EXPERIMENT NO: 6

NAME: SHRAVANI S RASAM
CLASS: D15A
ROLL NO: 45
TOPIC: BEHANCE:Behance is a platform for creative professionals to showcase portfolios, discover inspiration, and connect with others in the industry

AIM: To connect flutter UI with firebase database.

THEORY:

Introduction to Firebase and Flutter Integration
Firebase is a comprehensive platform developed by Google, designed to help developers build high-quality applications for both mobile and web. It provides essential services such as real-time databases, authentication, cloud storage, hosting, and much more. One of the most widely used Firebase services is the Firebase Realtime Database, which is a NoSQL cloud database that allows data to be stored and synced in real-time across all connected devices.

Flutter, on the other hand, is an open-source UI software development kit created by Google, which allows developers to build natively compiled applications for mobile, web, and desktop from a single codebase. Its rich set of pre-designed widgets and powerful tools makes Flutter an attractive option for developing visually appealing and performant applications. Integrating Firebase with Flutter allows developers to leverage the full potential of Firebase services in their applications.

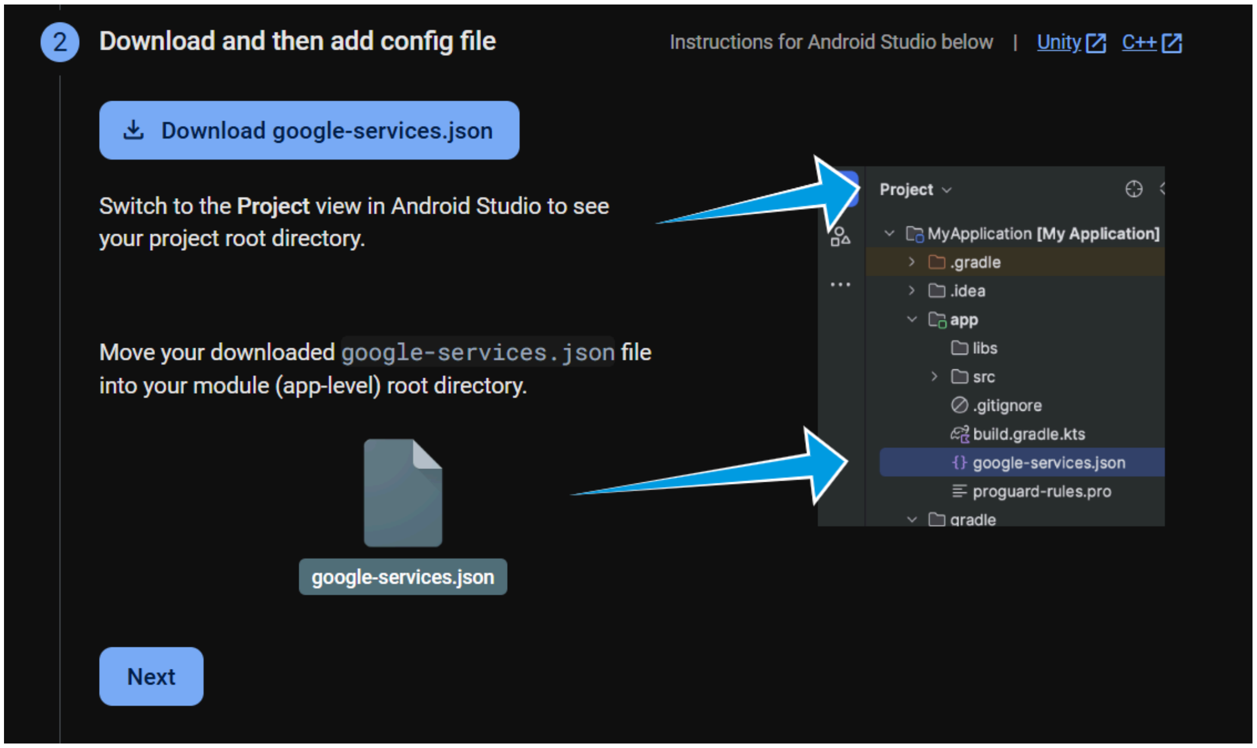
By using Firebase's Realtime Database, Flutter apps can achieve features such as real-time data synchronization, secure authentication, and cloud-based storage. This combination enables developers to create powerful, scalable, and feature-rich mobile and web applications.

