# MAD PWA LAb 6

**NAME:** Prajakta Upadhye **Batch :** C

**Class** : D15A **Roll No.** : 65

**Aim**: To connect flutter with the firebase database.

To connect Firebase with Flutter, follow these steps:

1. Create a Firebase Project:
   - Go to the [Firebase Console](https://console.firebase.google.com/).
   - Click on "Add project" and follow the setup instructions.

2. Add Your App to Firebase Project:
   - After creating the project, click on "Add app" and select the appropriate platform (iOS/Android).
   - Follow the setup instructions to download the configuration files.

3. Add Dependencies in `pubspec.yaml`:
   - Open your `pubspec.yaml` file and add the following dependencies:

```yaml
dependencies:
  firebase_core: ^latest_version
  firebase_auth: ^latest_version  # If you need authentication
  cloud_firestore: ^latest_version  # If you need Firestore
  firebase_storage: ^latest_version  # If you need Firebase Storage
```

   - Run `flutter pub get` in your terminal to download the dependencies.

4. Initialize Firebase in Your App:
   - In your main Dart file (usually `main.dart`), initialize Firebase in the `main` function:

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

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

5. Use Firebase Services:
   - Now, you can use Firebase services in your Flutter app. For example, if you're using Firestore:

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

final FirebaseFirestore _firestore = FirebaseFirestore.instance;

// Example: Reading data from Firestore
Future<void> fetchData() async {
  QuerySnapshot snapshot = await _firestore.collection('your_collection').get();
  // Process snapshot data
}
```

6. Configure Firebase Authentication (Optional):
   - If you're using Firebase Authentication, follow the documentation to set up authentication methods.

7. Configure Firebase Storage (Optional):
   - If you're using Firebase Storage, follow the documentation to upload/download files.

Remember to check for the latest versions of Firebase packages on [pub.dev](https://pub.dev) and update your dependencies accordingly.

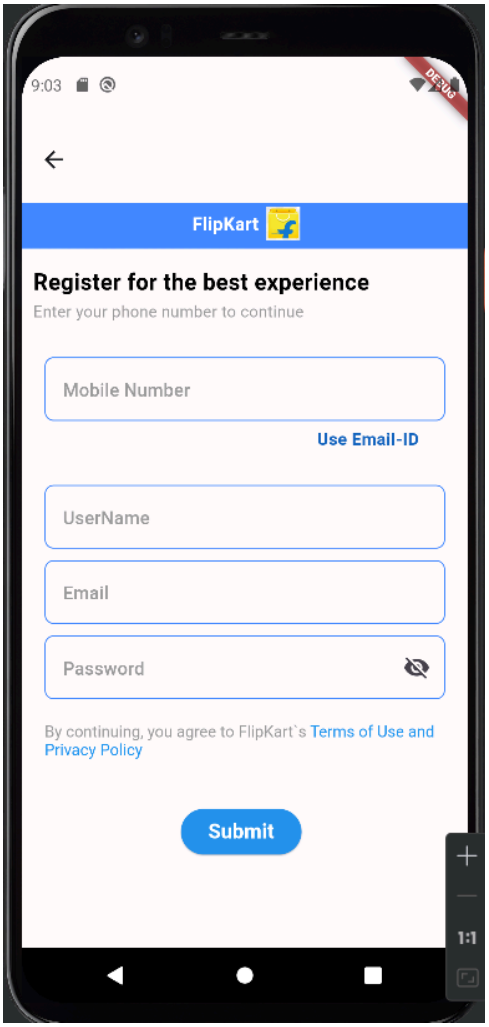
**CODE:**

```
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';
import 'package:firebase_core/firebase_core.dart';
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:flutter_application_1/firebase_options.dart';
import 'package:flutter_application_1/user_auth/firebase_auth_services.dart';
//import 'dart:ui' as ui;

