

MAD and PWA Lab 6

Name: Bhavesh Wadhvani

Class: D15A

Roll no:66

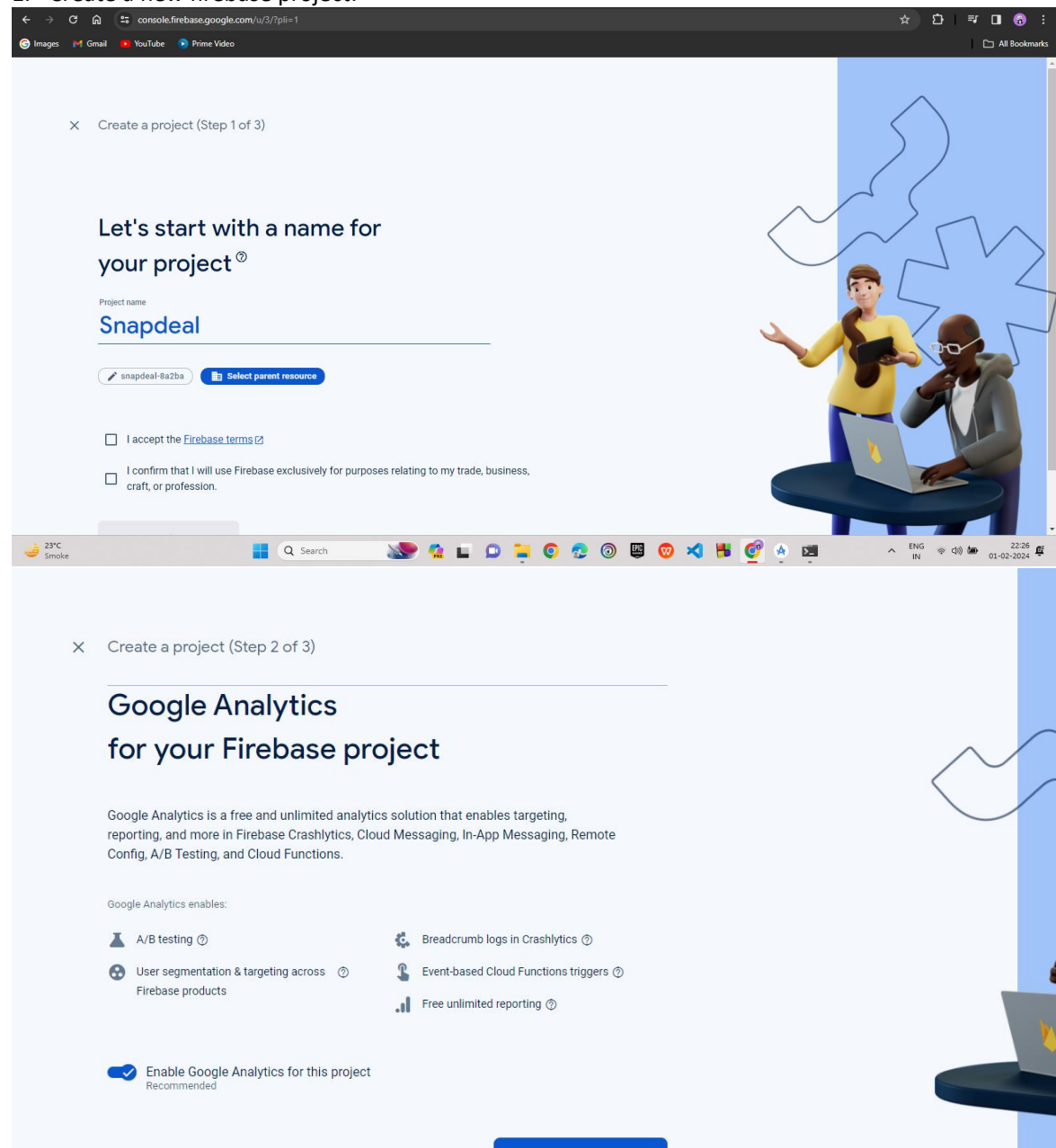
AIM: To connect flutter UI with firebase database.

THEORY:

Firebase is a great backend solution for anyone that wants to use authentication, databases, cloud functions, ads, and countless other features within an app.:

OUTPUT:


1. Create a new firebase project:




✕ Create a project (Step 3 of 3)

Configure Google Analytics

Choose or create a Google Analytics account ⓘ

 BhaveshWadhvani ▾

Automatically create a new property in this account ✎

Upon project creation, a new Google Analytics property will be created in your chosen Google Analytics account and linked to your Firebase project. This link will enable data flow between the products. Data exported from your Google Analytics property into Firebase is subject to the Firebase terms of service, while Firebase data imported into Google Analytics is subject to the Google Analytics terms of service. [Learn more](#) .

