

Name: Mahek Bhagwani
D15A Roll no:06

EXPERIMENT NO: 06

Aim: To Connect Flutter UI with firebase database

Theory:

Firebase is a Backend-as-a-Service (BaaS) app development platform that provides hosted backend services such as a realtime database, cloud storage, authentication, crash reporting, machine learning, remote configuration, and hosting for your static files.

CODE:

Firestore auth:

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

class FirestoreMethods {
  final FirebaseFirestore _firestore = FirebaseFirestore.instance;
  final FirebaseAuth _auth = FirebaseAuth.instance;
  Stream<QuerySnapshot<Map<String, dynamic>>> get meetingHistory => _firestore
    .collection('users')
    .doc(_auth.currentUser!.uid)
    .collection('meetings')
    .orderBy('createdAt', descending: true)
    .snapshots();
  void addToMeetingHistory(String meetingName) async {
    try {
      await _firestore
        .collection('users')
        .doc(_auth.currentUser!.uid)
        .collection('meetings')
        .add(
          {
            'meetingName': meetingName,
            'createdAt': DateTime.now(),
          },
        );
    } catch (e) {
      print(e);
    }
  }
}
```

```

void deleteMeetingHistory(String id) async {
  try {
    await _firestore
      .collection('users')
      .doc(_auth.currentUser!.uid)
      .collection('meetings')
      .doc(id)
      .delete();
  } catch (e) {}
}
}

```

Jitsi meet package methods:

```

import 'package:jitsi_meet_flutter_sdk/jitsi_meet_flutter_sdk.dart';
import 'package:zoom_meet/resources/auth_methods.dart';
import 'package:zoom_meet/resources/firestore_methods.dart';

```

```

class JitsiMeetMethods {
  final AuthMethods _authMethods = AuthMethods();
  final FirestoreMethods _firestoreMethods = FirestoreMethods();
  void createMeeting({
    required String roomName,
    required bool isAudioMuted,
    required bool isVideoMuted,
    String userName = "",
  }) async {
    try {
      String name;
      if (userName.isEmpty) {
        name = _authMethods.user.displayName!;
      } else {
        name = userName;
      }
      var options = JitsiMeetConferenceOptions(
        room: roomName,
        userInfo: JitsiMeetUserInfo(
          displayName: name,
          email: _authMethods.user.email,
          avatar: _authMethods.user.photoURL,
        ),
        configOverrides: {
          "startWithAudioMuted": isAudioMuted,
          "startWithVideoMuted": isVideoMuted,
        },
      ),
    }
  }
}