// void main() {
//   runApp(const MyApp());
// }
Future<void> main() async {
  WidgetsFlutterBinding.ensureInitialized();
  await Firebase.initializeApp(options: DefaultFirebaseOptions.currentPlatform);
  runApp(const MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        body: HomePage(),
      ),
    );
  }
}

class HomePage extends StatefulWidget {
  const HomePage({Key? key}) : super(key: key);

  @override
  State<HomePage> createState() => _HomePageState();
}

class _HomePageState extends State<HomePage> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(body: LoginScreen());
  }
```

```
}

class LoginScreen extends StatefulWidget {
  const LoginScreen({Key? key}) : super(key: key);

  @override
  State<LoginScreen> createState() => _LoginScreenState();
}

class _LoginScreenState extends State<LoginScreen> {
  final FirebaseAuthServices _auth = FirebaseAuthServices();
  // TextEditingController _mobilenoContoller = TextEditingController();
  // TextEditingController _usernameContoller = TextEditingController();
  TextEditingController _emailContoller = TextEditingController();
  TextEditingController _passwordContoller = TextEditingController();
  bool _passwordVisible = false;

  @override
  void dispose() {
    // TODO: implement dispose
    // _mobilenoContoller.dispose();
    // _usernameContoller.dispose();
    _emailContoller.dispose();
    _passwordContoller.dispose();
    super.dispose();
  }

  void _navigateToHomeScreen() {
    Navigator.of(context).push(
      MaterialPageRoute(
        builder: (context) => HomeScreen(),
      ),
    );
  }

  void _navigateToRegisterScreen() {
    Navigator.of(context).push(
      MaterialPageRoute(
        builder: (context) => RegisterScreen(),
      ),
    );
  }

