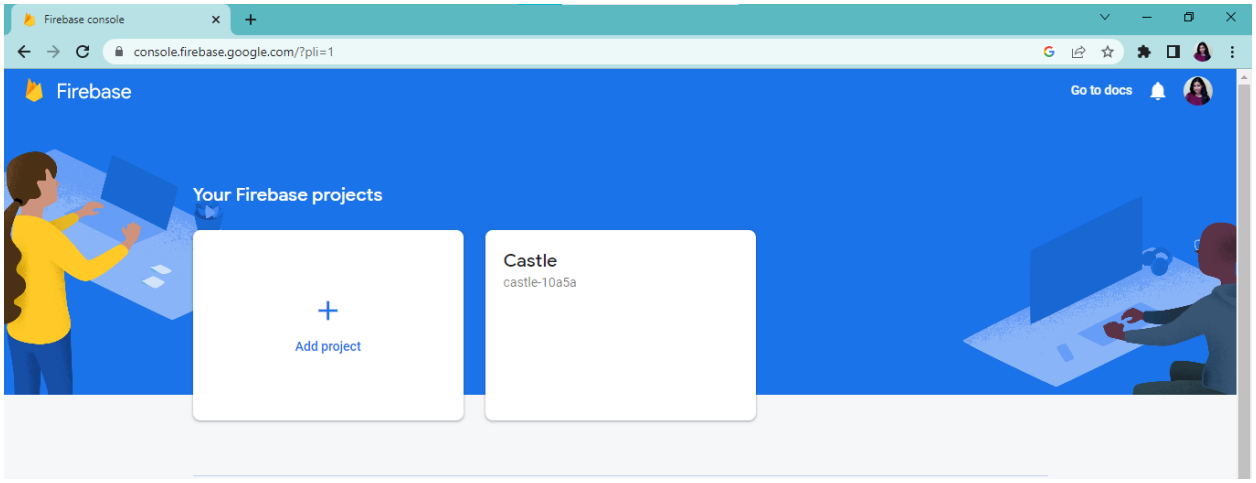
EXPERIMENT - 06

AIM: To Connect Flutter UI with firebase database

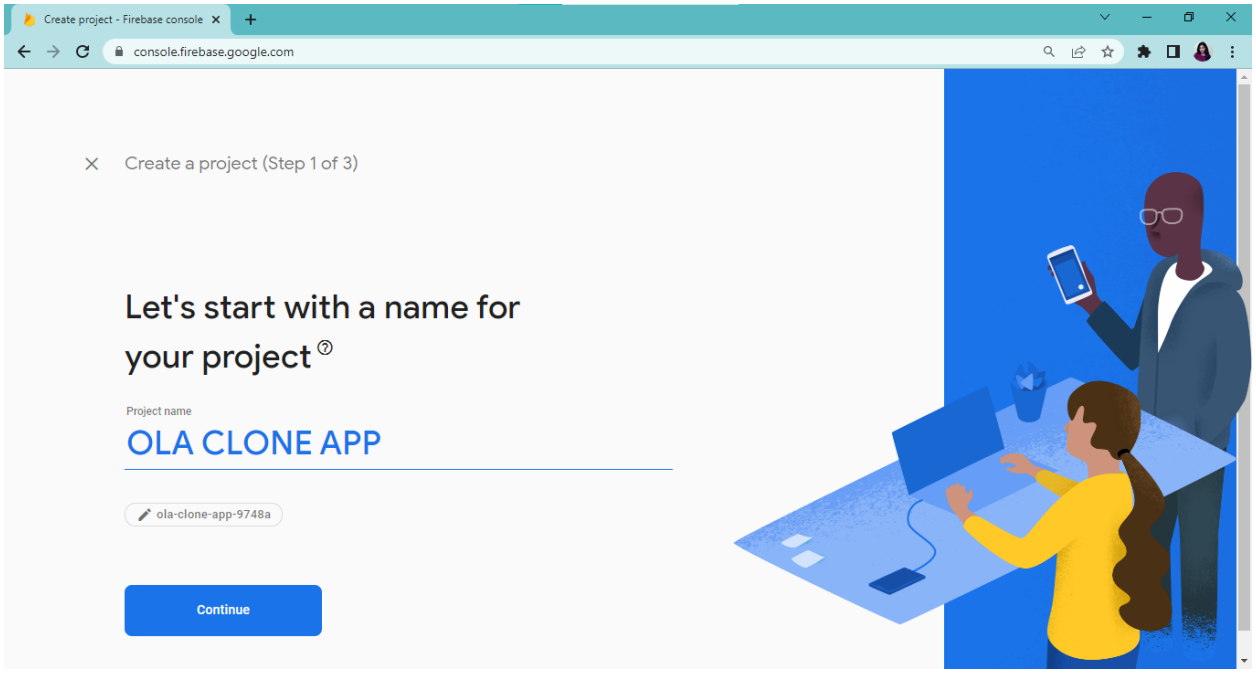
IMPLEMENTATION:

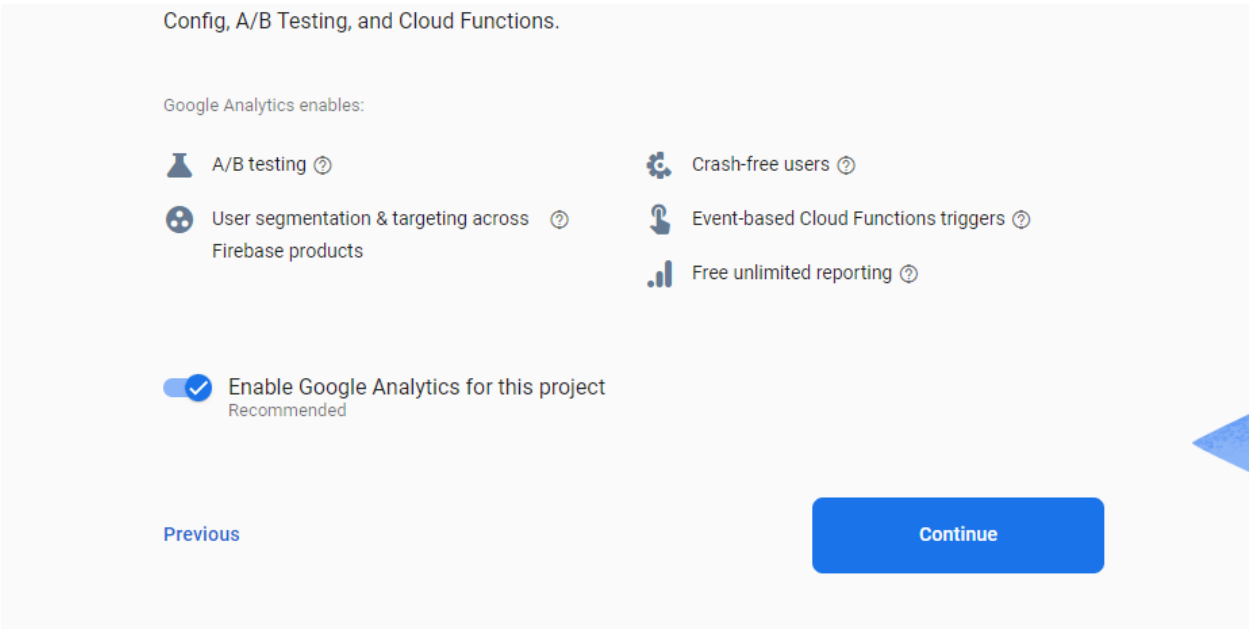
Connection with firebase

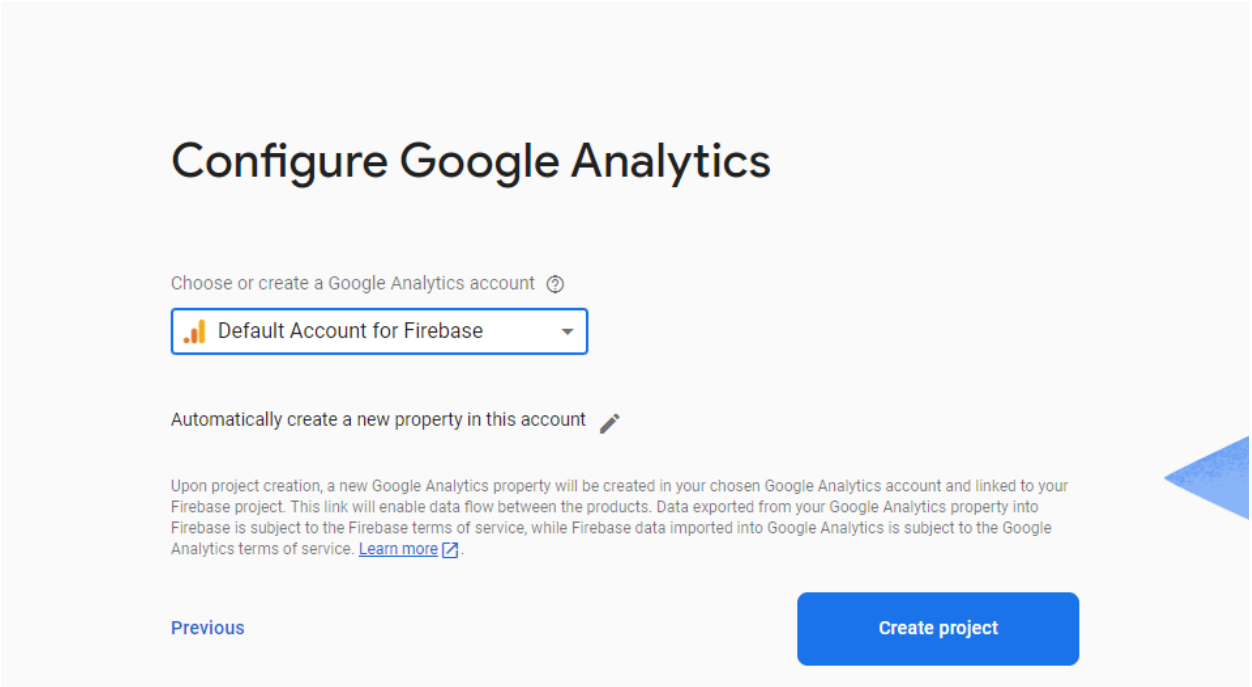
- **Go to firebase console**



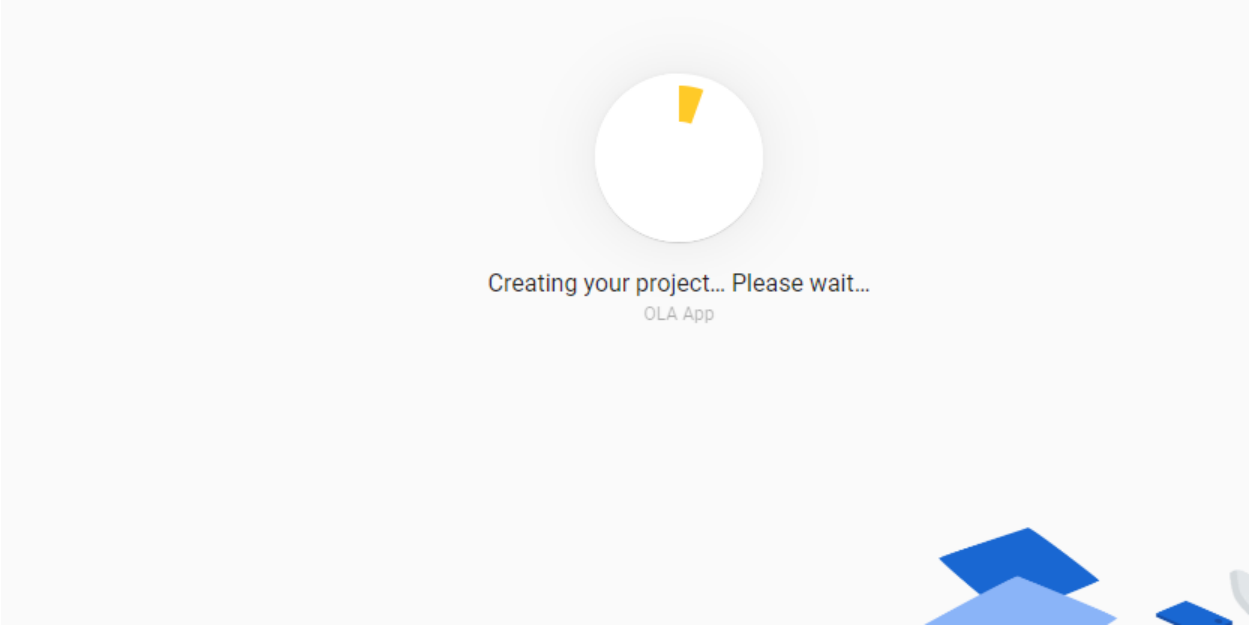
- **Click on "Add project"**
- **Name your project**



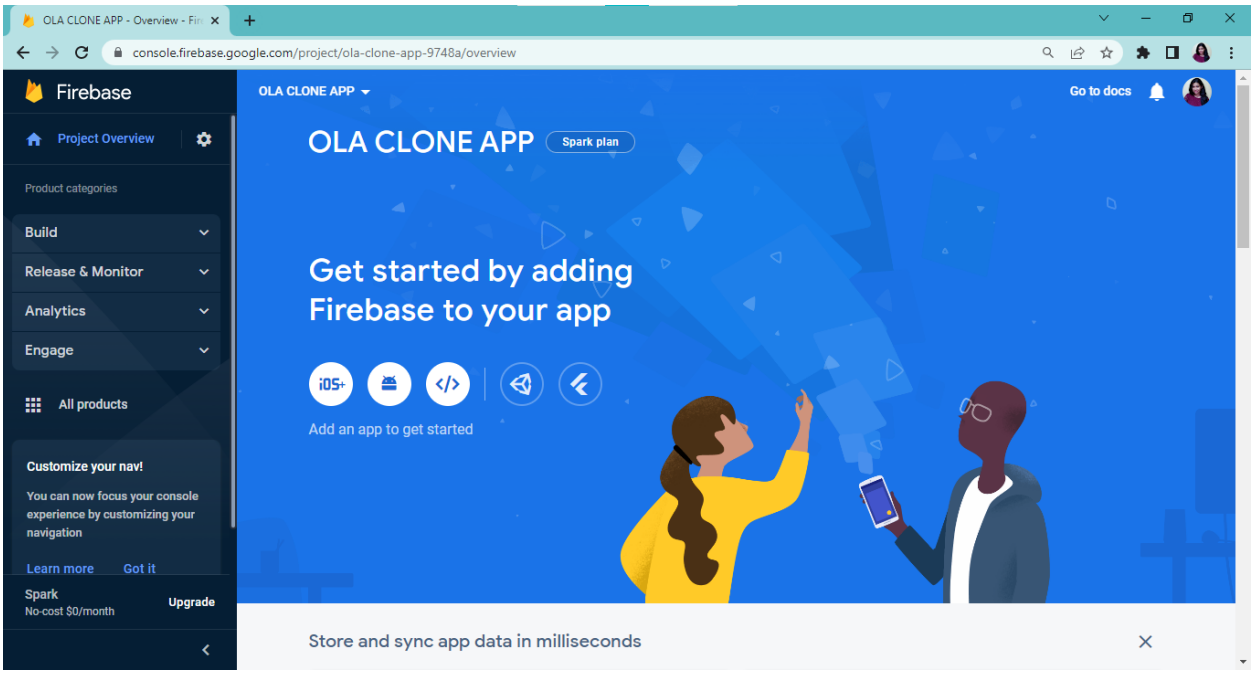

Config, A/B Testing, and Cloud Functions.
Google Analytics enables:
A/B testing
User segmentation & targeting across Firebase products
Crash-free users
Event-based Cloud Functions triggers
Free unlimited reporting
Enable Google Analytics for this project
Recommended
Previous
Continue


Configure Google Analytics
Choose or create a Google Analytics account
Default Account for Firebase
Automatically create a new property in this account
Upon project creation, a new Google Analytics property will be created in your chosen Google Analytics account and linked to your Firebase project. This link will enable data flow between the products. Data exported from your Google Analytics property into Firebase is subject to the Firebase terms of service, while Firebase data imported into Google Analytics is subject to the Google Analytics terms of service. Learn more.
Previous
Create project

