

MPL EXPERIMENT-6

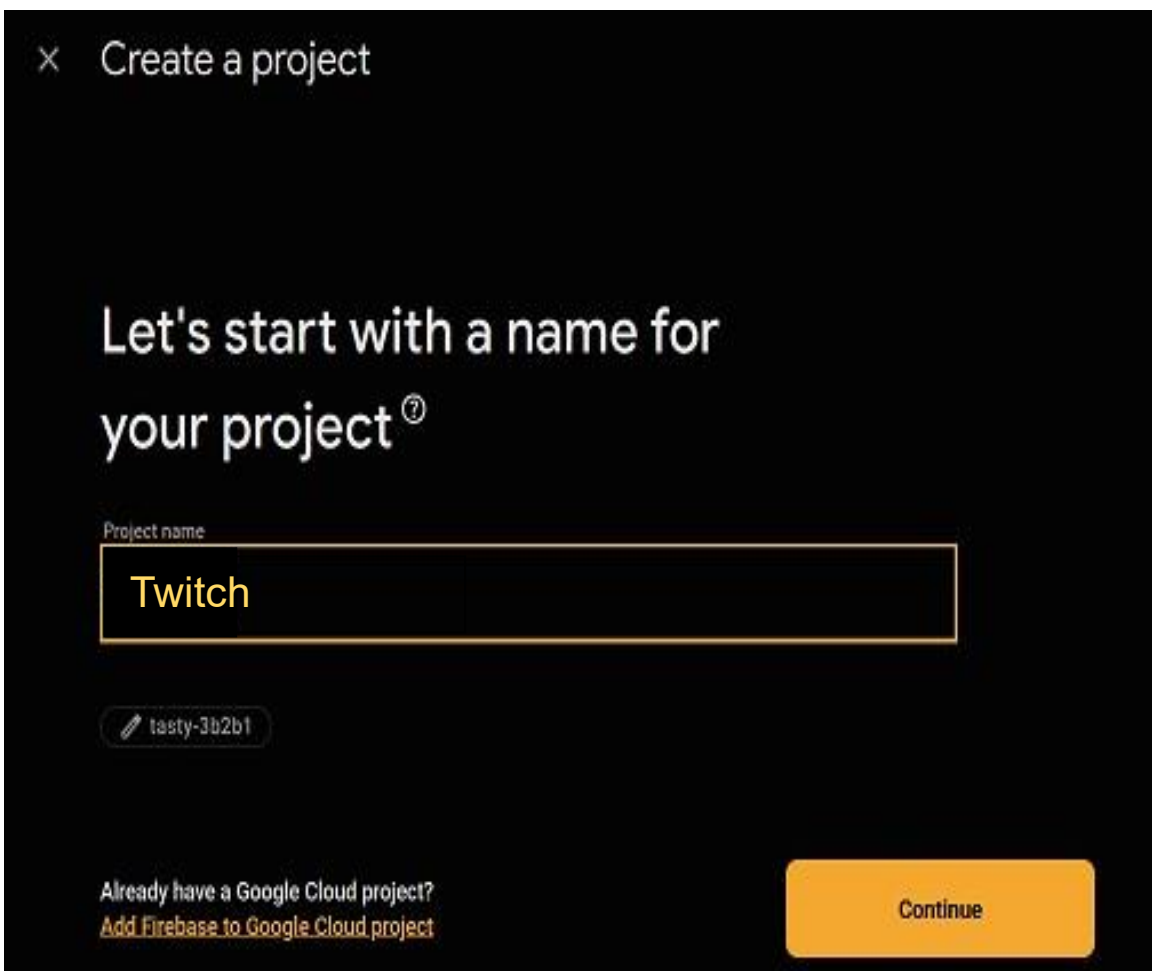
Prajwal Pandey

D15A / 32

Aim: To Connect Flutter UI with Firebase Database.