  @override
```

```
Widget build(BuildContext context) {
  return Padding(
    padding: const EdgeInsets.all(5.0),
    child: ListView(
      children: [
        Stack(
          children: [
            Container(
              color: Colors.blueAccent,
              height: 40.0,
              width: double.infinity,
              margin: const EdgeInsets.only(top: 40.0),
            ),
            Positioned(
              left: 150.0,
              top: 40.0 + (30.0 - 16.0) / 2,
              child: Text(
                "FlipKart",
                style: TextStyle(
                  color: Colors.white,
                  fontSize: 16.0,
                  fontWeight: FontWeight.bold,
                ),
              ),
            ),
            SizedBox(width: 8.0),
            Positioned(
              top: 43.0,
              left: 215.0,
              child: Image.asset(
                'assets/flipkartlogo.jpg',
                height: 30.0,
                width: 30.0,
              ),
            ),
          ],
        ),
        Padding(
          padding: const EdgeInsets.only(left: 10.0, top: 25.0),
          child: Text(
            "Log in for the best experience",
            style: TextStyle(
              color: Colors.black,
              fontSize: 20.0,
      fontWeight: FontWeight.bold,
    ),
  ),
),
Padding(
  padding: const EdgeInsets.only(left: 20.0, top: 50.0, right: 20.0),
  child: Container(
    decoration: BoxDecoration(
      border: Border.all(color: Colors.blueAccent),
      borderRadius: BorderRadius.circular(8.0),
    ),
    child: TextField(
      controller: _emailContoller,
      decoration: InputDecoration(
        hintText: 'Email',
        hintStyle: TextStyle(color: Colors.grey),
        border: InputBorder.none,
        contentPadding: EdgeInsets.all(15.0),
      ),
    ),
  ),
),
// Padding(
//   padding: const EdgeInsets.only(left: 20.0, top: 30.0, right: 20.0),
//   child: Container(
//     decoration: BoxDecoration(
//       border: Border.all(color: Colors.blueAccent),
//       borderRadius: BorderRadius.circular(8.0),
//     ),
//     child: TextField(
//       controller:  _passwordContoller,
//       decoration: InputDecoration(
//         hintText: 'Password',
//         hintStyle: TextStyle(color: Colors.grey),
//         border: InputBorder.none,
//         contentPadding: EdgeInsets.all(15.0),
//       ),
//     ),
//   ),
// ),
Padding(
  padding: const EdgeInsets.only(left: 20.0, top: 30.0, right: 20.0),
  child: Container(
    decoration: BoxDecoration(
      border: Border.all(color: Colors.blueAccent),
      borderRadius: BorderRadius.circular(8.0),
    ),
    child: TextField(
      controller: _passwordContoller,
      obscureText: !_passwordVisible, // Obfuscate the password
      decoration: InputDecoration(
        hintText: 'Password',
        hintStyle: TextStyle(color: Colors.grey),
        border: InputBorder.none,
        contentPadding: EdgeInsets.all(15.0),
        suffixIcon: IconButton(
          icon: Icon(
            _passwordVisible
                ? Icons.visibility
                : Icons.visibility_off,
          ),
          onPressed: () {
            setState(() {
              _passwordVisible = !_passwordVisible;
            });
          },
        ),
      ),
    ),
  ),
),
Padding(
  padding: const EdgeInsets.only(left: 20.0, top: 30.0, right: 20.0),
  child: RichText(
    text: TextSpan(
      text: 'By continuing, you agree to FlipKart`s ',
      style: TextStyle(color: Colors.grey),
      children: [
        TextSpan(
          text: 'Terms of Use and Privacy Policy',
          style: TextStyle(color: Colors.blue),
        ),
      ],
    ),
  ),
),
Padding(
  padding: const EdgeInsets.only(top: 20.0),
     child: ElevatedButton(
      onPressed: () => _navigateToRegisterScreen(),
      style: ElevatedButton.styleFrom(
       primary: Colors.blue,
      ),
      child: Text(
       'Register',
       style: TextStyle(
        color: Colors.white,
        fontSize: 18.0,
        fontWeight: FontWeight.bold,
       ),
      ),
     ),
    ),
    Padding(
     padding: const EdgeInsets.only(top: 10.0),
     child: ElevatedButton(
      onPressed: _signIn,
      style: ElevatedButton.styleFrom(
       primary: Colors.blue,
      ),
      child: Text(
       'Continue',
       style: TextStyle(
        color: Colors.white,
        fontSize: 18.0,
        fontWeight: FontWeight.bold,
       ),
      ),
     ),
    ),
   ],
  ),
 );
}

void _signIn() async {
 // String username = _usernameContoller.text;
 // String mobileno = _mobilenoContoller.text;
 String email = _emailContoller.text;
 String password = _passwordContoller.text;

 User? user = await _auth.signInWithEmailandPassword(email, password);
```

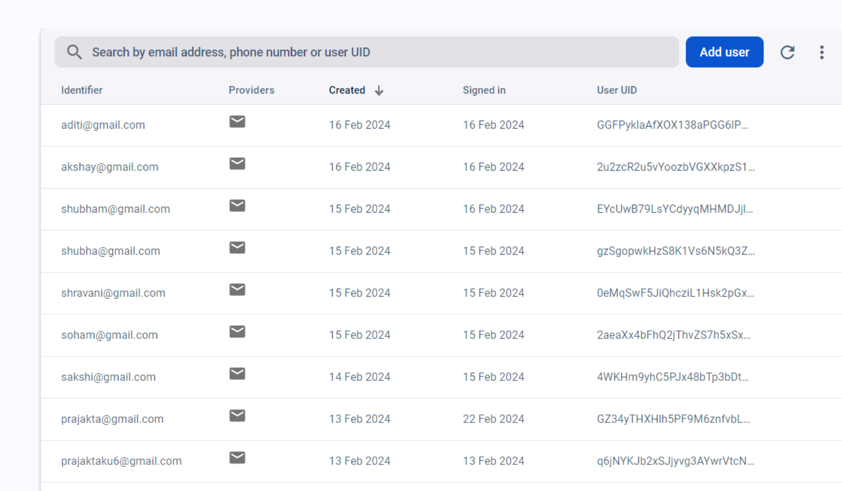
```
    if (user != null) {
      print("User succesfully Signed in");
      _showDialog("Success", "User successfully signed in!");
      Navigator.of(context).pushReplacement(
        MaterialPageRoute(
          builder: (context) => HomeScreen(),
        ),
      );
    } else {
      print("Error in registeration");
      _showDialog("Error", "Invalid email or password. Please try again.");
    }
  }