Setting Up Firebase in Flutter :
To connect a Flutter app with Firebase, the following steps are typically followed:

1. Creating a Firebase Project: To start using Firebase with Flutter, the first step is to create a Firebase project in the Firebase Console. Once the project is created, developers can associate their Flutter app with the Firebase project by following the platform-specific instructions for Android or iOS. This usually involves configuring API keys, downloading configuration files, and adding them to the Flutter project.

2. Integrating Firebase SDK in Flutter: After the Firebase project is set up, developers need to integrate Firebase's SDK into the Flutter app. This involves adding the necessary dependencies to the Flutter project's pubspec.yaml file. For Firebase's Realtime Database, the package firebase_database is used. Additionally, Firebase's core SDK (firebase_core) must also be included to initialize Firebase services.

3. Initializing Firebase: Before any Firebase functionality can be used, it is essential to initialize Firebase in the Flutter app. This is done by calling Firebase.initializeApp() in the main entry point of the app (usually in the main.dart file). Firebase needs to be initialized before interacting with any Firebase services, such as the Realtime Database, Cloud Firestore, or Authentication. Connecting Firebase to a Flutter app enables developers to create robust, scalable, and real-time applications with ease. Firebase's Realtime Database offers a powerful, cloud-based solution for managing data in realtime, while Firebase Authentication ensures secure access control. By integrating Firebase with Flutter, developers can take advantage of real-time data synchronization, offline support, and a wide range of other Firebase features, allowing them to build feature-rich apps that meet modern user expectations.



