

## Experiment-6

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

### Theory:

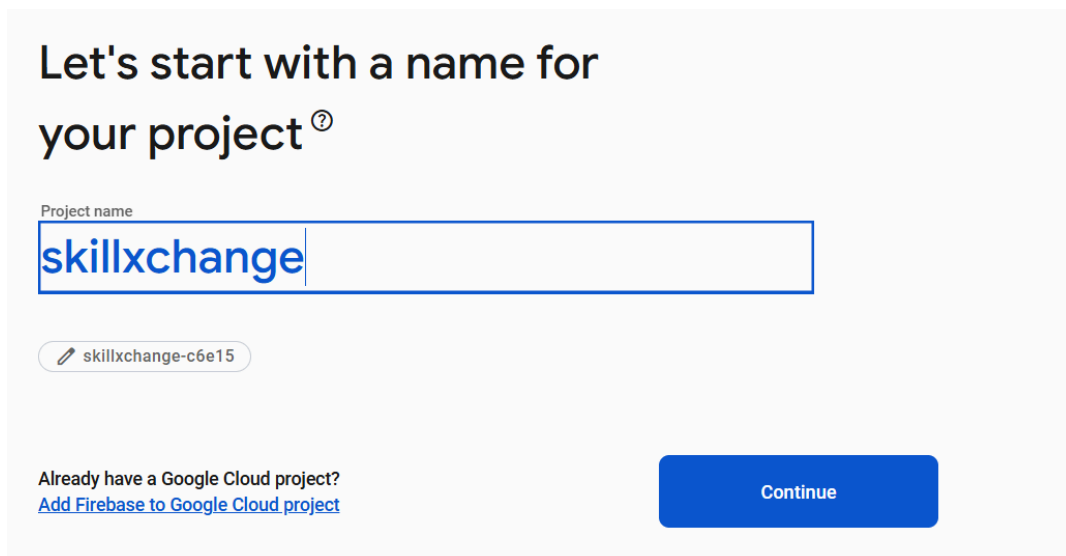
Firebase is a great backend solution for anyone that wants to use authentication, databases, cloud functions, ads, and countless other features within an app. In Flutter, Firebase integration is achieved using the `firebase_core` and `cloud_firestore` (or `firebase_database`) packages. After initializing Firebase in the app, data can be added, read, updated, and deleted directly through Firebase methods. Flutter widgets can be connected to database streams, ensuring that any changes in the database are instantly reflected in the UI.

This integration enhances app functionality by enabling persistent data storage, user-specific content, and real-time updates, making it essential for building modern mobile applications.

### Steps:

#### 1) Create Firebase Project


- Go to Firebase Console
- Click "Add Project"
- Give your project a name, accept terms, and click Continue



Let's start with a name for  
your project <sup>?</sup>

Project name

skillxchange

 skillxchange-c6e15

Already have a Google Cloud project?  
[Add Firebase to Google Cloud project](#)

Continue

#### 2) Register your app in Firebase



Then, add firebase SDK in your flutter project.

**3 Add Firebase SDK**

Instructions for Gradle | [Unity](#) [C++](#)

★ Are you still using the `buildscript` syntax to manage plug-ins? Learn how to [add Firebase plug-ins](#) using that syntax.

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

☒ Kotlin DSL (`build.gradle.kts`) ☐ Groovy (`build.gradle`)

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

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

```
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.kts` file, add both the `google-services` plug-in and any Firebase SDKs that you want to use in your app:

**Module (app-level) Gradle file** (`<project>/<app-module>/build.gradle.kts`):

```
plugins {
    id("com.android.application")
    // Add the Google services Gradle plugin
    id("com.google.gms.google-services")
    ...
}

dependencies {
    // Import the Firebase BoM
    implementation(platform("com.google.firebase:firebase-bom:33.12.0"))