  void _showDialog(String title, String content) {
    showDialog(
      context: context,
      builder: (BuildContext context) {
        return AlertDialog(
          title: Text(title),
          content: Text(content),
          actions: [
            TextButton(
              onPressed: () {
                Navigator.of(context).pop(); // Close the dialog
              },
              child: Text('OK'),
            ),
          ],
        );
      },
    );
  }
}

class RegisterScreen extends StatefulWidget {
  @override
  _RegisterScreenState createState() => _RegisterScreenState();
}

class _RegisterScreenState extends State<RegisterScreen> {
  final FirebaseAuthServices _auth = FirebaseAuthServices();
  TextEditingController _mobilenoContoller = TextEditingController();
TextEditingController _usernameContoller = TextEditingController();
TextEditingController _emailContoller = TextEditingController();
TextEditingController _passwordContoller = TextEditingController();
bool _passwordVisible = false;

@override
void dispose() {
  // TODO: implement dispose
  _mobilenoContoller.dispose();
  _usernameContoller.dispose();
  _emailContoller.dispose();
  _passwordContoller.dispose();
  super.dispose();
}

@override
Widget build(BuildContext context) {
  return Material(
    child: Scaffold(
      appBar: AppBar(),
      body: SingleChildScrollView(
        // Wrap with SingleChildScrollView
        physics: BouncingScrollPhysics(), // Enable scrolling
        child: Padding(
          padding: const EdgeInsets.only(top: 0.0),
          child: Stack(
            children: [
              Column(
                mainAxisAlignment: MainAxisAlignment.start,
                crossAxisAlignment: CrossAxisAlignment.start,
                children: [
                  Stack(
                    children: [
                      Container(
                        color: Colors.blueAccent,
                        height: 40.0,
                        width: double.infinity,
                        margin: const EdgeInsets.only(top: 10.0),
                      ),
                      Positioned(
                        left: 150.0,
                        top: 10.0 + (30.0 - 16.0) / 2,
                        child: Text(
                          "FlipKart",
        style: TextStyle(
          color: Colors.white,
          fontSize: 16.0,
          fontWeight: FontWeight.bold,
        ),
      ),
    ),
    SizedBox(
      width: 8.0, // Add some space between text and image
    ),
    Positioned(
      top: 13.0,
      left: 215.0,
      child: Image.asset(
        'assets/flipkartlogo.jpg',
        height: 30.0,
        width: 30.0,
      ),
    ),
  ],
),
Padding(
  padding: const EdgeInsets.only(left: 10.0, top: 15.0),
  child: Text(
    "Register for the best experience",
    style: TextStyle(
      color: Colors.black,
      fontSize: 20.0,
      fontWeight: FontWeight.bold,
    ),
  ),
),
Padding(
  padding: const EdgeInsets.only(left: 10.0, top: 1.0),
  child: Text(
    "Enter your phone number to continue",
    style: TextStyle(
      color: Colors.grey,
      fontSize: 14.0,
    ),
  ),
),
Padding(
  padding: const EdgeInsets.only(
      left: 20.0, top: 30.0, right: 20.0),
  child: Container(
    decoration: BoxDecoration(
      border: Border.all(color: Colors.blueAccent),
      borderRadius: BorderRadius.circular(8.0),
    ),
    child: TextField(
      controller: _mobilenoContoller,
      decoration: InputDecoration(
        hintText: 'Mobile Number',
        hintStyle: TextStyle(color: Colors.grey),
        border: InputBorder.none,
        contentPadding: EdgeInsets.all(15.0),
      ),
    ),
  ),
),
Padding(
  padding: const EdgeInsets.only(left: 260.0, top: 5),
  child: Text(