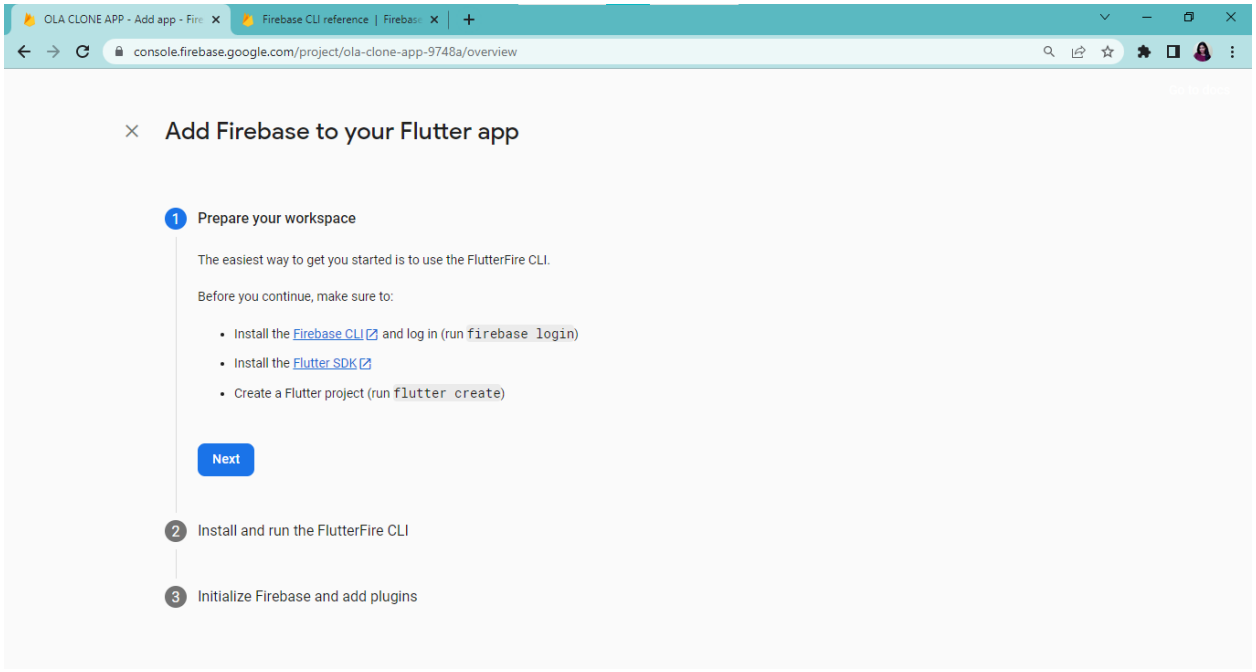
- **Now click on “create project”**



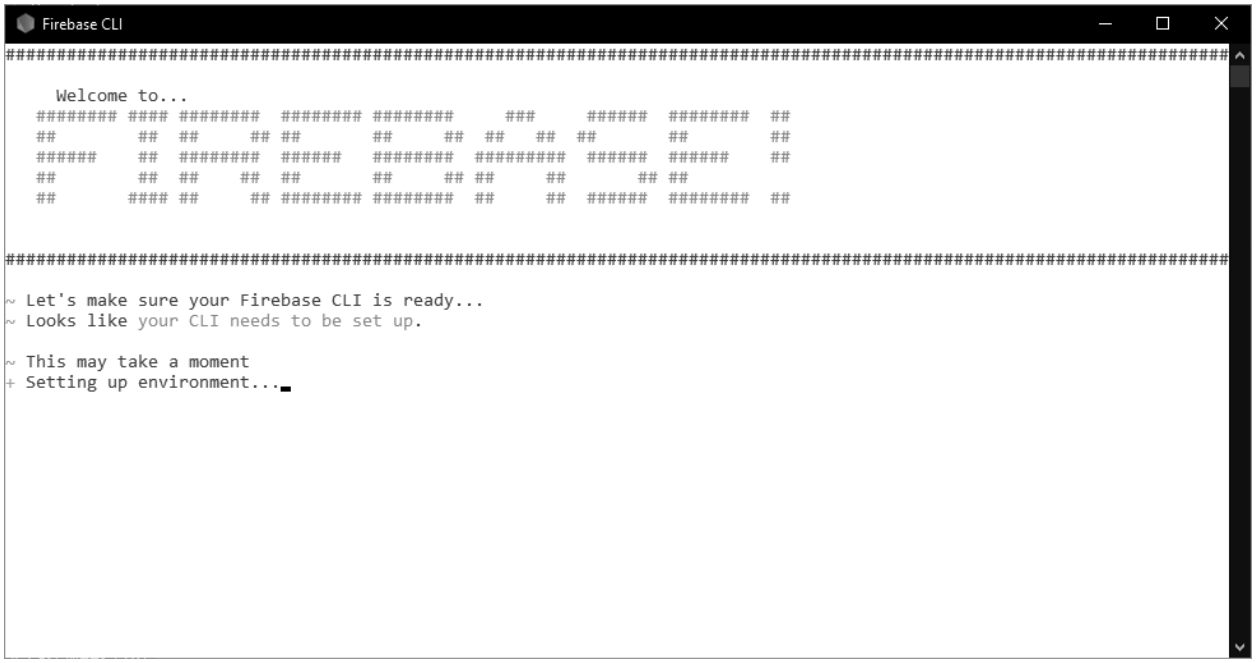
- **Your project is now ready**



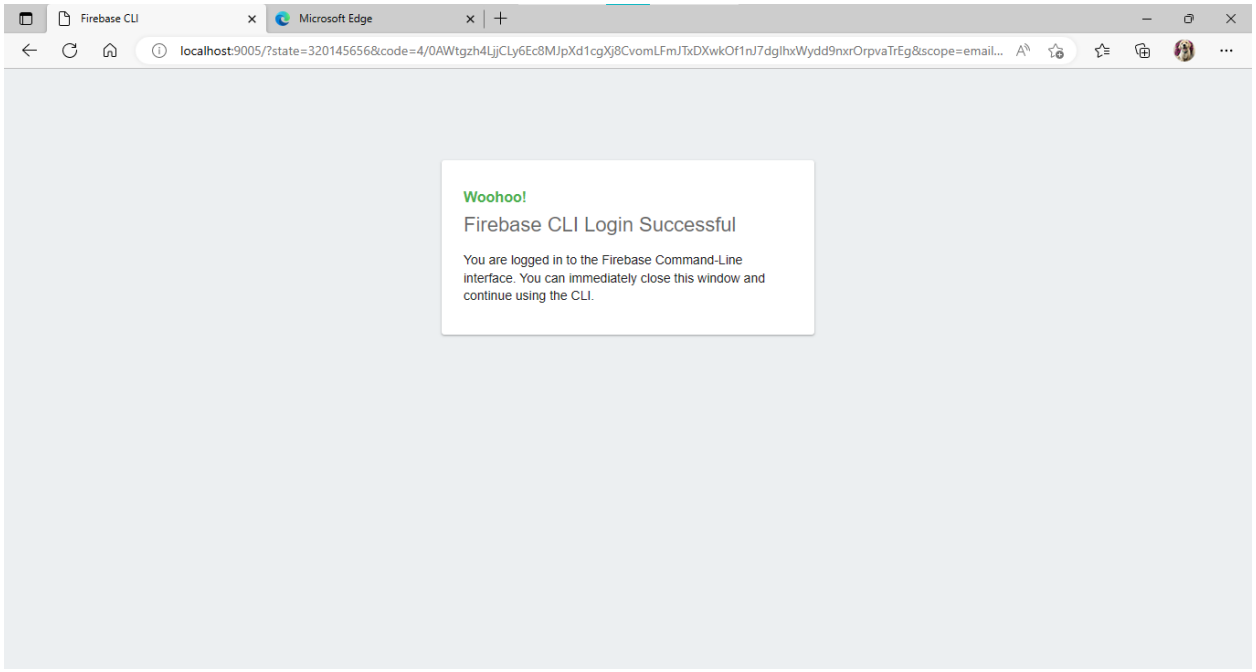
- **We have to connect flutter so click on flutter icon**