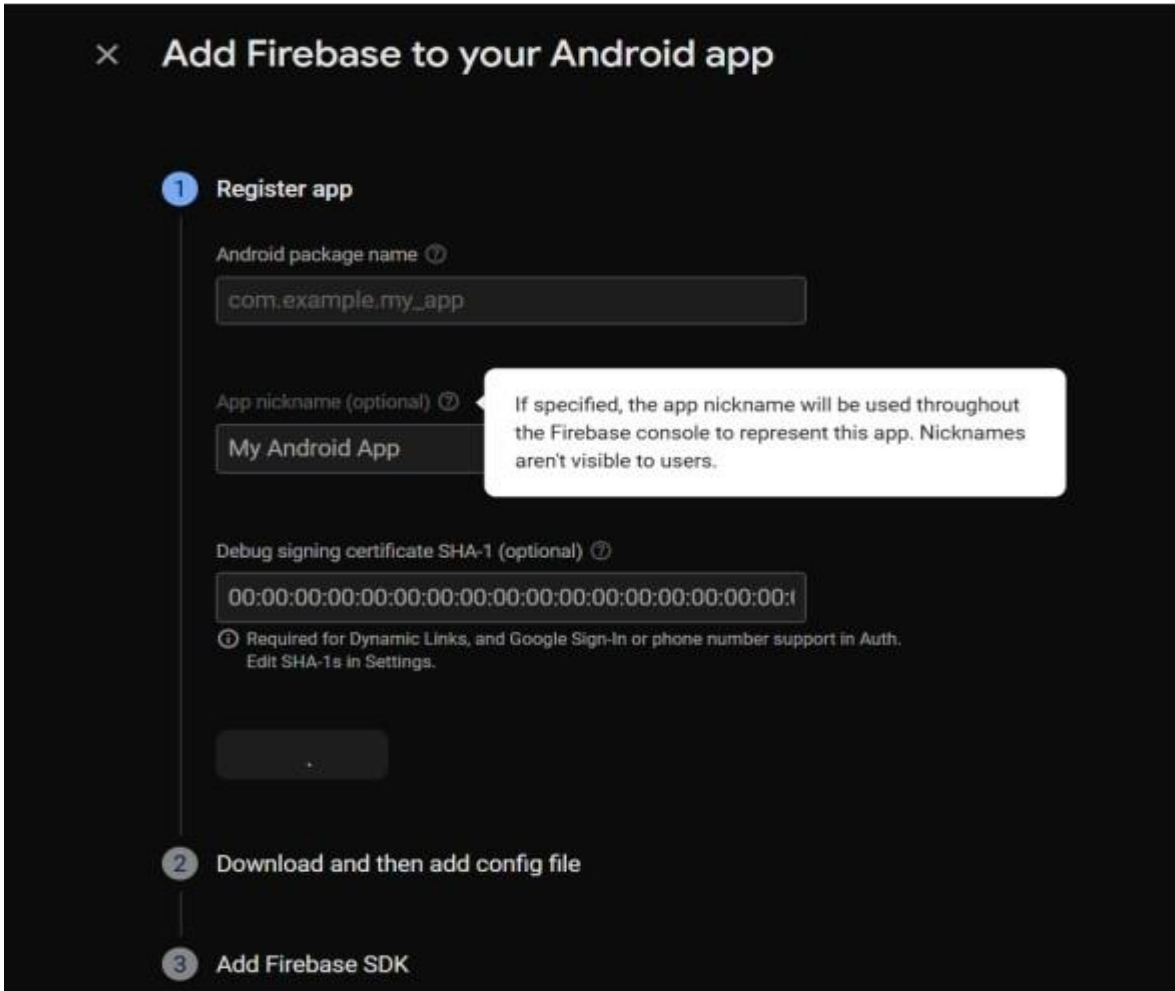
Step-1:

First, log in with your Google account to manage your Firebase projects. From within the Firebase dashboard, select the Create new project button and give it a name



The screenshot shows a dark-themed dialog box titled "Create a project" with a close button (X) in the top left corner. The main text reads "Let's start with a name for your project" followed by a question mark icon. Below this is a text input field labeled "Project name" containing the word "Twitch". Underneath the input field is a button with a pencil icon and the text "tasty-3b2b1". At the bottom left, there is a link that says "Already have a Google Cloud project? Add Firebase to Google Cloud project". At the bottom right is a large orange button labeled "Continue".

