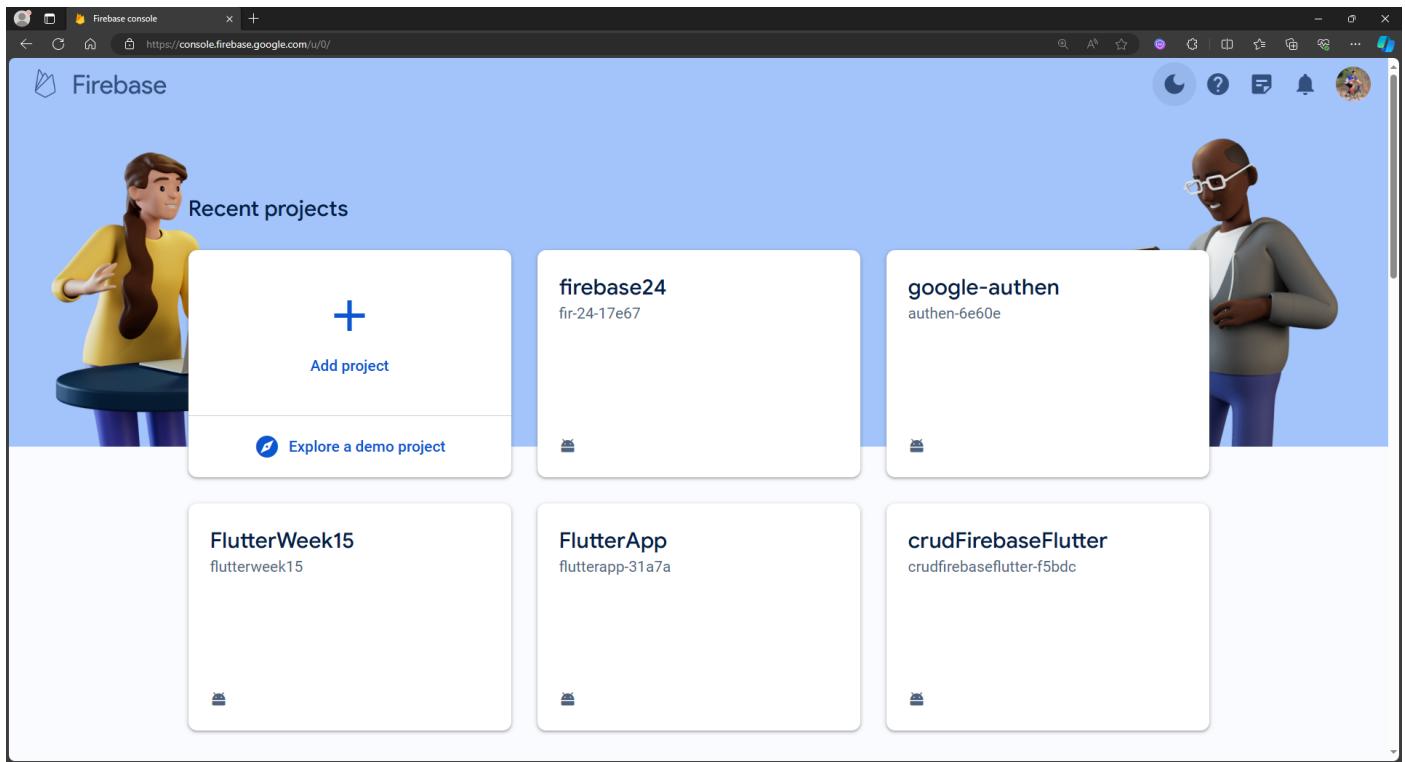


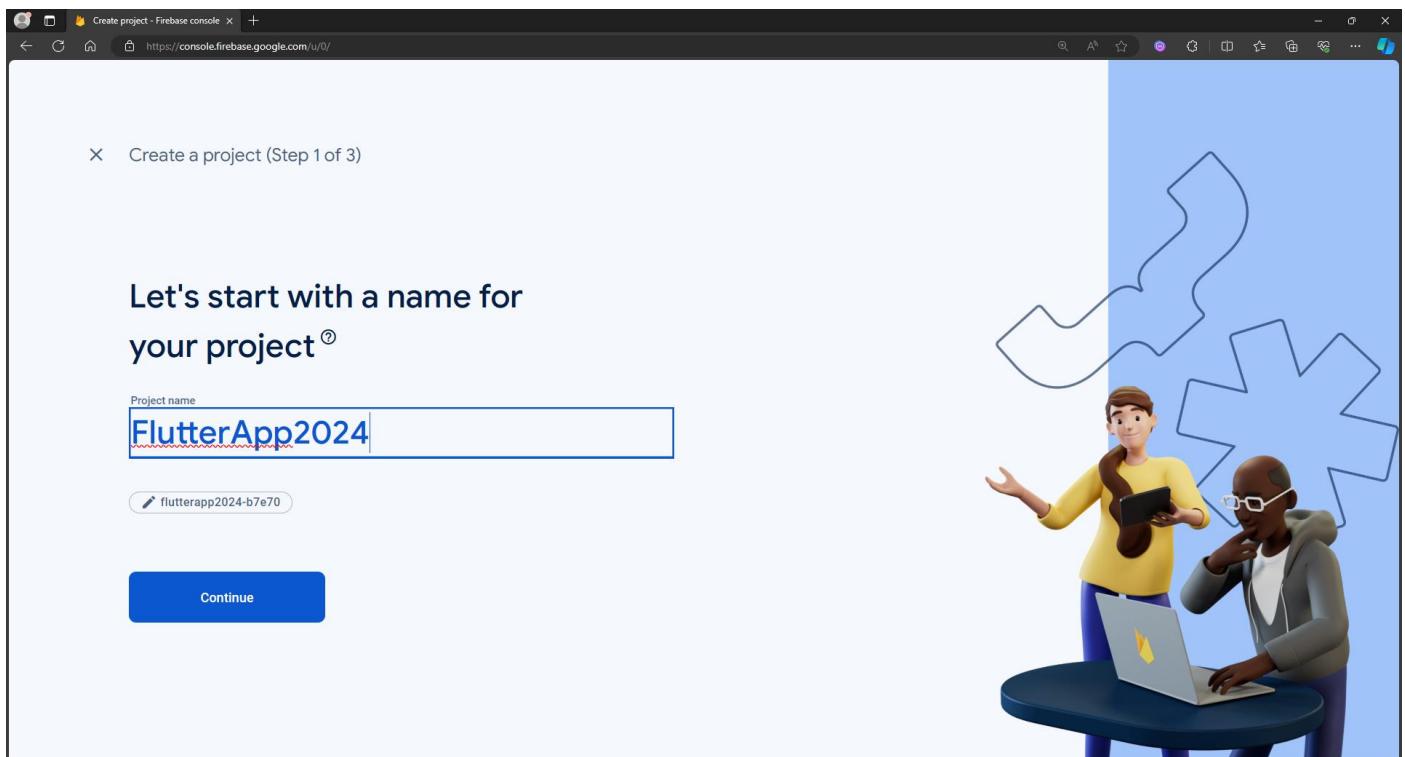
Flutter and Firestore Database: CRUD [2024]

