**Experiment No:06**

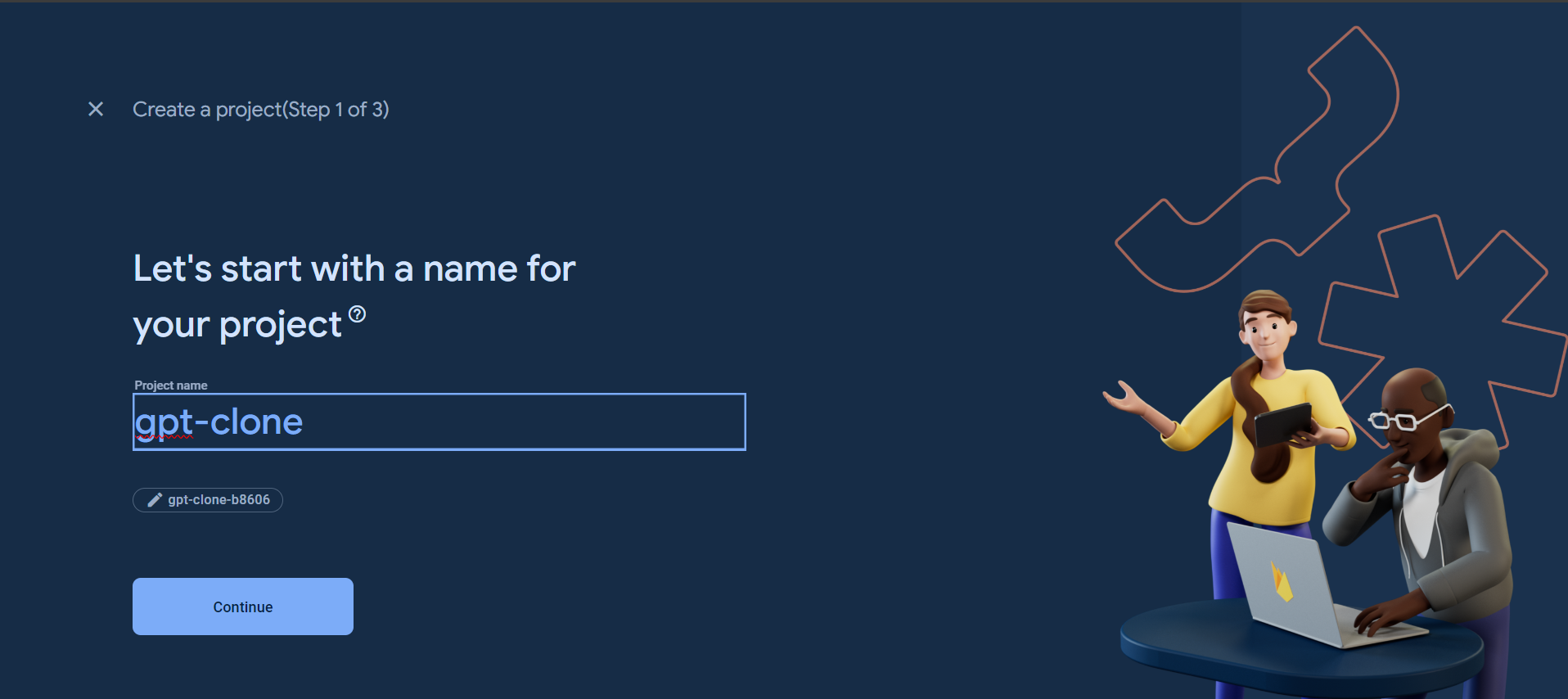
**Aim:How To Set Up Firebase with Flutter for iOS and Android Apps**

