MPL Experiment 6

Name: Mohit Patil   **Class:** D15A   **Roll no:**36

Aim: How To Set Up Firebase with Flutter for iOS and Android Apps
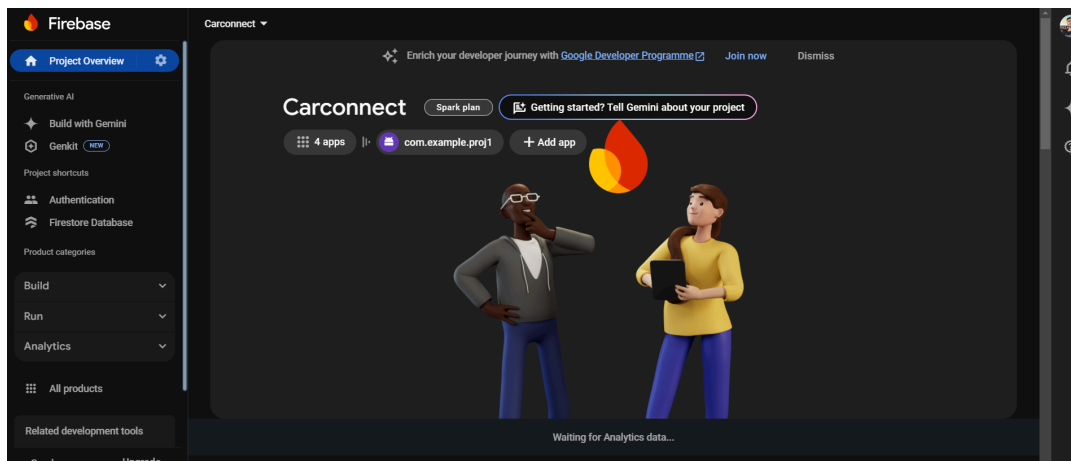
Theory: -

Flutter is an open-source UI toolkit developed by Google for building natively compiled applications for mobile, web, and desktop from a single codebase. Firebase, a Backend-as-a-Service (BaaS) platform, provides real-time database, authentication, and cloud storage services, making it a powerful backend solution for Flutter applications.
By integrating Firebase with Flutter, developers can store and retrieve data in real time, authenticate users, and manage cloud-based data efficiently. This is particularly useful for applications requiring dynamic content updates and user interactions.
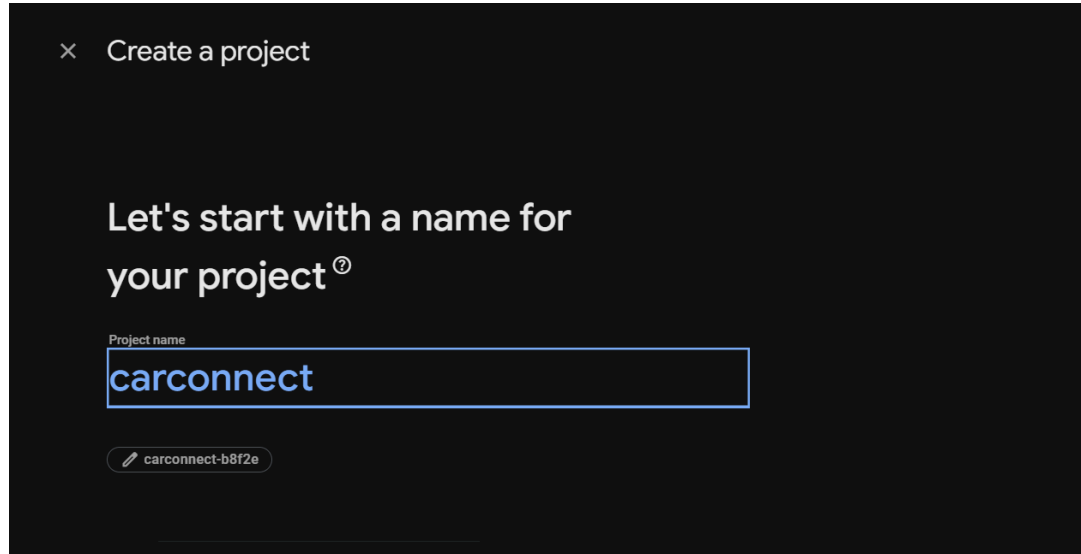
Steps to Connect Flutter UI with Firebase Database:

Step 1:
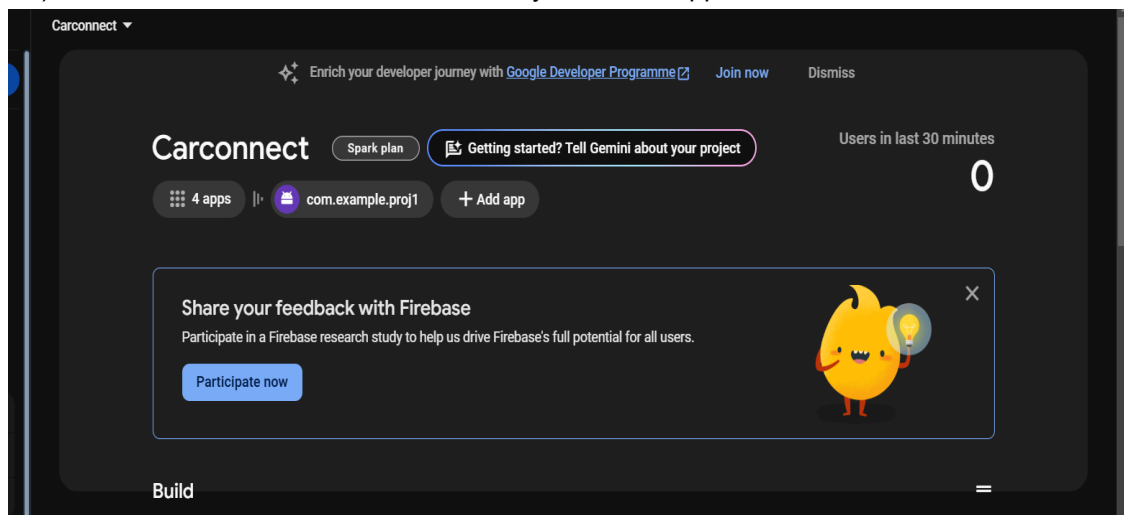1.1)   Go to Firebase Console and Create a Firebase Project

1.2)   Click on Create a Project and give it a suitable name.

Step 2:- Add Firebase to Your Flutter App

2.1)   Click on Android/iOS/Web based on your Flutter application

Step 3: - Add Firebase Authentication to Your App

3.1)   Add Firebase Authentication Dependencies

```
dependencies:
  flutter:
    sdk: flutter
  firebase_core: ^3.11.0
  firebase_auth: ^5.4.2   # For authentication
  cloud_firestore: ^5.6.3   # For Firestore, if you need it
  firebase_messaging: ^15.2.2
  http: ^0.13.3
  image_picker: ^1.0.4
  tflite_flutter: ^0.11.0
  image: ^3.2.0
  url_launcher: ^6.1.14
```

3.2)   Enable Authentication in Firebase Console Go to Firebase Console → Authentication. Click on Sign-in method and enable Email/Password (or any other method like Google). Click Save

3.3)

Implement Authentication in Flutter Modify main.dart

import 'package:firebase_core/firebase_core.dart'; import 'package:firebase_auth/firebase_auth.dart';

void main() async { WidgetsFlutterBinding.ensureInitialized(); await Firebase.initializeApp();
runApp(MyApp());
}

Step 4: -Configure Firebase Realtime Database

1. Go to Firebase Console → Realtime Database.
2. Click Create Database → Choose location → Set rules (for development, set read/write to true).
3. Click Publish.

CODE:

Login page.dart

```dart
import 'package:flutter/material.dart';
import 'package:google_fonts/google_fonts.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:google_sign_in/google_sign_in.dart';

class LoginPage extends StatefulWidget {
  const LoginPage({super.key});

  @override
  _LoginPageState createState() => _LoginPageState();
}

class _LoginPageState extends State<LoginPage> {
  final _formKey = GlobalKey<FormState>();
  final TextEditingController _emailController = TextEditingController();
  final TextEditingController _passwordController = TextEditingController();
  final FirebaseAuth _auth = FirebaseAuth.instance;

  // Use the renderButton method for Google Sign-In on web
  void _signInWithGoogle() async {
    final GoogleSignIn googleSignIn = GoogleSignIn();

    try {
```

```dart
        GoogleSignInAccount? googleUser = await googleSignIn.signIn();
        if (googleUser == null) return;

        final GoogleSignInAuthentication googleAuth =
            await googleUser.authentication;

        final AuthCredential credential = GoogleAuthProvider.credential(
          accessToken: googleAuth.accessToken,
          idToken: googleAuth.idToken,
        );

        await _auth.signInWithCredential(credential);
        ScaffoldMessenger.of(context).showSnackBar(
          const SnackBar(content: Text('Google Sign-In Successful!')),
        );
      } catch (e) {
        print("Google Sign-In Error: $e");
        ScaffoldMessenger.of(context).showSnackBar(
          SnackBar(content: Text('Google Sign-In Failed: $e')),
        );
      }
    }

    void _signInWithEmailAndPassword() async {
      if (_formKey.currentState!.validate()) {
        try {
          await _auth.signInWithEmailAndPassword(
            email: _emailController.text.trim(),
            password: _passwordController.text.trim(),
          );
          ScaffoldMessenger.of(context).showSnackBar(
            const SnackBar(content: Text('Login Successful!')),
          );

          // Redirect to Home Page after login
          Navigator.pushReplacementNamed(context, '/home');
        } catch (e) {
          ScaffoldMessenger.of(context).showSnackBar(
            SnackBar(content: Text('Login Failed: $e')),
          );
        }
      }
    }

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