Setting up a new Firebase project

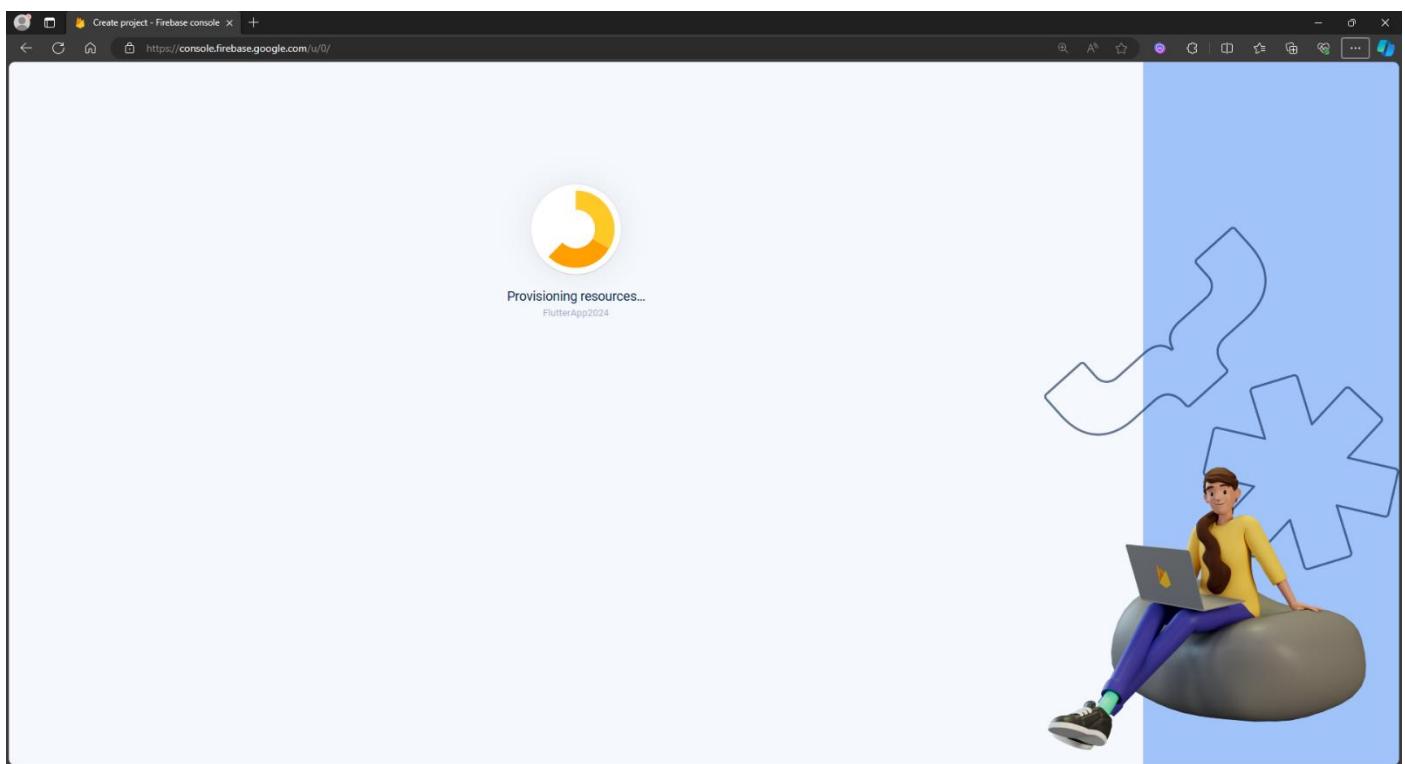
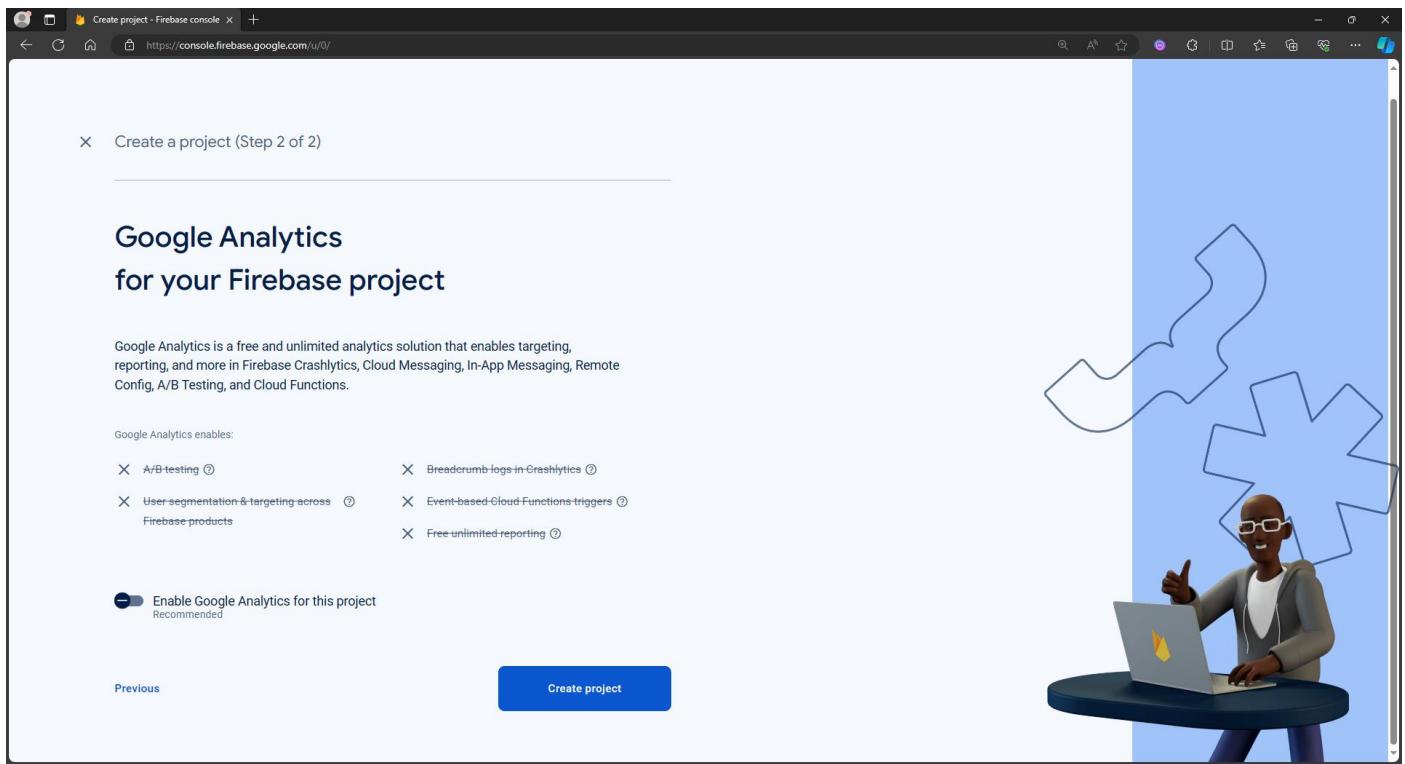
1. Launch your web browser, go to <https://console.firebaseio.google.com/>, login, and click on the “+” button.



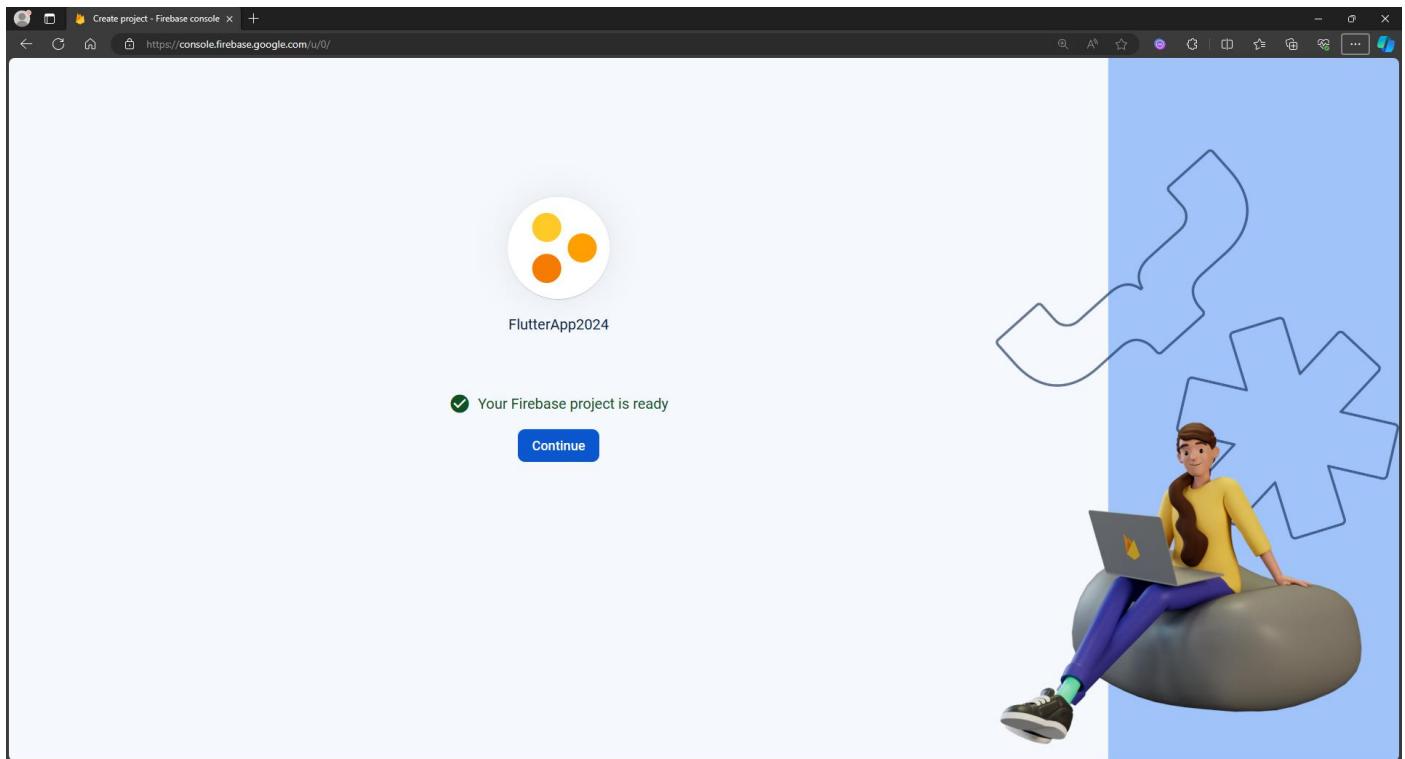
2. Give your project a name then click on the “Continue” button:



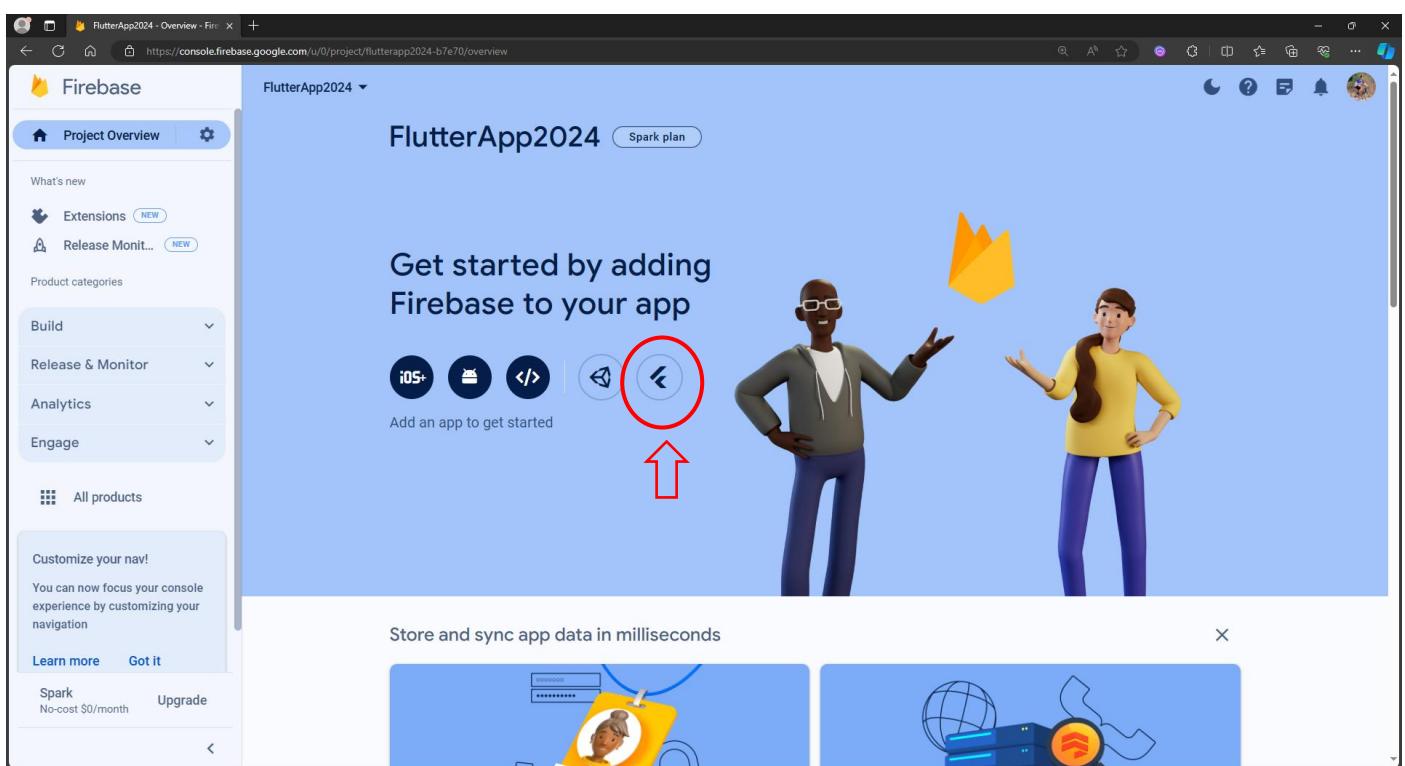
3. To keep this tutorial short and clean, we will disable Google Analytics but you can reenable it later as you want. Click on the “Create project” button to initialize your project.