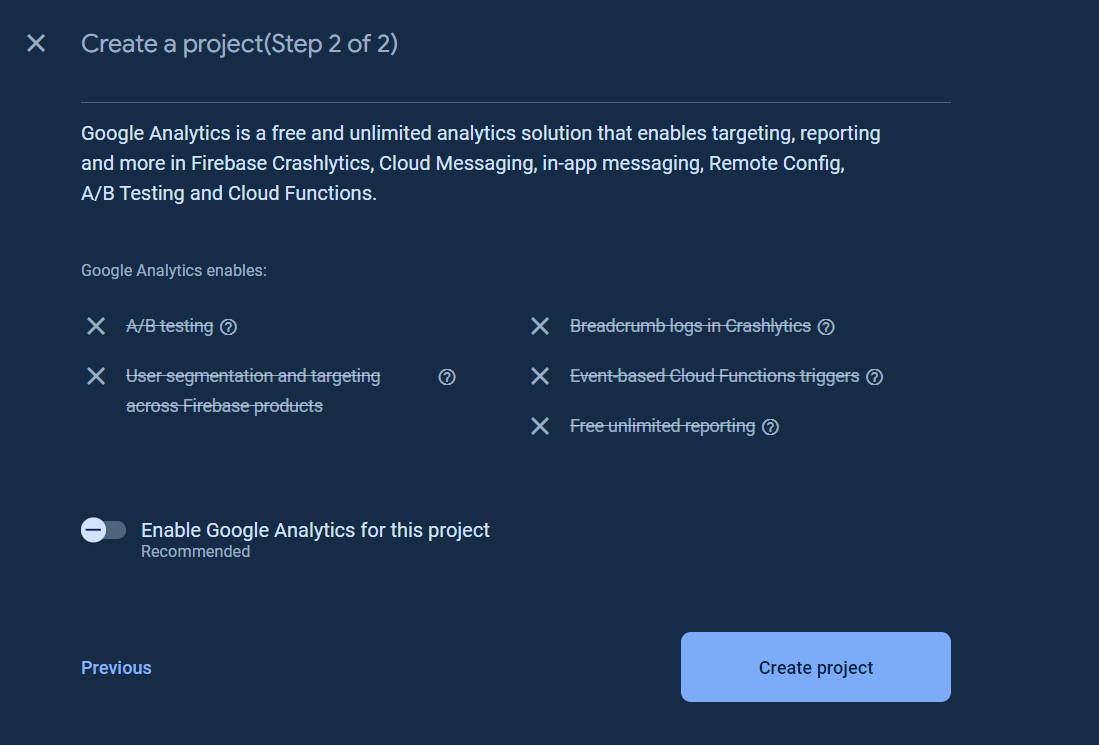
**Theory:**

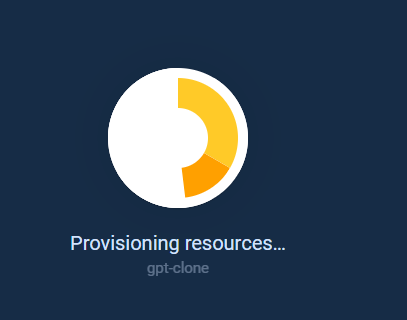
Firebase is a comprehensive platform provided by Google for developing mobile and web applications. It offers a wide range of features, including authentication, real-time database, cloud storage, hosting, and more, making it a popular choice for app development. Integrating Firebase with Flutter for both iOS and Android apps involves several steps:

**1. Create a Firebase Project:**

* Start by creating a new project on the Firebase console (console.firebase.google.com).
* Follow the prompts to set up your project, including providing a project name and enabling Google Analytics.