In order to add Android support to our Flutter application, select the Android logo from the dashboard. This brings us to the following screen:



The most important thing here is to match up the Android package name that you choose here with the one inside of our application.

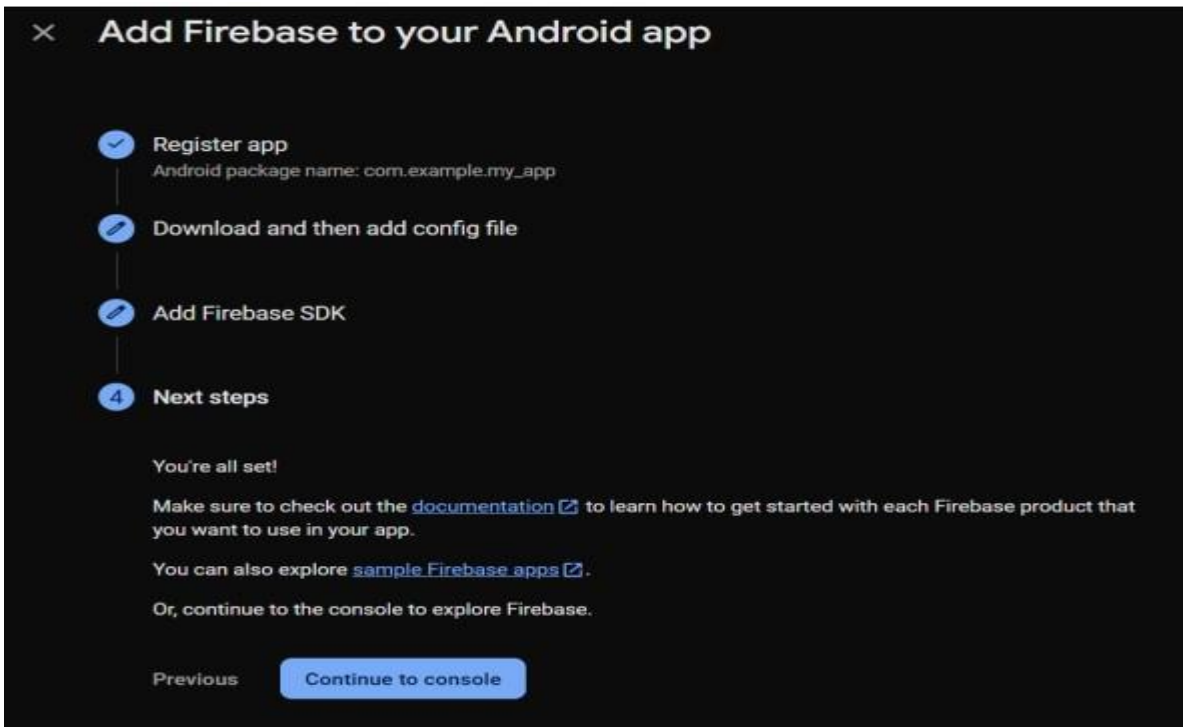
Downloading the Config File

The next step is to add the Firebase configuration file into our Flutter project. This is important as it contains the API keys and other critical information for Firebase to use.