[Previous](#)

Create project



Snapdeal

✓ Your Firebase project is ready

Continue

2. DataBase Connection Part:

The screenshot displays a development environment. The top half shows the Firebase console for a project named 'Snapdeal'. The console's left sidebar includes navigation options like 'Project Overview', 'Extensions', and 'Product categories'. The main area of the console features a 'Get started by adding Firebase to your app' banner with illustrations of two people and a large yellow Firebase logo. Below this, there are icons for various platforms (iOS, Android, Web, etc.) and a 'Store and sync app data in milliseconds' section.

The bottom half of the image shows an IDE with two open files: 'main.dart' and 'build.gradle'. The 'build.gradle' file is currently selected and shows the following configuration:

```
31 sourceSets {
32     main.java.srcDirs += 'src/main/kotlin'
33 }
34
35 lintOptions {
36     disable 'InvalidPackage'
37 }
38
39 defaultConfig {
40     applicationId "com.bhaveshwadhvani.snapdeal"
41     minSdkVersion 16
42     targetSdkVersion 28
43     versionCode flutterVersionCode.toInteger()
44     versionName flutterVersionName
45     multiDexEnabled true
46 }
47
```

× Add Firebase to your Android app

✓ Register app

Android package name: com.bhaveshwadhvani.snapdeal

2 Download and then add config file

Instructions for Android Studio below | [Unity](#) [C++](#)

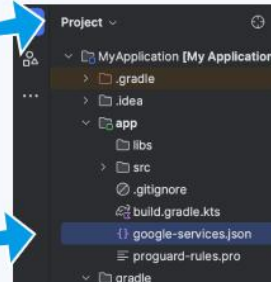
Download google-services.json

Switch to the Project view in Android Studio to see your project root directory.

Move your downloaded google-services.json file into your module (app-level) root directory.



Next



3 Add Firebase SDK

Download and then add config file

3 Add Firebase SDK

Instructions for Gradle | [Unity](#) | [C++](#)

★ Are you still using the `buildscript` syntax to manage plugins? Learn how to [add Firebase plugins](#) using that syntax.

1. To make the `google-services.json` config values accessible to Firebase SDKs, you need the Google services Gradle plugin.

☐ Kotlin DSL (`build.gradle.kts`) ☒ Groovy (`build.gradle`)

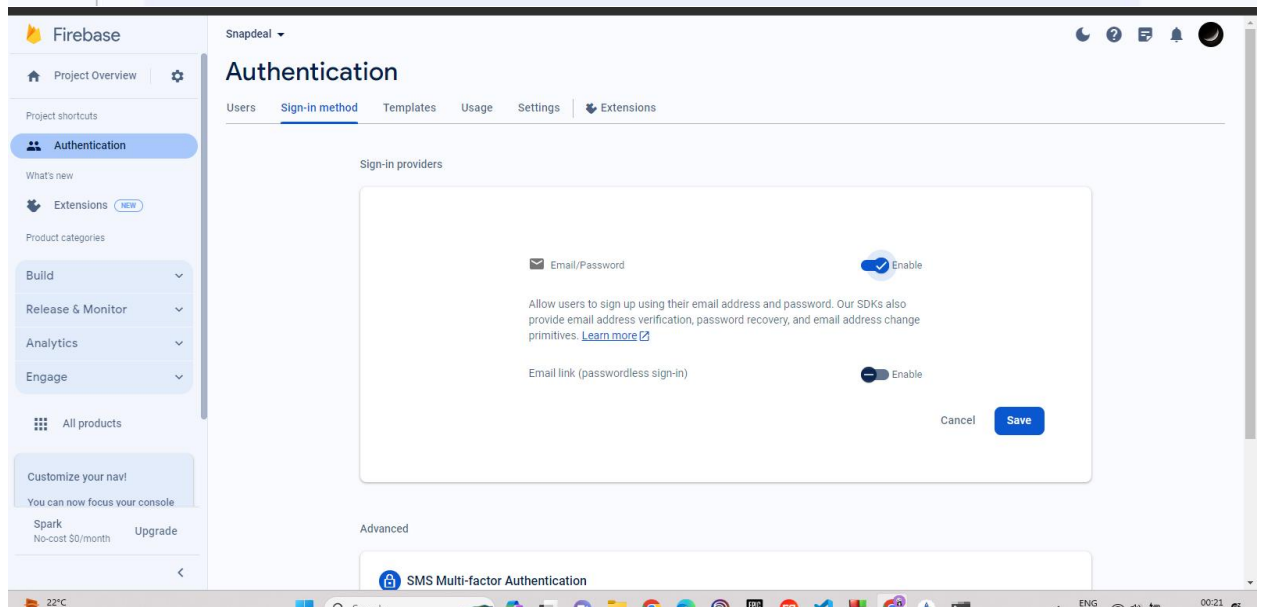
Add the plugin as a dependency to your **project-level** `build.gradle` file:

Root-level (project-level) Gradle file (`<project>/build.gradle`):

```
plugins {  
    // ...  
  
    // Add the dependency for the Google services Gradle plugin  
    id 'com.google.gms.google-services' version '4.4.0' apply false  
}
```

2. Then, in your **module (app-level)** `build.gradle` file, add both the `google-services` plugin and any Firebase SDKs that you want to use in your app:

Module (app-level) Gradle file (`<project>/<app-module>/build.gradle`):



What's new

Extensions NEW

Product categories

Build

Release & Monitor

Analytics

Engage

All products

Customize your nav!

You can now focus your console experience by customizing your navigation

Learn more Got it

Spark

No-cost \$0/month Upgrade

Snapdeal

Create database

Set name and location Secure rules

After you define your data structure, you will need to write rules to secure your data.
[Learn more](#)

☒ Start in production mode

Your data is private by default. Client read/write access will only be granted as specified by your security rules.

☐ Start in test mode

Your data is open by default to enable quick setup. However, you must update your security rules within 30 days to enable long-term client read/write access.

```
rules_version = '2';
service cloud.firestore {
  match /databases/{database}/documents {
    match /(document=**) {
      allow read, write: if false;
    }
  }
}
```

All third party reads and writes will be denied

Cancel Create

Watch later Share

22°C Haze

Search

ENG IN 00:23 02-02-2024

Storage

Store and retrieve user-generated files like

Set up Cloud Storage

Secure rules for Cloud Storage Set Cloud Storage location

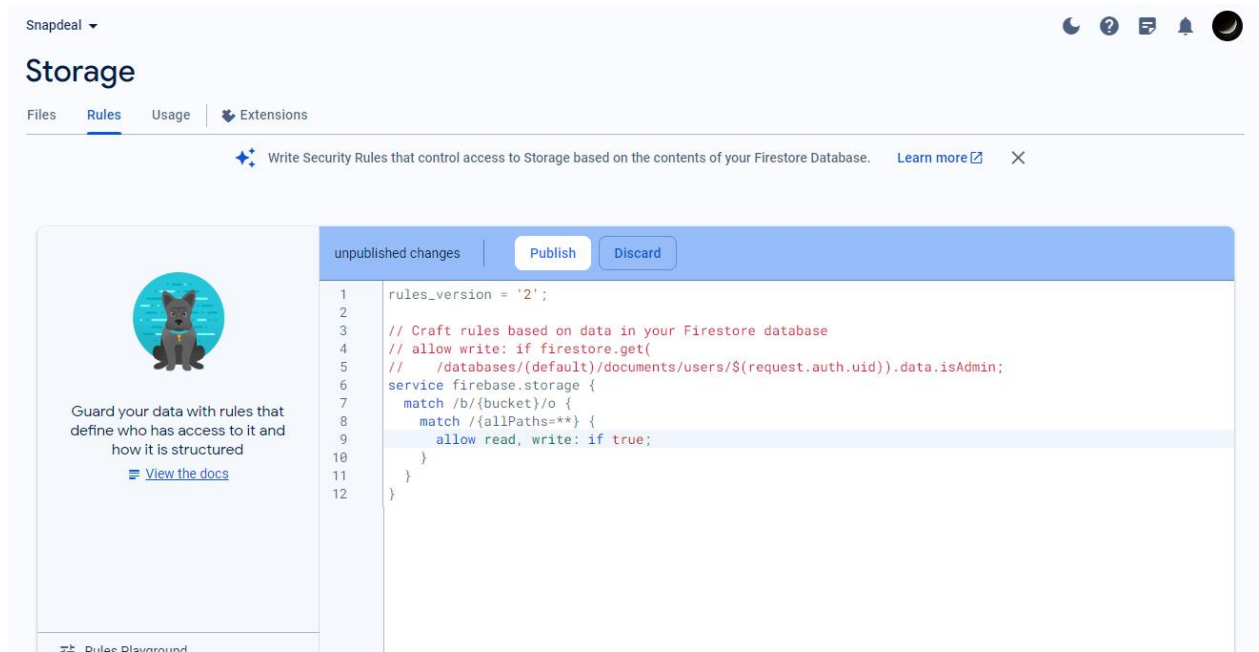
Your location setting is where your default Cloud Storage bucket and its data will be stored.