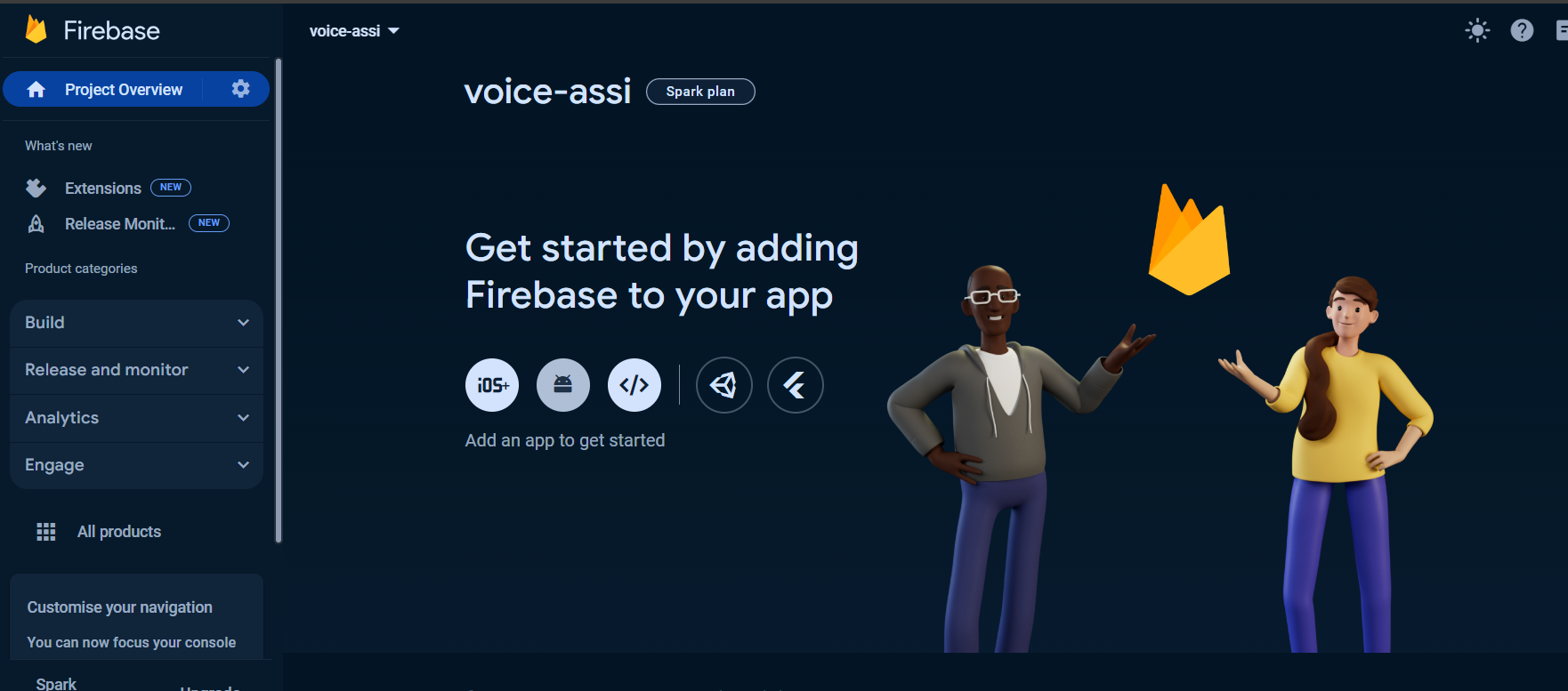


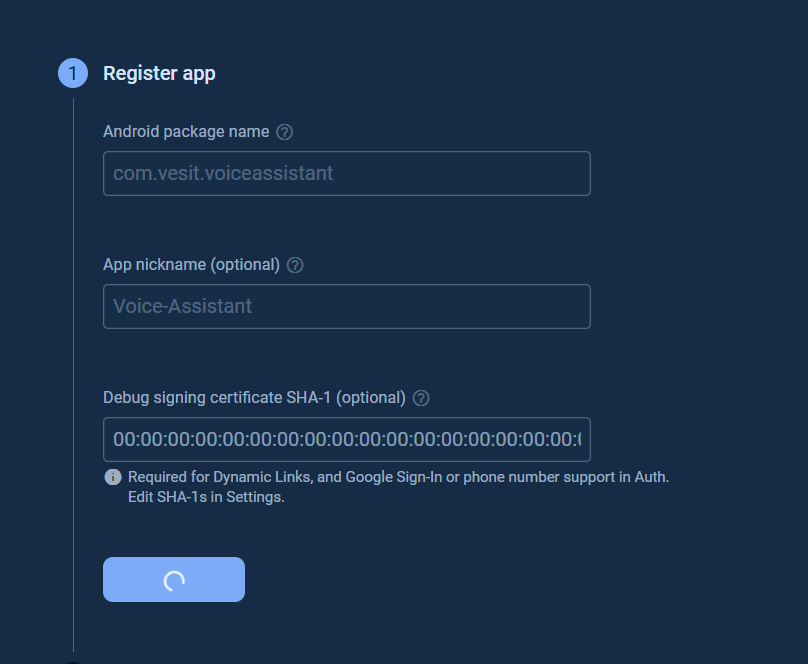


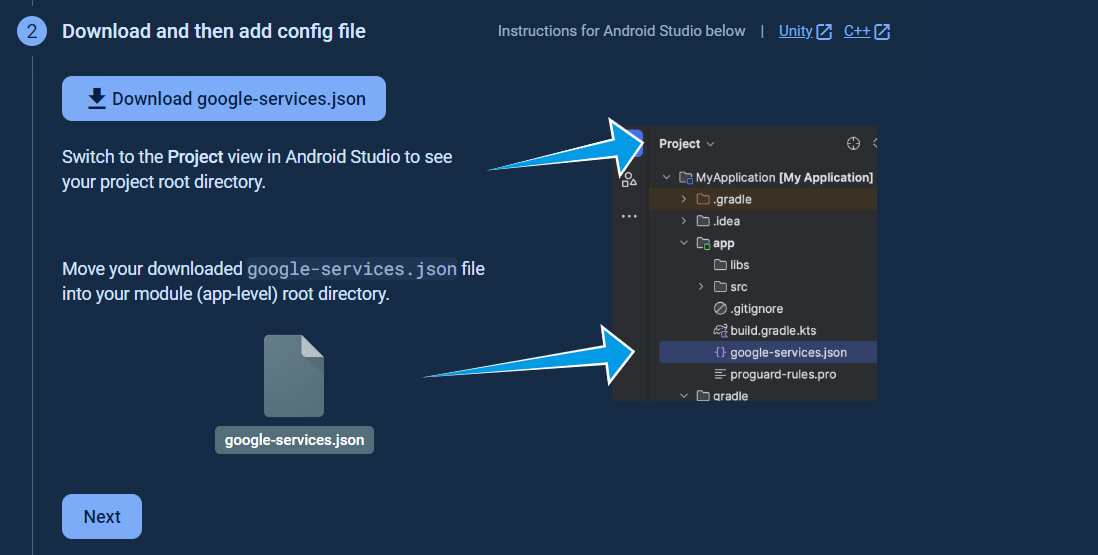


**2. Add Your App to Firebase:**

* After creating the project, add your Flutter app to it by clicking on the "Add App" button and selecting the appropriate platform (iOS or Android).
* Follow the instructions to register your app with Firebase. For Android, you'll need to provide your app's package name, and for iOS, you'll need to upload your app's GoogleService-Info.plist file.

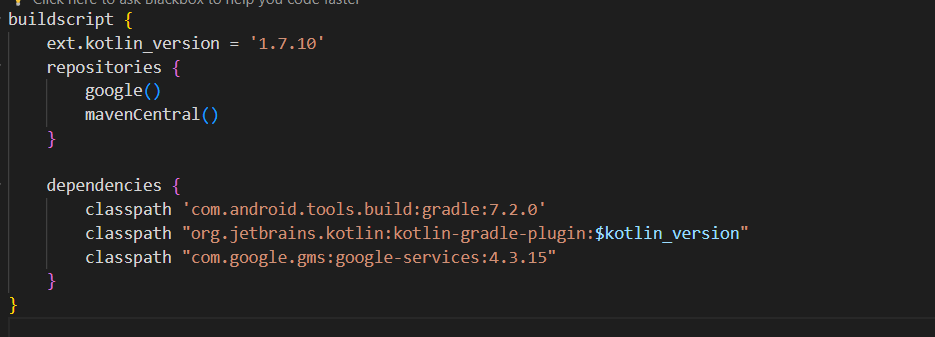