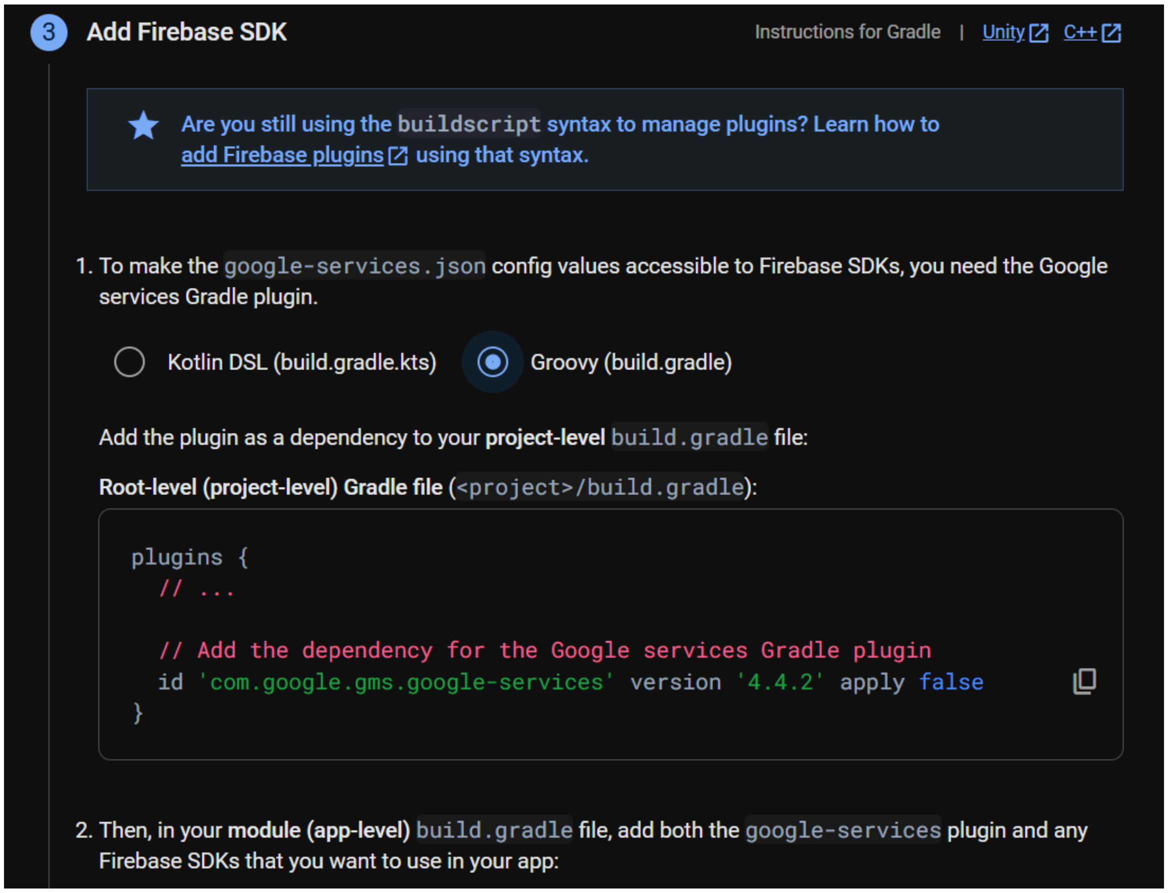
2 Download and then add config file

Instructions for Android Studio below | Unity C++

Download google-services.json

Switch to the **Project** view in Android Studio to see your project root directory.

Move your downloaded `google-services.json` file into your module (app-level) root directory.


3 Add Firebase SDK

Instructions for Gradle | Unity C++

Are you still using the `buildscript` syntax to manage plugins? Learn how to add Firebase plugins using that syntax.

1. To make the `google-services.json` config values accessible to Firebase SDKs, you need the Google services Gradle plugin.

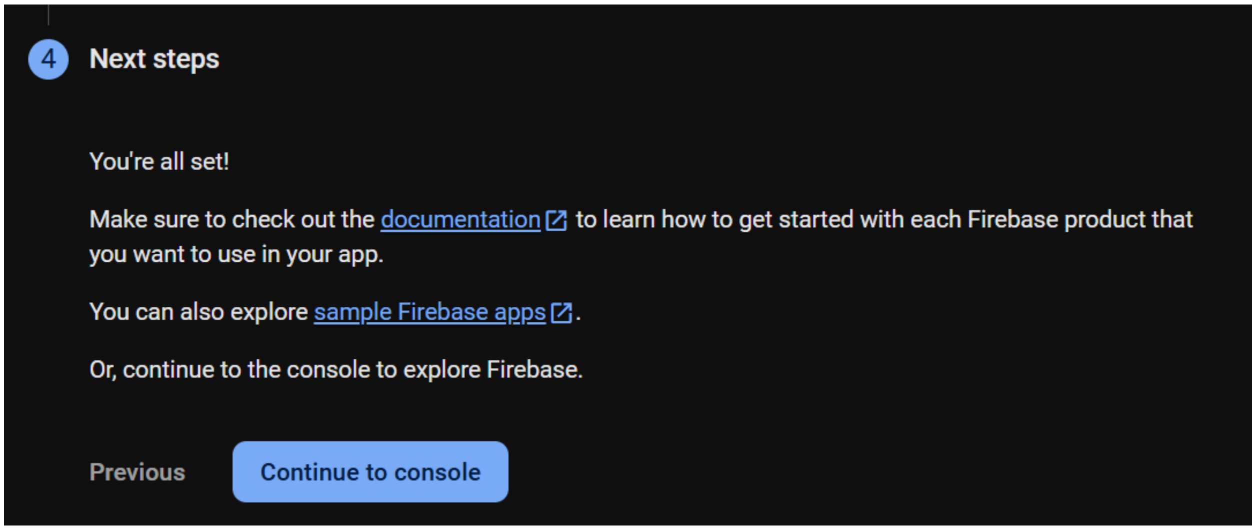
   Kotlin DSL (build.gradle.kts) [✓ Groovy (build.gradle)]

   Add the plugin as a dependency to your **project-level** `build.gradle` file:

   Root-level (project-level) Gradle file (`<project>/build.gradle`):

```
plugins {
  // ...

  // Add the dependency for the Google services Gradle plugin
  id 'com.google.gms.google-services' version '4.4.2' apply false
}
```

2. Then, in your **module (app-level)** `build.gradle` file, add both the `google-services` plugin and any Firebase SDKs that you want to use in your app:

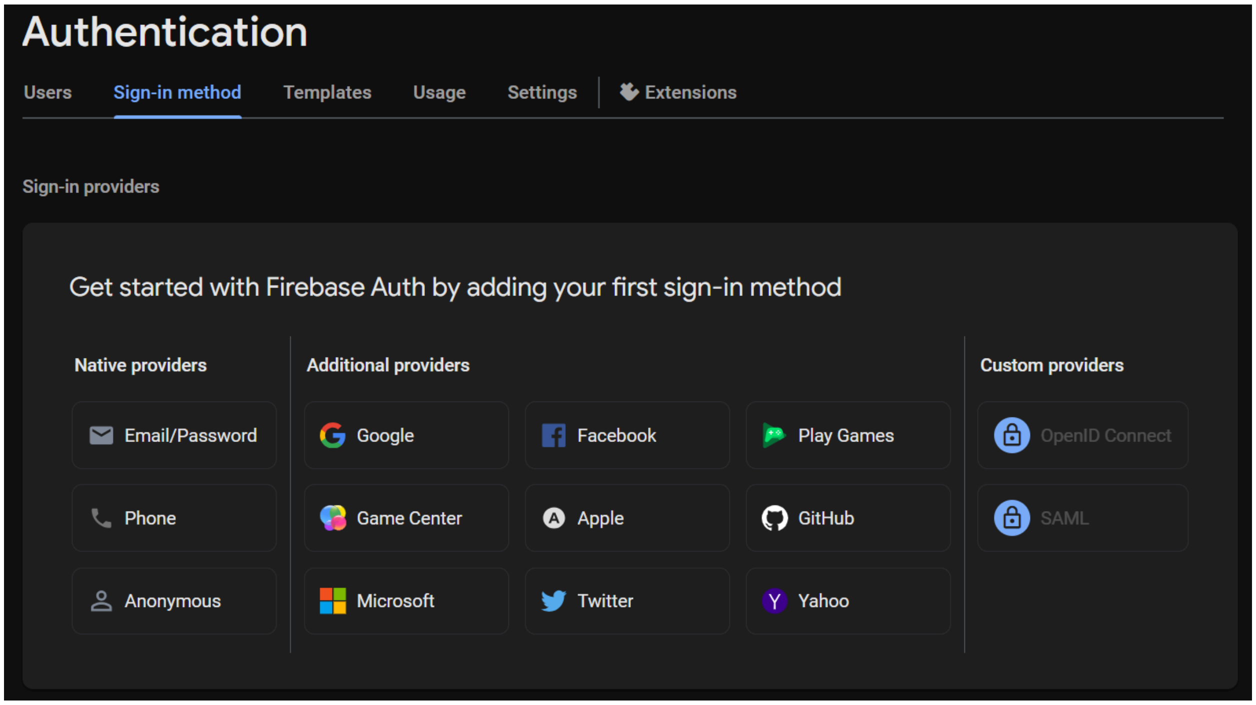

4 Next steps

You're all set!

Make sure to check out the documentation to learn how to get started with each Firebase product that you want to use in your app.

You can also explore sample Firebase apps.

Or, continue to the console to explore Firebase.