    // TODO: Add the dependencies for Firebase products you want to use
    // When using the BoM, don't specify versions in Firebase dependencies
    // https://firebase.google.com/docs/android/setup#available-libraries
}
```

By using the Firebase Android BoM, your app will always use compatible Firebase library versions. [Learn more](#)

3. After adding the plug-in and the desired SDKs, sync your Android project with the Gradle files.

[Previous](#) [Next](#)

### 3) Initialize Firebase in Flutter

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

- 4) For running as web app, we need to initialize flutter app with dependencies in the `Firebase.initializeApp()` since, it does not fetches dependencies from `google-services.json` file,

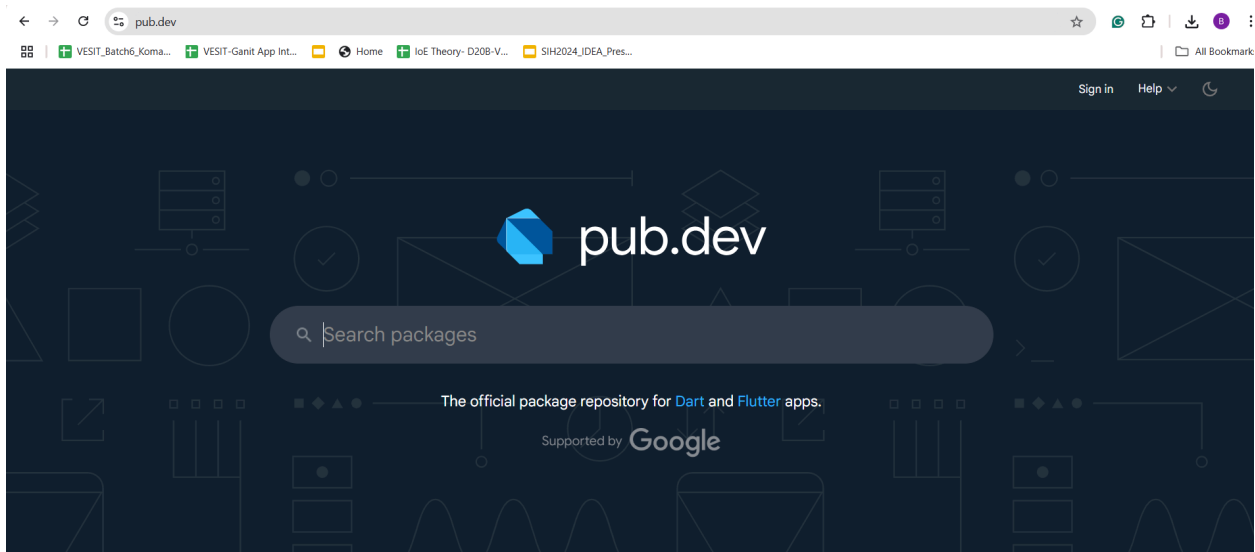
```
Run | Debug | Profile
void main() async {
  WidgetsFlutterBinding.ensureInitialized();
  await Firebase.initializeApp(
    options: const FirebaseOptions(
      apiKey: "AIzaSyBJbTuuEaIn75nHibMH1l6TsMFfcAI1vI",
      authDomain: "skillxchange-2381a.firebaseio.com",
      projectId: "skillxchange-2381a",
      storageBucket: "skillxchange-2381a.firebaseio.com",
      messagingSenderId: "415393619969",
      appId: "1:415393619969:web:fc45b978d48170d7e1e000",
      measurementId: "G-YVD3LM7M6L")
  );
  runApp(MyApp());
}
```

- 5) Add Firebase packages to Flutter

```
flutter pub add firebase_core
```

```
flutter pub add firebase_auth # or other services like cloud_firestore
```

- a) Go to `pub.dev`
- b) Search for `firebase_core` and `firebase_auth`
- c) Download and configure in project



## firebase\_auth 5.5.2

Published 14 days ago • firebase.google.com Dart 3 compatible

SDK FLUTTER PLATFORM ANDROID IOS MACOS WEB WINDOWS

4.1K

Readme Changelog Example Installing Versions Scores

### Use this package as a library

Depend on it

Run this command:

With Flutter:

```
$ flutter pub add firebase_auth
```

## firebase\_core 3.13.0

Published 14 days ago • firebase.google.com Dart 3 compatible

SDK FLUTTER PLATFORM ANDROID IOS MACOS WEB WINDOWS

3.8K

Readme Changelog Example Installing Versions Scores

### Use this package as a library


Depend on it

Run this command:

With Flutter:

```
$ flutter pub add firebase_core
```


## Output:



### Sign up

Create an account for buying new shoes

Have an account? [Login](#)



### Login

Login into your account for buying new shoes

Forgot Password?

Don't have an account? [Create one](#)

Signup Successful!

<input type="text"/> Search by email address, phone number or user UID			Add user	↺	⋮
Identifier	Providers	Created ↓	Signed in	User UID	
brijeshrsharma145@g...	✉	14 Apr 2025	14 Apr 2025	ug0MkvHyTCfuvhqOp2CINJf8...	
2022.brijesh.sharma@v...	✉	13 Apr 2025	14 Apr 2025	5F4FC5MbCvejmtZW3AuUXH...	

Rows per page
50
1 – 2 of 2
<
>

**Conclusion:**

This experiment successfully demonstrates how to integrate Firebase services with a Flutter application. By implementing user signup, login, and data submission features, we explored the use of Firebase Authentication for secure user management and Cloud Firestore for real-time database storage. The flow between screens and backend interaction highlights the power and simplicity of using Firebase in Flutter apps, making it an ideal choice for building scalable and responsive mobile applications.