Cloud Storage location

nam5 (us-central)

Blaze Plan customers can choose other locations for additional buckets [click here to upgrade](#) Cancel Done

Introducing Cloud Storage for Firebase



:

Google-services.json

```
{
  "project_info": {
    "project_number": "560299755679",
    "project_id": "bhaveshs-826b8",
    "storage_bucket": "bhaveshs-826b8.appspot.com"
  },
  "client": [
    {
      "client_info": {
        "mobilesdk_app_id":
"1:560299755679:android:7701dcc4e602a6bc493aaa",
        "android_client_info": {
          "package_name": "com.furqan.flutter_ecommerce_ui_kit"
        }
      },
      "oauth_client": [],
      "api_key": [
        {
          "current_key": "AlzaSyCIVw9QJIN2fi3_7thVm7HdmA__SWTYMtw"
        }
      ],
      "services": {
        "appinvite_service": {
          "other_platform_oauth_client": []
        }
      }
    }
  ],
  "configuration_version": "1"
}
```

Add dependencies related to firebase in pubsec.yaml:

name: olx_clone
description: "A new Flutter project."

version: 1.0.0+1

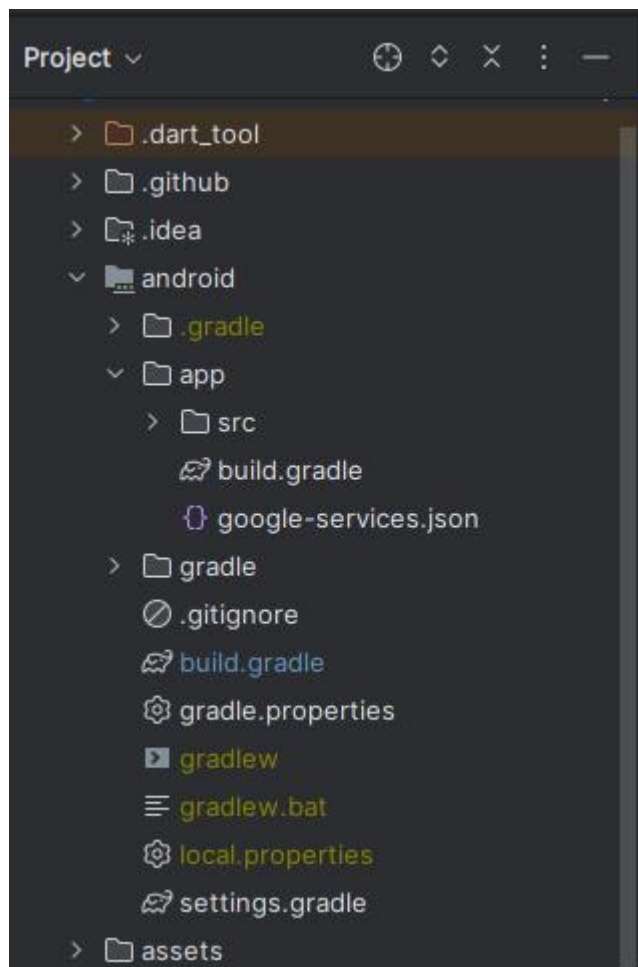
environment:
 sdk: ">=2.12.0 <3.0.0"

dependencies:
 flutter:
 sdk: flutter
 flutter_localizations:
 sdk: flutter
 firebase_core: ^2.25.4
 firebase_auth: ^4.17.4
 cloud_firestore: ^4.2.0