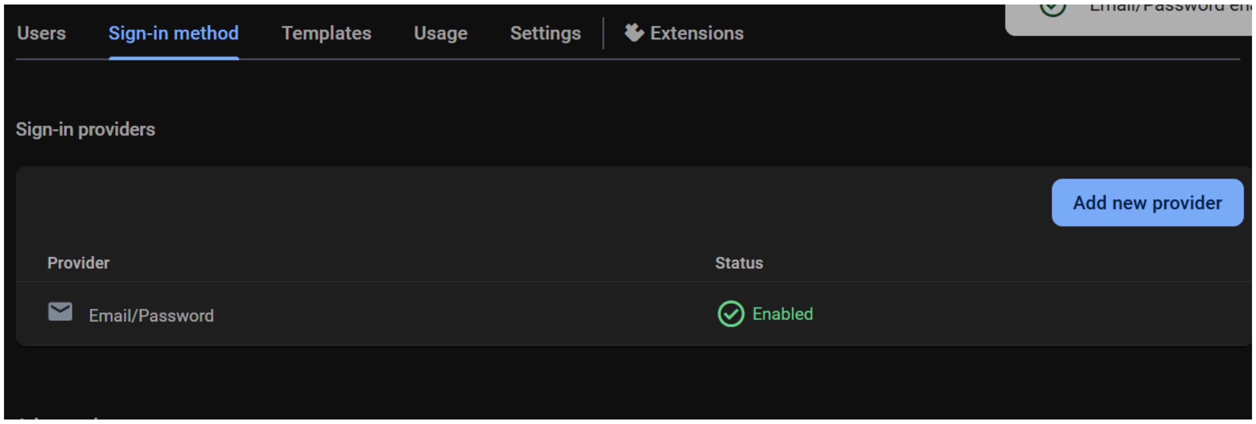
Authentication

Users | Sign-in method | Templates | Usage | Settings | Extensions

Sign-in providers

Get started with Firebase Auth by adding your first sign-in method

Native providers	Additional providers			Custom providers
Email/Password	Google	Facebook	Play Games	OpenID Connect
Phone	Game Center	Apple	GitHub	SAML
Anonymous	Microsoft	Twitter	Yahoo	



