

**EXPERIMENT NO :- 06**

**AIM:** - To connect Flutter UI with Firebase Database

**Theory:** -

Firebase is a cloud-based platform by Google that provides backend services like authentication, real-time databases, and cloud storage. In Flutter, Firebase can be integrated using the Firebase SDK to store and retrieve data dynamically. The Cloud Firestore database enables real-time data synchronization, making it ideal for Flutter applications.

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

**1. Create a Firebase Project**

- Go to Firebase Console.
- Click on "Add Project" → Configure settings → Create the project.

**2. Add Firebase to Flutter App**

- Open your Flutter project.

Run:

```
flutter pub add firebase_core firebase_firestore
```

- 
- Configure Firebase in `android/app/google-services.json` (for Android) and `ios/Runner/GoogleService-Info.plist` (for iOS).

**3. Initialize Firebase in Flutter**

Modify `main.dart`:

```
import 'package:firebase_core/firebase_core.dart';  
import 'package:flutter/material.dart';
```

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

**4. Connect Firestore Database**

- In Firebase Console → Firestore → Create a database.

Create a Firestore instance in Flutter:

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

```
FirebaseFirestore firestore = FirebaseFirestore.instance;
```

**5. Perform CRUD Operations**

**Add Data:**

```
firestore.collection('users').add({'name': 'John', 'age': 25});
```

**Retrieve Data:**

```
firestore.collection('users').get().then((snapshot) {  
  
  for (var doc in snapshot.docs) {  
    print(doc.data());  
  }  
});
```

**Update Data:**

```
firestore.collection('users').doc('docId').update({'age': 26});
```

**Delete Data:**

```
firestore.collection('users').doc('docId').delete();
```

**6. Run the App & Test**

Execute `flutter run` and verify Firebase data operations in Firestore.

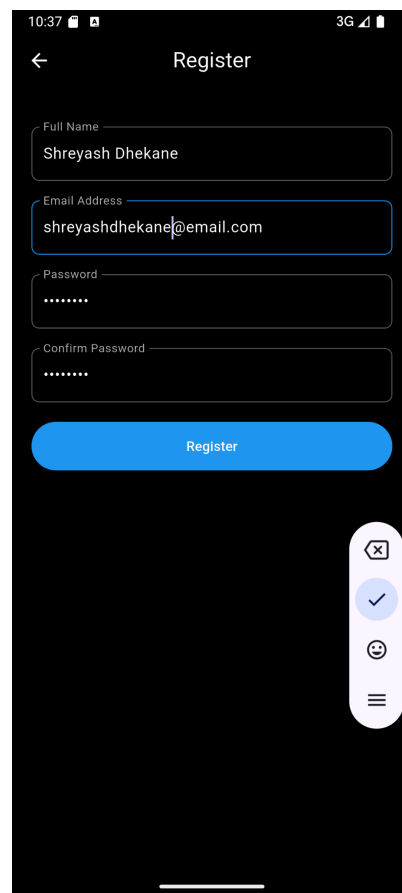
This setup enables a Flutter UI to interact with Firebase in real-time, ensuring seamless data storage and retrieval.

**Code:****build.gradle**

```
buildscript {  
    ext.kotlin_version = '2.1.0' // Update to a  
    newer version  
  
    repositories {  
        google()  
        mavenCentral()  
    }  
  
    dependencies {  
        classpath  
        "com.android.tools.build:gradle:8.1.2"  
  
        classpath  
        "com.google.gms:google-services:4.4.2" //  
        ADD THIS LINE  
  
        classpath  
        "org.jetbrains.kotlin:kotlin-gradle-plugin:$kotlin  
        _version"    }  
  
    allprojects {  
        repositories {  
            google()  
            mavenCentral()  
        }  
    }  
}
```

```
}  
  
rootProject.buildDir = "../build"  
  
subprojects {  
    project.buildDir =  
    "${rootProject.buildDir}/${project.name}"  
}  
  
subprojects {  
    project.evaluationDependsOn(":app")  
}  
  
tasks.register("clean", Delete) {  
    delete rootProject.buildDir  
}
```

### Screenshots:



The first screenshot shows the Firebase Project Overview page for 'myfitnesspal'. The main content area displays 'Get started by adding Firebase to your app' with buttons for 'Spark plan', 'Getting started? Tell Gemini about your project', and 'Add an app to get started'. The left sidebar contains navigation options like 'Project Overview', 'Generative AI', 'Build with Gemini', 'Genkit', 'Product categories', 'Build', 'Run', 'Analytics', 'All products', 'Related development tools', and 'Spark'.

The second screenshot shows the 'Add app to your Android app' wizard. Step 1 is 'Register app' with the package name 'com.example.myfitnesspal'. Step 2 is 'Download and then add config file', which includes a 'Download google-services.json' button and instructions to move the file to the 'app' directory in the project structure. A diagram shows the file being moved to the 'app' directory. Step 3 is 'Add Firebase SDK' and Step 4 is 'Next steps'.

The third screenshot shows the Firebase Project Overview page again, but now with the 'myfitnesspal' app listed under the 'Add app' button. The main content area displays 'Waiting for Analytics data...' and 'Accelerate app development'.