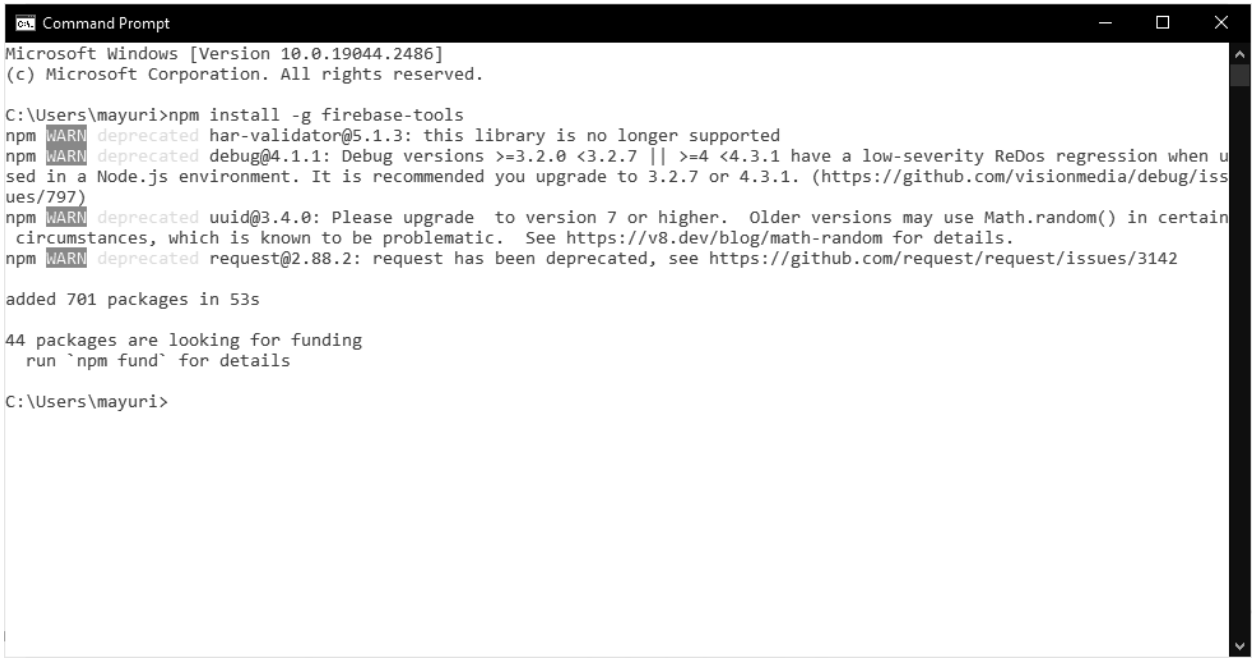
- **Install firebase cli and flutter sdk**
- **Set up your firebase CLI**



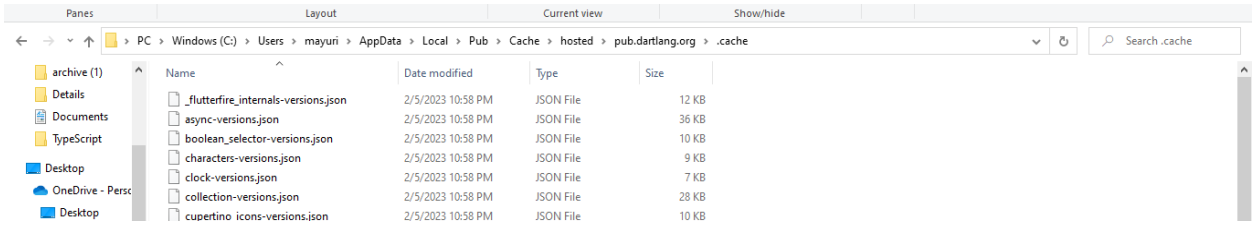
- **Login to the account with which you made the firebase project**



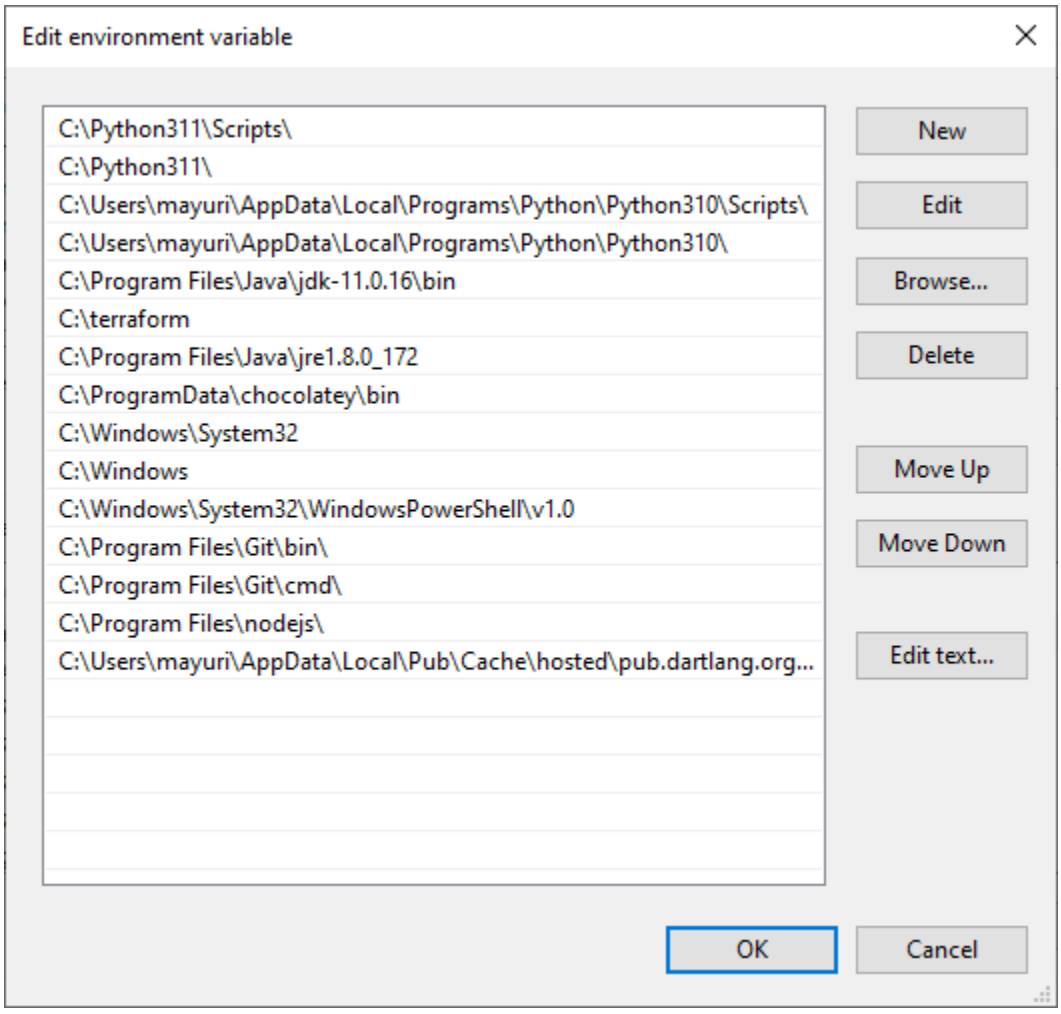
- **Make sure that you have node js installed in your system**
- **Run this command in cmd to install firebase tools**



