

MPL LAB 6

AIM: To setup Firebase in Flutter Project

Theory:

Setting up flutter Firebase in IOS app:

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 this article, you will create a Firebase project for iOS and Android platforms using Flutter.

Prerequisites for Firebase setup:

1. A Google account to use Firebase.
2. Developing for iOS will require XCode.
3. To download and install Flutter.
4. To download and install Android Studio and Visual Studio Code.
5. It is recommended to install plugins for your code editor:
6. Flutter and Dart plugins installed for Android Studio.
7. Flutter extension installed for Visual Studio Code.

Code:

Firebase.options file

```
import 'package:firebase_core/firebase_core.dart' show FirebaseOptions;
import 'package:flutter/foundation.dart'
    show defaultTargetPlatform, kIsWeb, TargetPlatform;

/// Default [FirebaseOptions] for use with your Firebase apps.
///
/// Example:
/// ```dart
/// import 'firebase_options.dart';
/// // ...
/// await Firebase.initializeApp(
///   options: DefaultFirebaseOptions.currentPlatform,
/// );
/// ```
class DefaultFirebaseOptions {
  static FirebaseOptions get currentPlatform {
    if (kIsWeb) {
      return web;
    }
    switch (defaultTargetPlatform) {
      case TargetPlatform.android:
        return android;
    }
  }
}
```

```

    case TargetPlatform.iOS:
      return ios;
    case TargetPlatform.macOS:
      throw UnsupportedError(
        'DefaultFirebaseOptions have not been configured for macos - '
        'you can reconfigure this by running the FlutterFire CLI again.',
      );
    case TargetPlatform.windows:
      throw UnsupportedError(
        'DefaultFirebaseOptions have not been configured for windows - '
        'you can reconfigure this by running the FlutterFire CLI again.',
      );
    case TargetPlatform.linux:
      throw UnsupportedError(
        'DefaultFirebaseOptions have not been configured for linux - '
        'you can reconfigure this by running the FlutterFire CLI again.',
      );
    default:
      throw UnsupportedError(
        'DefaultFirebaseOptions are not supported for this platform.',
      );
  }
}

static const FirebaseOptions web = FirebaseOptions(
  apiKey: 'AIzaSyD- VN8edivHrRv5xd9QVgnYv-pV1Rl0WI',
  appId: '1:724141443535:web:eada6c562a8f679026eaba',
  messagingSenderId: '724141443535',
  projectId: 'whatsapp-new-clone-3fa2b',
  authDomain: 'whatsapp-new-clone-3fa2b.firebaseio.com',
  storageBucket: 'whatsapp-new-clone-3fa2b.appspot.com',
  measurementId: 'G-L3LR8LLGB7',
);

static const FirebaseOptions android = FirebaseOptions(
  apiKey: 'AIzaSyCNJoulwHlxCqCPu-9yUpVcXuMOOduCShY',
  appId: '1:724141443535:android:5ef3e32a07637f1926eaba',
  messagingSenderId: '724141443535',
  projectId: 'whatsapp-new-clone-3fa2b',
  storageBucket: 'whatsapp-new-clone-3fa2b.appspot.com',
);

static const FirebaseOptions ios = FirebaseOptions(
  apiKey: 'AIzaSyClZNaP9NVYF18N3gVMCqW81ElLsds8mLk',
  appId: '1:724141443535:ios:ac7cca22ed2f848926eaba',
  messagingSenderId: '724141443535',
  projectId: 'whatsapp-new-clone-3fa2b',
  storageBucket: 'whatsapp-new-clone-3fa2b.appspot.com',
  iosBundleId: 'com.example.whatsapp',
);

```

```
);  
}
```

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.  
  cloud_firestore: ^4.9.2  
  country_picker: ^2.0.21  
  cupertino_icons: ^1.0.2  
  firebase_auth: ^4.10.0  
  firebase_core: ^2.16.0  
  firebase_storage: ^11.2.7  
  flutter_native_splash: ^2.3.2  
  flutter_hooks: ^0.18.0  
  hooks_riverpod: ^2.4.1  
  cached_network_image: ^3.3.1  
  dio: ^5.4.0  
  mobile_chat_ui: ^1.0.0  
  flutter_chat_ui: ^1.6.12  
  file_picker: ^6.1.1  
  open_filex: ^4.4.0  
  bubble: ^1.2.1  
  get_storage: ^2.1.1  
  intl: ^0.19.0  
  google_fonts: ^6.1.0  
  http: ^1.2.0  
  uuid: ^4.3.3  
  flutter_launcher_icons: ^0.13.1  
  url_launcher: ^6.0.9  
  image_picker: ^1.0.7  
  fluttertoast: ^8.2.4  
  story_view: ^0.16.5  
  
dev_dependencies:  
  flutter_test:  
    sdk: flutter  
  
  # The "flutter_lints" package or other dev dependencies go here if needed  
  
flutter_native_splash:  
  color: "#FFFFFF"  
  color_dark: "#111b21"  
  image: assets/images/splash_light.png  
  image_dark: assets/images/splash_dark.png  
  android_12:  
    color1: "#FFFFFF"  
    color_dark1: "#111b21"
```

```
image1: assets/images/splash_light.png
image_dark1: assets/images/splash_dark.png
icon_background_color: "#FFFFFF"
icon_background_color_dark: "#111B21"
```

OUTPUT:

Firestore for authentication:

Search by email address, phone number, or user UID					Add user	↺	⋮
Identifier	Providers	Created ↓	Signed In	User UID			
9136562312@mobile.c...	✉	Feb 23, 2024		EFFmaw0MVWdovlr6fMu9Uk...			
977344527@mobile.com	✉	Feb 23, 2024		xJCHRoP1j7dIRnS0nS5PuXIV...			
9819149555@mobile.c...	✉	Feb 23, 2024		5BkMNI3RzASvGerj32LWZso...			
9619132050@mobile.c...	✉	Feb 23, 2024		OrleGSJqy2OKk2cHlARhoo17...			
					Rows per page:	50 ▼	1 – 4 of 4 < >

Users using whatsapp.

Conclusion:

Successfully implemented the Firestore setup in flutter project.