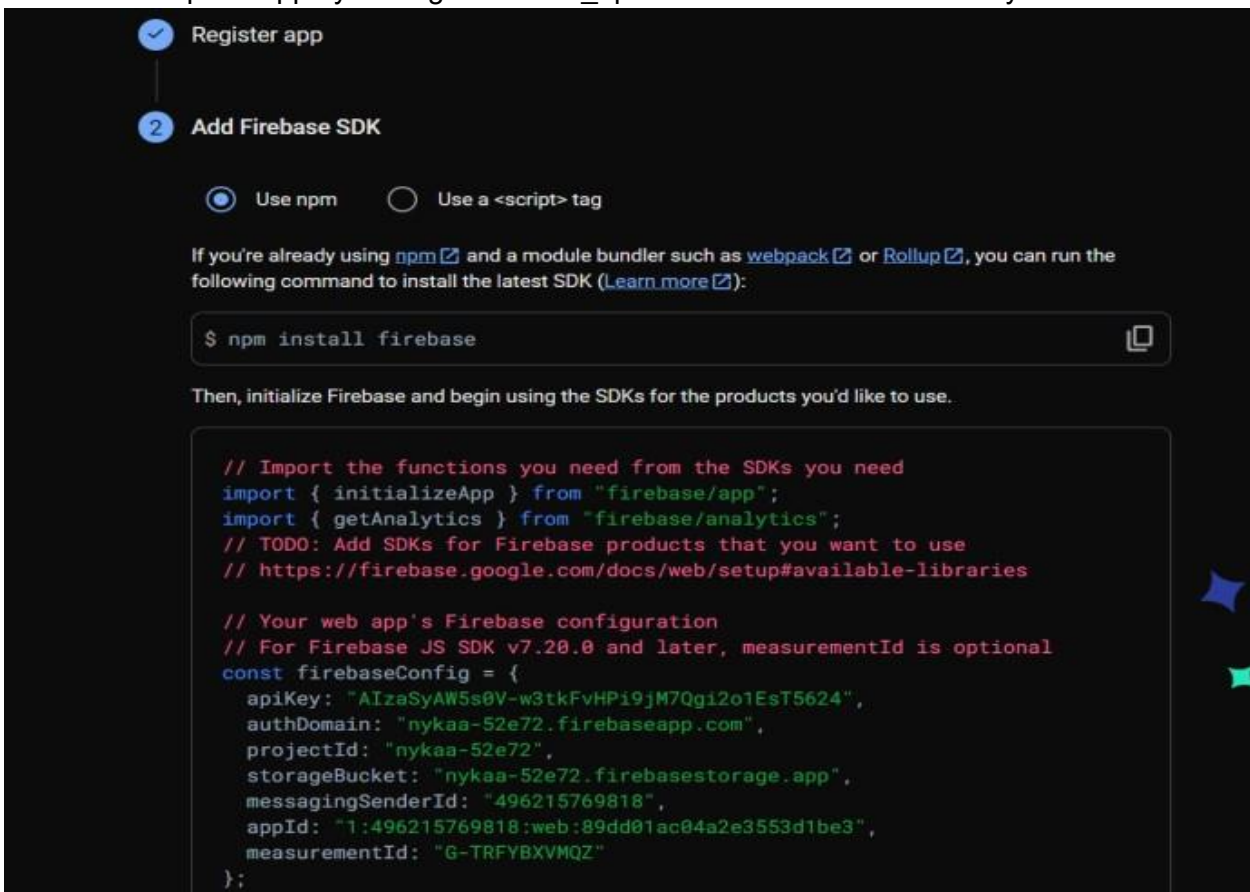
Select Download **google-services.json** from this page

```
In android/build.gradle adddependencies { classpath
'com.google.gms.google-services:4.4.2'
}
```

```
In app/build.gradle adddependencies{ implementation
platform('com.google.firebase:firebase-bom:33.9.0')
}
```



Now let us setup the app by adding a `firebase_option.dart` file in the lib directory similar to :



In pubspec.yaml:

```
dependencies: flutter:
sdk: flutter
# The following adds the Cupertino Icons font to your application.
# Use with the CupertinoIcons class for iOS style icons.
cupertino_icons: ^1.0.8
firebase_core: ^3.11.0 firebase_auth:
^5.4.2 cloud_firestore: ^5.6.3
firebase_storage: ^12.4.2
```