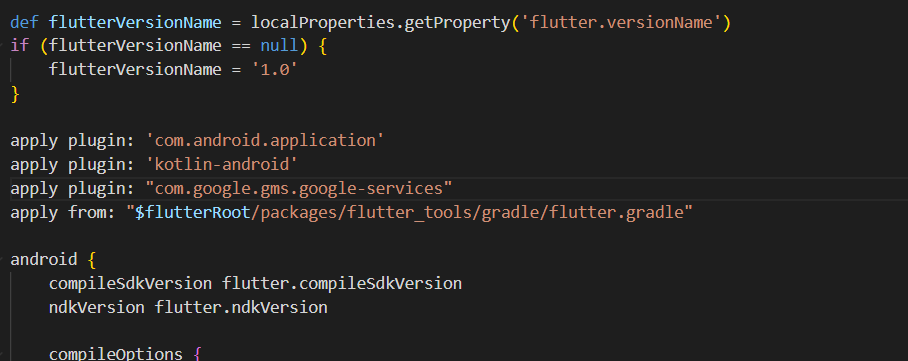




**3. Configure Firebase SDK:**

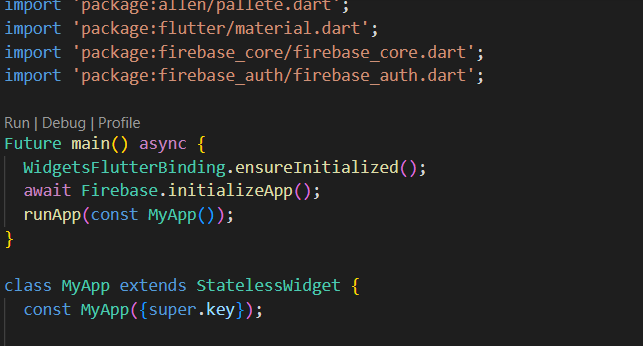
* For Android:
  + Add the Firebase SDK to your Flutter project by adding the required dependencies in your build.gradle file.
  + Update the android/build.gradle file to include the Google services classpath.
  + Add the google-services.json file to the android/app directory.
* For iOS:
  + Add the Firebase SDK to your Flutter project by adding the required dependencies in your Podfile.
  + Add the GoogleService-Info.plist file to the ios/Runner directory.





