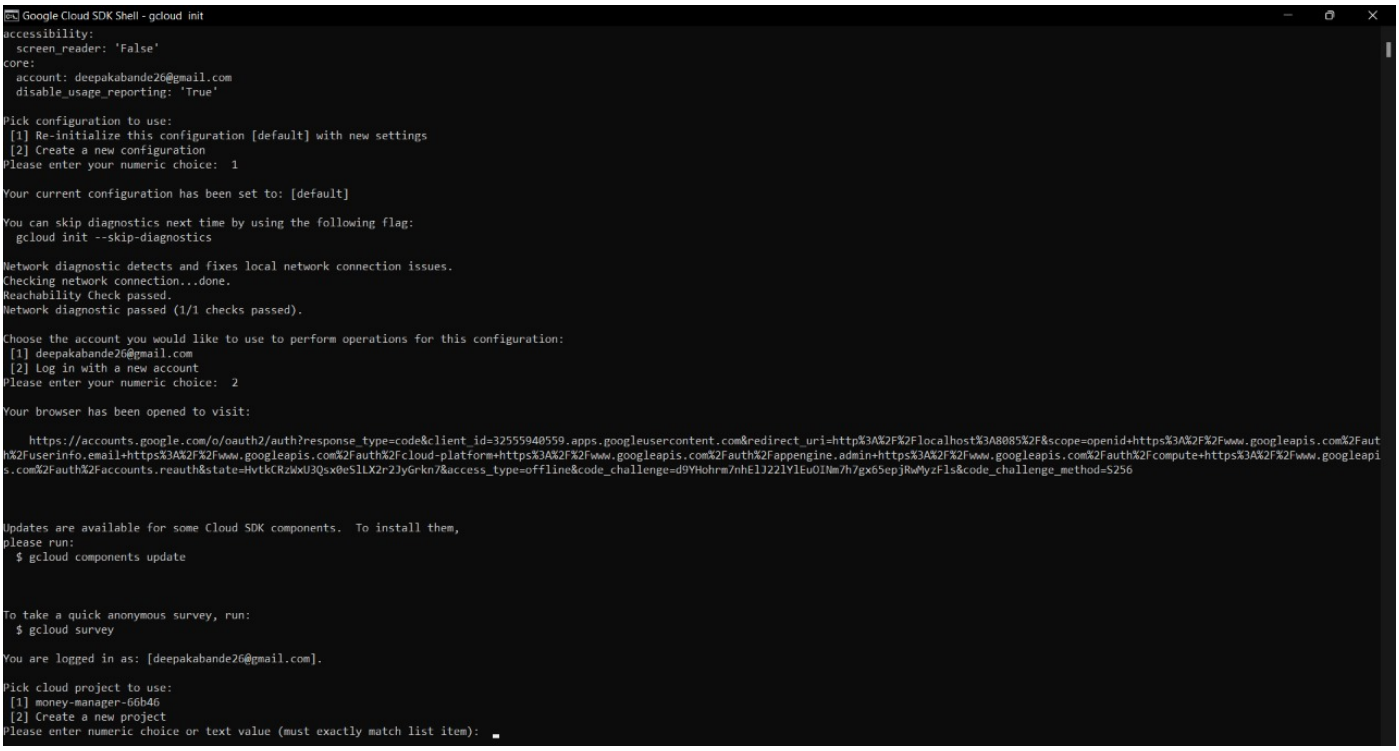


b. You can view the created project in the dashboard:

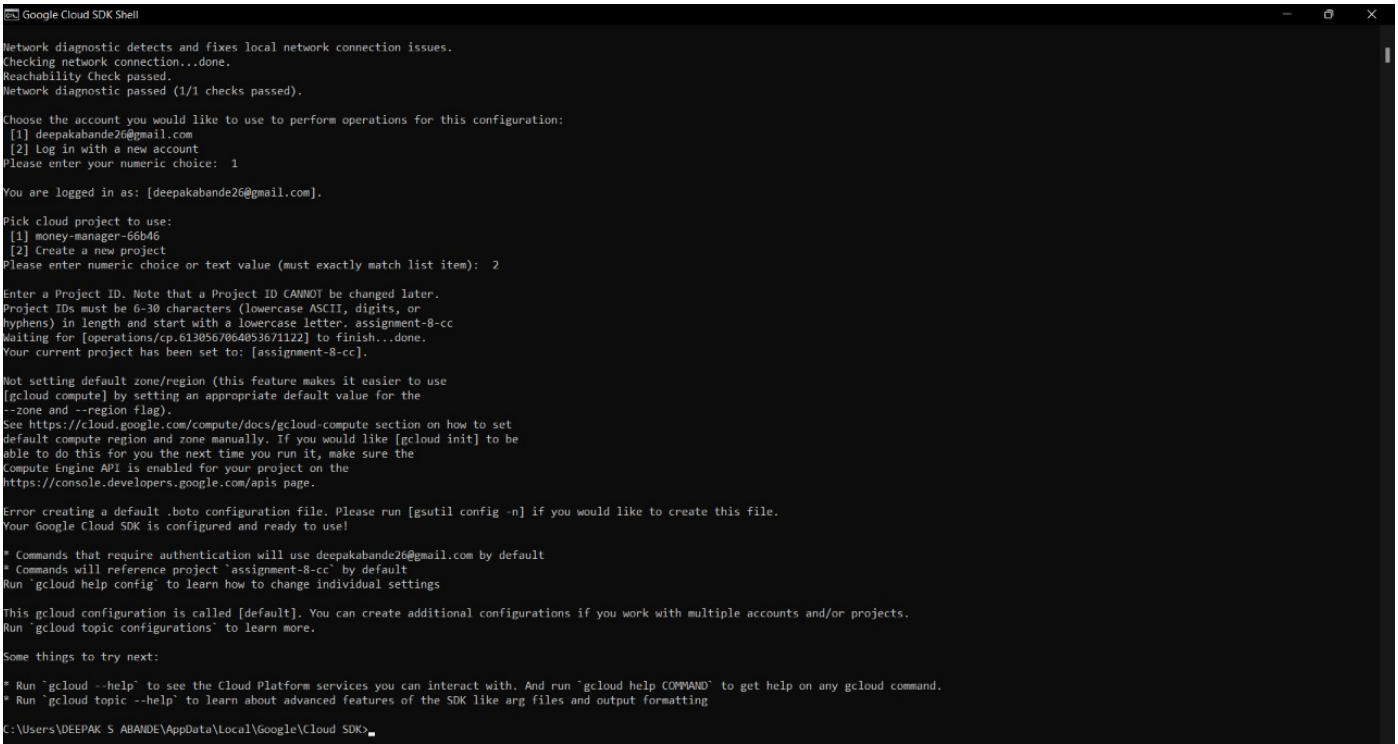


3. Setup Google App Engine:

a. Open the GAE SDK shell and type the command `'gcloud init'`. Then selectappropriate configuration an account:

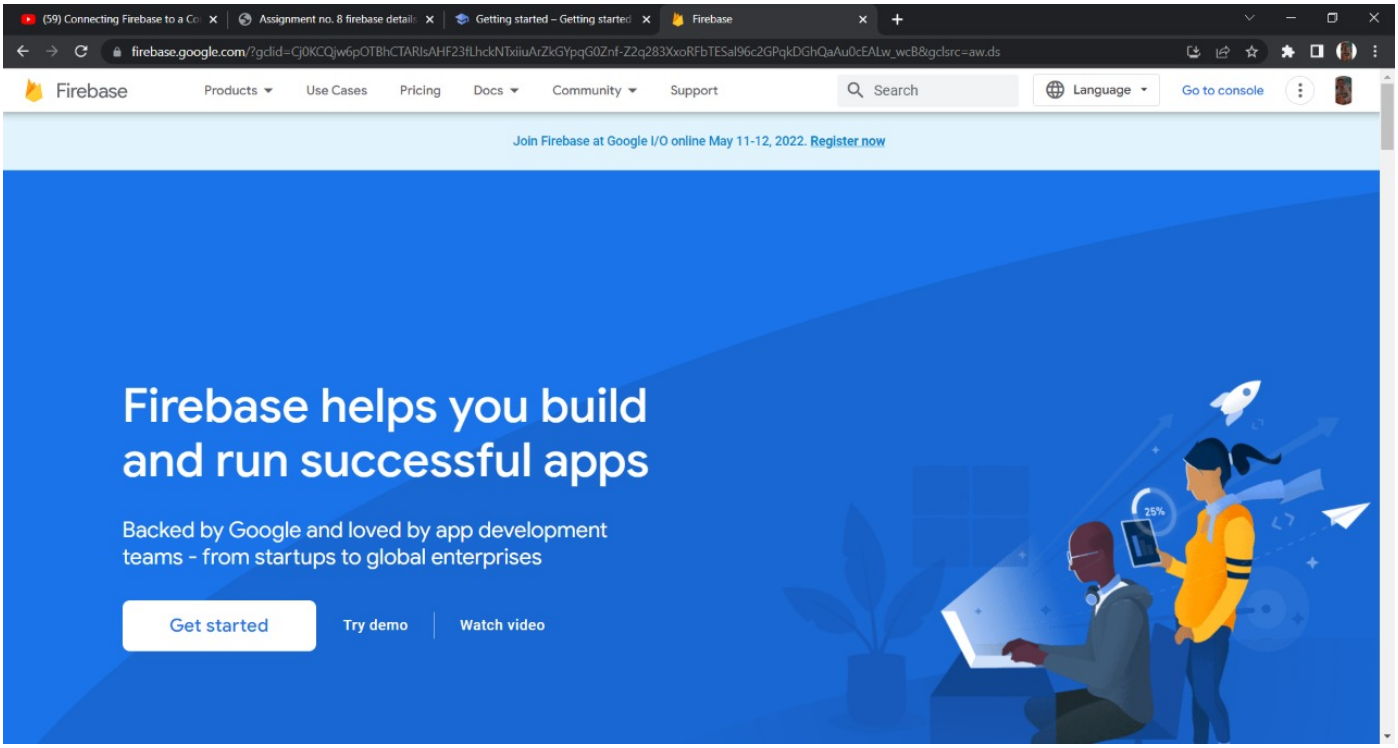


b. From the list that appears select the appropriate project that we created in GC