- **Copy this path**



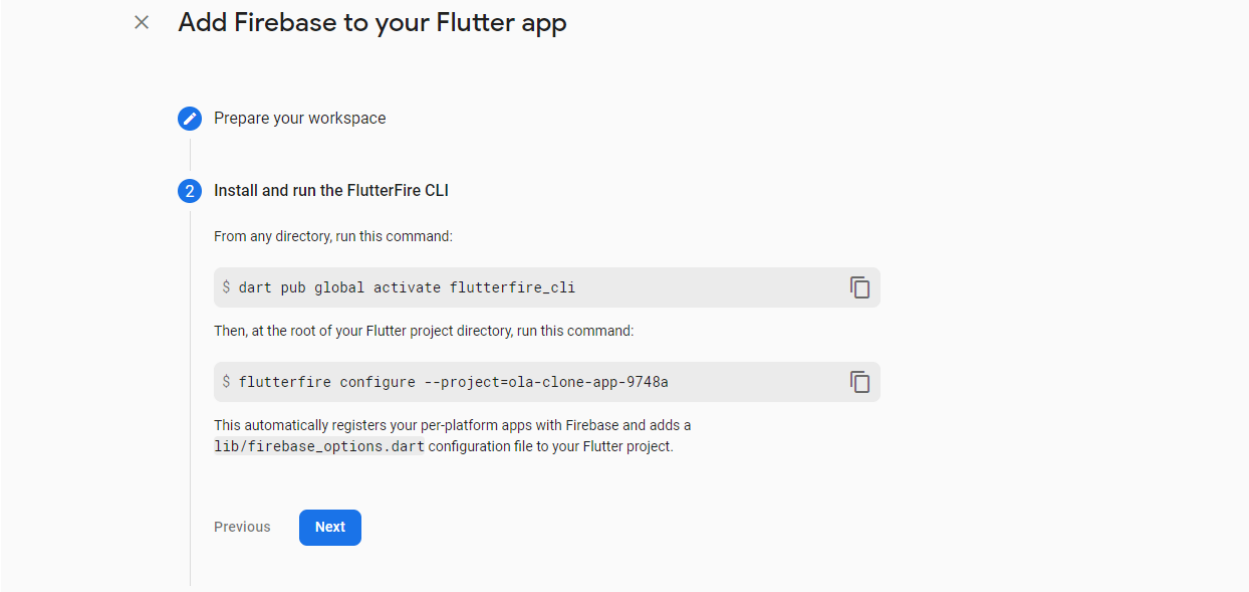
- **Edit this path into your system variables**



- **Check if your firebase is installed or not**

```
C:\Users\mayuri>firebase -V
11.22.0

C:\Users\mayuri>
```

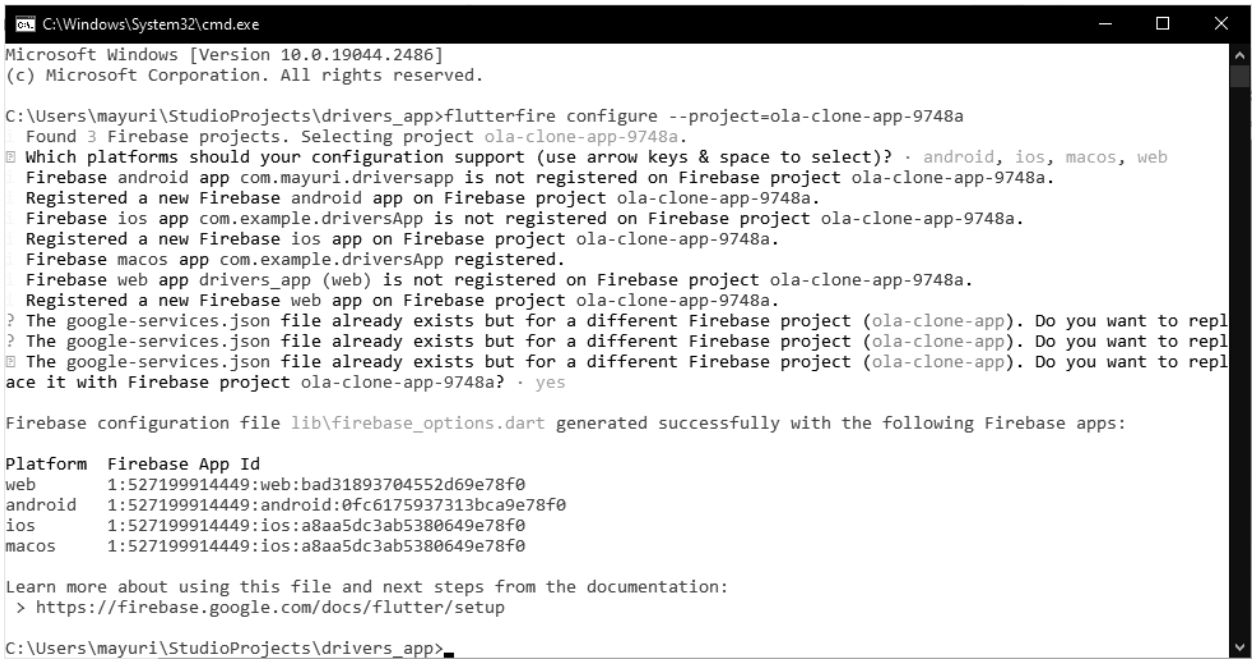
- **Now go back to flutter console**



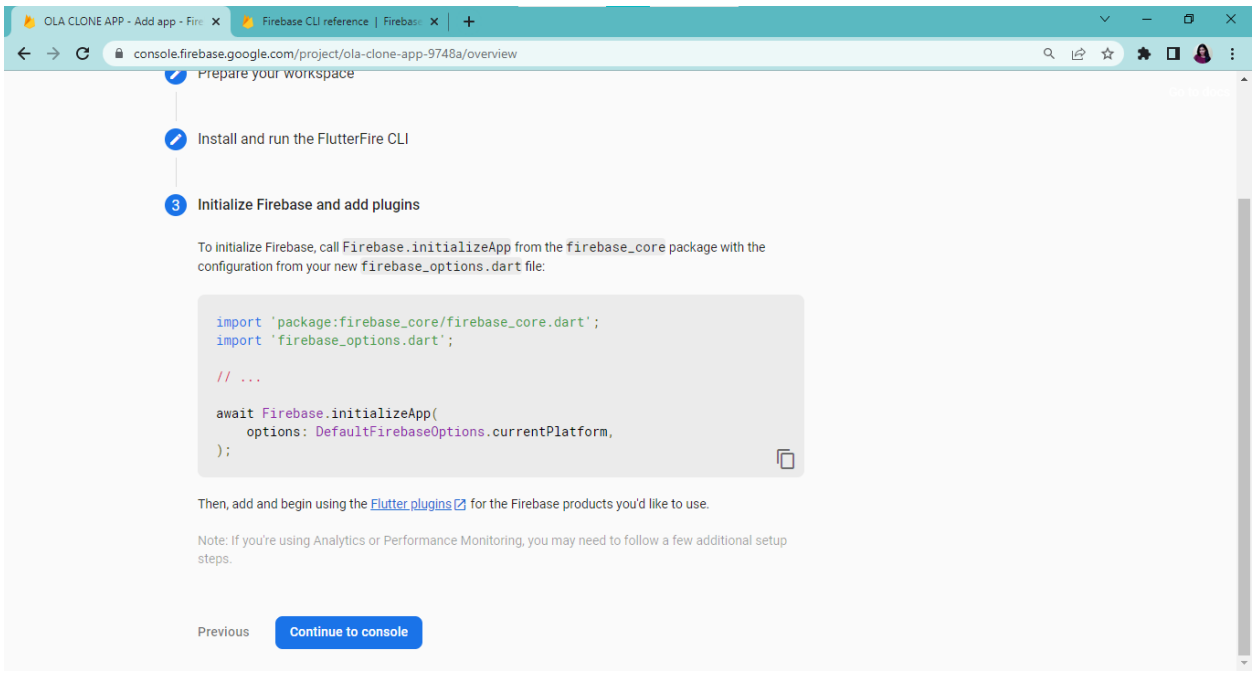
- **Run the first command on cmd**

```
C:\Users\mayuri>dart pub global activate flutterfire_cli
Package flutterfire_cli is currently active at version 0.2.7.
Resolving dependencies...
The package flutterfire_cli is already activated at newest available version.
To recompile executables, first run `dart pub global deactivate flutterfire_cli`.
Installed executable flutterfire.
Activated flutterfire_cli 0.2.7.

C:\Users\mayuri>
```

- **Now go to your root project and open cmd and run the second command**



- **Copy the two lines below**



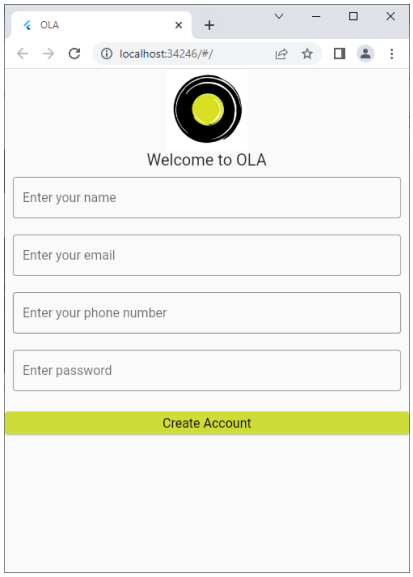
- **Paste it in your main.dart**

```
{
  WidgetsFlutterBinding.ensureInitialized();

  await Firebase.initializeApp(
    options: DefaultFirebaseOptions.currentPlatform,
  );
  runApp(
```