    "Use Email-ID",
    style: TextStyle(
      color: Colors.blue[800],
      fontSize: 15.0,
      fontWeight: FontWeight.bold,
    ),
  ),
),
Padding(
  padding: const EdgeInsets.only(
      left: 20.0, top: 30.0, right: 20.0),
  child: Container(
    decoration: BoxDecoration(
      border: Border.all(color: Colors.blueAccent),
      borderRadius: BorderRadius.circular(8.0),
    ),
    child: TextField(
      controller: _usernameContoller,
      decoration: InputDecoration(
        hintText: 'UserName',
        hintStyle: TextStyle(color: Colors.grey),
        border: InputBorder.none,
        contentPadding: EdgeInsets.all(15.0),
      ),
    ),
  ),
),
Padding(
  padding: const EdgeInsets.only(
      left: 20.0, top: 10.0, right: 20.0),
  child: Container(
    decoration: BoxDecoration(
      border: Border.all(color: Colors.blueAccent),
      borderRadius: BorderRadius.circular(8.0),
    ),
    child: TextField(
      controller: _emailContoller,
      decoration: InputDecoration(
        hintText: 'Email',
        hintStyle: TextStyle(color: Colors.grey),
        border: InputBorder.none,
        contentPadding: EdgeInsets.all(15.0),
      ),
    ),
  ),
),
Padding(
  padding: const EdgeInsets.only(
      left: 20.0, top: 10.0, right: 20.0),
  child: Container(
    decoration: BoxDecoration(
      border: Border.all(color: Colors.blueAccent),
      borderRadius: BorderRadius.circular(8.0),
    ),
    child: TextField(
      controller: _passwordContoller,
      obscureText:
          !_passwordVisible, // Obfuscate the password
      decoration: InputDecoration(
        hintText: 'Password',
        hintStyle: TextStyle(color: Colors.grey),
        border: InputBorder.none,
        contentPadding: EdgeInsets.all(15.0),
        suffixIcon: IconButton(
          icon: Icon(
            _passwordVisible
                ? Icons.visibility
                : Icons.visibility_off,
          ),
          onPressed: () {
            setState(() {
              _passwordVisible = !_passwordVisible;
            });
          },
        ),
      ),
    ),
  ),
),
Padding(
  padding: const EdgeInsets.only(
      left: 20.0, top: 20.0, right: 20.0),
  child: RichText(
    text: TextSpan(
      text: 'By continuing, you agree to FlipKart`s ',
      style: TextStyle(color: Colors.grey),
      children: [
        TextSpan(
          text: 'Terms of Use and Privacy Policy',
          style: TextStyle(color: Colors.blue),
        ),
      ],
    ),
  ),
),
Padding(
  padding: const EdgeInsets.only(top: 40.0, left: 140.0),
  child: ElevatedButton(
    onPressed: () {
      _register();
      String mobileNumber = _mobilenoContoller.text;
      String userName = _usernameContoller.text;
      String email = _emailContoller.text;
      String password = _passwordContoller.text;

      // Print the values for testing (you can replace this with your logic)
      print('Mobile Number: $mobileNumber');
      print('User Name: $userName');
      print('Email: $email');
      print('Password: $password');

      //  Navigator.of(context).pushReplacement(
              //   MaterialPageRoute(
              //     builder: (context) => HomeScreen(),
              //   ),
              // );
              // Add the action for the blue button here
              // For example, you can navigate to another screen
              print('Blue button pressed!');
            },
            style: ElevatedButton.styleFrom(
              primary: Colors.blue,
            ),
            child: Text(
              'Submit',
              style: TextStyle(
                color: Colors.white,
                fontSize: 18.0,
                fontWeight: FontWeight.bold,
              ),
            ),
          ),
        ),