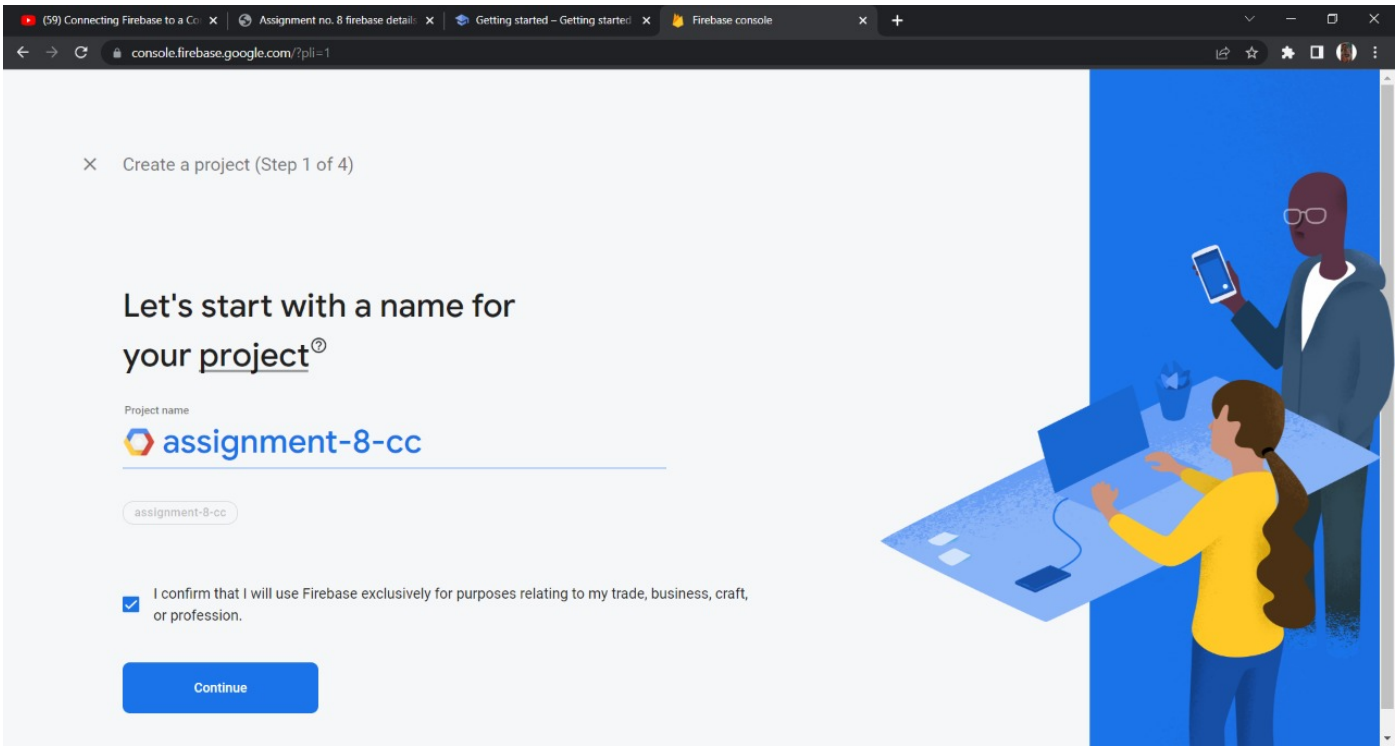


4. Adding Firebase to the project:

a. Go to the Firebase console and click on Create a project:

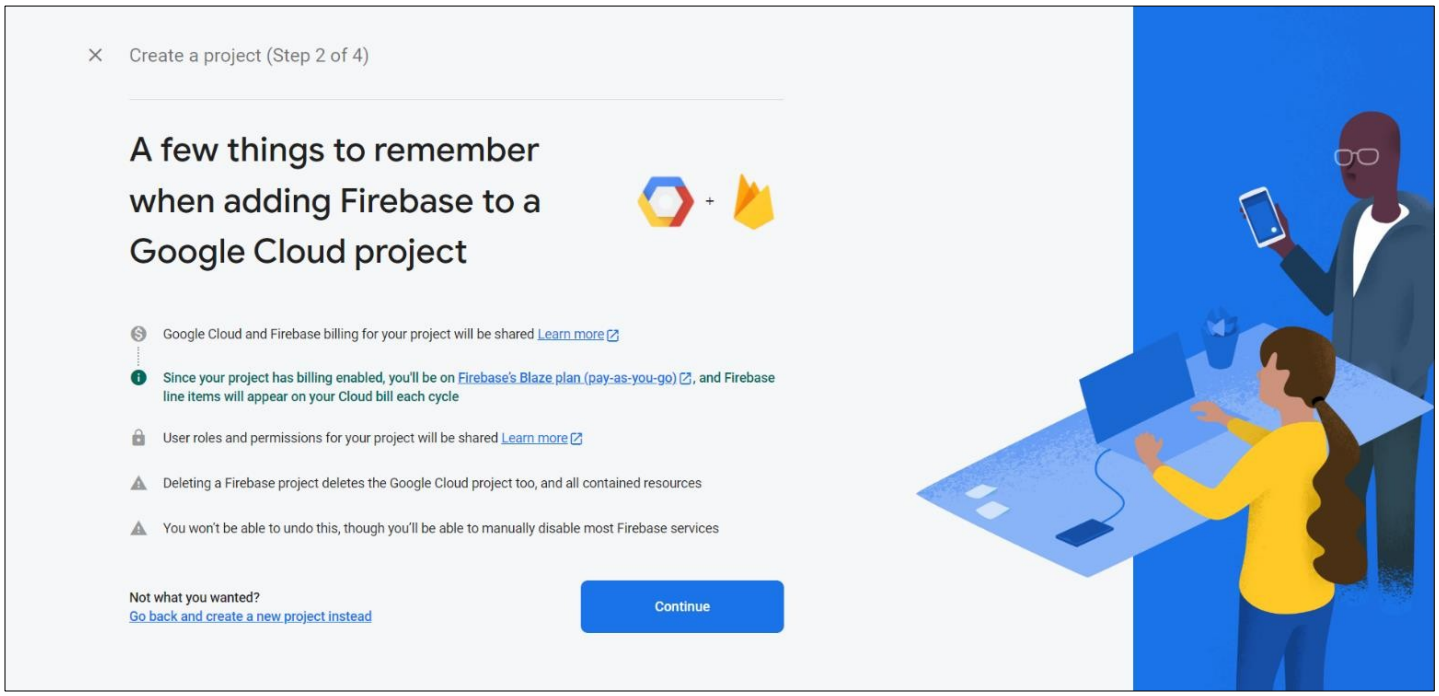


b. In the next window, add the project name that we created in Google Cloud:

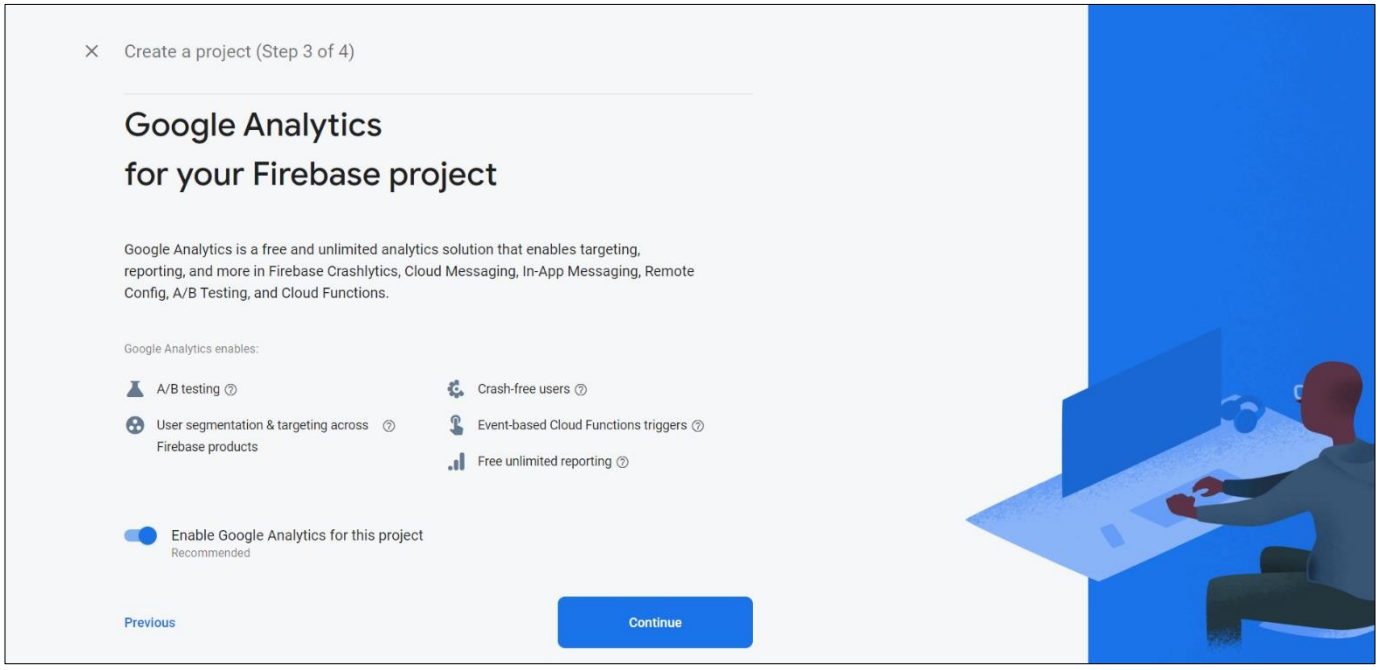


c. Confirm Firebase billing plan:

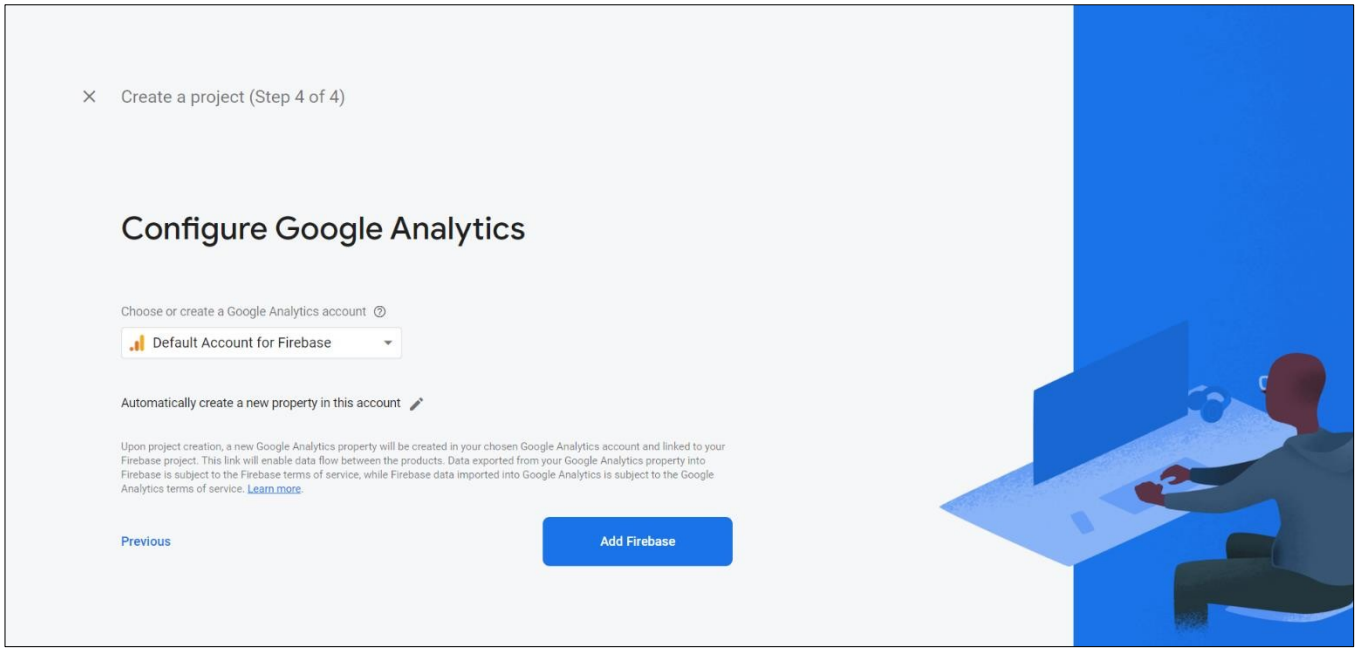
d. The next window shows some instructions. Read those and click Continue:



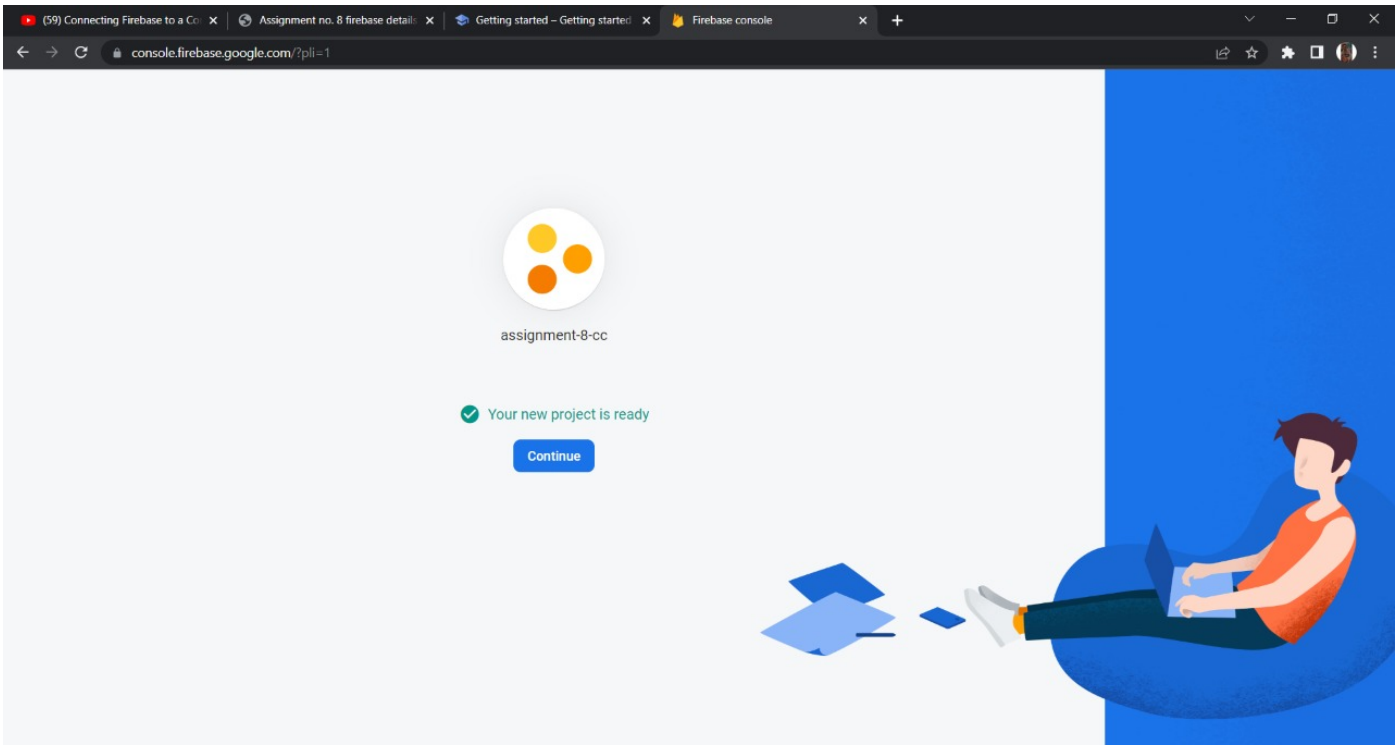
e. Enable Google Analytics for the project:



f. Configure Google Analytics by selecting Default Account for Firebase:

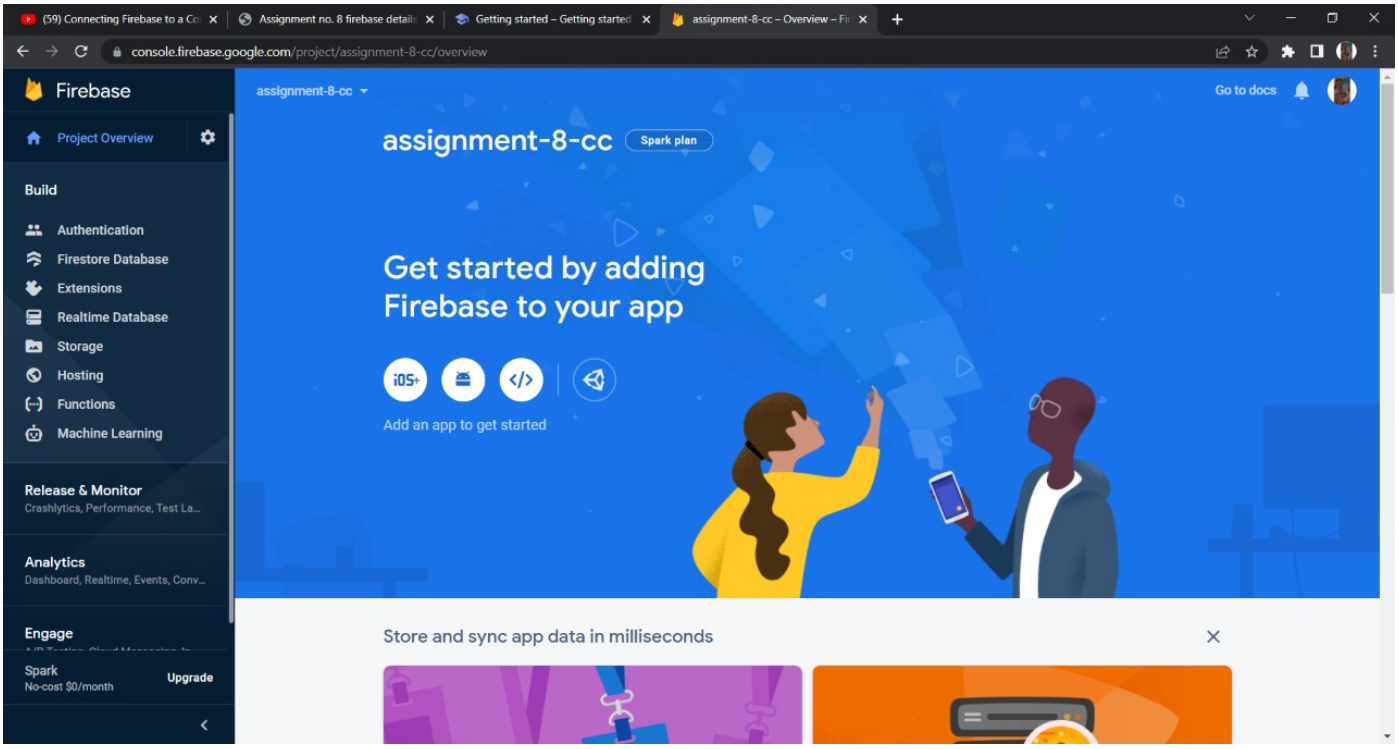


g. We have successfully added Firebase to our project.

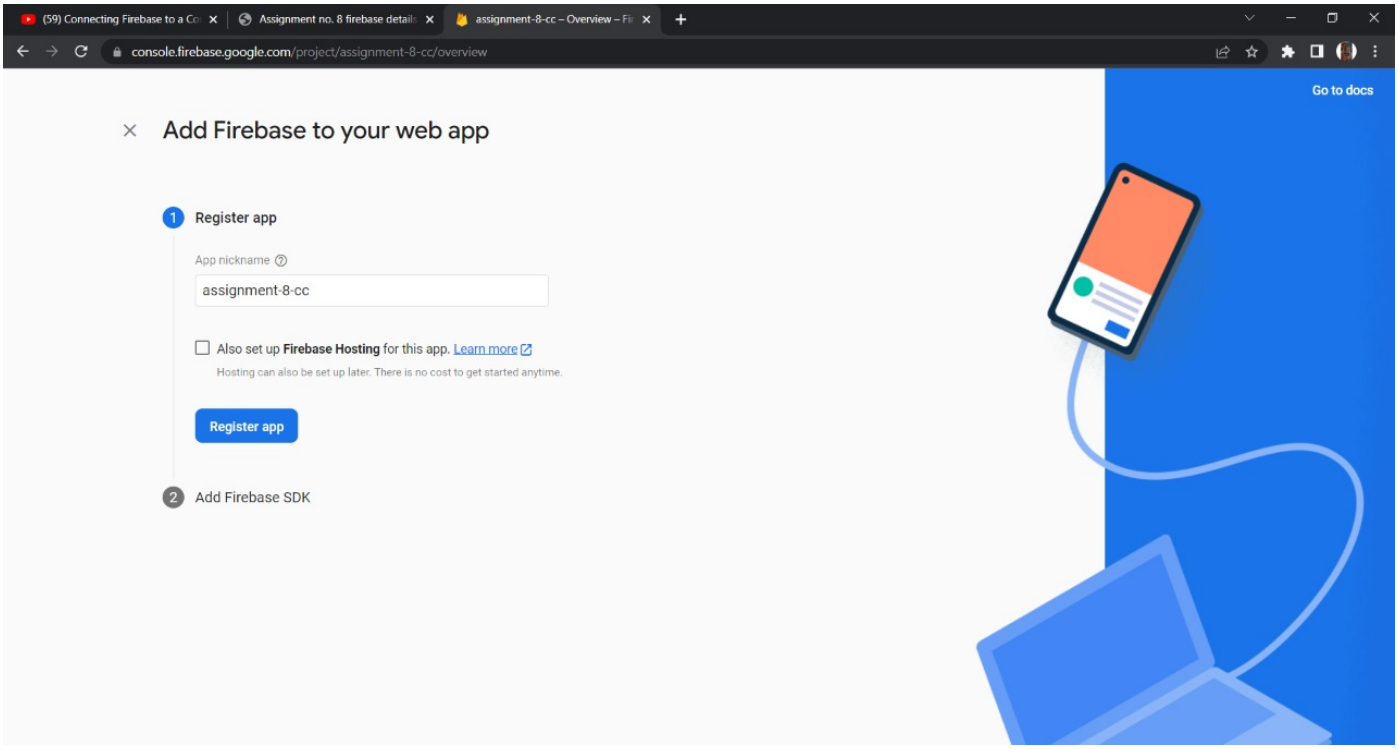


5. Adding an App to the Firebase project:

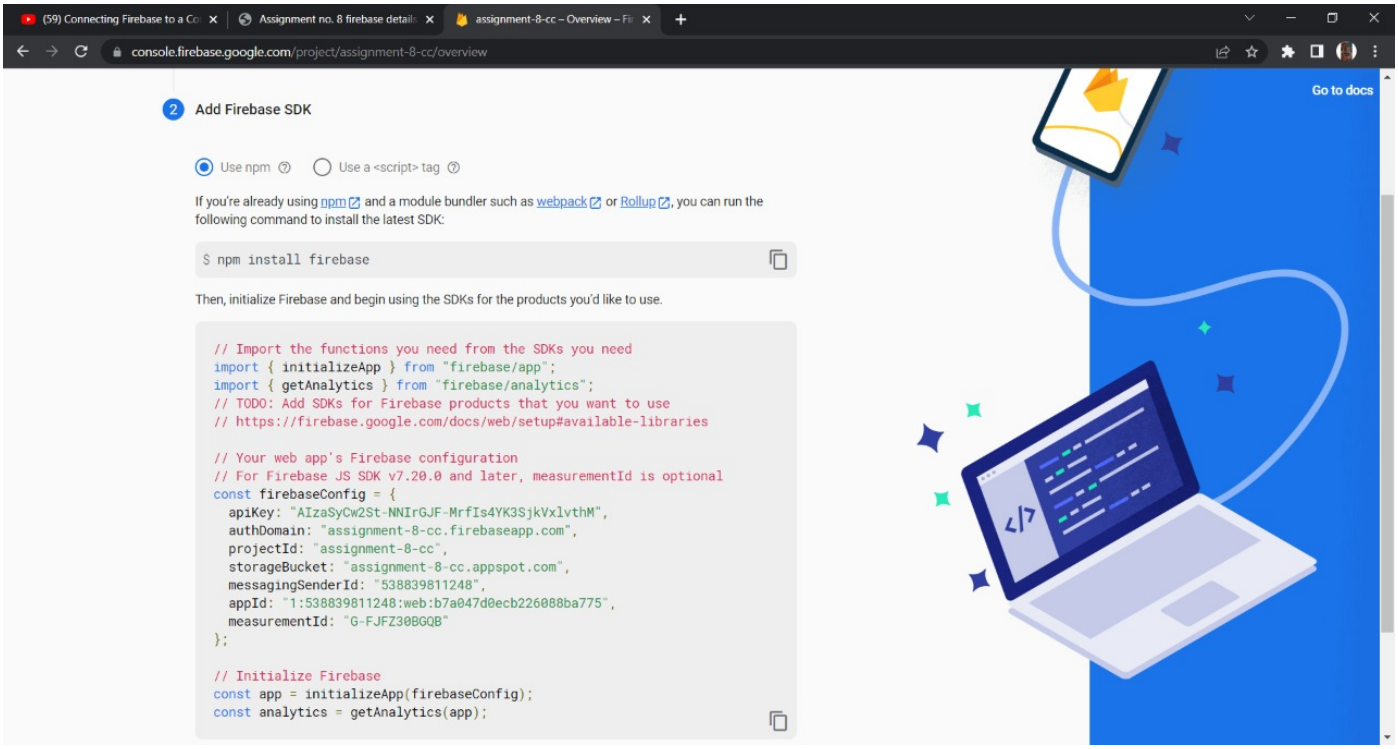
a. From the console, go to your project and click '</>' to add your app:



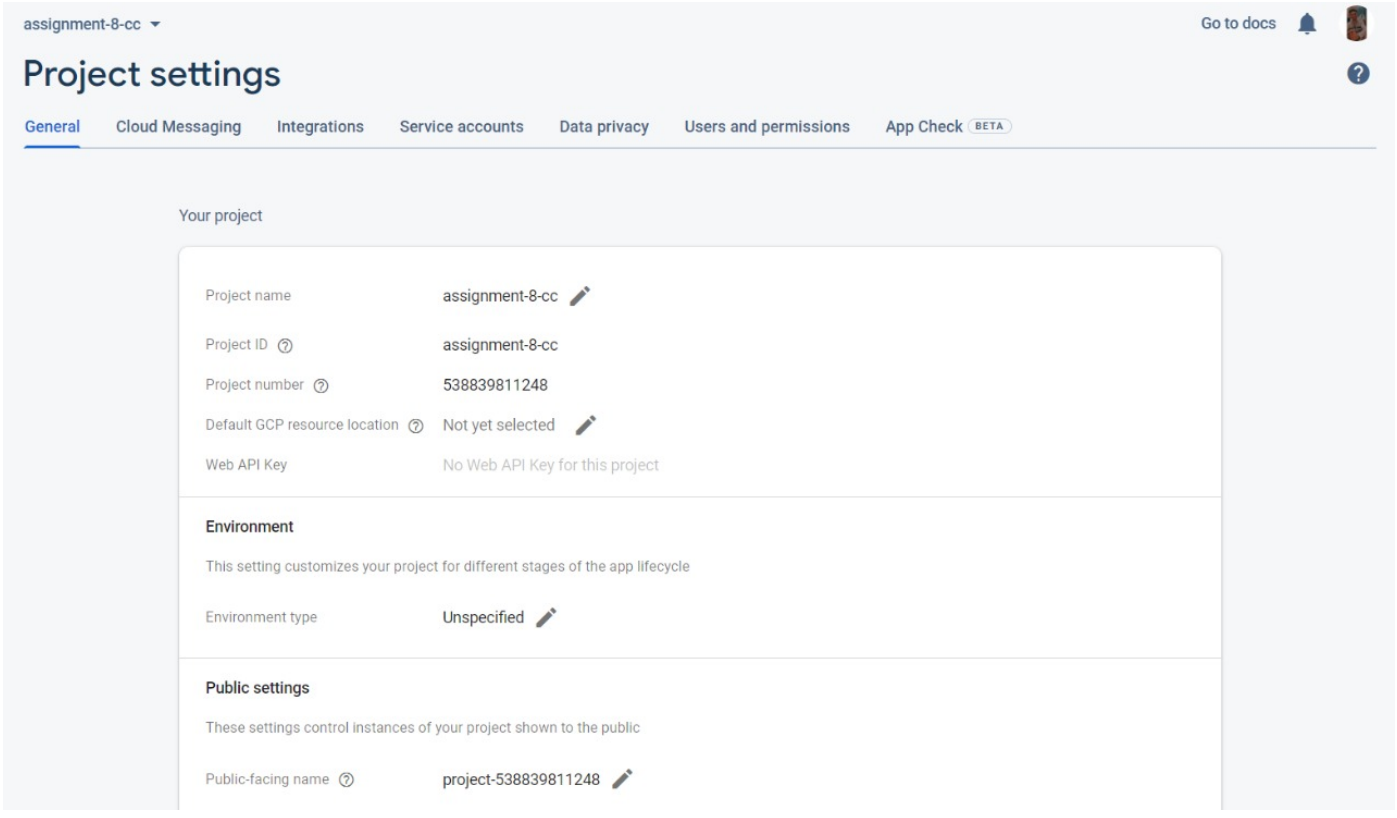
b. Add a nickname for your app and click Register app:

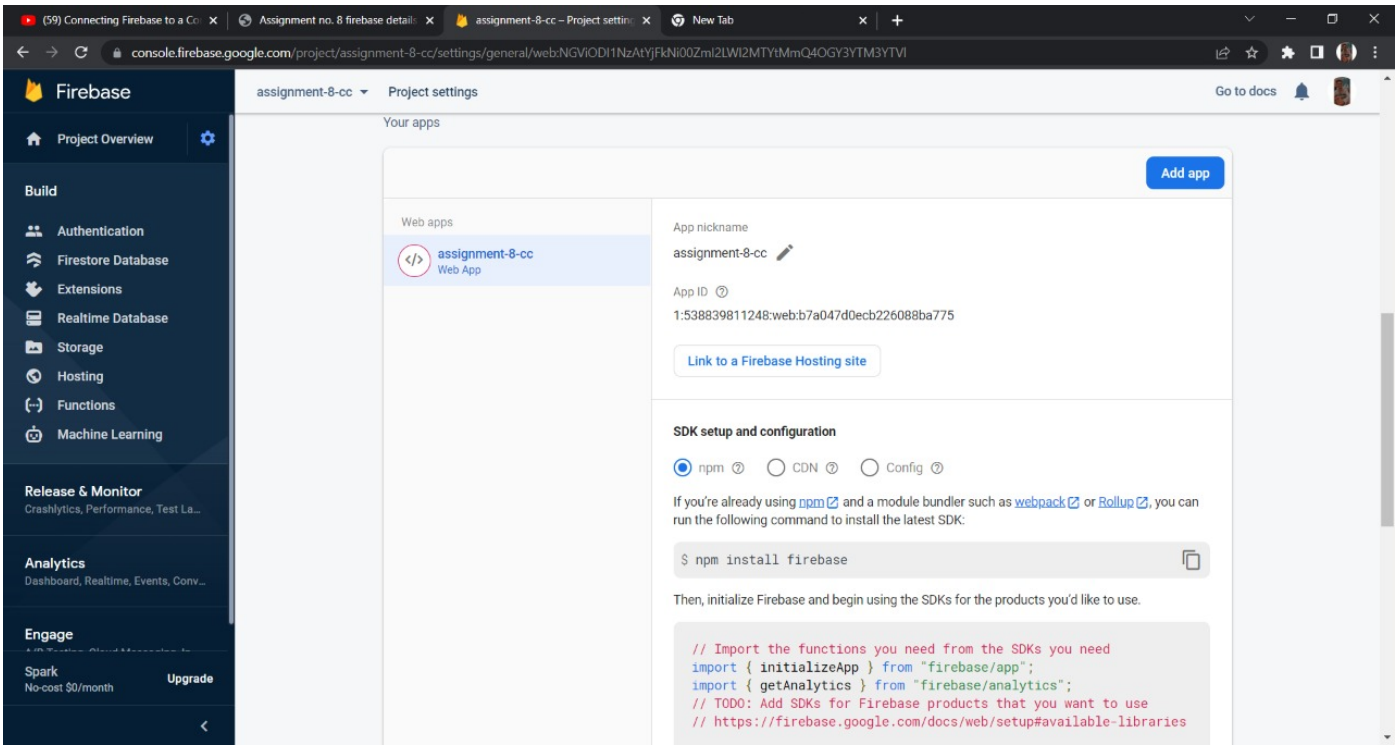


c. You will receive further configuration details then click Continue to console:



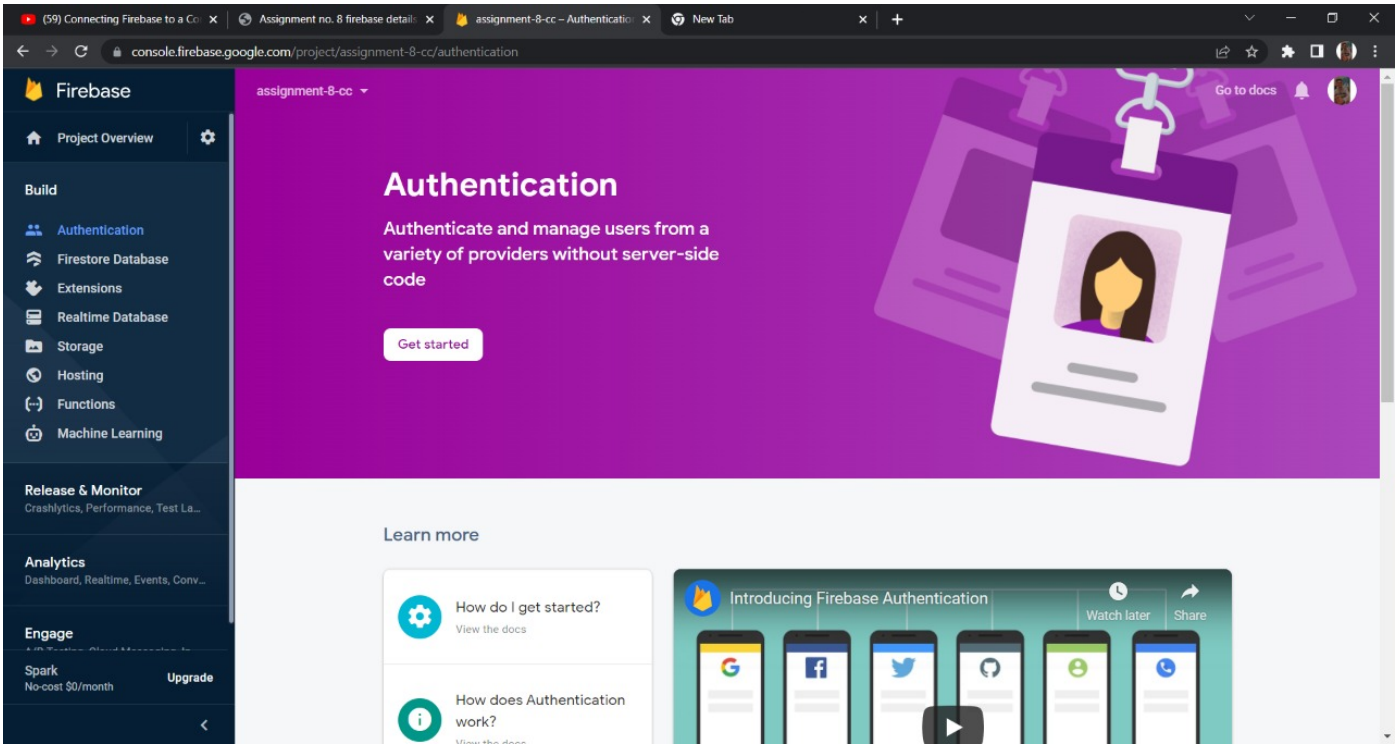
d. You will see the app on the console, click on it to view all its details.



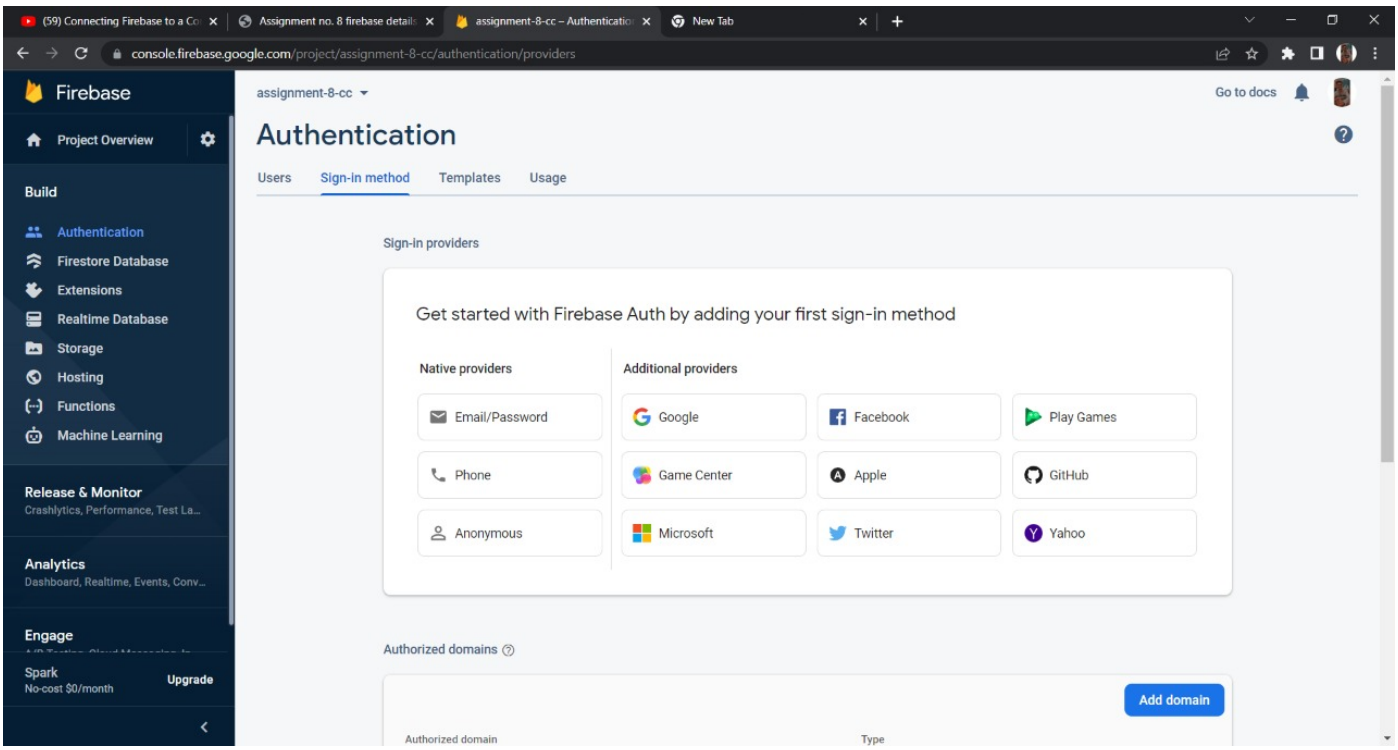

(59) Connecting Firebase to a Co
Assignment no. 8 firebase details
assignment-8-cc – Project settings
New Tab
console.firebase.google.com/project/assignment-8-cc/settings/general/web:NGViODI1NzAtYjFkNi00Zml2LWI2MTYtMmQ4OGY3YTM3YTVl
Firebase
assignment-8-cc
Project settings
Go to docs
Project Overview
Your apps
Build
Authentication
Firestore Database
Extensions
Realtime Database
Storage
Hosting
Functions
Machine Learning
Release & Monitor
Crashlytics, Performance, Test La...
Analytics
Dashboard, Realtime, Events, Conv...
Engage
Spark
No-cost $0/month
Upgrade
Add app
Web apps
assignment-8-cc
Web App
App nickname
assignment-8-cc
App ID
1:538839811248:web:b7a047d0ecb226088ba775
Link to a Firebase Hosting site
SDK setup and configuration
npm
CDN
Config
If you're already using npm and a module bundler such as webpack or Rollup, you can run the following command to install the latest SDK:
$ npm install firebase
Then, initialize Firebase and begin using the SDKs for the products you'd like to use.
// Import the functions you need from the SDKs you need
import { initializeApp } from "firebase/app";
import { getAnalytics } from "firebase/analytics";
// TODO: Add SDKs for Firebase products that you want to use
// https://firebase.google.com/docs/web/setup#available-libraries

6. Authentication in Firebase:

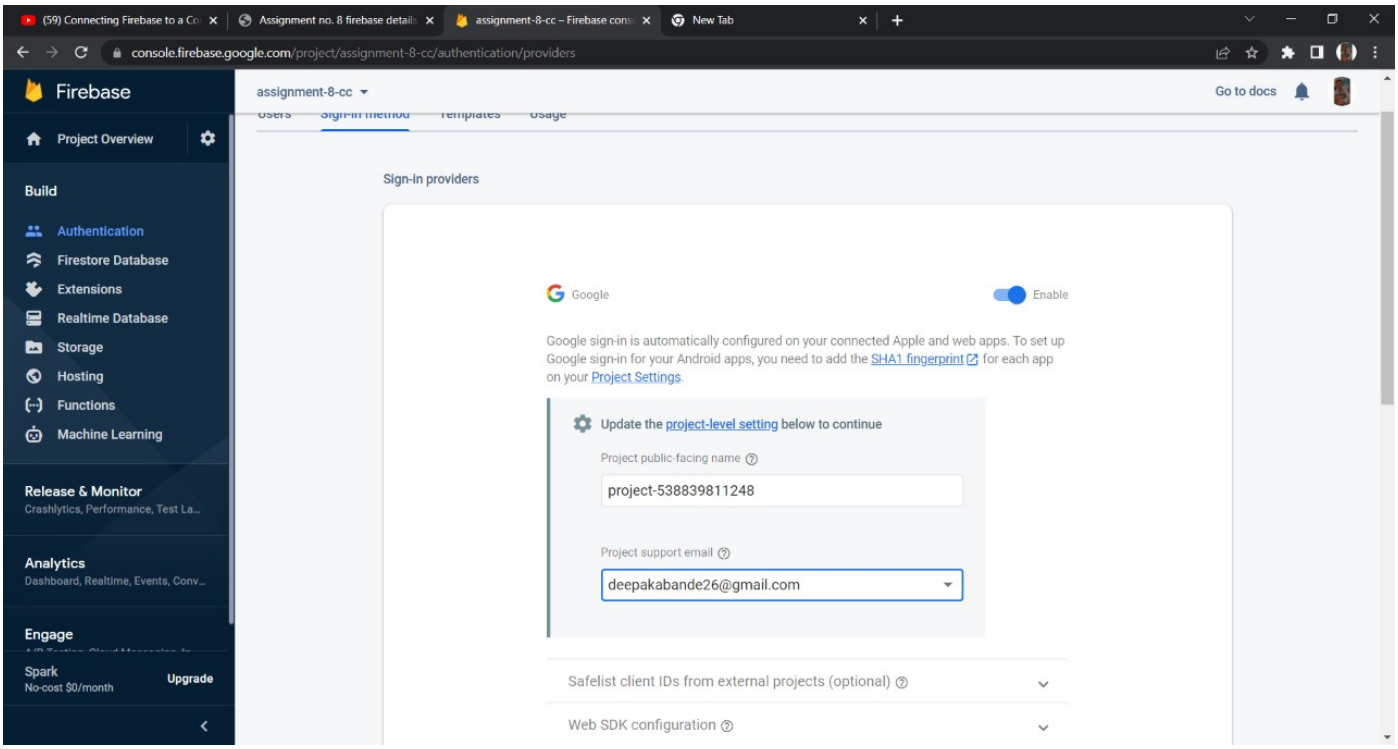
a. Go to the project's console and select Authentication:



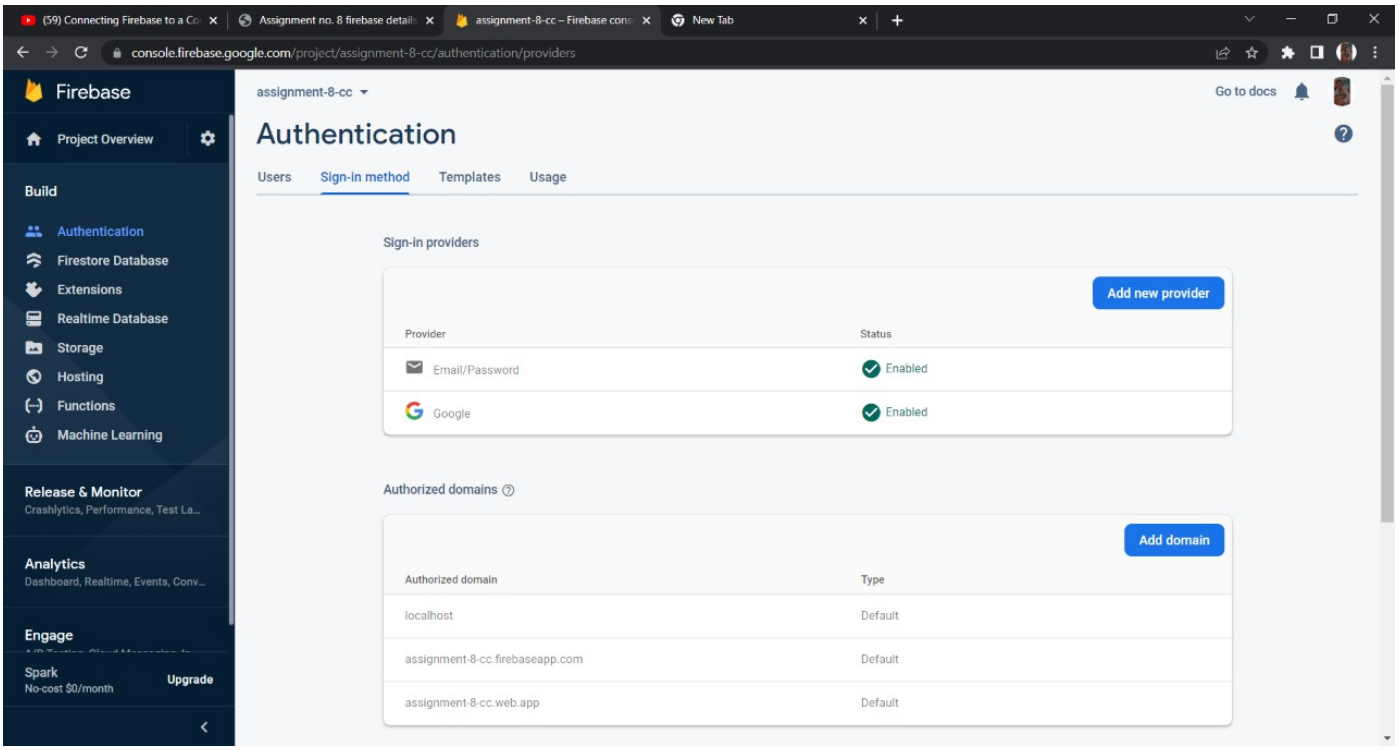
b. In the Sign-in methods you will see various options, select any one option:



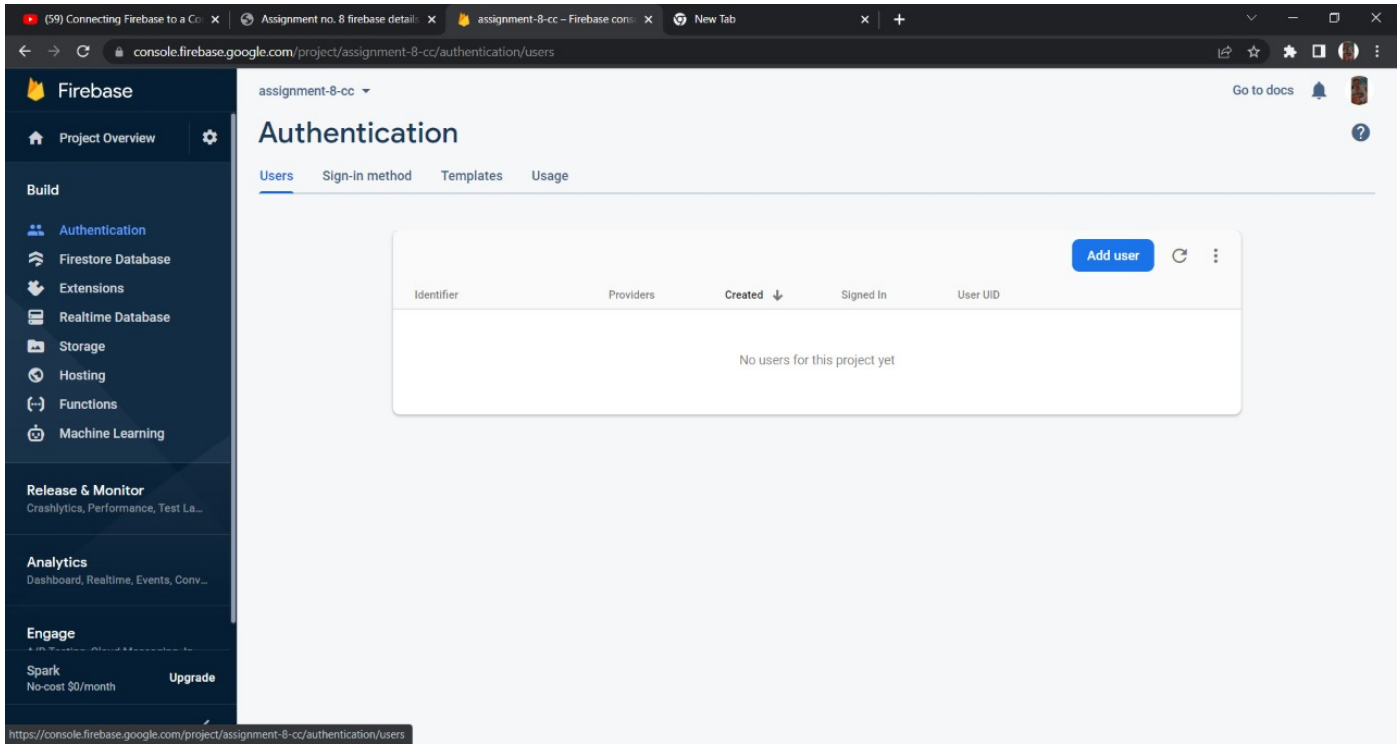
c. Perform appropriate configuration for that platform. You can also add domains:



d. Then you can see the added sign-in methods and the domains:

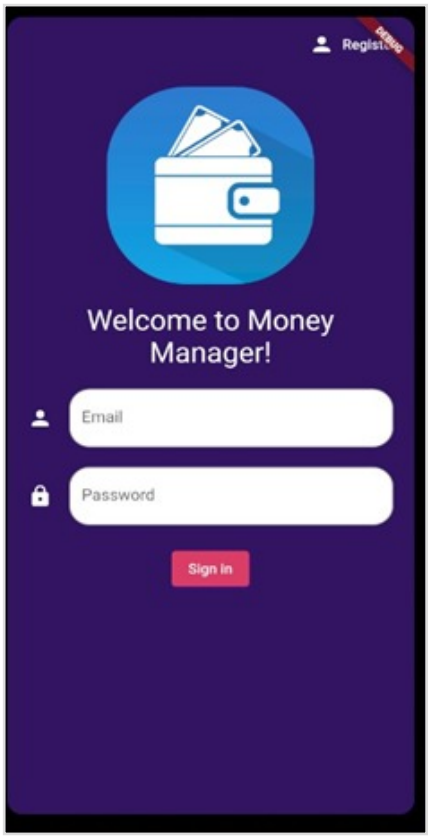


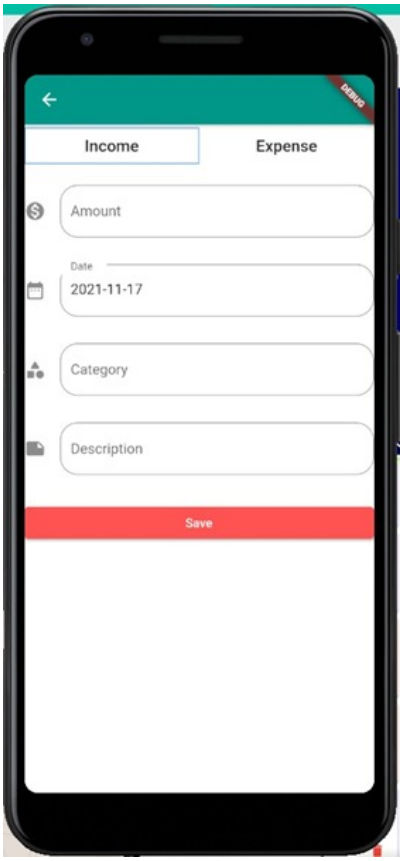
e. Also, when the users login to your application, their details will be visible at theUsers tab.