Firestore connection Code:

(options is imported from the file: firebase_options.dart)

```
void main() async {
  WidgetsFlutterBinding.ensureInitialized(); await
  Firebase.initializeApp( options:
  DefaultFirebaseOptions.currentPlatform,
  );
  CloudinaryService.init();
  runApp( MultiProvider(
  providers: [
    ChangeNotifierProvider(create: (context) => CartProvider()),
  ],
  child: MyApp(),
  ),
  );
}
```

Authentication code:

```
import 'package:firebase_auth/firebase_auth.dart'; import
'package:cloud_firestore/cloud_firestore.dart';

class AuthService { final FirebaseAuth _auth =
FirebaseAuth.instance;
final FirebaseFirestore _firestore = FirebaseFirestore.instance;

Future<User?> registerWithEmailAndPassword(String email, String password, String
username) async {
  try {
    UserCredential credential = await _auth.createUserWithEmailAndPassword(
email: email, password: password,
```

```

    );
    User? user = credential.user;
    if (user != null) {      await
_firestore.collection('users').doc(user.uid).set({
        'email': email,
        'username': username, // Add username field
        'likedRecipes': [],
        'savedRecipes': [],
    });
    }

    return user;  }
catch (e) {
    print(e.toString());
    return null;
}
}

Future<User?> signInWithEmailAndPassword(String email, String password) async {
try {
    UserCredential credential = await _auth.signInWithEmailAndPassword(
email: email,      password: password,
    );
    User? user = credential.user;
    if (user != null) {      final userDoc = await
_firestore.collection('users').doc(user.uid).get();      if (!userDoc.exists) {
await _Firestore.collection('users').doc(user.uid).set({
        'email': email,
        'likedRecipes': [],
        'savedRecipes': [],
    });
    }
    }
    return user;  }
catch (e) {
    print(e.toString());
    return null;
}
}

Stream<User?> authStateChanges() { return
FirebaseAuth.instance.authStateChanges();
}

```

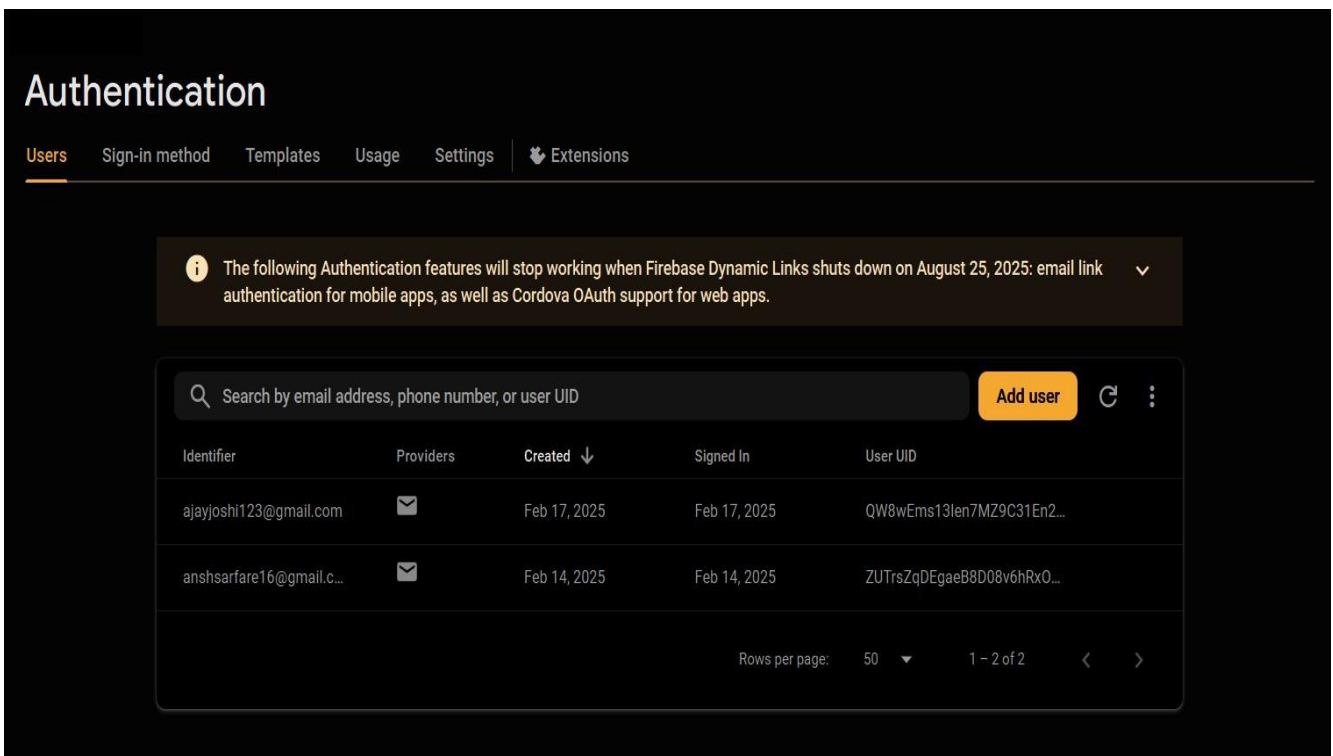
```

// Sign out
Future<void> signOut() async {
await _auth.signOut();
}



// Get the currently logged-in user
User? getCurrentUser() { return
_auth.currentUser;
}
}