4. It will take a while to complete.

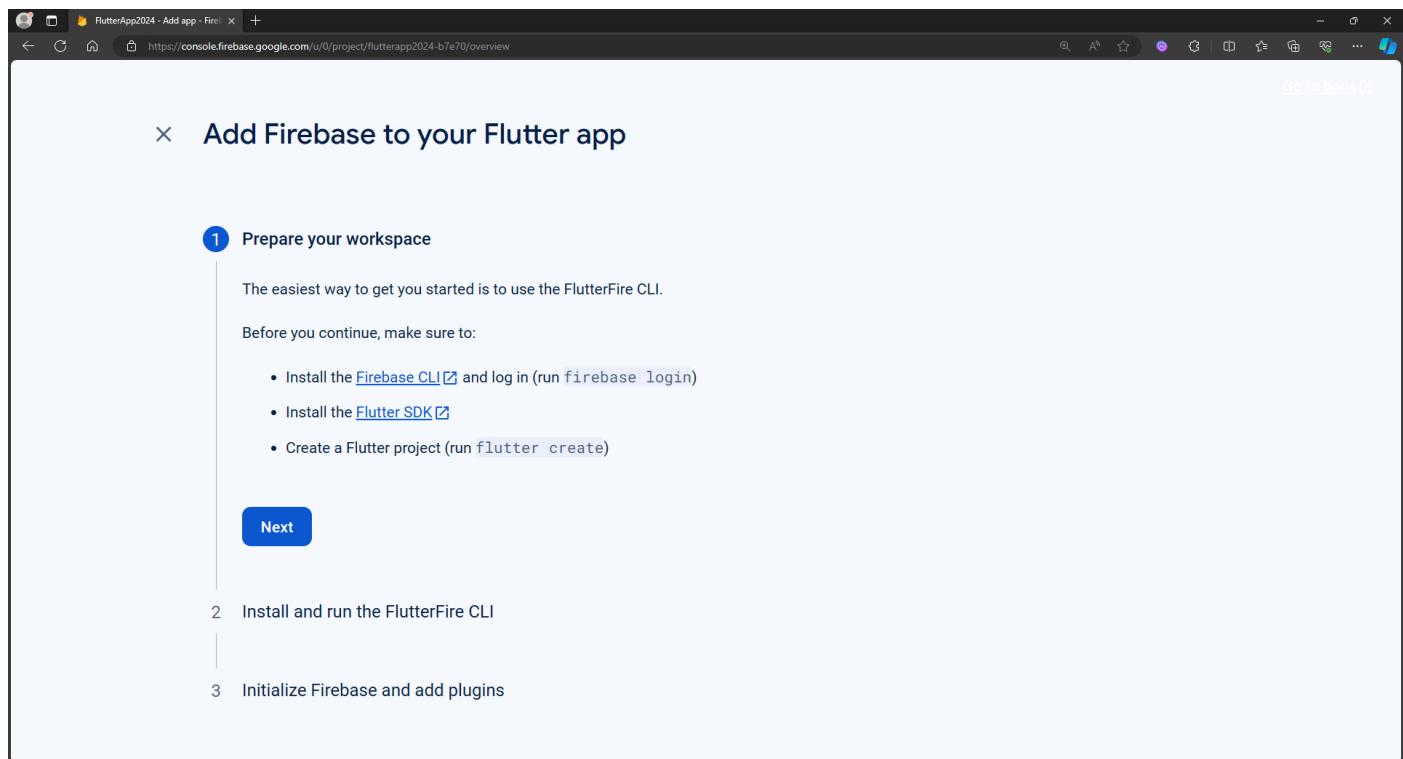


Here's the project's dashboard and Adding Firebase to Your App with Flutter



Step for Add Firebase to Your App with Flutter

1. Prepare your Workspace



1 Prepare your workspace

The easiest way to get you started is to use the FlutterFire CLI.

Before you continue, make sure to:

- Install the [Firebase CLI](#) and log in (run `firebase login`)
- Install the [Flutter SDK](#)
- Create a Flutter project (run `flutter create`)

[Next](#)

2 Install and run the FlutterFire CLI

3 Initialize Firebase and add plugins

Step : Install [Firebase CLI](#) (*windows)

To use `npm` (the Node Package Manager) to install the Firebase CLI, follow these steps:

1. Install [Node.js](#) using [nvm-windows](#) (the Node Version Manager). Installing Node.js automatically installs the `npm` command tools.

★ Note: The Firebase CLI requires [Node.js v18.0.0 or later](#). Some Firebase features might require specific versions of Node.js, so check each Firebase product's getting started page for any specific Node.js requirements.

2. Install the Firebase CLI via `npm` by running the following command: *** install nodejs**

```
$ npm install -g firebase-tools
```

This command enables the globally available `firebase` command.

★ Note: If the `npm install -g firebase-tools` command fails, you might need to [change npm permissions](#).

3. Continue to [log in and test the CLI](#).

Command for Install Firebase CLI

C:\> **npm install -g firebase-tools**

```
Microsoft Windows [Version 10.0.19045.4170]
(c) Microsoft Corporation. All rights reserved.

D:\2566-2\Mobile APP 2566\MobileApp2024\firebaselab>code .

D:\2566-2\Mobile APP 2566\MobileApp2024\firebaselab>npm install -g firebase-tools
[ ] - reify:@google-cloud/pubsub: timing reifyNode:node_modules/firebase-tools/node_modules/exegesis Completed in 264
```

Login and Test firebase CLI

[Log in and test the Firebase CLI](#)

After installing the CLI, you must authenticate. Then you can confirm authentication by listing your Firebase projects.

1. Log into Firebase using your Google account by running the following command:

```
$ firebase login
```

This command connects your local machine to Firebase and grants you access to your Firebase projects.

★ **Note:** The `firebase login` command opens a web page that connects to `localhost` on your machine. If you're using a remote machine and don't have access to `localhost`, run the command with the flag `--no-localhost`.

★ **Note:** You can also [use the Firebase CLI with CI systems](#).

2. Test that the CLI is properly installed and accessing your account by listing your Firebase projects. Run the following command:

```
$ firebase projects:list
```

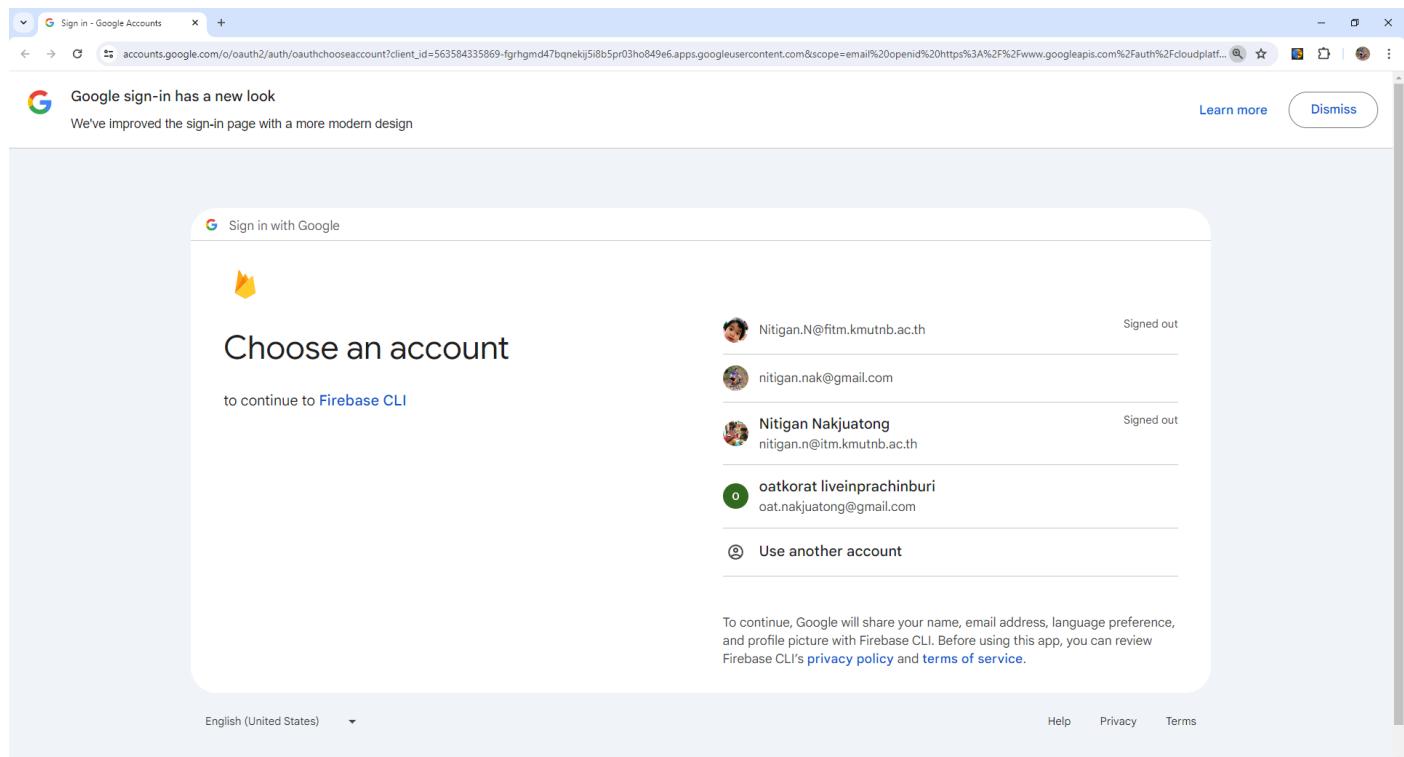
The displayed list should be the same as the Firebase projects listed in the [Firebase console](#).