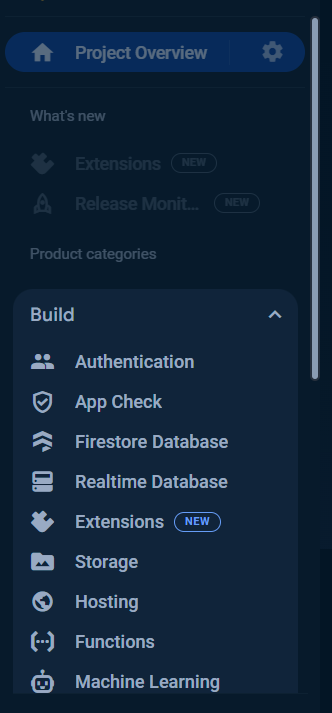
**4. Initialize Firebase:**

* In your Flutter app, initialize Firebase by importing the necessary Firebase libraries and calling Firebase.initializeApp() in your main() function.



**5. Add Firebase Services:**

* Depending on your app's requirements, add the Firebase services you need, such as Authentication, Firestore, Realtime Database, Cloud Storage, etc.
* Follow the documentation provided by Firebase to set up and configure each service according to your app's needs.

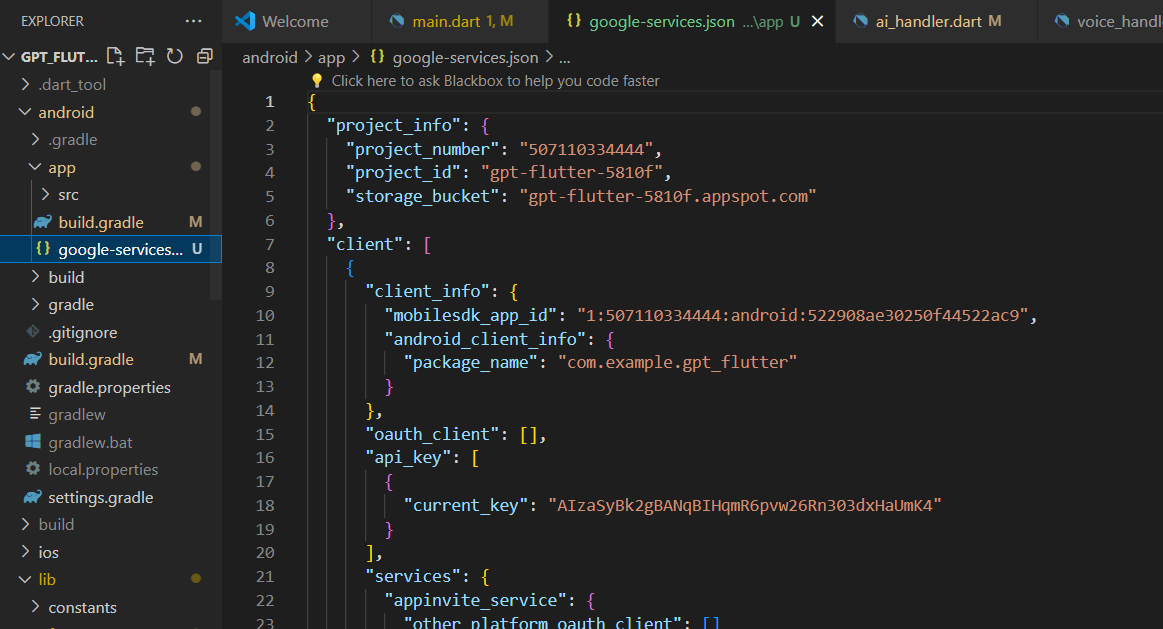


**6. Test Firebase Integration:**

* Verify that Firebase is correctly integrated into your Flutter app by testing various Firebase features and services.
* Use Firebase Authentication to authenticate users, Firestore or Realtime Database to store and retrieve data, and other Firebase services as required.

**7. Handle Platform-Specific Configurations (Optional):**

* Depending on your app's requirements, you may need to handle platform-specific configurations for Firebase services, such as handling deep links, push notifications, etc.

****

**Conclusion:**

**I have successfully understood and implemented ,How To Set Up Firebase with Flutter for iOS and Android Apps**