MPL Experiment 6

**Name:** Omkar Gholap **Class:** D15A **Roll no:**36

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

**Theory: -**

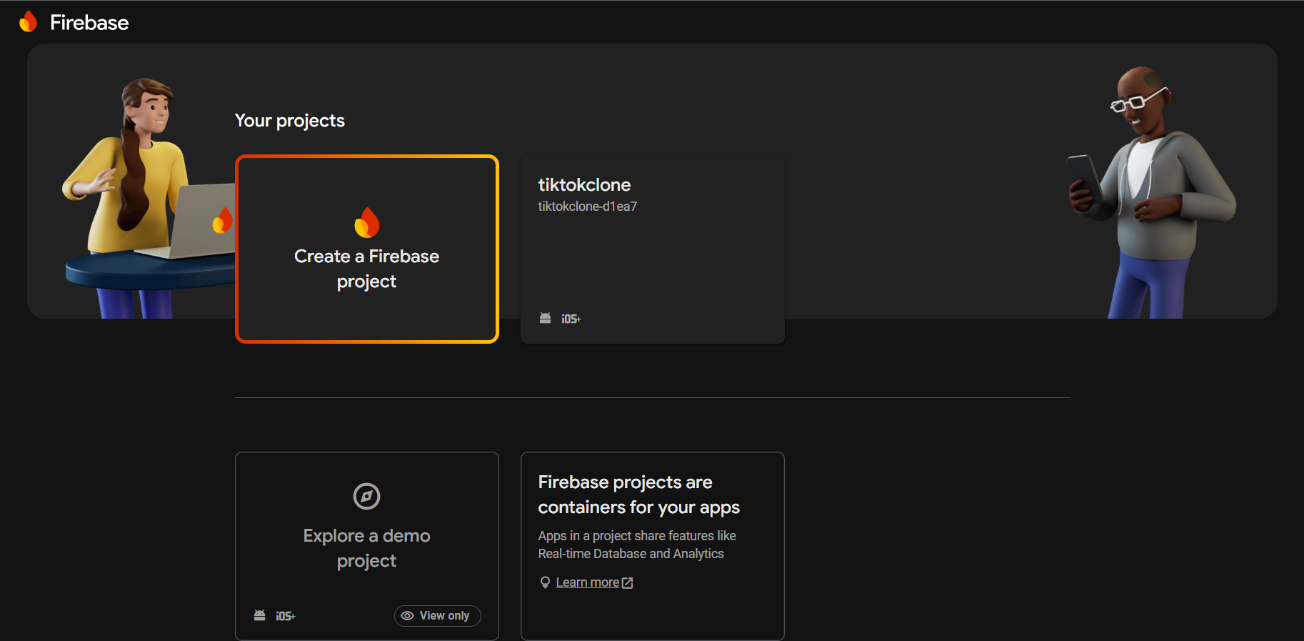
Flutter is an open-source UI toolkit developed by Google for building natively compiled applications for mobile, web, and desktop from a single codebase. Firebase, a Backend-as-a- Service (BaaS) platform, provides real-time database, authentication, and cloud storage services, making it a powerful backend 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.