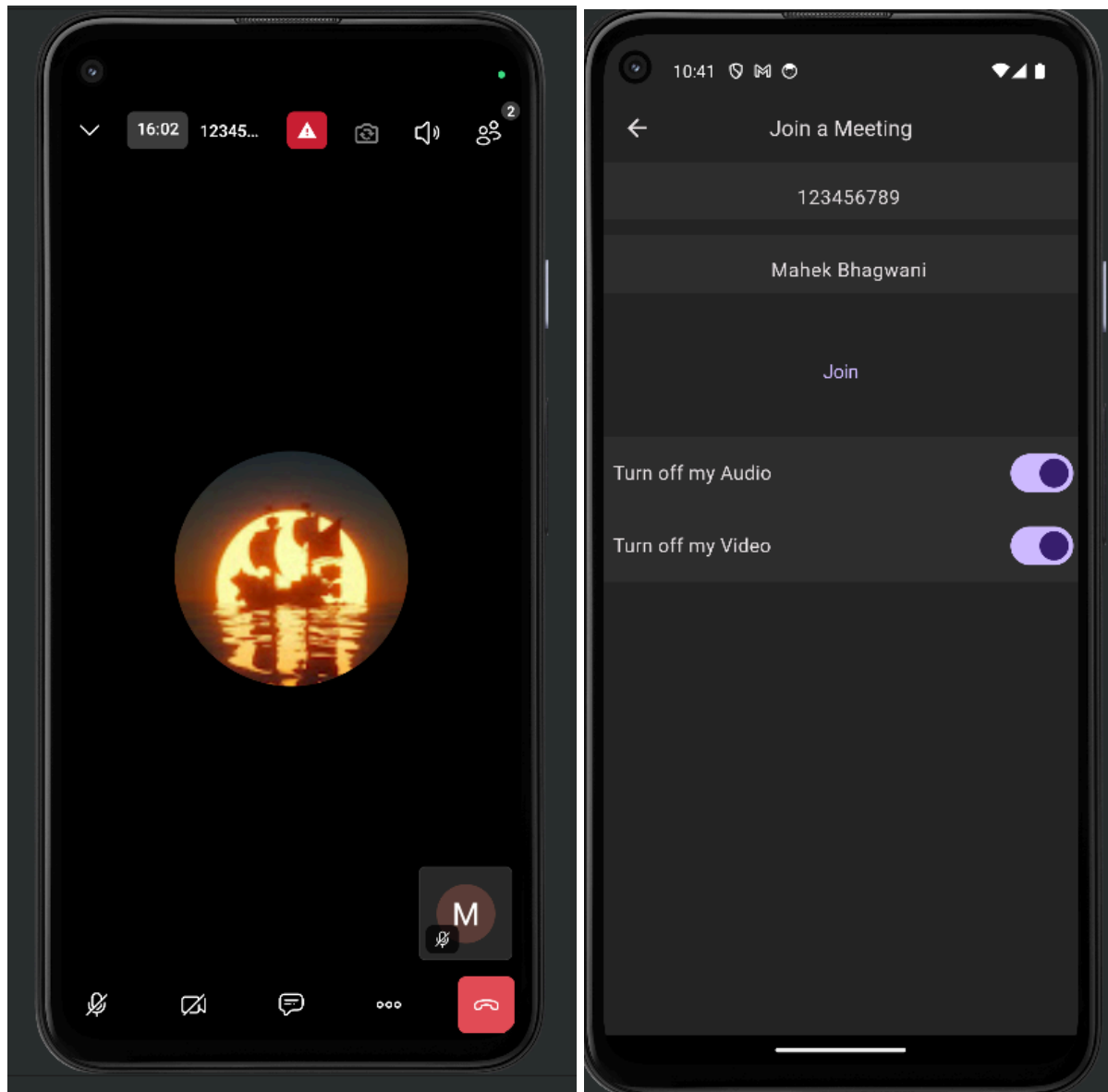
```