Users | Sign-in method | Templates | Usage | Settings | Extensions

Sign-in providers

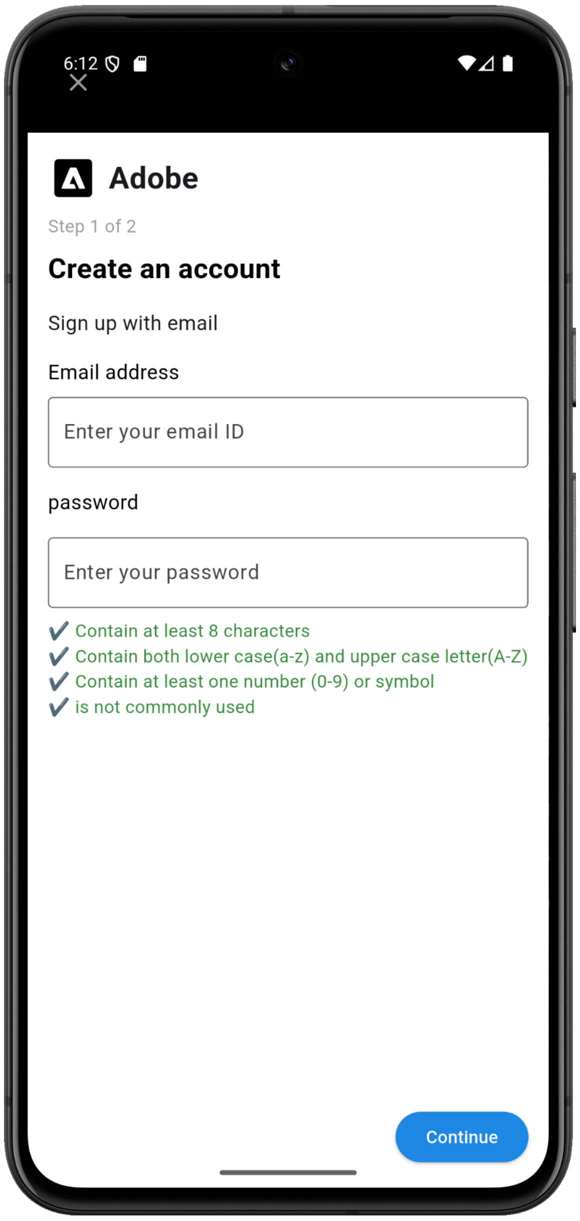
Add new provider

Provider	Status
Email/Password	Enabled

Login.dart

```
import
'package:behnace/Screens/Landing.dart';
import 'package:flutter/material.dart';
import 'package:provider/provider.dart';
import
'package:firebase_auth/firebase_auth.dart';
import '../Widgets/theme.dart';
import '../Widgets/auth_provider.dart';
import  '../Screens/Landing.dart';
// import '../Screens/Profile.dart';

class Login extends StatefulWidget {
  @override
  _LoginState createState() =>
_LoginState();
}

class _LoginState extends State<Login> {
  final TextEditingController _emailController
= TextEditingController();
  final TextEditingController
_passwordController =
TextEditingController();
  bool _isLoading = false;
  String? _errorMessage;

  Future<void> _signInWithEmailPassword()
async {
    setState(() {
      _isLoading = true;
      _errorMessage = null;
    });

    try {
      // Sign in the user
      UserCredential userCredential = await
FirebaseAuth.instance
          .signInWithEmailAndPassword(
        email: _emailController.text.trim(),
        password:
_passwordController.text.trim(),
      );

      // Extract user details correctly
      User? user = userCredential.user;

      if (user != null) {
        // Store email using Provider

Provider.of<AuthProviderFunction>(context,
listen: false).setEmail(
            user.email!);

        // Navigate if login is successful
        Navigator.pushReplacement(
          context,
          MaterialPageRoute(builder: (context)
=> Landing()),
        );
      } else {
        setState(() {
          _errorMessage = "User not found.
Please check your credentials.";
        });
      }
    } catch (e) {
      setState(() {
        _errorMessage = e.toString(); //
Display the actual Firebase error
      });
    } finally {
      setState(() {
        _isLoading = false;
      });
    }
  }



    @override
    Widget build(BuildContext context) {
      return Scaffold(
        backgroundColor: Colors.white,
        body: Column(
          crossAxisAlignment:
CrossAxisAlignment.start,
        children: [
          Container(
            color: AppColors.background,
            padding: EdgeInsets.all(16),
            child: Row(
              children: [
                IconButton(
                  icon: Icon(Icons.close, color:
AppColors.hint),
                  onPressed: () =>
Navigator.pop(context),
                ),
              ],
            ),
          ),
          Padding(
            padding: const
EdgeInsets.all(16.0),
            child: Column(
              crossAxisAlignment:
CrossAxisAlignment.start,
              children: [
                Row(
                  children: [
                    Icon(Icons.adobe, size: 40,
color: Colors.black),
                    SizedBox(width: 8),
                    Text(
                      'Adobe',
                      style: TextStyle(
                          fontSize: 24, fontWeight:
FontWeight.bold),
                    ),
                  ],
                ),
                SizedBox(height: 8),
                Text('Step 1 of 2',