```dart
body: Container(
  color: Colors.black,
  child: Center(
    child: Padding(
      padding: const EdgeInsets.all(16.0),
      child: Card(
        shape: RoundedRectangleBorder(
          borderRadius: BorderRadius.circular(20),
        ),
        elevation: 10,
        color: Colors.white,
        child: Padding(
          padding: const EdgeInsets.all(30.0),
          child: Form(
            key: _formKey,
            child: Column(
              mainAxisSize: MainAxisSize.min,
              crossAxisAlignment: CrossAxisAlignment.stretch,
              children: [
                Column(
                  children: [
                    Image.asset(
                      'assets/c1.png', // Ensure image is in assets folder
                      height: 150,
                    ),
                    const SizedBox(height: 10),
                    Text(
                      'CarConnect',
                      style: GoogleFonts.alegreyaSans(
                        fontSize: 40,
                        fontWeight: FontWeight.bold,
                        color: Colors.black,
                      ),
                    ),
                  ],
                ),
                const SizedBox(height: 20),
                TextFormField(
                  controller: _emailController,
                  decoration: InputDecoration(
                    labelText: 'Email',
                    border: OutlineInputBorder(
                      borderRadius: BorderRadius.circular(12),
                    ),
                    prefixIcon:
                        const Icon(Icons.email, color: Colors.black),
                  ),
```

```dart
              validator: (value) {
                if (value == null ||
                    value.isEmpty ||
                    !value.contains('@')) {
                  return 'Enter a valid email';
                }
                return null;
              },
            ),
            const SizedBox(height: 15),
            TextFormField(
              controller: _passwordController,
              obscureText: true,
              decoration: InputDecoration(
                labelText: 'Password',
                border: OutlineInputBorder(
                  borderRadius: BorderRadius.circular(12),
                ),
                prefixIcon:
                    const Icon(Icons.lock, color: Colors.black),
              ),
              validator: (value) {
                if (value == null || value.length < 6) {
                  return 'Password must be at least 6 characters';
                }
                return null;
              },
            ),
            const SizedBox(height: 20),
            ElevatedButton(
              style: ElevatedButton.styleFrom(
                padding: const EdgeInsets.symmetric(vertical: 14),
                shape: RoundedRectangleBorder(
                  borderRadius: BorderRadius.circular(12),
                ),
                backgroundColor: Colors.black,
              ),
              onPressed: _signInWithEmailAndPassword,
              child: const Text(
                'Login',
                style: TextStyle(fontSize: 20, color: Colors.white),
              ),
            ),
            const SizedBox(height: 15),
            ElevatedButton(
              style: ElevatedButton.styleFrom(
                backgroundColor: Colors.white,
```

```dart
                    padding: const EdgeInsets.symmetric(vertical: 14),
                    shape: RoundedRectangleBorder(
                      borderRadius: BorderRadius.circular(12),
                      side: const BorderSide(color: Colors.black),
                    ),
                  ),
                  onPressed: _signInWithGoogle,
                  child: const Text(
                    'Sign in with Google',
                    style: TextStyle(fontSize: 20, color: Colors.black),
                  ),
                ),
              ],
            ),
          ),
        ),
      ),
    ),
  );
  }
}
```


Cloud Firestore — Add database — Ask Gemini how to get started with Firestore

Data | Rules | Indexes | Disaster recovery (NEW) | Usage | Extensions

Protect your Cloud Firestore resources from abuse, such as billing fraud or phishing   Configure App Check   ✕

Panel view   Query builder   ⋮

⌂ > Cars > 5i296ic2qZEuCg   ☁ More in Google Cloud ∨

📶 (default)	📁 Cars	📄 5i296ic2qZEuCgvNSAcy
+ Start collection	+ Add document	+ Start collection
Cars >	**5i296ic2qZEuCgvNSAcy** >	+ Add field
	EpgfF1QBJJ0iSU0rSnq0	engine: "2.2L"
	Ik57aNmuX5phxxmI18ZS	fuel: "Diesel"
	K2ROKDNGs2VAF5tBa8qC	horsepower: "200 HP"
	MolnicipBWupxqjmc8tz	mileage: "15 kmpl"
	NArBeFn7MsKGRJhCljqm	name: "Mahindra XUV700"
	NwNXQH4uK8e3JjQZ85oV	price: 1400000
	RUDCV3QO4RklVkiFgX66	seating: "7"

Start collection

Cars

Cars

Add document

EpgfF1QBJJ0iSU0rSnq0
Ik57aNmuX5phxxmI18ZS
K2ROKDNGs2VAF5tBa8qC
MolnicipBWupxqjmc8tz
NArBeFn7MsKGRJhCljqm
NwNXQH4uK8e3JjQZ85oV
RUDCV3QO4RklVkiFgX66
UVAqJTiUDlymlclvuget
UdRHGFygTOfbuMsNQ4Vm
ZNRwgCJ0Ux8rH8MZfDXF
jEKe7pKlEVHimK045b1d
lPuS63gxa6PxwQis2kJo
oy9KBxrfD5nr6zwBbCS9
yenTSarfMYsMBWYtqRtx

EpgfF1QBJJ0iSU0rSnq0

Start collection

Add field

Engine: "2.8L Diesel"

Fuel Type: "Diesel"

Horsepower: "201 hp"

Mileage: "14 kmpl"

Name: "Toyota Fortuner"

Price: 3500000

Seating: "7"

Torque: "500 Nm"

Transmission: "Automatic"