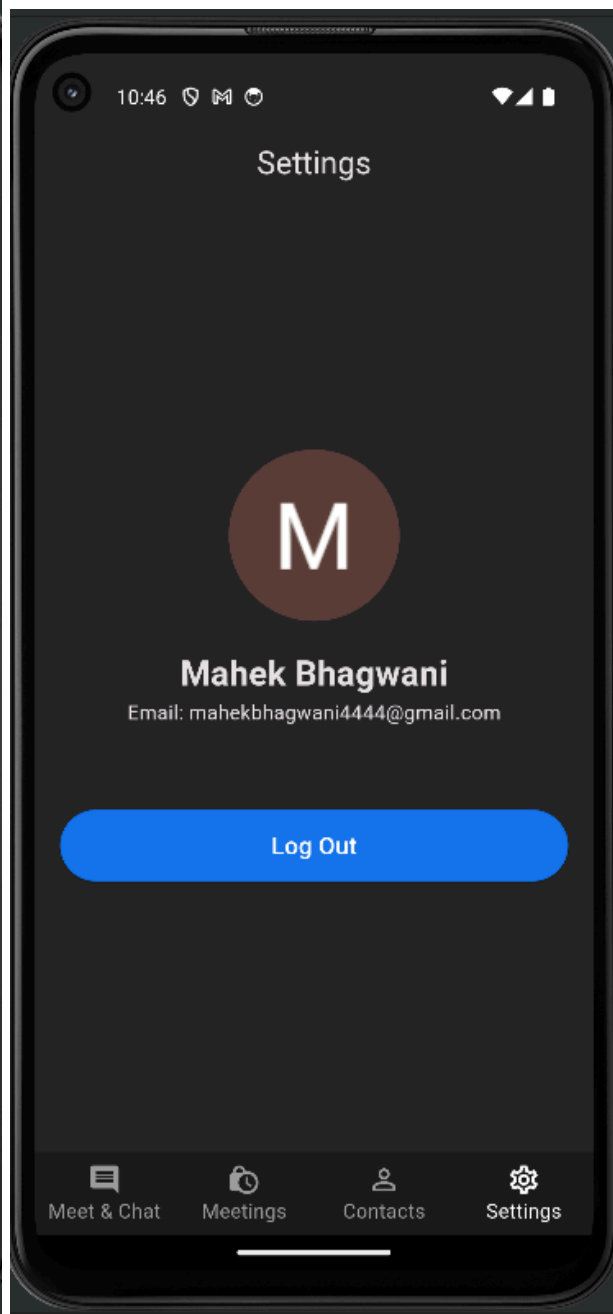
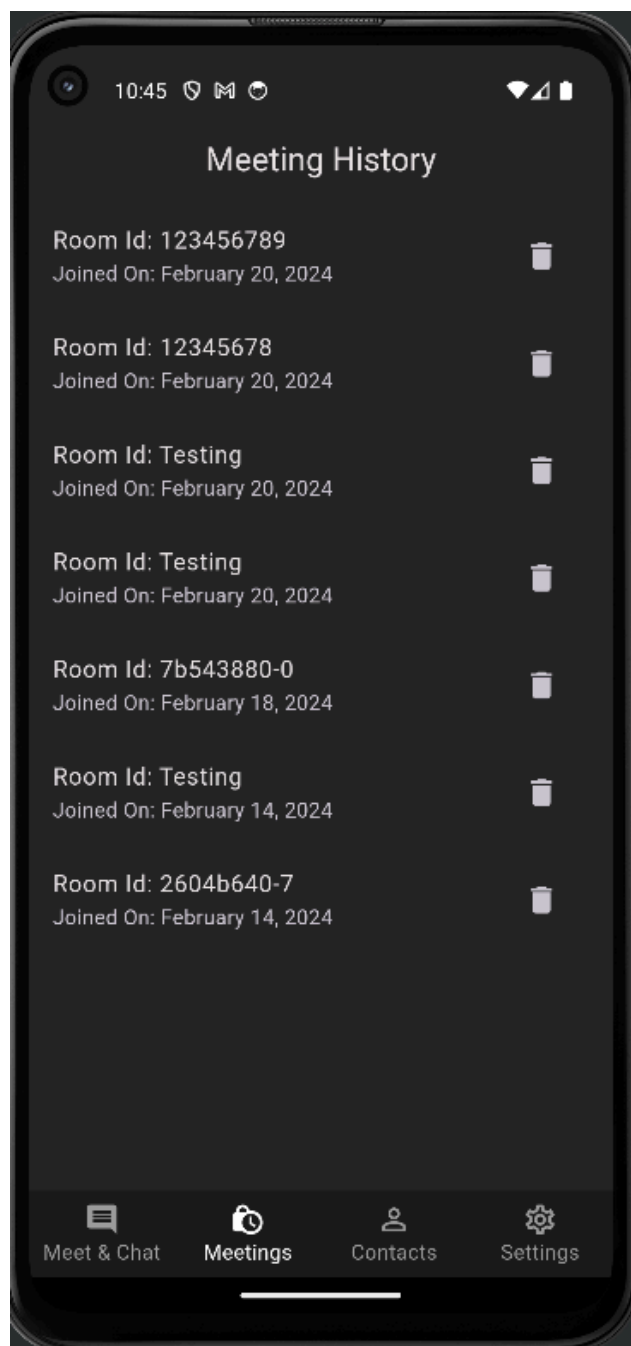
```