        // Padding(
        //   padding: const EdgeInsets.only(top: 0.0, left: 100.0),
        //   child: TextButton(
        //     onPressed: () {
        //       // Add the action for the text button here
        //       // For example, you can navigate to the login screen
        //       print('Text button pressed!');
        //     },
        //     child: Text(
        //       'Already have an account? Login',
        //       style: TextStyle(
        //         color: Colors.blue,
        //         fontSize: 16.0,
        //         fontWeight: FontWeight.bold,
        //       ),
        //     ),
        //   ),
        // ),
      ],
    ),
  ],
),
          ),
        ),
      ),
    );
  }

  void _register() async {
  String username = _usernameContoller.text;
  String mobileno = _mobilenoContoller.text;
  String email = _emailContoller.text;
  String password = _passwordContoller.text;

  try {
    User? user = await _auth.registerWithEmailandPassword(email, password);

    if (user != null) {
      print("User successfully registered");
      Navigator.of(context).pushReplacement(
        MaterialPageRoute(
          builder: (context) => HomeScreen(),
        ),
      );
      addUserDetails(
        _usernameContoller.text.trim(),
        _mobilenoContoller.text.trim(),
        _emailContoller.text.trim(),
        _passwordContoller.text.trim(),
      );
    } else {
      print("Error in registration");
    }
  } catch (e) {
    print("Error during registration: $e");
    // Display a pop-up indicating that the user already exists
    _showErrorDialog("User Already Exists",
        "The provided email is already registered. Please use a different email or login.");
  }
}

void _showErrorDialog(String title, String message) {
  showDialog(
    context: context,
    builder: (BuildContext context) {
      return AlertDialog(
        title: Text(title),
        content: Text(message),
        actions: [
          TextButton(
            onPressed: () {
              Navigator.of(context).pop(); // Close the dialog
            },
            child: Text('OK'),
          ),
        ],
      );
    },
  );
}


  Future addUserDetails(
      String username, String email, String mobileno, String password) async {
    await FirebaseFirestore.instance.collection('accountinfo').add({
      'username': username,
      'email': email,
      'mobileno': mobileno,
      'password': password,
      'address': '',
    });
  }
}
```



Register for the best experience

Enter your phone number to continue

By continuing, you agree to FlipKart`s Terms of Use and Privacy Policy



| Identifier | Providers | Created ↓ | Signed in | User UID |
|---|---|---|---|---|
| aditi@gmail.com | ✉ | 16 Feb 2024 | 16 Feb 2024 | GGFPyklaAfXOX138aPGG6IP... |
| akshay@gmail.com | ✉ | 16 Feb 2024 | 16 Feb 2024 | 2u2zcR2u5vYoozbVGXXkpzS1... |
| shubham@gmail.com | ✉ | 15 Feb 2024 | 16 Feb 2024 | EYcUwB79LsYCdyyqMHMDJjl... |
| shubha@gmail.com | ✉ | 15 Feb 2024 | 15 Feb 2024 | gzSgopwkHzS8K1Vs6N5kQ3Z... |
| shravani@gmail.com | ✉ | 15 Feb 2024 | 15 Feb 2024 | 0eMqSwF5JiQhcziL1Hsk2pGx... |
| soham@gmail.com | ✉ | 15 Feb 2024 | 15 Feb 2024 | 2aeaXx4bFhQ2jThvZS7h5xSx... |
| sakshi@gmail.com | ✉ | 14 Feb 2024 | 15 Feb 2024 | 4WKHm9yhC5PJx48bTp3bDt... |
| prajakta@gmail.com | ✉ | 13 Feb 2024 | 22 Feb 2024 | GZ34yTHXHlh5PF9M6znfvbL... |
| prajaktaku6@gmail.com | ✉ | 13 Feb 2024 | 13 Feb 2024 | q6jNYKJb2xSJjyvg3AYwrVtcN... |

## CONCLUSION:

Thus we connected firebase to flutter.