C:\> firebase login and type Y

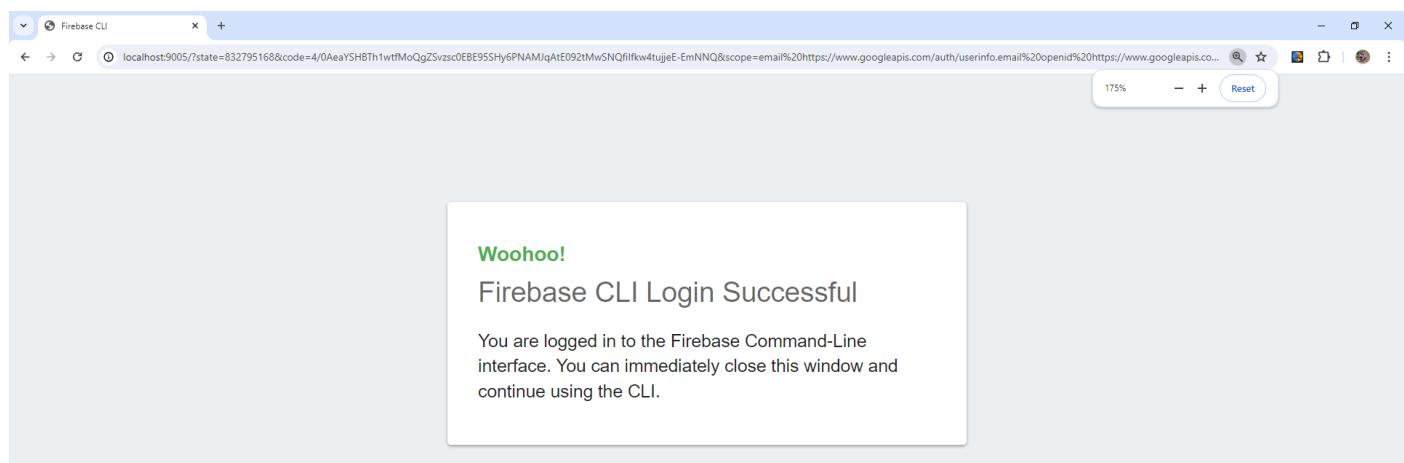
```
Select C:\WINDOWS\system32\cmd.exe - "node" "C:\Users\ADMINI\AppData\Roaming\npm\node_modules\firebase-tools\lib\bin\firebase.js" login
D:\2566-2\Mobile APP 2566\MobileApp2024\firebaselab>firebase login
i Firebase optionally collects CLI and Emulator Suite usage and error reporting information to help improve our products. Data is collected in accordance with Google's privacy policy (https://policies.google.com/privacy) and is not used to identify you.
? Allow Firebase to collect CLI and Emulator Suite usage and error reporting information? (Y/n)
```

```
Select C:\WINDOWS\system32\cmd.exe - "node" "C:\Users\ADMINI\AppData\Roaming\npm\node_modules\firebase-tools\lib\bin\firebase.js" login
D:\2566-2\Mobile APP 2566\MobileApp2024\firebaselab>firebase login
i Firebase optionally collects CLI and Emulator Suite usage and error reporting information to help improve our products. Data is collected in accordance with Google's privacy policy (https://policies.google.com/privacy) and is not used to identify you.
? Allow Firebase to collect CLI and Emulator Suite usage and error reporting information? yes
i To change your data collection preference at any time, run firebase logout and log in again.
Visit this URL on this device to log in:
https://accounts.google.com/o/oauth2/auth?client\_id=563584335869-fgrhgm47bqneki5j5i8b5pr03ho849e6.apps.googleusercontent.com&scope=email&openid%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fc1oudplatform&projects.readonly%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Ffirebase%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fc1oud-platform&response\_type=code&state=832795168&redirect\_uri=http%3A%2F%2Flocalhost%3A9005
Waiting for authentication...
```

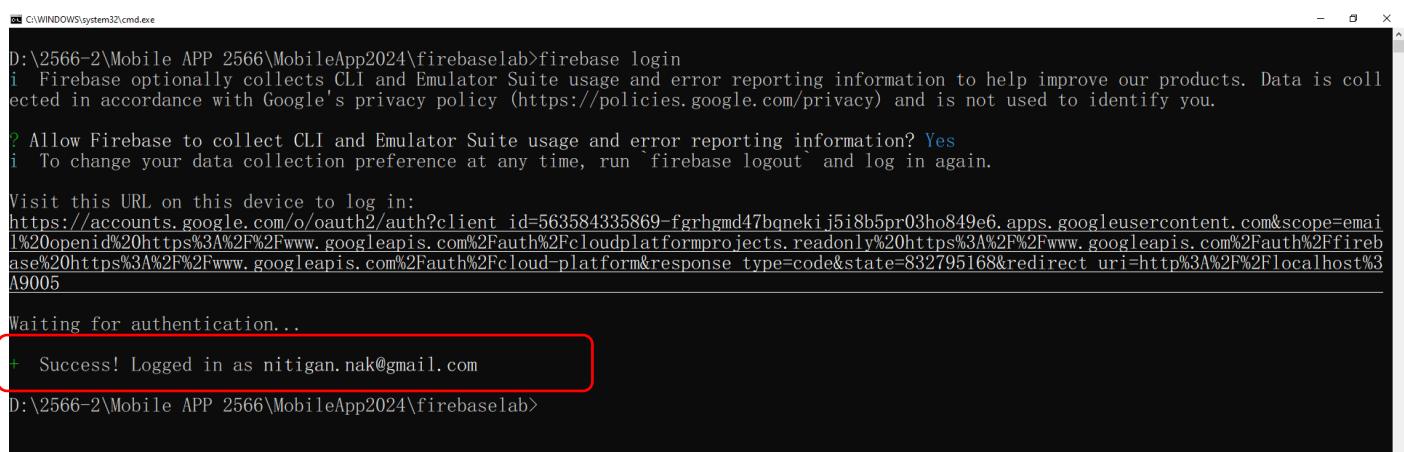
Choose Account



Click Continue and Allow

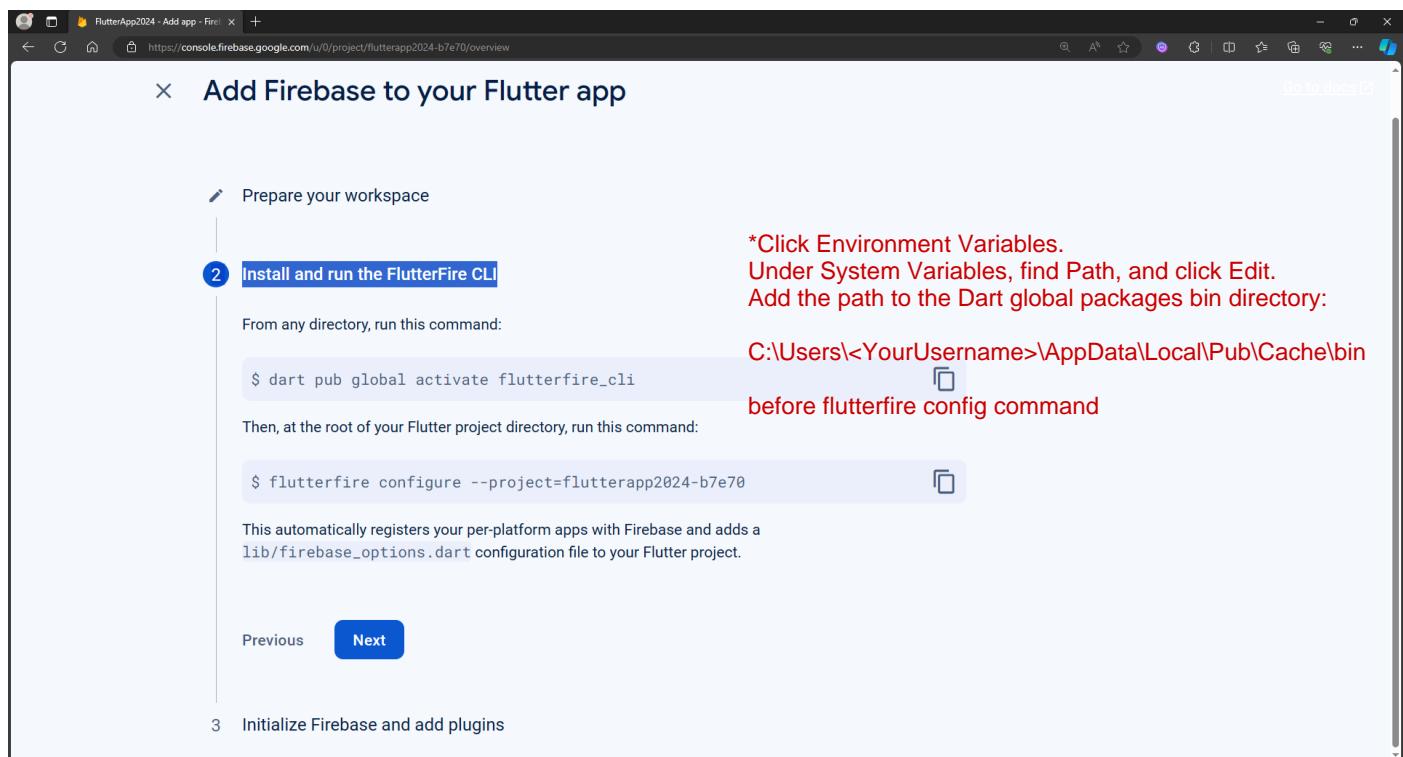


Click enter CLI



2. Install and run FlutterFire CLI

* install Git



2 Install and run the FlutterFire CLI

From any directory, run this command:

```
$ dart pub global activate flutterfire_cli
```

Then, at the root of your Flutter project directory, run this command:

```
$ flutterfire configure --project=flutterapp2024-b7e70
```

This automatically registers your per-platform apps with Firebase and adds a lib.firebaseio_options.dart configuration file to your Flutter project.