```

Authentication:



The screenshot shows the Firebase Authentication console. At the top, there's a navigation bar with tabs: Users (selected), Sign-in method, Templates, Usage, Settings, and Extensions. Below the navigation bar, there's a warning message: "The following Authentication features will stop working when Firebase Dynamic Links shuts down on August 25, 2025: email link authentication for mobile apps, as well as Cordova OAuth support for web apps." Below the warning, there's a search bar with the text "Search by email address, phone number, or user UID" and an "Add user" button. Below the search bar, there's a table with the following columns: Identifier, Providers, Created, Signed In, and User UID. The table contains two rows of user data.

Identifier	Providers	Created ↓	Signed In	User UID
ajayjoshi123@gmail.com		Feb 17, 2025	Feb 17, 2025	QW8wEms13len7MZ9C31En2...
anshsarfare16@gmail.c...		Feb 14, 2025	Feb 14, 2025	ZUTrsZqDEgaeB8D08v6hRxO...

At the bottom right of the table, there's a pagination control showing "Rows per page: 50" and "1 - 2 of 2".

Firestore Database:

Cloud Firestore

Add database

Ask Gemini how to get started with Firestore

Data Rules Indexes Disaster Recovery Usage Extensions

Protect your Cloud Firestore resources from abuse, such as billing fraud or phishing

Configure App Check

Panel view Query builder

users > ZUTrsZqDEgaeB

More in Google Cloud

(default)	users	ZUTrsZqDEgaeB8D08v6hRx0Z0Tf1
<div>+ Start collection</div> <div>addresses</div> <div>cart</div> <div>communityPosts</div> <div>orders</div> <div>users</div>	<div>+ Add document</div> <div>7bzCfhXx2UFS1PjJpPm6</div> <div>QW8wEms13Ien7MZ9C31En2uiH0f2</div> <div>ZUTrsZqDEgaeB8D08v6hRx0Z0Tf1</div>	<div>+ Start collection</div> <div>+ Add field</div> <div>email: "anshsarfare16@gmail.com"</div> <div>likedRecipes</div> <div>0 "52807"</div> <div>1 "52951"</div> <div>savedRecipes</div> <div>0 "52807"</div> <div>1 "52928"</div> <div>username: "Ansh"</div>