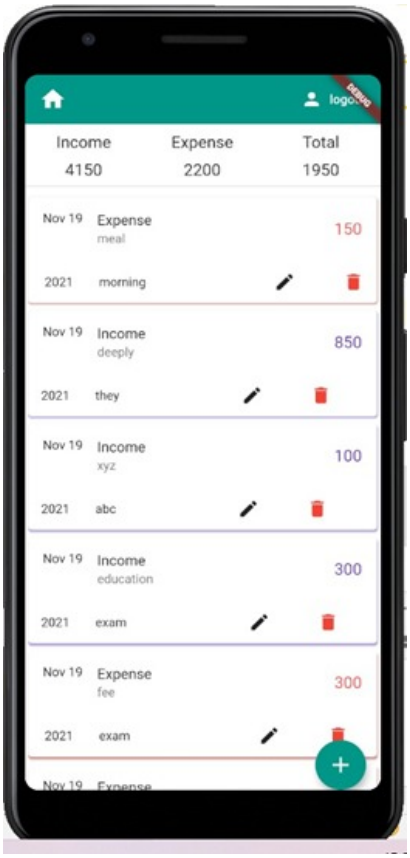


7. Installing dependencies and Running application locally:

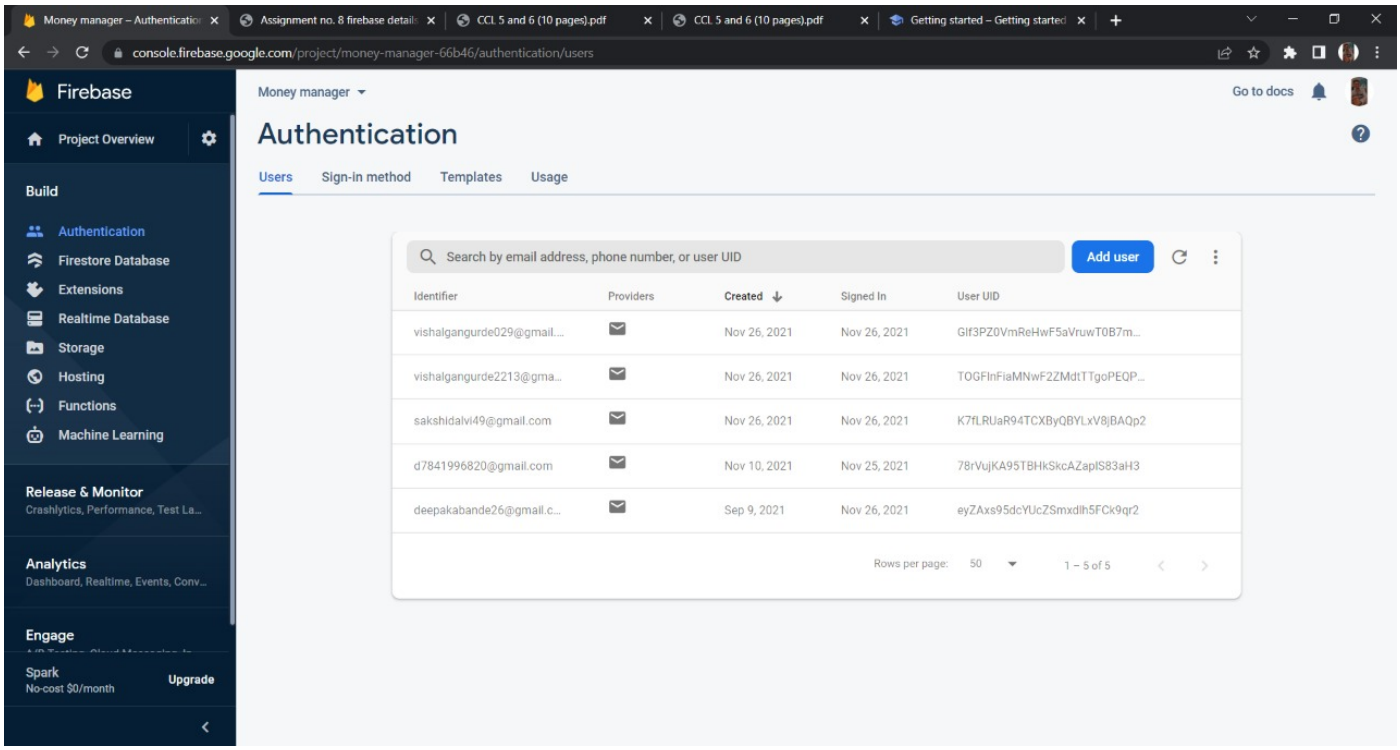
A. Our application



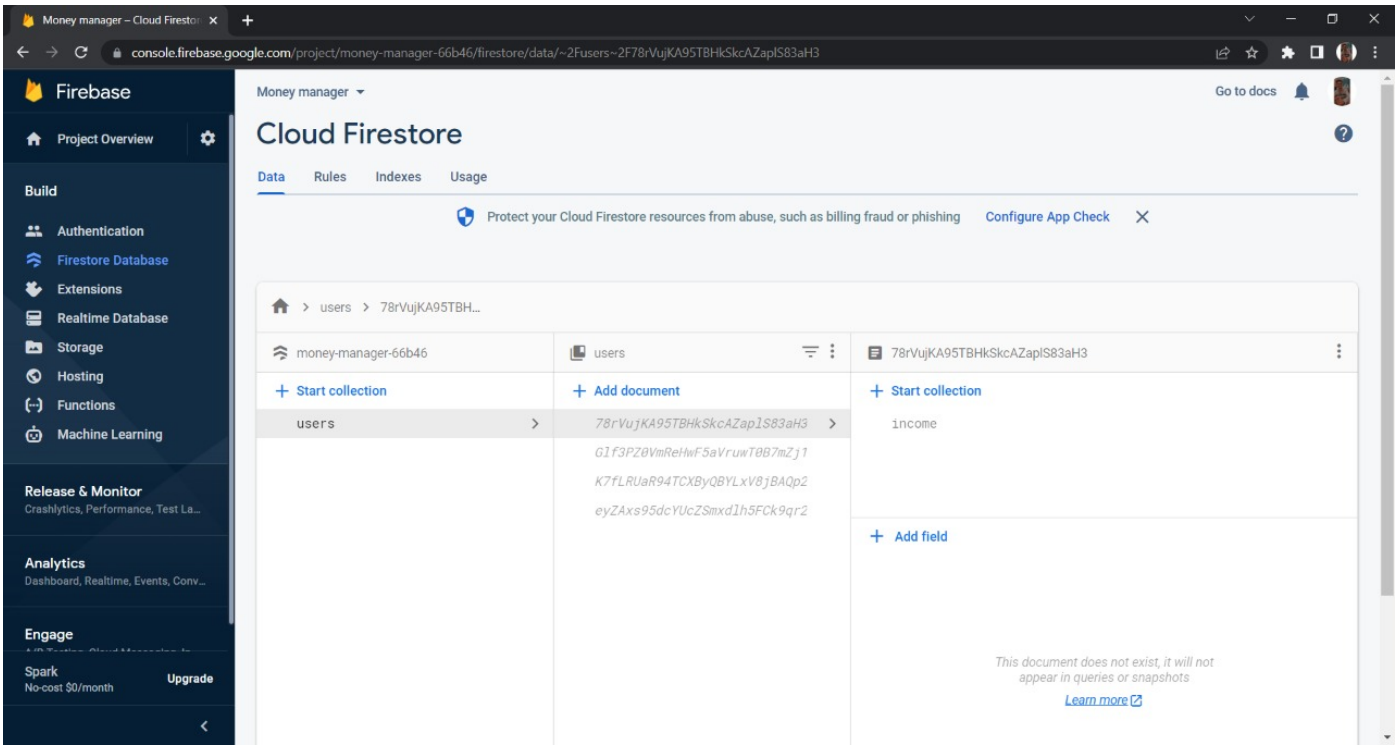




B. Authentication



C. Database



Conclusion:

Performed authentication using firebase to store user credentials and also successfully stored data on Firestore.