Previous Next

3 Initialize Firebase and add plugins

C:\> dart pub global activate flutterfire_cli



```
dart pub global activate flutterfire_cli
Package flutterfire_cli is currently active at version 0.2.7.
The package flutterfire_cli is already activated at newest available version.
To recompile executables, first run `dart pub global deactivate flutterfire_cli`.
Installed executable flutterfire.
Activated flutterfire_cli 0.2.7.
```

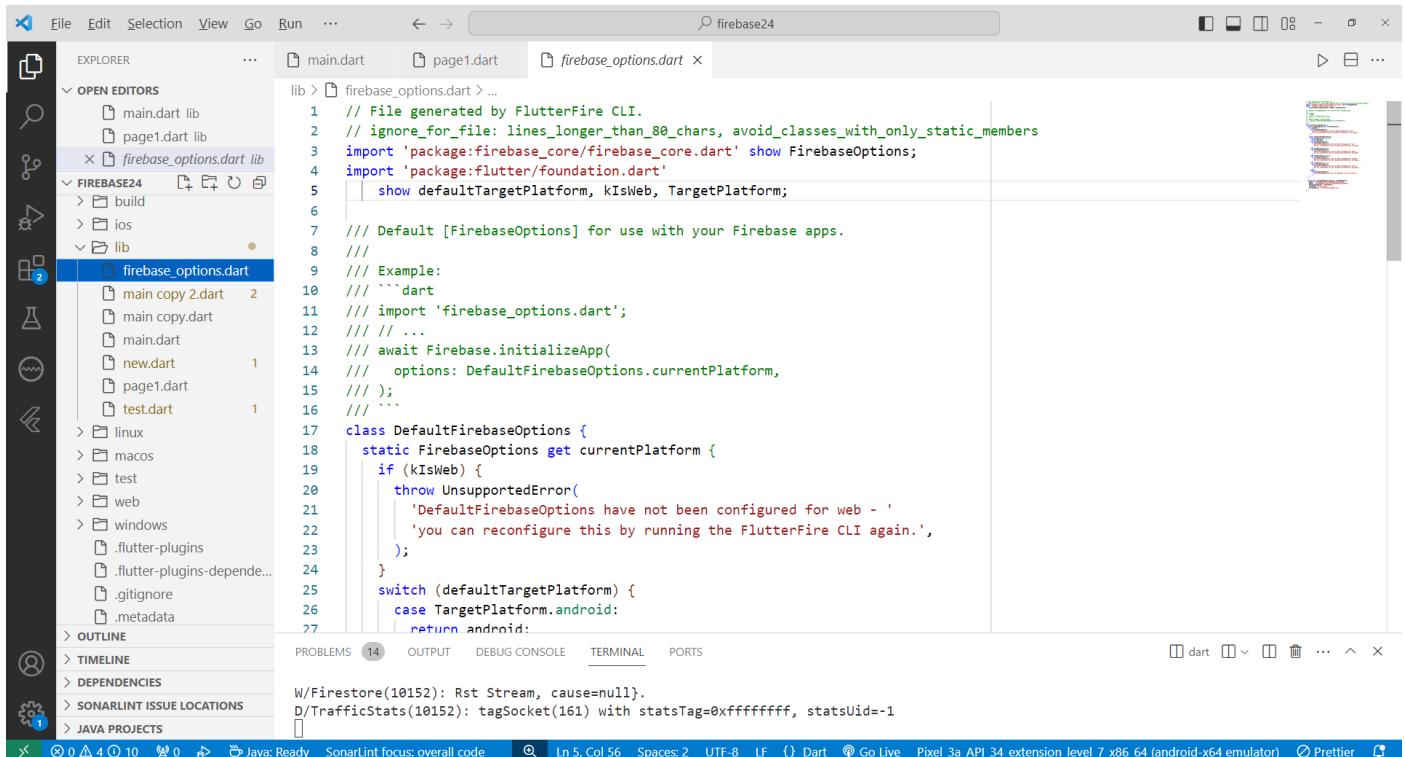
C:\> flutterfire configure --project=flutterapp2024-b7e70 (use your project)



```
flutterfire configure --project=flutterapp2024-b7e70
i Found 13 Firebase projects. Selecting project flutterapp2024-b7e70.
? Which platforms should your configuration support (use arrow keys & space to select)? >
✓ android
```

** select your project's platform

This automatically registers your per-platform apps with Firebase and adds a `lib/firebase_options.dart` configuration file to your Flutter project.

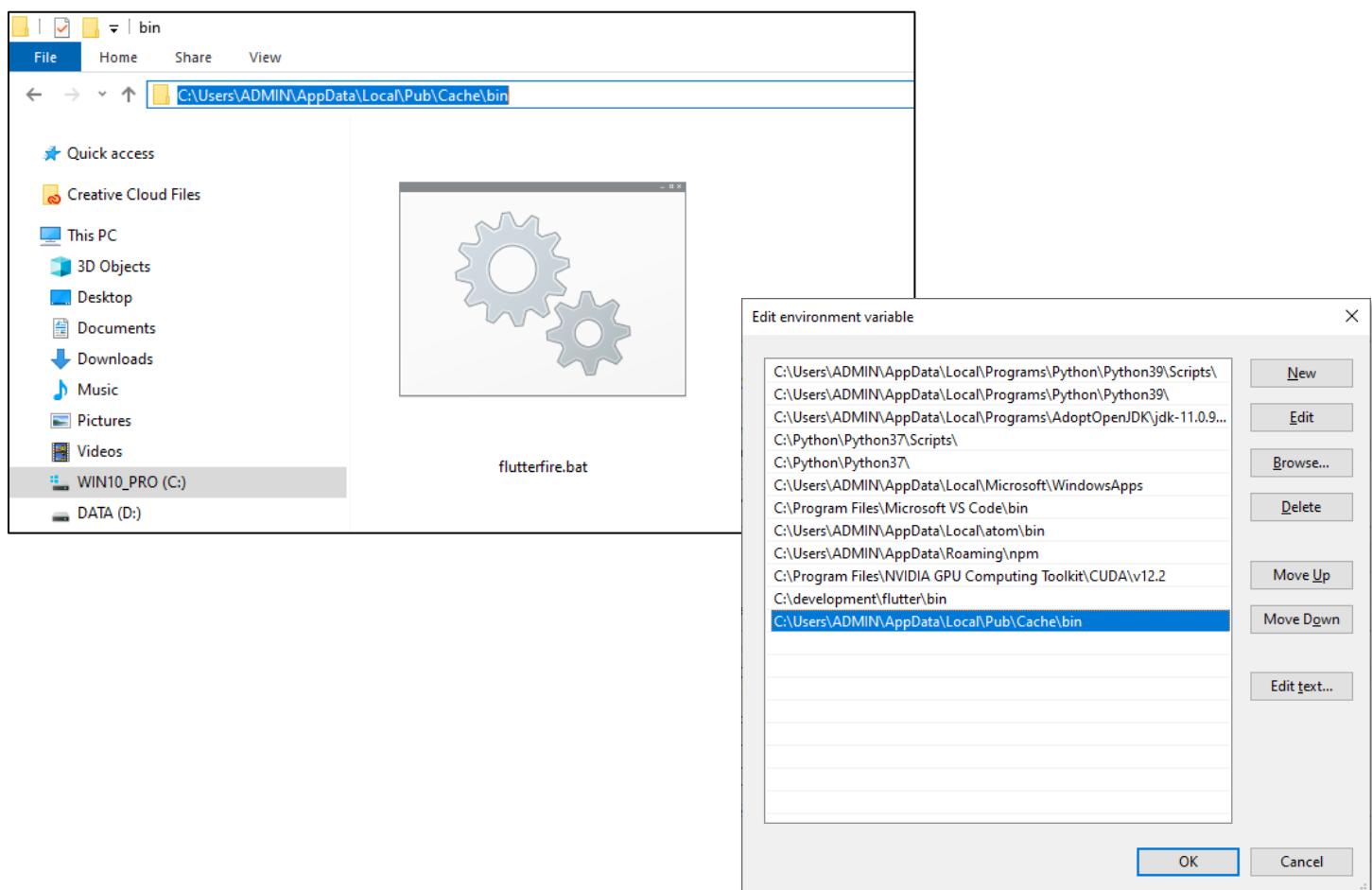


The screenshot shows the VS Code interface with the following details:

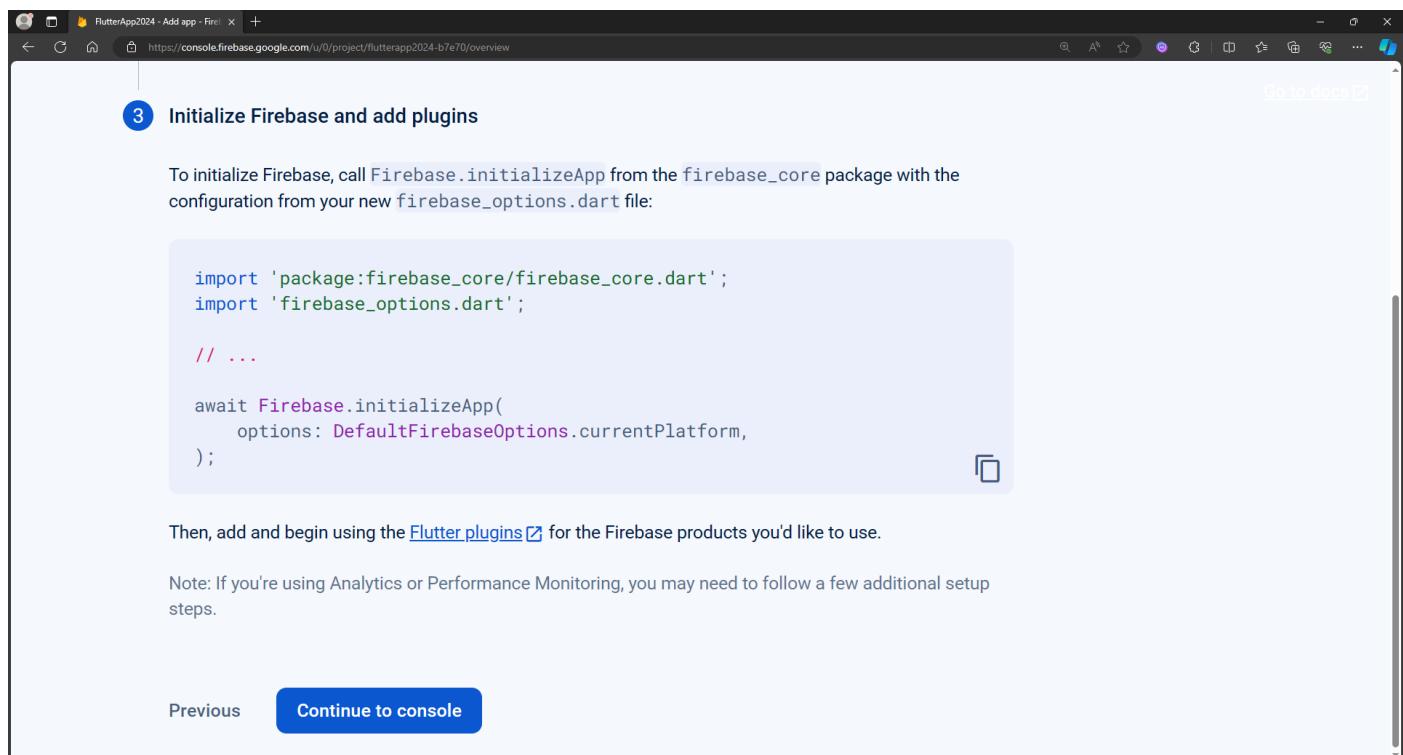
- File Explorer:** Shows the project structure with files like `main.dart`, `page1.dart`, and `firebase_options.dart`.
- Editor:** Displays the content of `firebase_options.dart`. The code is generated by the FlutterFire CLI and handles the configuration of FirebaseOptions based on the target platform (web, android, or iOS).
- Bottom Status Bar:** Shows Java: Ready, SonarLint focus: overall code, and various terminal and file-related icons.