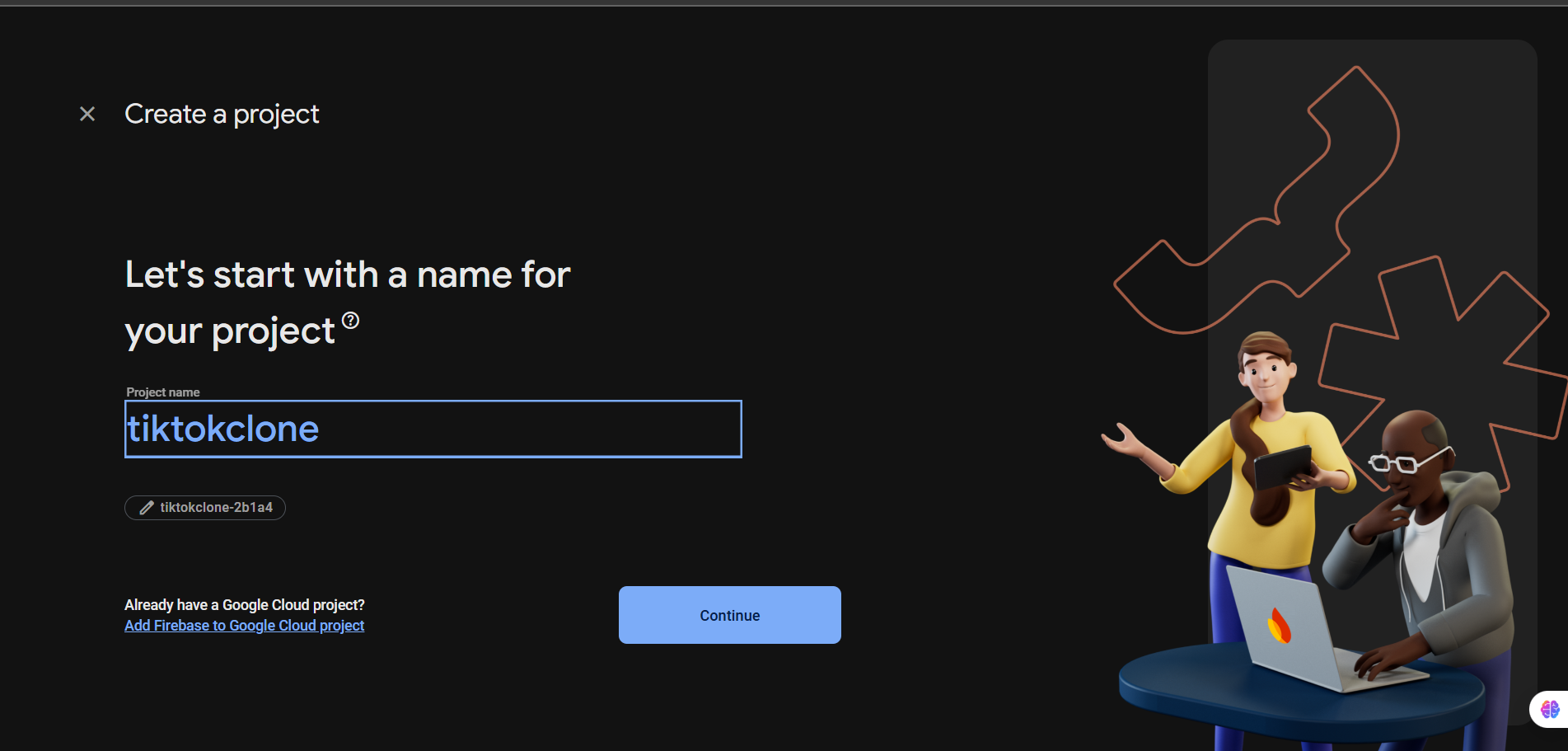
**Steps to Connect Flutter UI with Firebase Database:**