                    style: TextStyle(color:
AppColors.hint, fontSize: 14)),
                SizedBox(height: 8),
                Text(
                  'Create an account',
                  style:
                  TextStyle(fontSize: 22,
                      fontWeight: FontWeight.bold,
                      color: Colors.black),
                ),
                SizedBox(height: 16),
                Text('Sign up with email', style:
TextStyle(fontSize: 16)),
                SizedBox(height: 16),
                Text('Email address',
                    style: TextStyle(fontSize: 16,
color: Colors.black)),
                SizedBox(height: 8),
                TextField(
                  controller: _emailController,
                  keyboardType:
TextInputType.emailAddress,
                  decoration: InputDecoration(
                    border: OutlineInputBorder(
                        borderSide:
BorderSide(color: AppColors.hint)),
                    hintText: 'Enter your email ID',
                  ),
                ),
                SizedBox(height: 16),
                Text('Password',
                    style: TextStyle(fontSize: 16,
color: Colors.black)),
                SizedBox(height: 8),
                TextField(
                  controller: _passwordController,
                  obscureText: true,
                  decoration: InputDecoration(
                    border: OutlineInputBorder(
                        borderSide:
BorderSide(color: AppColors.hint)),
                    hintText: 'Enter your
password',
                  ),
                ),
                SizedBox(height: 8),
                if (_errorMessage != null)
                  Text("Invalid email or
password",
                      style: TextStyle(color:
Colors.red, fontSize: 14)),
                  SizedBox(height: 16),
                  Text('✔ Contain at least 8
characters',
                      style: TextStyle(color:
Colors.green[700], fontSize: 14)),
                  Text('✔ Contain both lower and
upper case letters',
                      style: TextStyle(color:
Colors.green[700], fontSize: 14)),
                  Text('✔ Contain at least one
number (0-9) or symbol',
                      style: TextStyle(color:
Colors.green[700], fontSize: 14)),
                  Text('✔ Is not commonly used',
                      style: TextStyle(color:
Colors.green[700], fontSize: 14)),
                ],
              ),
            ),
            Spacer(),
            Padding(
              padding: const
EdgeInsets.all(16.0),
              child: Align(
                alignment:
Alignment.bottomRight,
                child: ElevatedButton(
                  style: ElevatedButton.styleFrom(
                    backgroundColor:
AppColors.primary,
                    foregroundColor: Colors.white,
                  ),
                  onPressed: _isLoading ? null :
_signInWithEmailPassword,
                  child: _isLoading
                      ?
CircularProgressIndicator(color:
Colors.white)
                      : Text('Continue'),
                ),
              ),
            ),
          ],
        ),
```

```
      );
    }
  }
```