** If cannot run flutterfire configure , go to set Path : where flutterfire.bat

Example with C:\Users\ADMIN\AppData\Local\Pub\Cache\bin



3. Initialize Firebase and add plugins

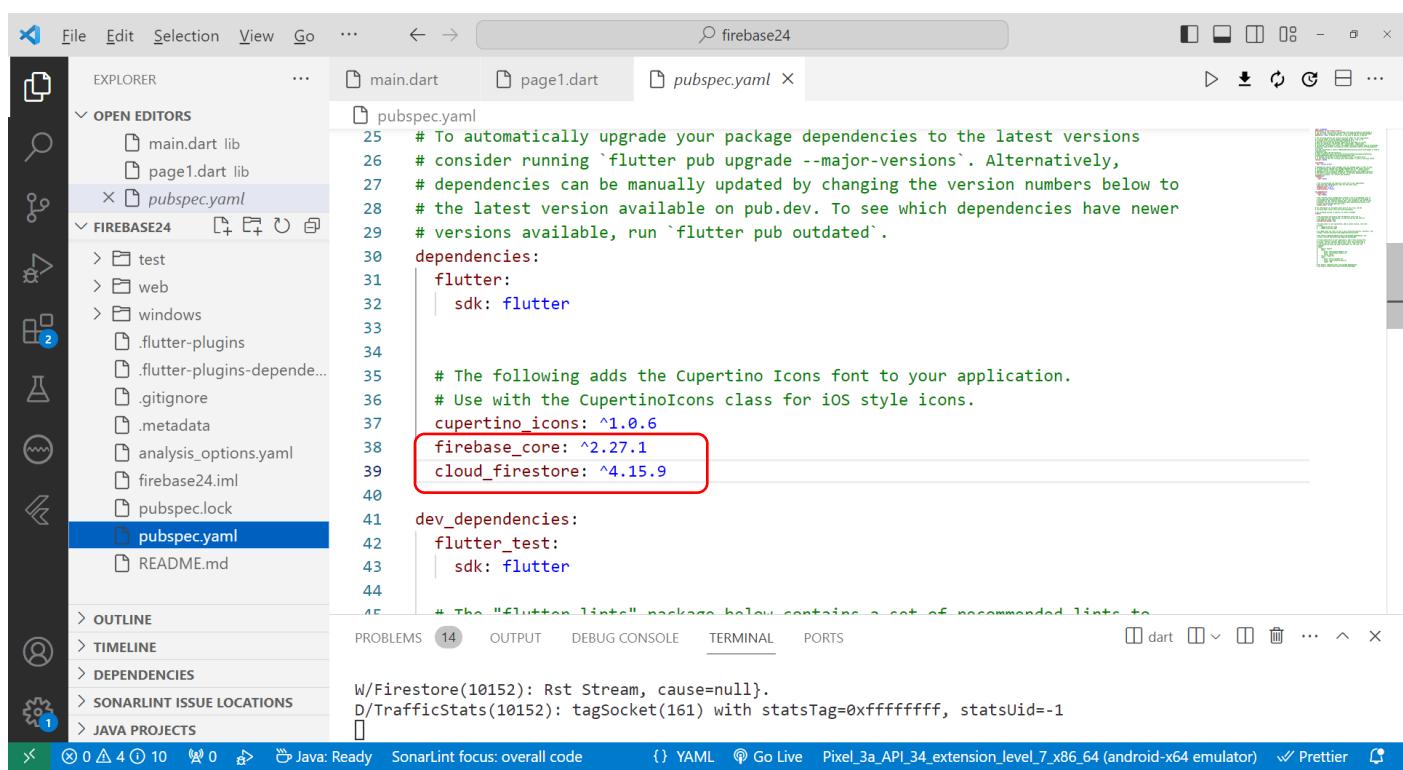


The screenshot shows the Firebase console with the URL <https://console.firebaseio.google.com/u/0/project/flutterapp2024-b7e70/overview>. The page is titled '3 Initialize Firebase and add plugins'. It contains instructions and code snippets for initializing Firebase and adding plugins. Below the code, it says 'Then, add and begin using the [Flutter plugins](#) for the Firebase products you'd like to use.' and 'Note: If you're using Analytics or Performance Monitoring, you may need to follow a few additional setup steps.' At the bottom, there are 'Previous' and 'Continue to console' buttons.

```
import 'package:firebase_core/firebase_core.dart';
import 'firebase_options.dart';

// ...

await Firebase.initializeApp(
    options: DefaultFirebaseOptions.currentPlatform,
);
```



The screenshot shows the VS Code interface with the file 'pubspec.yaml' open. The file contains the following code, with the 'firebase_core' and 'cloud_firestore' dependencies highlighted by a red box:

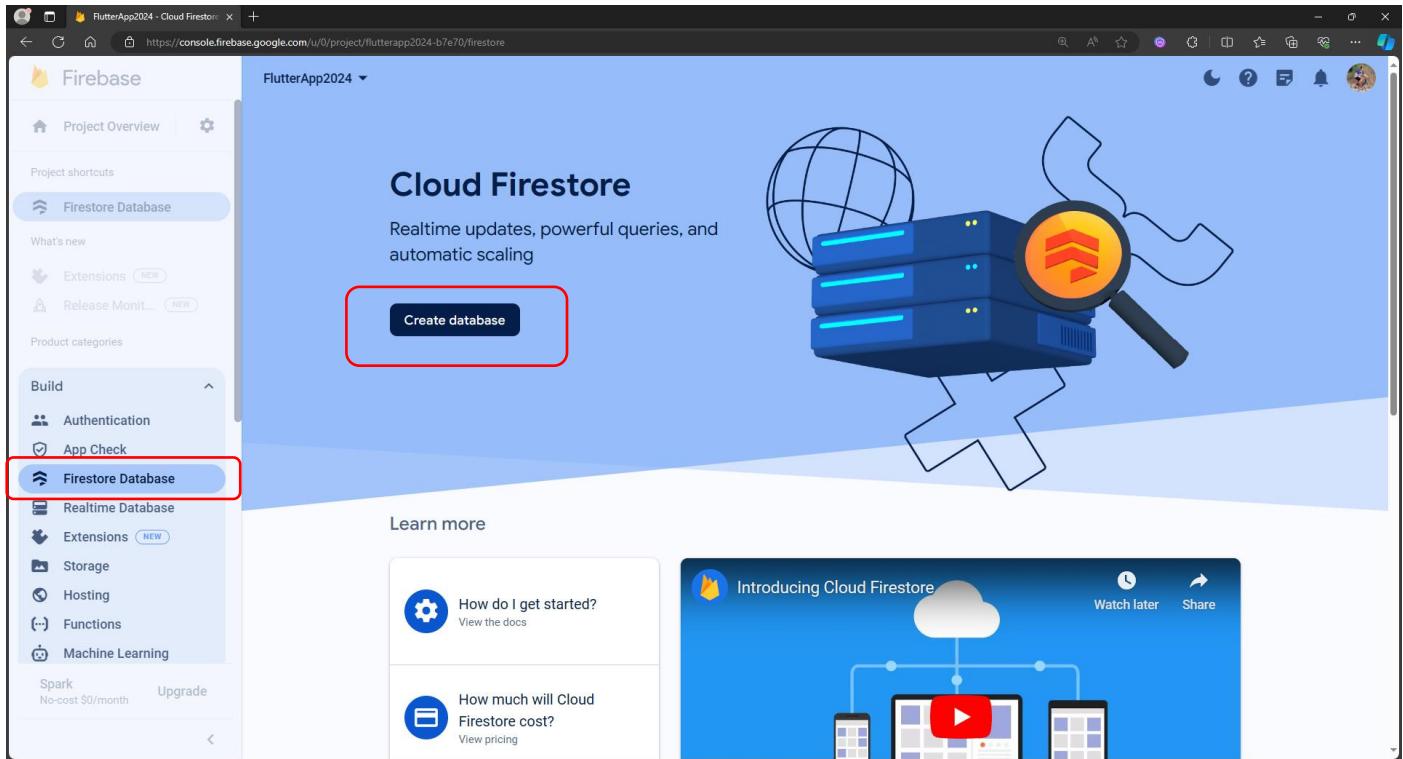
```
25 # To automatically upgrade your package dependencies to the latest versions
26 # consider running `flutter pub upgrade --major-versions`. Alternatively,
27 # dependencies can be manually updated by changing the version numbers below to
28 # the latest version available on pub.dev. To see which dependencies have newer
29 # versions available, run `flutter pub outdated`.
30 dependencies:
31   flutter:
32     | sdk: flutter
33
34
35 # The following adds the Cupertino Icons font to your application.
36 # Use with the CupertinoIcons class for iOS style icons.
37 cupertino_icons: ^1.0.6
38 firebase_core: ^2.27.1
39 cloud_firestore: ^4.15.9
40
41 dev_dependencies:
42   flutter_test:
43     | sdk: flutter
44
45 # The "flutter_lint" package below contains a set of recommended lints to
```

The VS Code interface includes the Explorer, Editor, and Terminal panes. The Editor pane shows the 'pubspec.yaml' file with the highlighted code. The Terminal pane at the bottom shows some log output.