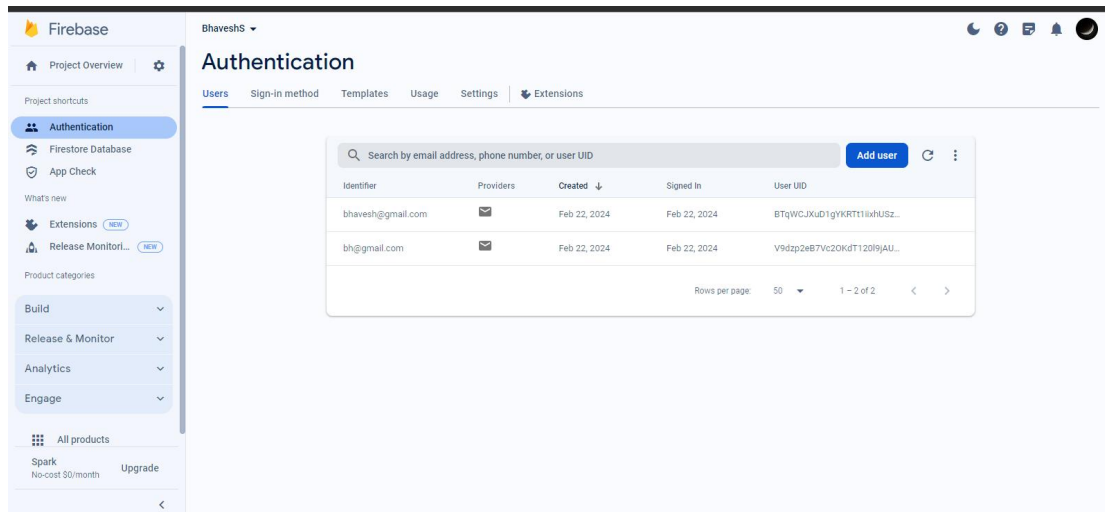
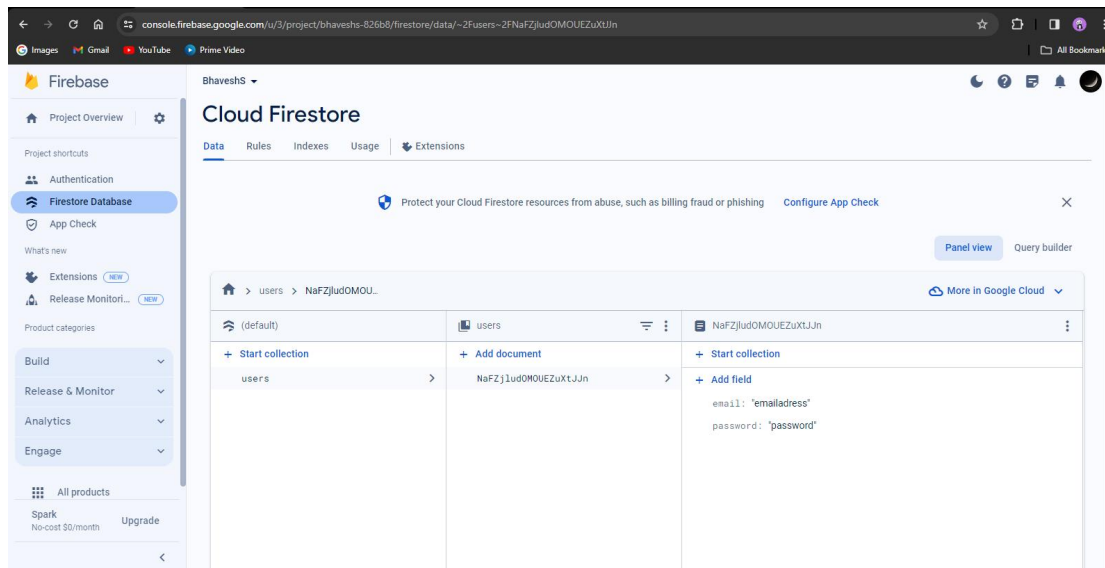
intl: 0.18.1
carousel_slider: ^4.0.0
cached_network_image: ^3.0.0
flutter_rating_stars: ^1.0.3+4
provider: ^6.0.3
flutter_secure_storage: ^5.0.2
fluttertoast: ^8.0.7

The following adds the Cupertino Icons font to your application.
Use with the CupertinoIcons class for iOS style icons.
cupertino_icons: ^1.0.2

dev_dependencies:
 flutter_test:
 sdk: flutter

**Build.gradle:**

```
dependencies {  
    classpath 'com.google.gms:google-services:4.4.1'  
    classpath 'com.android.tools.build:gradle:7.0.0'  
    classpath "org.jetbrains.kotlin:kotlin-gradle-plugin:$kotlin_version"  
    classpath 'com.google.gms:google-services:4.3.15'  
}  
}
```



CONCLUSION:

We have implemented navigation ,routing and gestures in the pages.