**Step 1:**

1.1) Go to Firebase Console and Create a Firebase Project

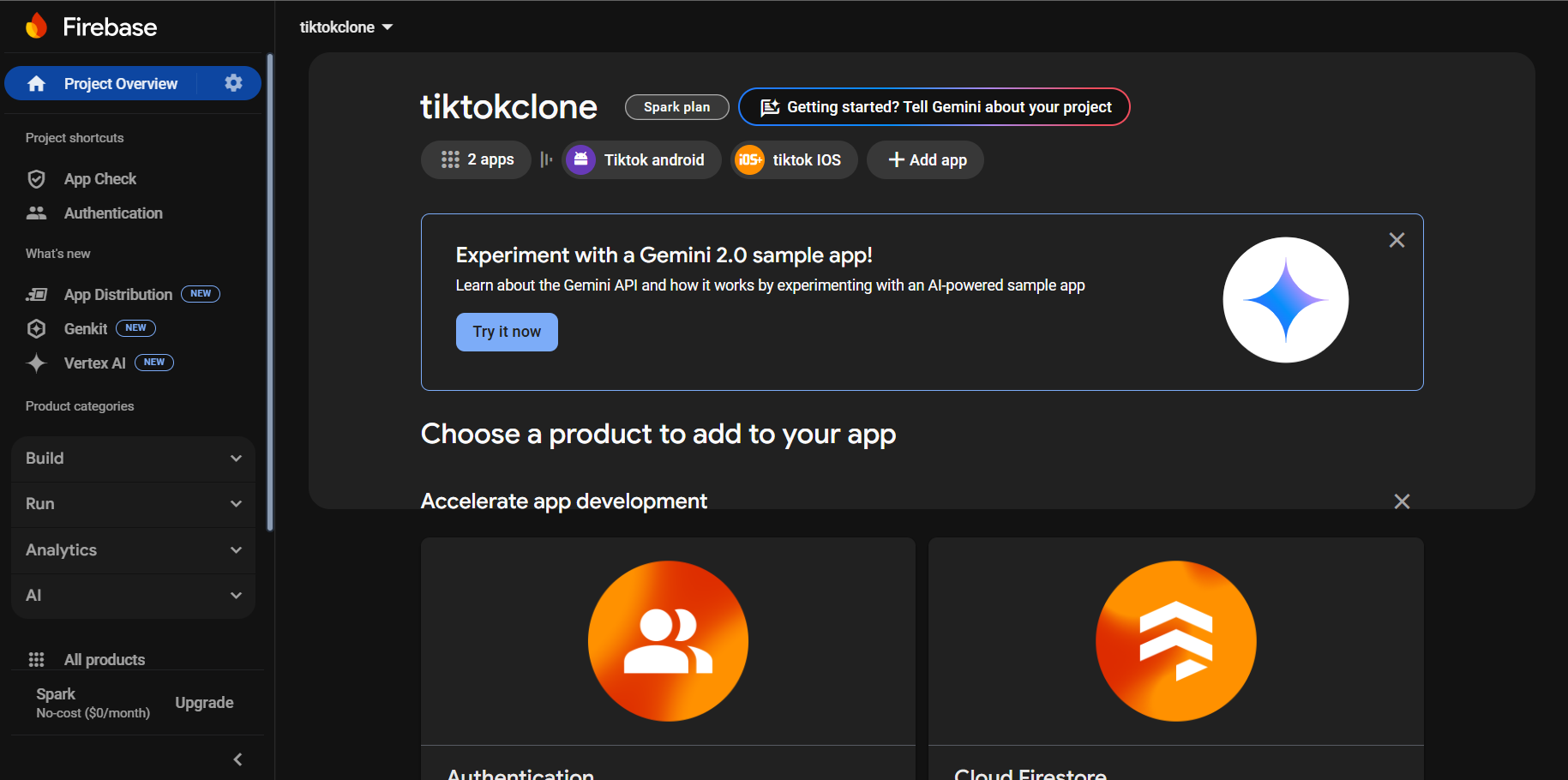


1.2) Click on Create a Project and give it a suitable name.



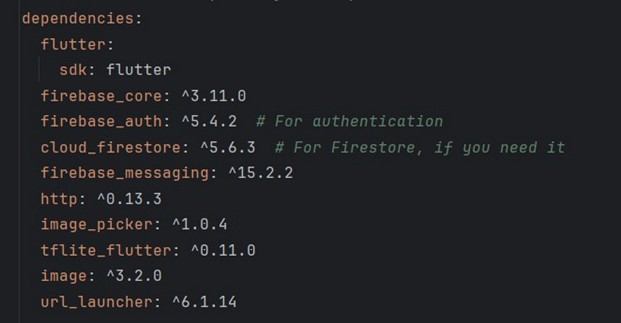
**Step 2:- Add Firebase to Your Flutter App**