6:12

Adobe

Step 1 of 2

Create an account

Sign up with email

Email address

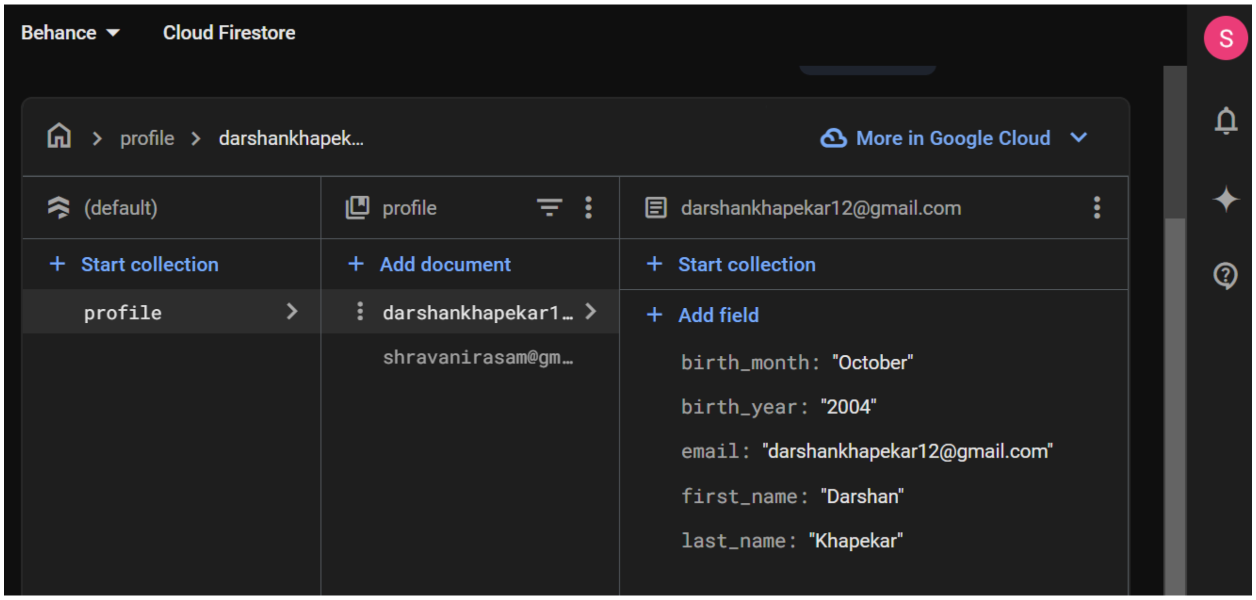
Enter your email ID

password

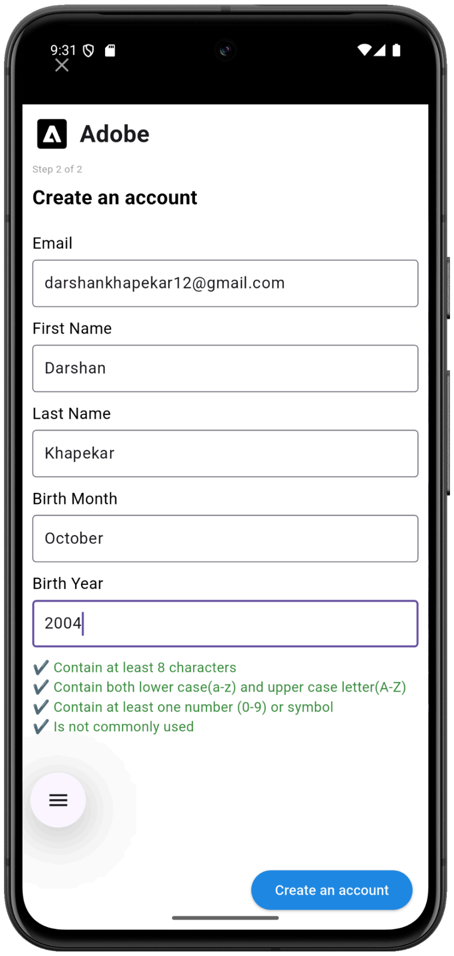
Enter your password

✔ Contain at least 8 characters
✔ Contain both lower case(a-z) and upper case letter(A-Z)
✔ Contain at least one number (0-9) or symbol
✔ is not commonly used

Continue

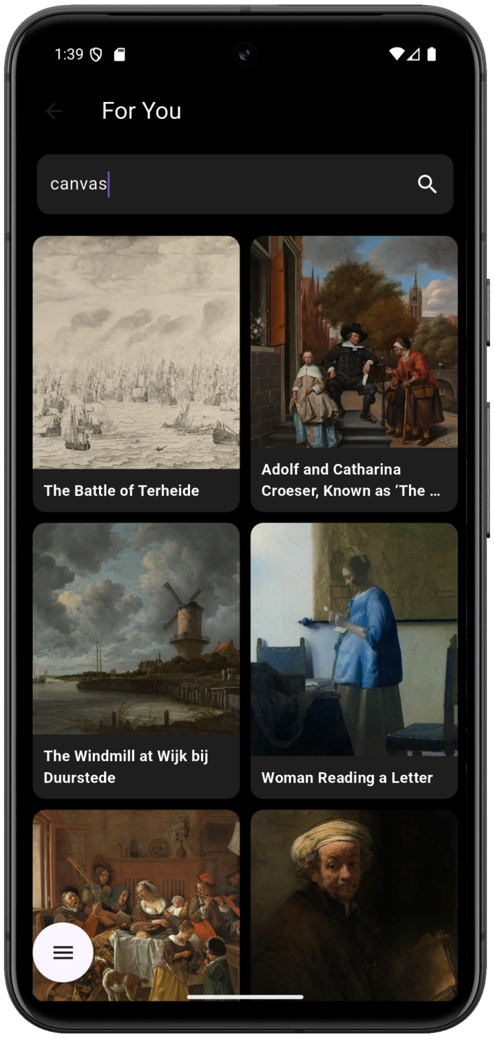


```
birth_month: "October"
birth_year: "2004"
email: "darshankhapekar12@gmail.com"
first_name: "Darshan"
last_name: "Khapekar"
```



Email	darshankhapekar12@gmail.com
First Name	Darshan
Last Name	Khapekar
Birth Month	October
Birth Year	2004

✔ Contain at least 8 characters
✔ Contain both lower case(a-z) and upper case letter(A-Z)
✔ Contain at least one number (0-9) or symbol
✔ Is not commonly used


1:39
For You
canvas
The Battle of Terheide
Adolf and Catharina Croeser, Known as 'The ...
The Windmill at Wijk bij Duurstede
Woman Reading a Letter