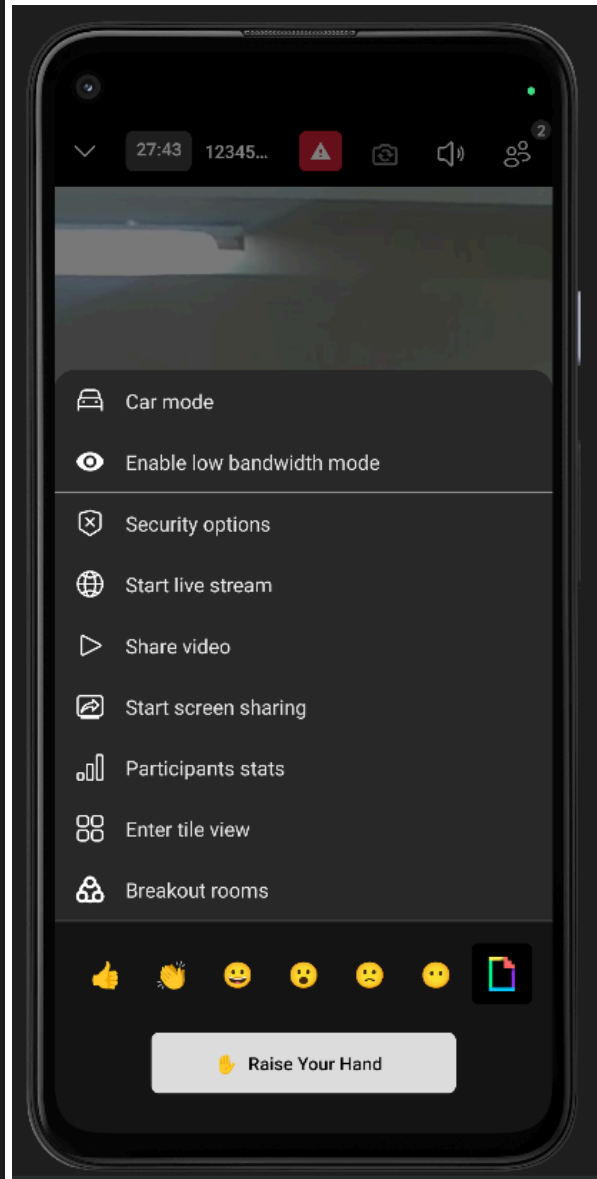
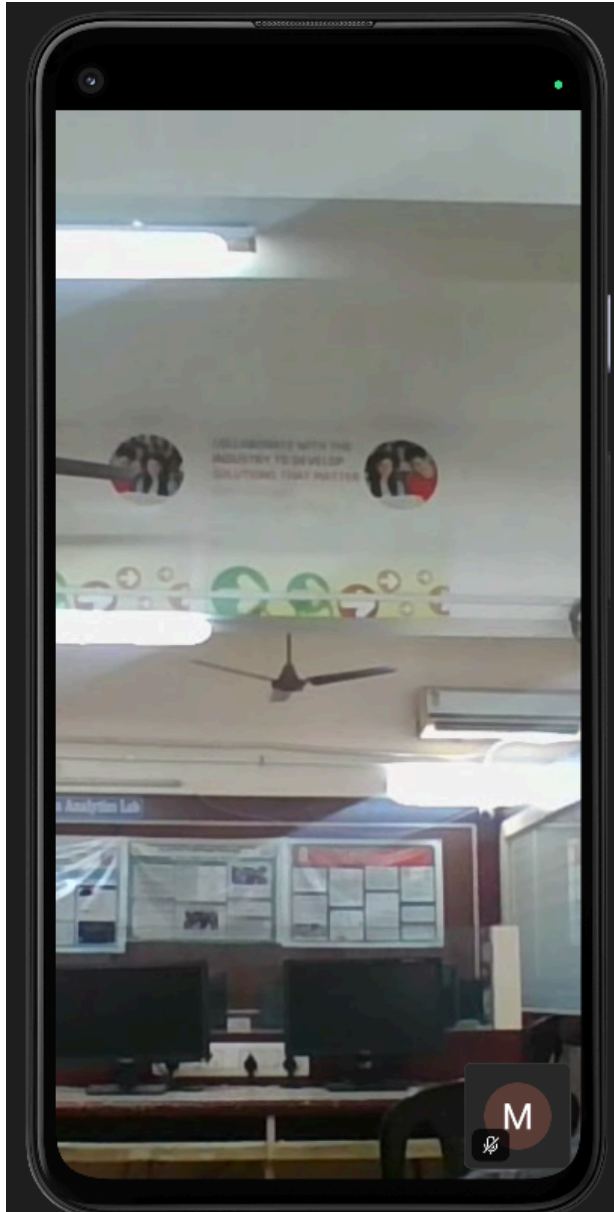
);
_firestoreMethods.addToMeetingHistory(roomName);
var jitsiMeet = JitsiMeet();
await jitsiMeet.join(options);
} catch (e) {
  print(e.toString());
}
}
}
}

```

Output:









Firestore Console:

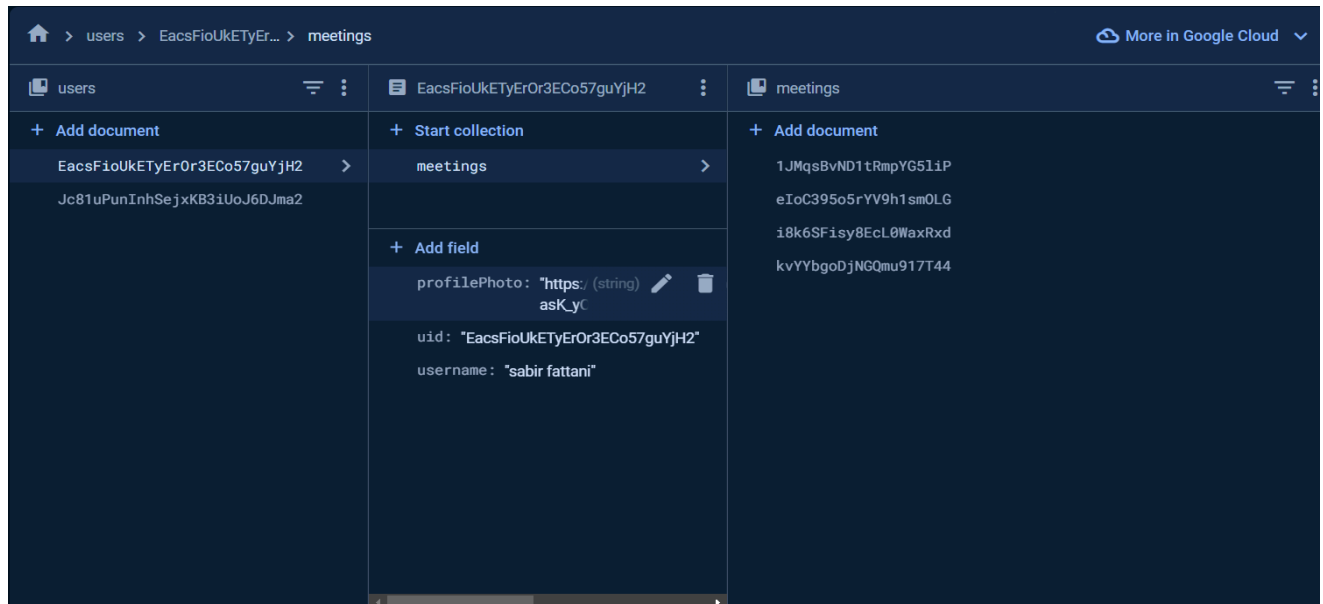
Firestore Authentication:

Search by email address, phone number or user UID					Add user	↺	⋮
Identifier	Providers	Created ↓	Signed in	User UID			
mahekbhagwani4444@...		14 Feb 2024	14 Feb 2024	Jc81uPunInhSejxKB3iUoJ6DJ...			