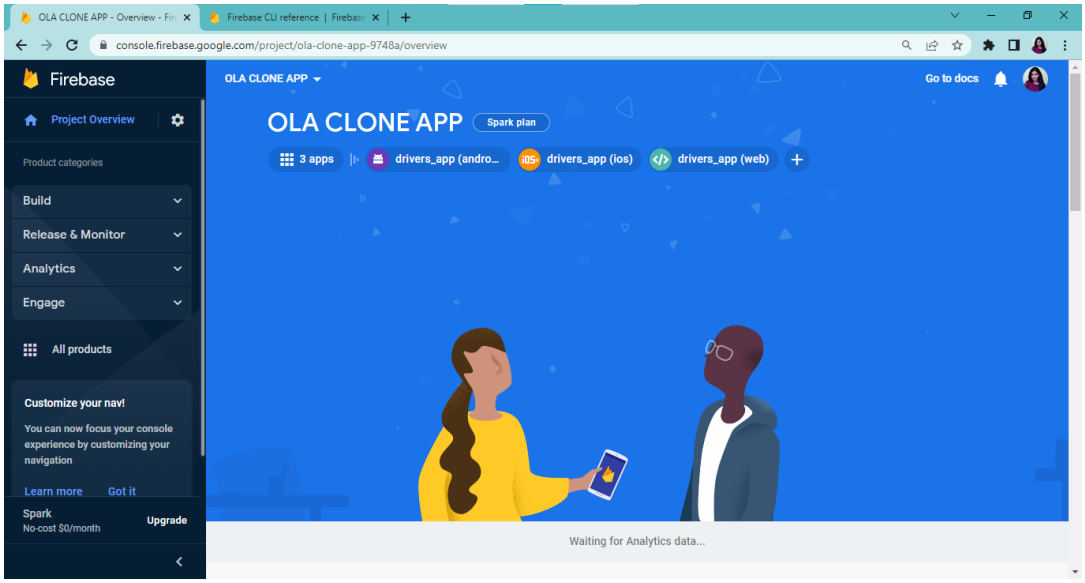
- **Import the packages in pubspec.yaml**

```
# Use with the CupertinoIcons class for iOS style
cupertino_icons: ^1.0.2
firebase_core: ^2.4.1
firebase_auth: ^4.2.5
firebase_database: ^10.0.9
```

- **Now run your project**
- **(If the project runs without any error, then your firebase has been connected successfully)**



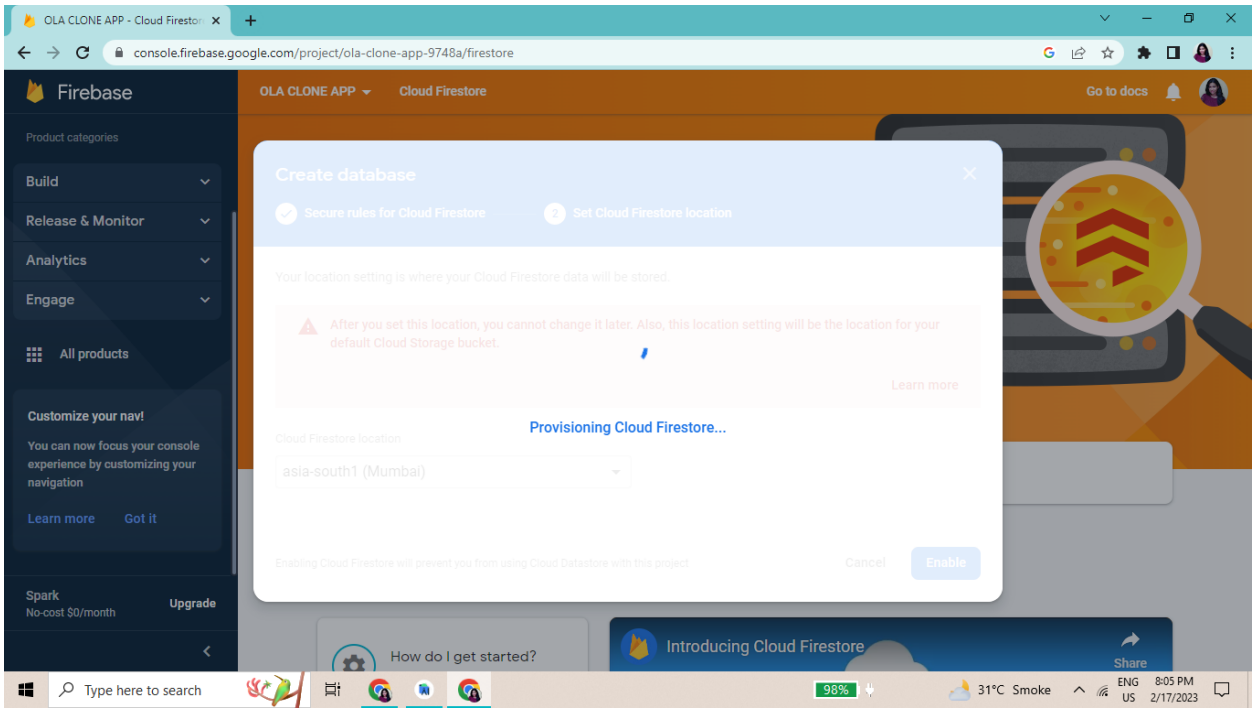
- **Refresh your Firebase console**



- **Firebase has been connected to our flutter project successfully.**

Uploading Data to Firebase

- **Create database in firestore**



- **Add these dependencies**

```
dependencies:
  flutter:
    sdk: flutter
  cloud_firestore: ^4.3.2



  # The following adds the Cupertino Icons font to your applicati
  # Use with the CupertinoIcons class for iOS style icons.
  cupertino_icons: ^1.0.2
  firebase_core: ^2.4.1
  firebase_auth: ^4.2.5
```

➢ **<u>Connecting login screen to firebase</u>**

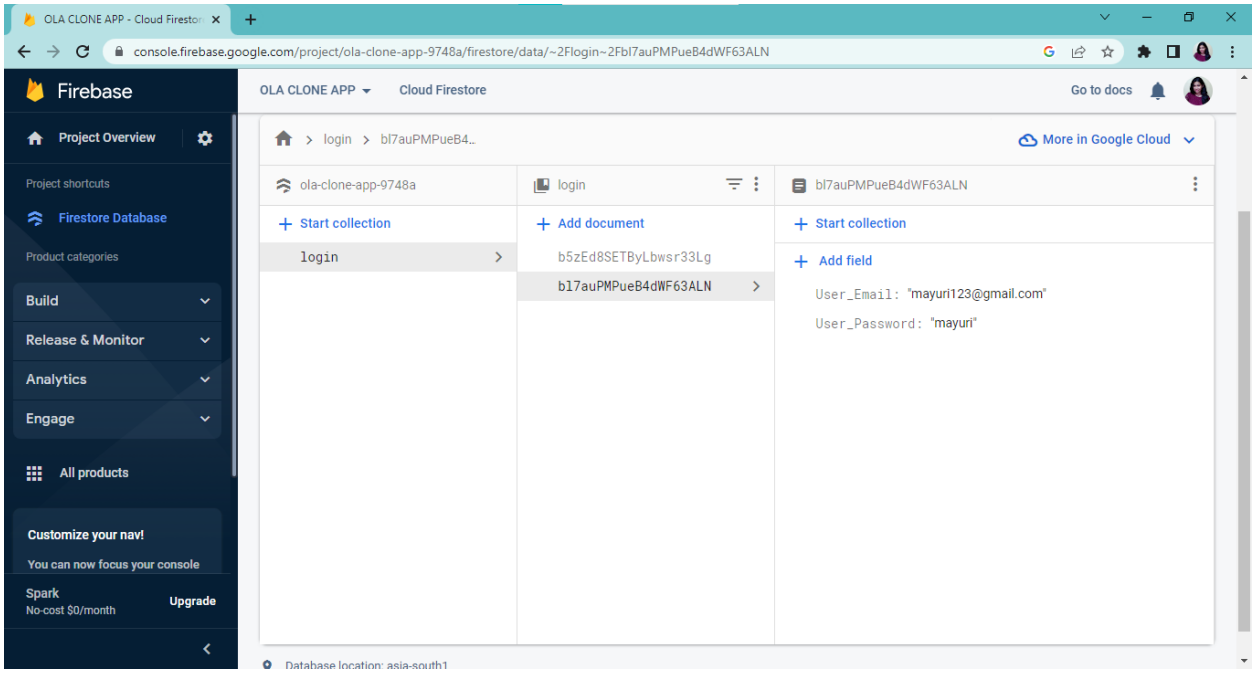
- **Add this in your login page**

```
class _MyStatefulWidgetState extends State<MyStatefulWidget> {
  TextEditingController emailController = TextEditingController();
  TextEditingController passwordController = TextEditingController();

  final _formKey = GlobalKey<FormState>();

  @override
```

