



TIERS LIMITED SUMMER INTERNSHIP 2024

MOBILE APP DEVELOPMENT

Introduction to Firebase

CLASS # 18



TIERS
Limited

Overview

In this class, we will learn about Firebase, a powerful backend platform for building web and mobile applications. We will focus on integrating Firebase into a Flutter project, setting the stage for future classes where we will explore various Firebase services such as Authentication, Firestore, and Cloud Storage.

Objectives

- Understand what Firebase is and its key features.
- Learn how to set up a Firebase project.
- Integrate Firebase into a Flutter project.

What is Firebase?

Firebase is a comprehensive platform provided by Google for building mobile and web applications. It offers a variety of tools and services to help developers build high-quality apps, including Cloud Database.

Setting Up Firebase Project

Step 1: Create a Firebase Project

1. Go to the Firebase Console.
2. Click on "Add project" and follow the instructions to create a new project.

Step 2: Register Your App with Firebase

1. In the Firebase Console, navigate to your project.
2. Click on the Android or iOS icon to add a new app.
3. Follow the instructions to register your app with Firebase. You'll need to provide your app's package name (e.g., com.example.myapp).

Step 3: Download the Firebase Configuration File

For Android:

- Download the google-services.json file and place it in the android/app directory of your Flutter project.

Integrating Firebase with a Flutter Project

Step 1: Add Firebase SDK Dependencies

Open your `pubspec.yaml` file and add the following dependencies:

```
dependencies:
  flutter:
    sdk: flutter
  firebase_core: latest_version
```

Step 2: Configure Firebase for Android

1. Open the `android/build.gradle` file and add the following line in the `dependencies` section:

```
buildscript {
  repositories {
    google()
    mavenCentral()
  }

  dependencies {
    classpath 'com.android.tools.build:gradle:7.3.1'
    classpath 'com.google.gms:google-services:4.4.2'
  }
}
```

2. Open the `android/app/build.gradle` file and add the following lines at the bottom of the file:

```
apply plugin: 'com.android.application'
apply plugin: 'com.google.gms.google-services'

dependencies {
  implementation platform('com.google.firebase:firebase-bom:32.6.0')
  implementation 'com.google.firebase:firebase-appcheck-debug:18.0.0'
}
```

Step 3: Initialize Firebase in Flutter

1. Open your Flutter project and edit your file `main.dart` to configure Firebase options.

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

```

class DefaultFirebaseOptions {
static FirebaseOptions get currentPlatform {
  if (kIsWeb) {
    throw UnsupportedError(
      'DefaultFirebaseOptions have not been configured for windows - '
      'you can reconfigure this by running the FlutterFire CLI again.',
    );
  }

  switch (defaultTargetPlatform) {
    case TargetPlatform.android:
      return android;

    case TargetPlatform.iOS:
      throw UnsupportedError(
        'DefaultFirebaseOptions have not been configured for iOS'
        'you can reconfigure this by running the FlutterFire CLI'
        'again.',
      );

    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 android = FirebaseOptions(
  apiKey: "API_KEY_HERE",
  appId: 'APP_ID_HERE',
  messagingSenderId: 'MESSAGING_ID_HERE',
  projectId: 'PROJECT_ID_HERE',
);
}

```

Exercises:

- Set up a Firebase project and register an Android and iOS app with it.
- Integrate Firebase into a new Flutter project and verify the integration by running the example code provided.