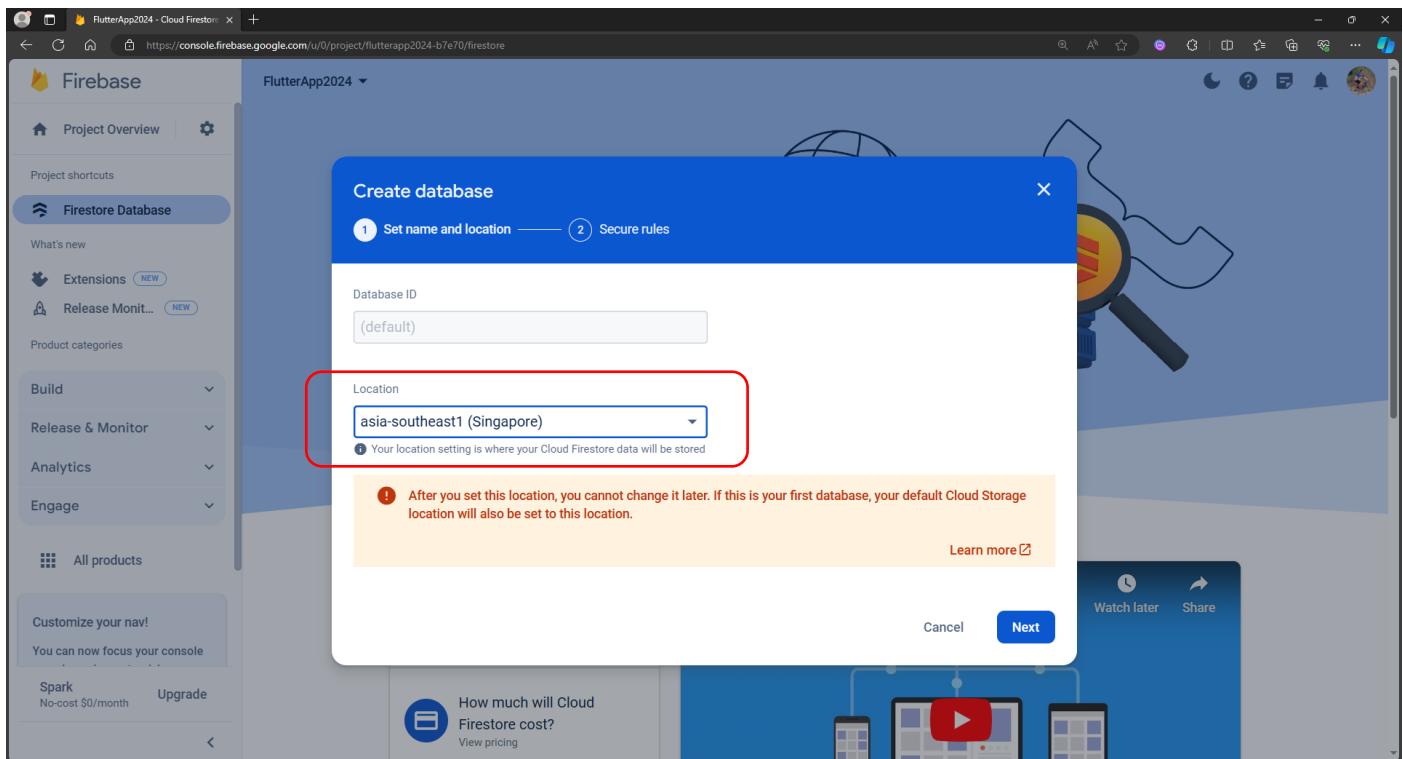
Creating a Firestore database

This section is intended to be as clear as possible. If you feel tedious, just move faster.

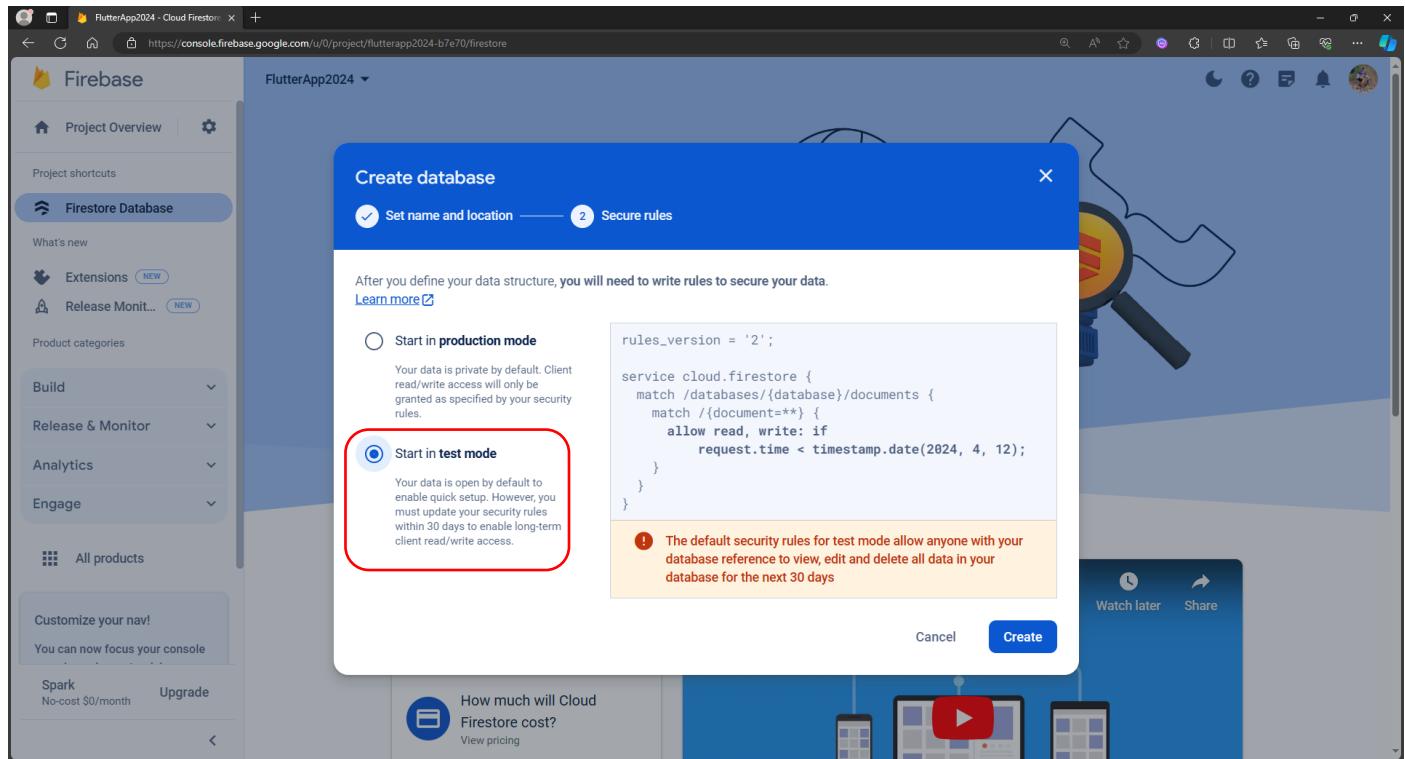
1. Go to your Firebase project's dashboard, select “Firestore Database” then click on the “Create database” button:



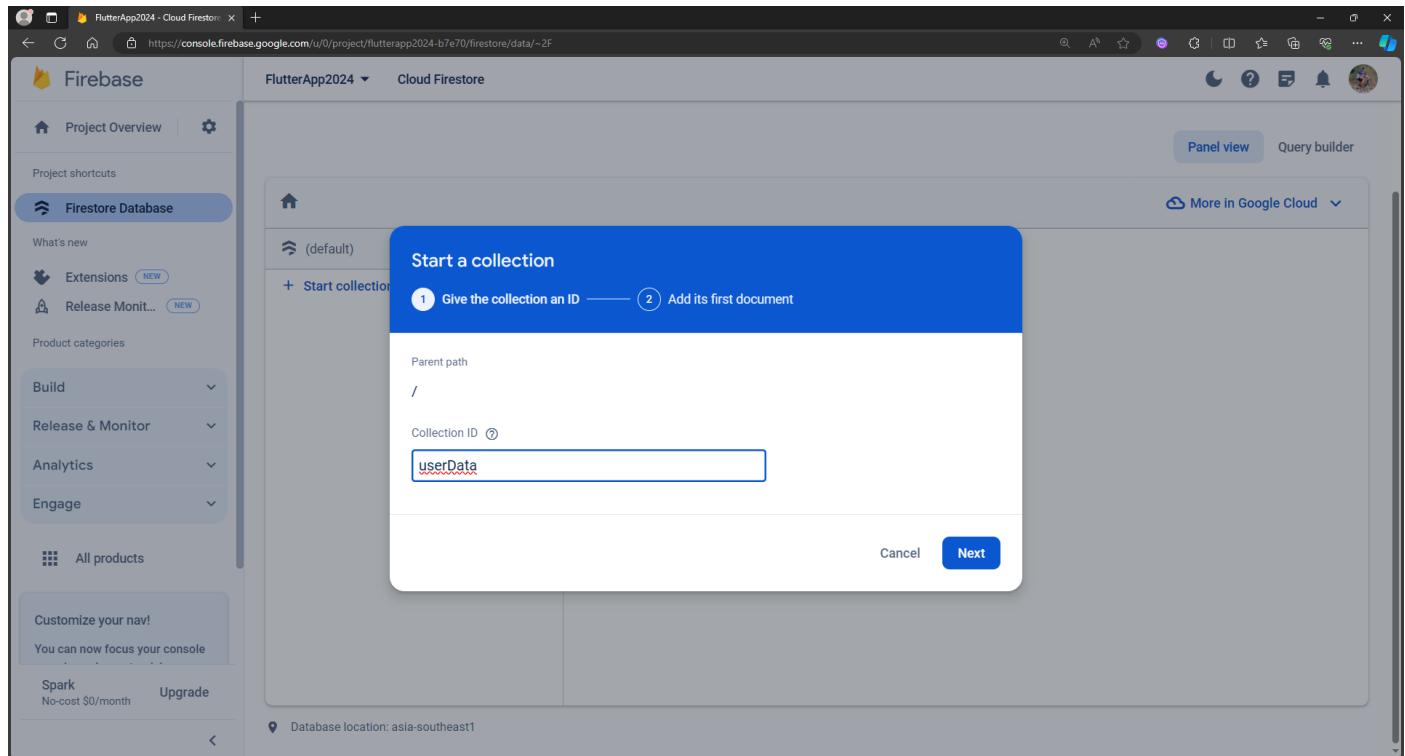
2. Select a location then click the “Next” button. The closer the location is, the lower the latency will be and the faster your app will function.