- **Mapping our data to firebase**

```
onPressed: () {
  if (_formKey.currentState!.validate()) {
    ScaffoldMessenger.of(context).showSnackBar(
        const SnackBar(content: Text("You have successfully logged in")));

  }
  Map <String,dynamic> data= {"User_Email": emailController.text,"User_Password": passwordController.text};
  FirebaseFirestore.instance.collection("login").add(data);

  Navigator.push(context, MaterialPageRoute(builder: (context) => const MainScreen())
  );
```

- **Input data in your login screen**
- **The data is added to firestore database**



➢ Connecting to Registration Screen

Code:

```
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:drivers_app/authentication/signup_screen.dart';
import 'package:flutter/material.dart';

import '../mainScreens/main_screen.dart';
import 'car_info_screen.dart';
import 'login_screen.dart';

class SignUpScreen extends StatefulWidget {
  const SignUpScreen({Key? key}) : super(key: key);

  @override
  State<SignUpScreen> createState() => _SignUpScreenState();
}

class _SignUpScreenState extends State<SignUpScreen> {

  @override
  Widget build(BuildContext context) {
    return Scaffold(
```

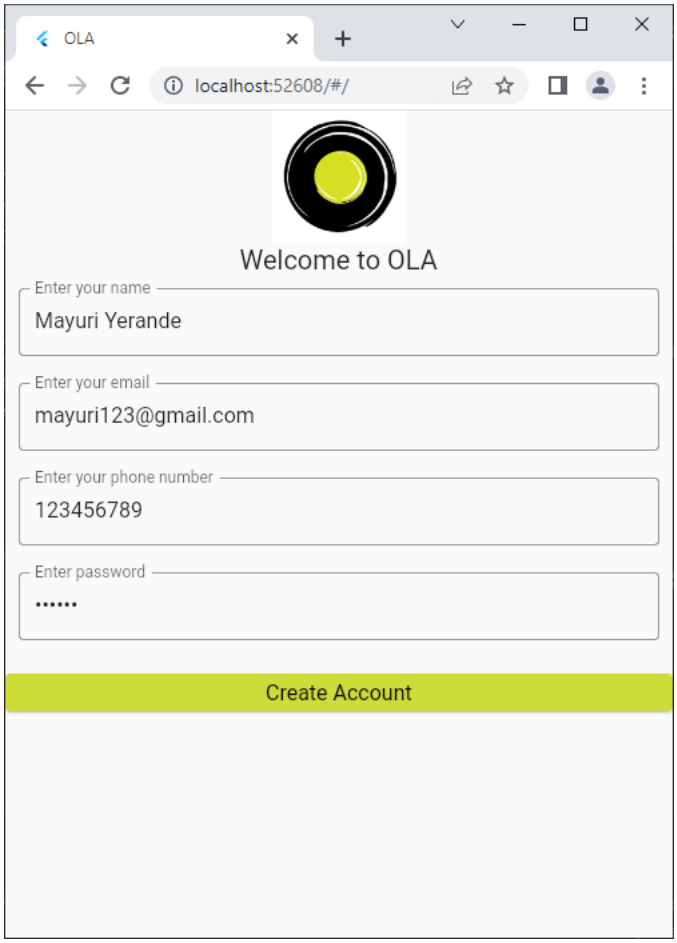
```
      body: MyStatefulWidget(
      ),
    );
  }
}

class MyStatefulWidget extends StatefulWidget {
  const MyStatefulWidget({Key? key}) : super(key: key);

  @override
  State<MyStatefulWidget> createState() => _MyStatefulWidgetState();
}

class _MyStatefulWidgetState extends State<MyStatefulWidget> {
  TextEditingController nameTextEditingController = TextEditingController();
  TextEditingController emailTextEditingController = TextEditingController();
  TextEditingController phoneTextEditingController = TextEditingController();
  TextEditingController passwordTextEditingController = TextEditingController();

  final _formKey = GlobalKey<FormState>();

  @override
  Widget build(BuildContext context) {
    return Form(

      key: _formKey,
      child: ListView(

        children: <Widget>[
          Container(
            width: 100,
            height: 100,
            decoration: BoxDecoration(
              image: DecorationImage(
                image: AssetImage("images/ola.png"),
              ),
            )
          ),
          Container(
            child: Center(
              child: Text("Welcome to OLA",style: TextStyle(fontSize:(20)),),
            ),
          ),
          Container(
            padding: const EdgeInsets.fromLTRB(10, 10, 10, 10),
            child: Column(
              crossAxisAlignment: CrossAxisAlignment.start,
              children: [
                TextFormField(
                  controller: nameTextEditingController,
                  decoration: const InputDecoration(
```

```
          border: OutlineInputBorder(),
          labelText: 'Enter your name',
        ),
        validator: (value) {
          if (value == null || value.isEmpty) {
            return "Please enter some text";
          }
          return null;
        },
      ),
    ],
  ),
),
Container(
  padding: const EdgeInsets.fromLTRB(10, 10, 10, 10),
  child: TextFormField(
    obscureText: true,
    controller: emailTextEditingController,
    decoration: const InputDecoration(
      border: OutlineInputBorder(),
      labelText: 'Enter your email',
    ),
    validator: (value) {
      if (value == null || value.isEmpty) {
        return "Please enter some text";
      }
      return null;
    },
  ),
),
Container(
  padding: const EdgeInsets.fromLTRB(10, 10, 10, 10),
  child: TextFormField(
    obscureText: true,
    controller: phoneTextEditingController,
    decoration: const InputDecoration(
      border: OutlineInputBorder(),
      labelText: 'Enter your phone number',
    ),
    validator: (value) {
      if (value == null || value.isEmpty) {
        return "Please enter some text";
      }
      return null;
    },
  ),
),
Container(
  padding: const EdgeInsets.fromLTRB(10, 10, 10, 10),
  child: TextFormField(
    obscureText: true,
```

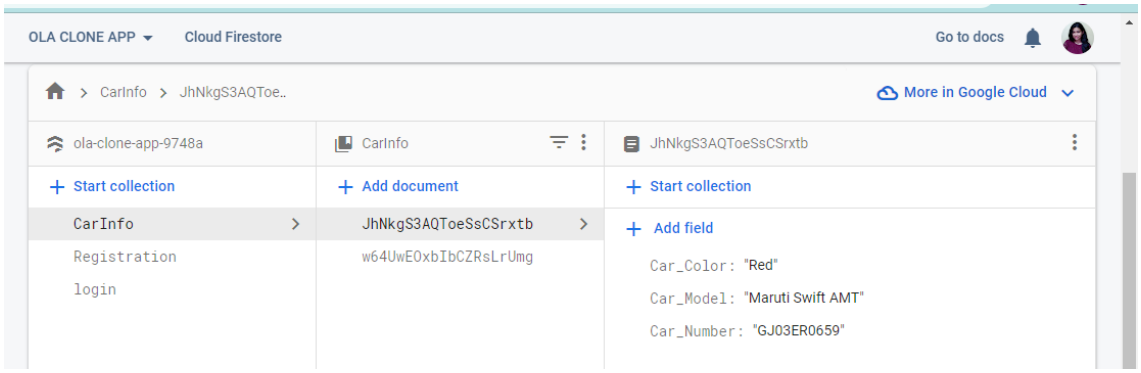
```
            controller: passwordTextEditingController,
            decoration: const InputDecoration(
              border: OutlineInputBorder(),
              labelText: 'Enter password',
            ),
            validator: (value) {
              if (value == null || value.isEmpty) {
                return "Please enter some text";
              }
              return null;
            },
          ),
        ),
        const SizedBox(height: 15),
        Container(
            child: ElevatedButton(

              style: ElevatedButton.styleFrom(
                backgroundColor: Colors.lime,
              ),

              child: const Text(
                'Create Account',
                style: TextStyle(fontSize: (16),color: Colors.black),
              ),
              onPressed: () {
                if (_formKey.currentState!.validate()) {
                  ScaffoldMessenger.of(context).showSnackBar(
                      const SnackBar(content: Text("Processing data,Your Account has been created
successfully")));
                  Navigator.push(context, MaterialPageRoute(builder: (context) => const CarInfoScreen())
                  );

                }
                Map <String,dynamic> data= {"User_Name":
nameTextEditingController.text,"User_Email":
emailTextEditingController.text,"User_Phone_Number":
phoneTextEditingController,"user_Password": passwordTextEditingController};
                FirebaseFirestore.instance.collection("Registration").add(data);



              },

            )),


      ],
    ));
  }
}
```