vu4s2122003@pvppco...		13 Feb 2024	13 Feb 2024	EacsFioUkETyErOr3EC057guY...			
					Rows per page	50	1 - 2 of 2

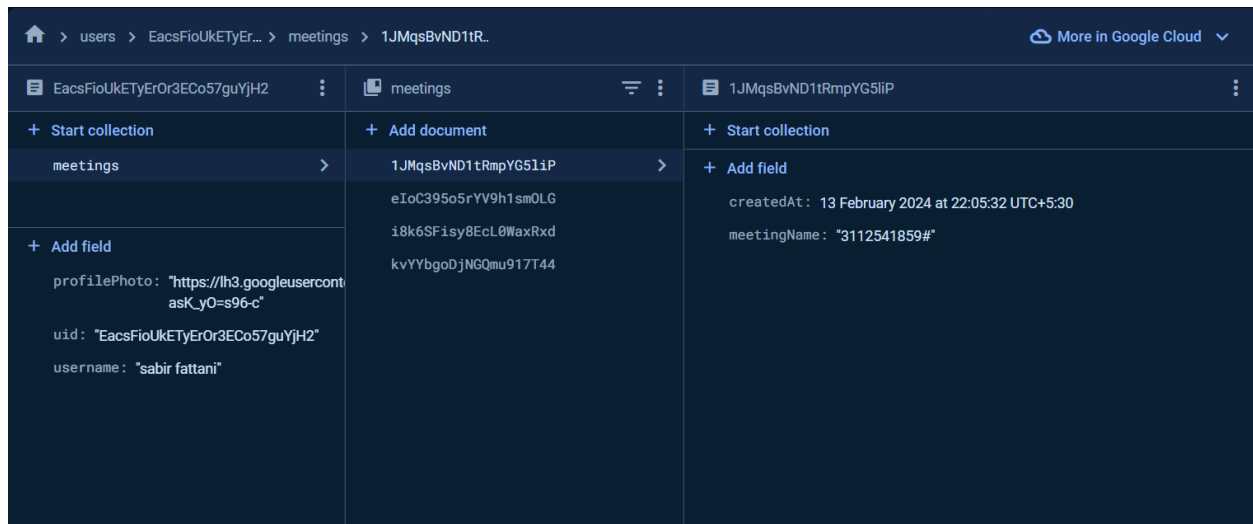
I have created a firestore database syncing to my project
This includes all the user details like user id user name profile photo.

🏠 > users > EacsFioUkETyEr...			☁ More in Google Cloud	
📁 (default)	📁 users	⌵ ⋮	📁 EacsFioUkETyErOr3EC057guYjH2	⋮
+ Start collection	+ Add document		+ Start collection	
users	EacsFioUkETyErOr3EC057guYjH2		meetings	
	Jc81uPunInhSejxKB3iUoJ6DJma2		+ Add field	
			profilePhoto: "https://lh3.googleusercontent.com/a/ACg8ocJ5DGpopkXNqqplaskK_y0=s96-c"	
			uid: "EacsFioUkETyErOr3EC057guYjH2"	
			username: "sabir fattani"	

Then in users you can check their meeting history that which users created how many meetings it will store the meeting id



Ahead after clicking on a particular meeting you can get more information about that particular meeting like meeting name, when was it created how many minutes the meeting continued



Conclusion: Here, Successfully Connected Flutter UI with Firebase database