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



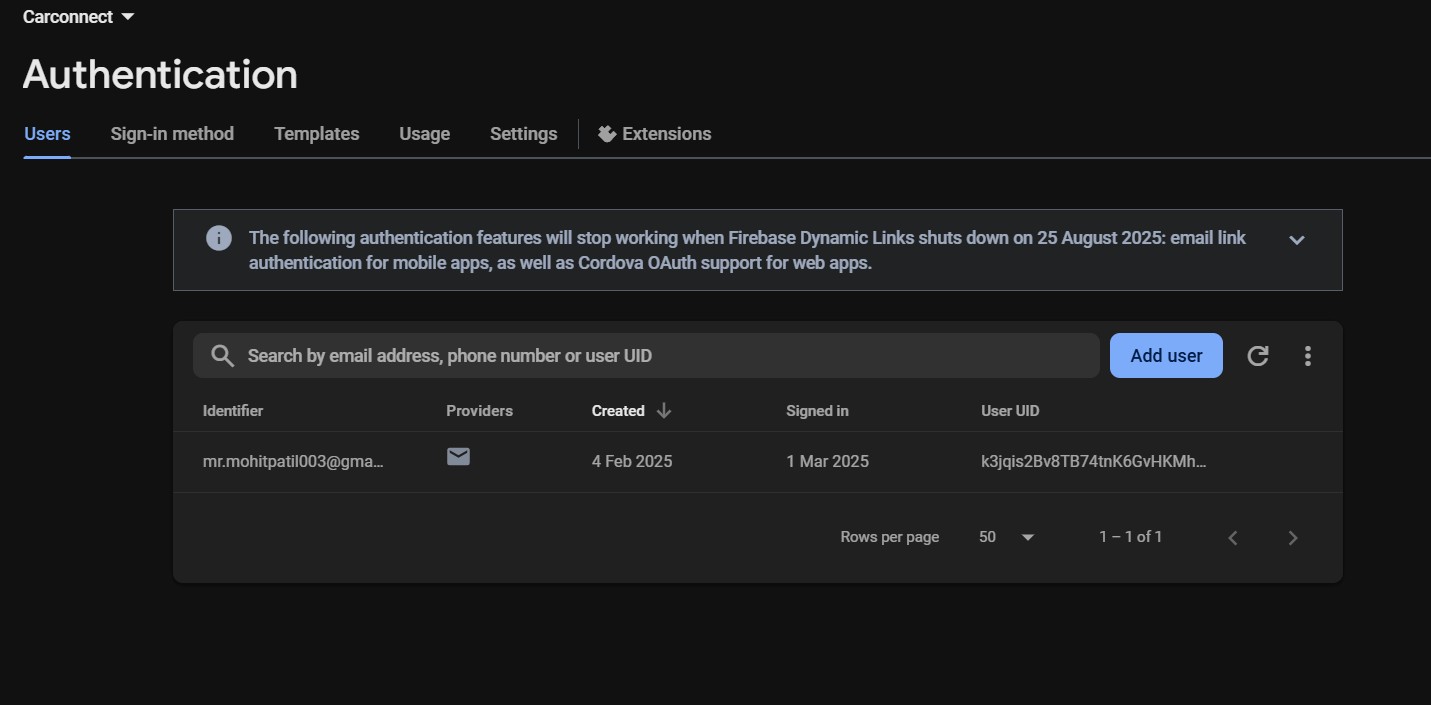
**Step 3: - Add Firebase Authentication to Your App**

3.1) Add Firebase Authentication Dependencies



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**

|  |
| --- |
| **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 {** |

|  |
| --- |
| **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(** |

|  |
| --- |
| **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),** |
| **),** |

|  |
| --- |
| **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,** |
| **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),** |
| **),** |
| **),** |
| **],** |
| **),** |
| **),** |
| **),** |
| **),** |
| **),** |
| **),** |
| **),** |
| **);** |
| **}** |
| **}** |
|  |