Output:

(Input data)



(Data reflected in database)



➢ Uploading Car Info screen data to firebase

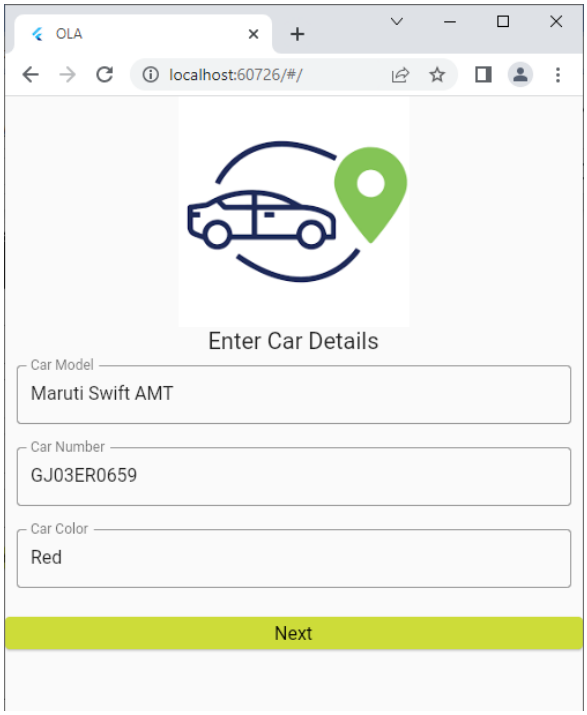
Code:

```
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:flutter/material.dart';
import 'login_screen.dart';
class CarInfoScreen extends StatefulWidget {
  const CarInfoScreen({Key? key}) : super(key: key);

  @override
  State<CarInfoScreen> createState() => _CarInfoScreenState();
}

class _CarInfoScreenState extends State<CarInfoScreen> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(

      body: MyStatefulWidget(
      ),
    );
  }
}

class MyStatefulWidget extends StatefulWidget {
  const MyStatefulWidget({Key? key}) : super(key: key);

  @override
  State<MyStatefulWidget> createState() => _MyStatefulWidgetState();
}

class _MyStatefulWidgetState extends State<MyStatefulWidget> {
  TextEditingController carModelTextEditingController = TextEditingController();
  TextEditingController carNumberTextEditingController = TextEditingController();
  TextEditingController carColorTextEditingController = TextEditingController();
  final _formKey = GlobalKey<FormState>();

  @override
  Widget build(BuildContext context) {
    return Form(

      key: _formKey,
      child: ListView(

        children: <Widget>[
          Container(
            width: 600,
            height: 200,
            decoration: BoxDecoration(
```

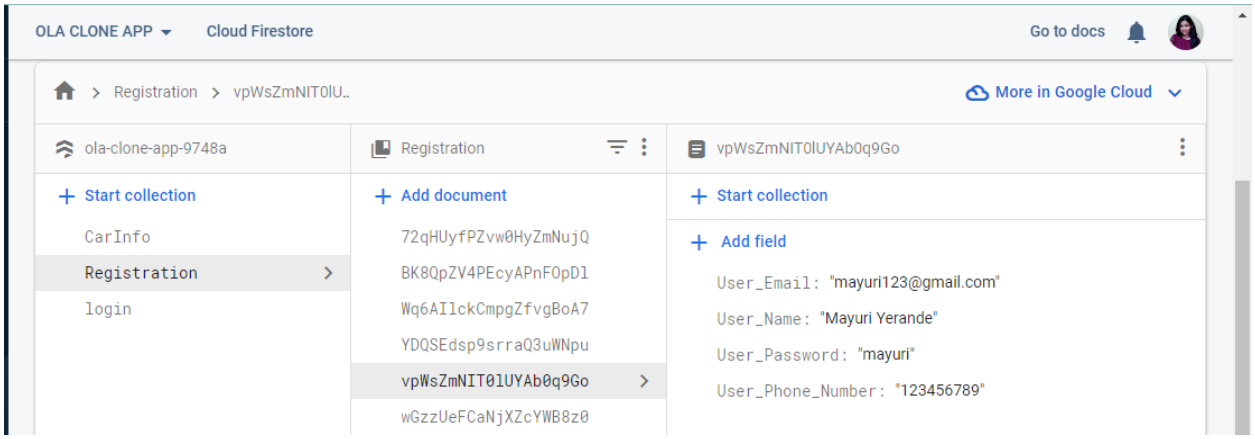
```
      image: DecorationImage(
        image: AssetImage("images/car_info.png"),
      ),
    )
),
Container(
  child: Center(
    child: Text("Enter Car Details",style: TextStyle(fontSize:(20)),),
  ),
),
Container(
  padding: const EdgeInsets.fromLTRB(10, 10, 10, 10),
  child: Column(
    crossAxisAlignment: CrossAxisAlignment.start,
    children: [
      TextFormField(
        controller: carModelTextEditingController,
        decoration: const InputDecoration(
          border: OutlineInputBorder(),
          labelText: 'Car Model',
        ),
        validator: (value) {
          if (value == null || value.isEmpty) {
            return "Please enter some text";
          }
          return null;
        },
      ),
    ],
  ),
),
Container(
  padding: const EdgeInsets.fromLTRB(10, 10, 10, 10),
  child: TextFormField(
    controller: carNumberTextEditingController,
    decoration: const InputDecoration(
      border: OutlineInputBorder(),
      labelText: 'Car Number',
    ),
    validator: (value) {
      if (value == null || value.isEmpty) {
        return "Please enter some text";
      }
      return null;
    },
  ),
),
Container(
  padding: const EdgeInsets.fromLTRB(10, 10, 10, 10),
  child: TextFormField(
    controller: carColorTextEditingController,
```

```
            decoration: const InputDecoration(
              border: OutlineInputBorder(),
              labelText: 'Car Color',
            ),
            validator: (value) {
              if (value == null || value.isEmpty) {
                return "Please enter some text";
              }
              return null;
            },
          ),
        ),
        const SizedBox(height: 15),
        Container(
            child: ElevatedButton(

              style: ElevatedButton.styleFrom(
                backgroundColor: Colors.lime,
              ),

              child: const Text(
                'Next',
                style: TextStyle(fontSize: (16),color: Colors.black),
              ),
              onPressed: () {
                if (_formKey.currentState!.validate()) {
                  ScaffoldMessenger.of(context).showSnackBar(
                      const SnackBar(content: Text("Processing data")));

                }
                Map <String,dynamic> data2 = {"Car_Model":
carModelTextEditingController.text,"Car_Color":carColorTextEditingController.text,"Car_Numb
er": carNumberTextEditingController.text};
                FirebaseFirestore.instance.collection("CarInfo").add(data2);

                Navigator.push(context, MaterialPageRoute(builder: (context) => const LoginScreen())
                );


              },

            )),
        const SizedBox(height: 15),

      ],
    ));
  }
}
```

Input:



- **Data reflected in database**



CONCLUSION: We created a project in the firebase console. Then we imported the required packages to run it and Then we connected our project to firebase with some commands mentioned. The data which we enter in our App is successfully getting added to our database. Thus we successfully connected our project to firebase.