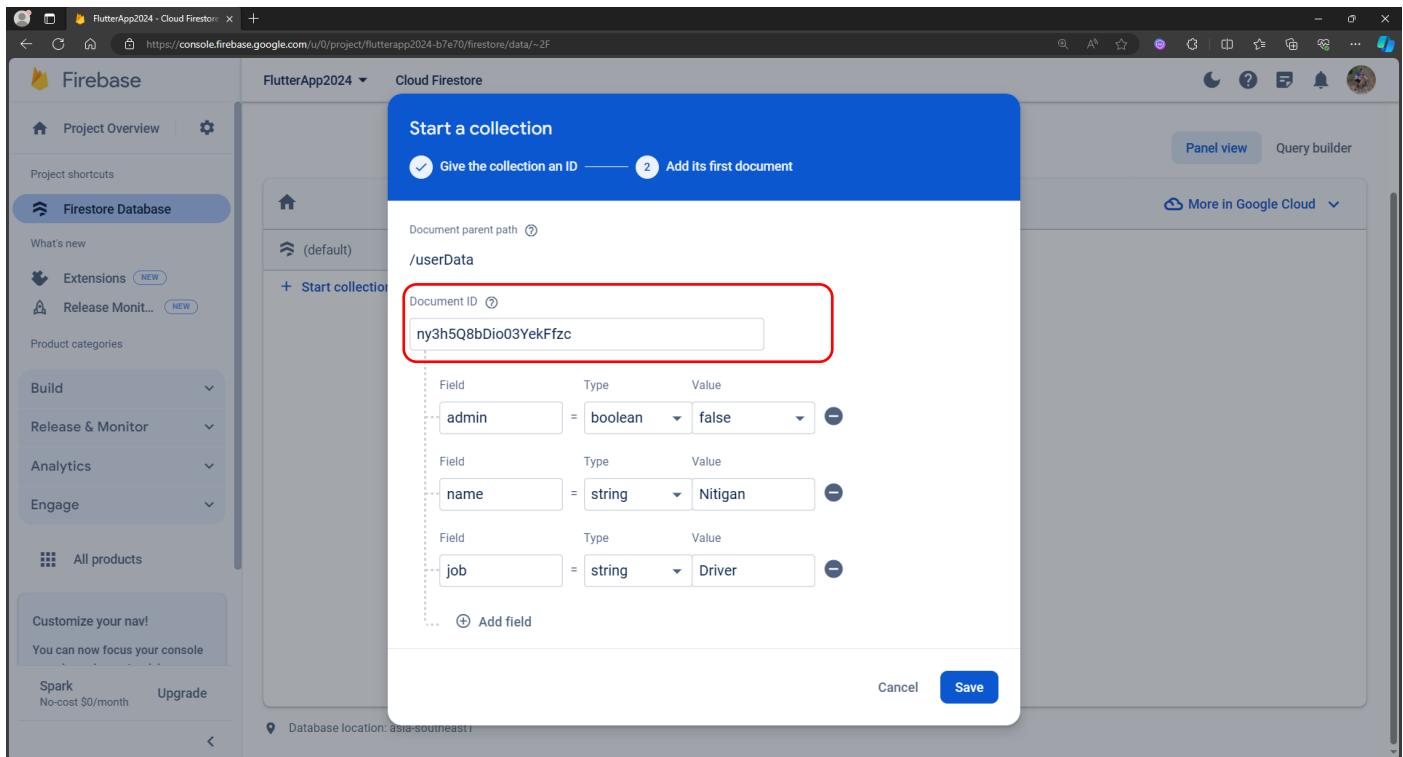
3. Select “Start in test mode” then click on the “Create” button (you can change to “production mode” later if you want)



4. Click “+ Start Collection” to create a new collection.



5. Click Auto ID and create field for data and Fill in the fields then click “Save”.



Start a collection

Give the collection an ID Add its first document

Document parent path

+ Start collection

Document ID

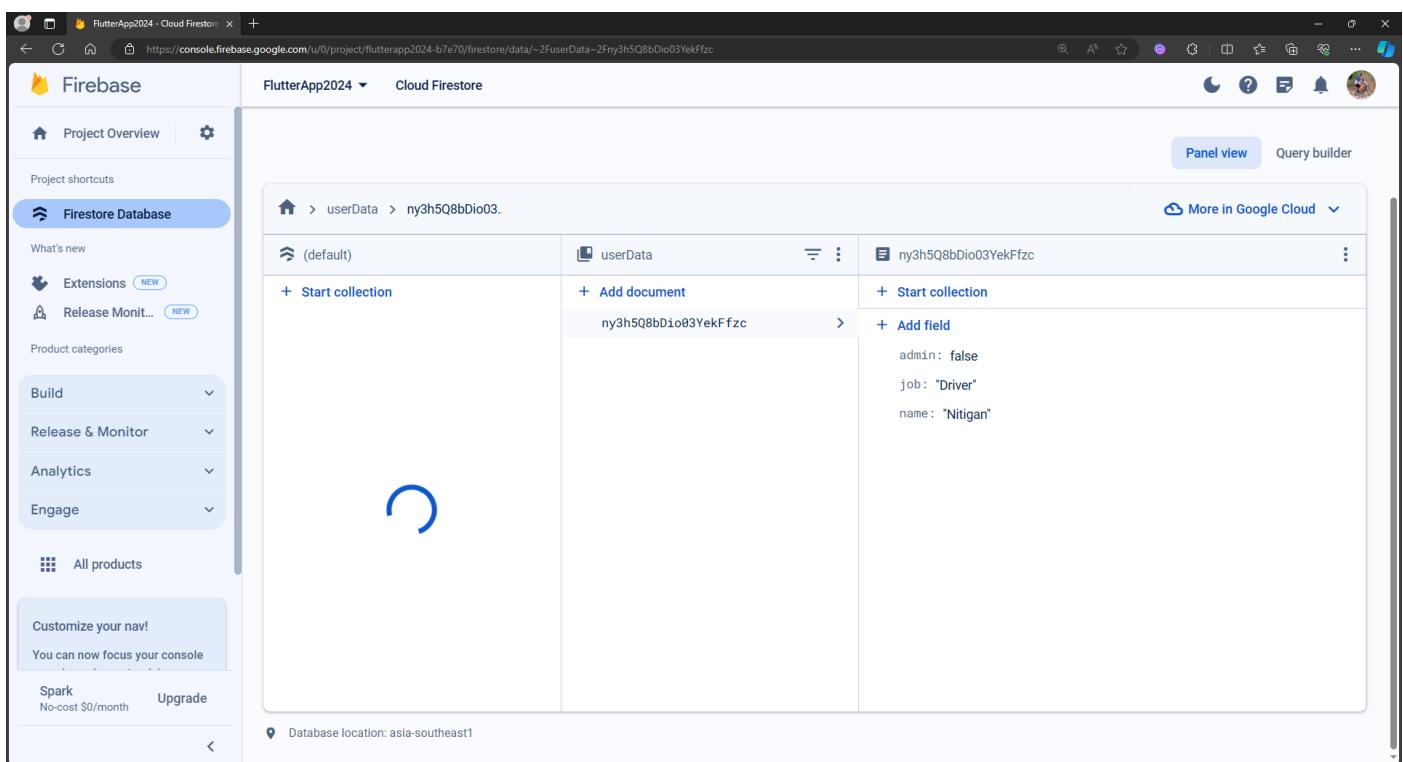
Field	Type	Value
admin	boolean	false
name	string	Nitigan
job	string	Driver

+ Add field

Cancel Save

Database location: asia-southeast1

6. Now we get a Firestore database ready to use



FlutterApp2024 - Cloud Firestore

Project Overview

Firestore Database

userData > ny3h5Q8bDio03...

(default)

+ Start collection

+ Add document

ny3h5Q8bDio03YekFfzc >

userData

+ Start collection

+ Add field

admin: false

job: "Driver"

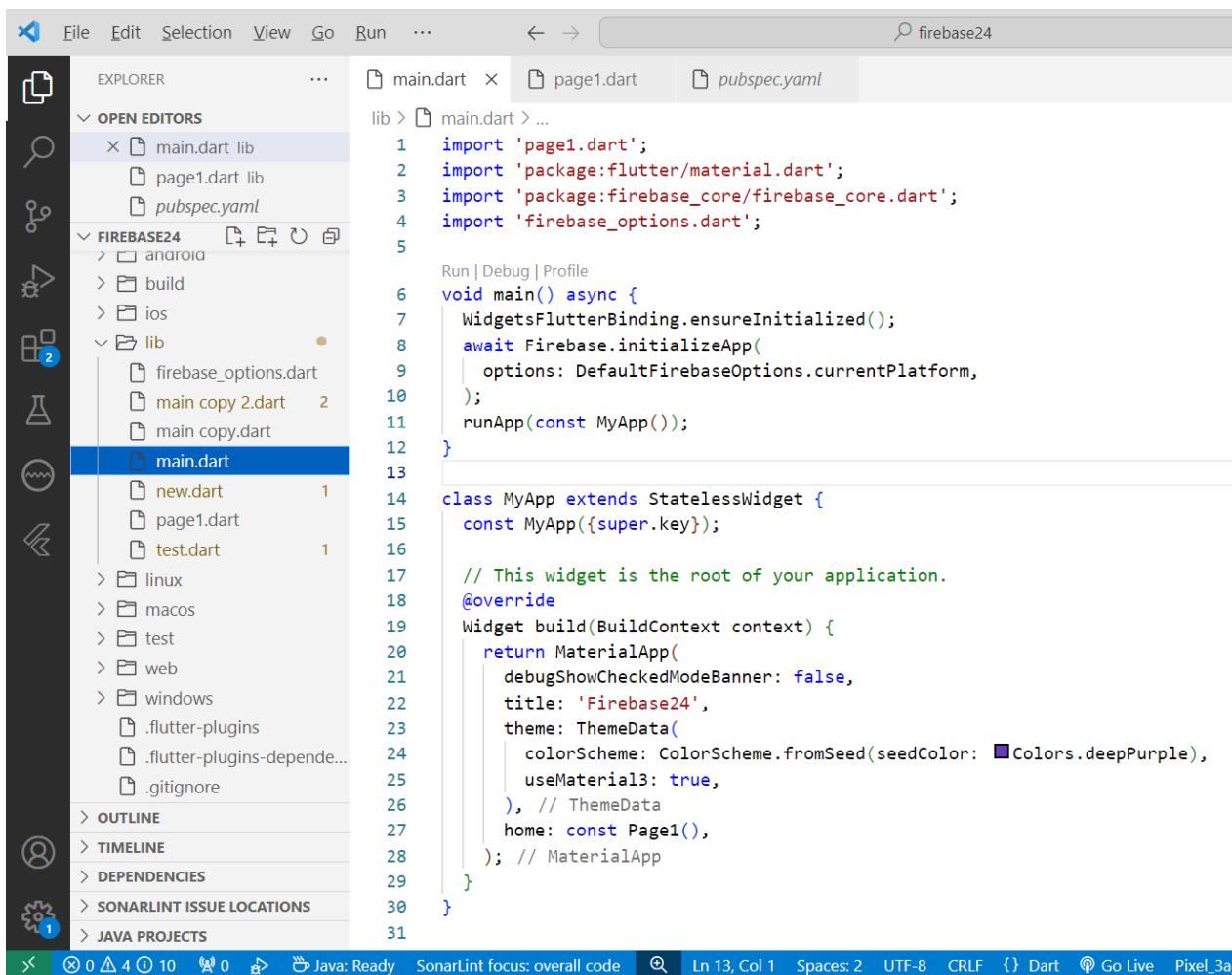
name: "Nitigan"

Customize your nav!

Spark No-cost \$0/month Upgrade

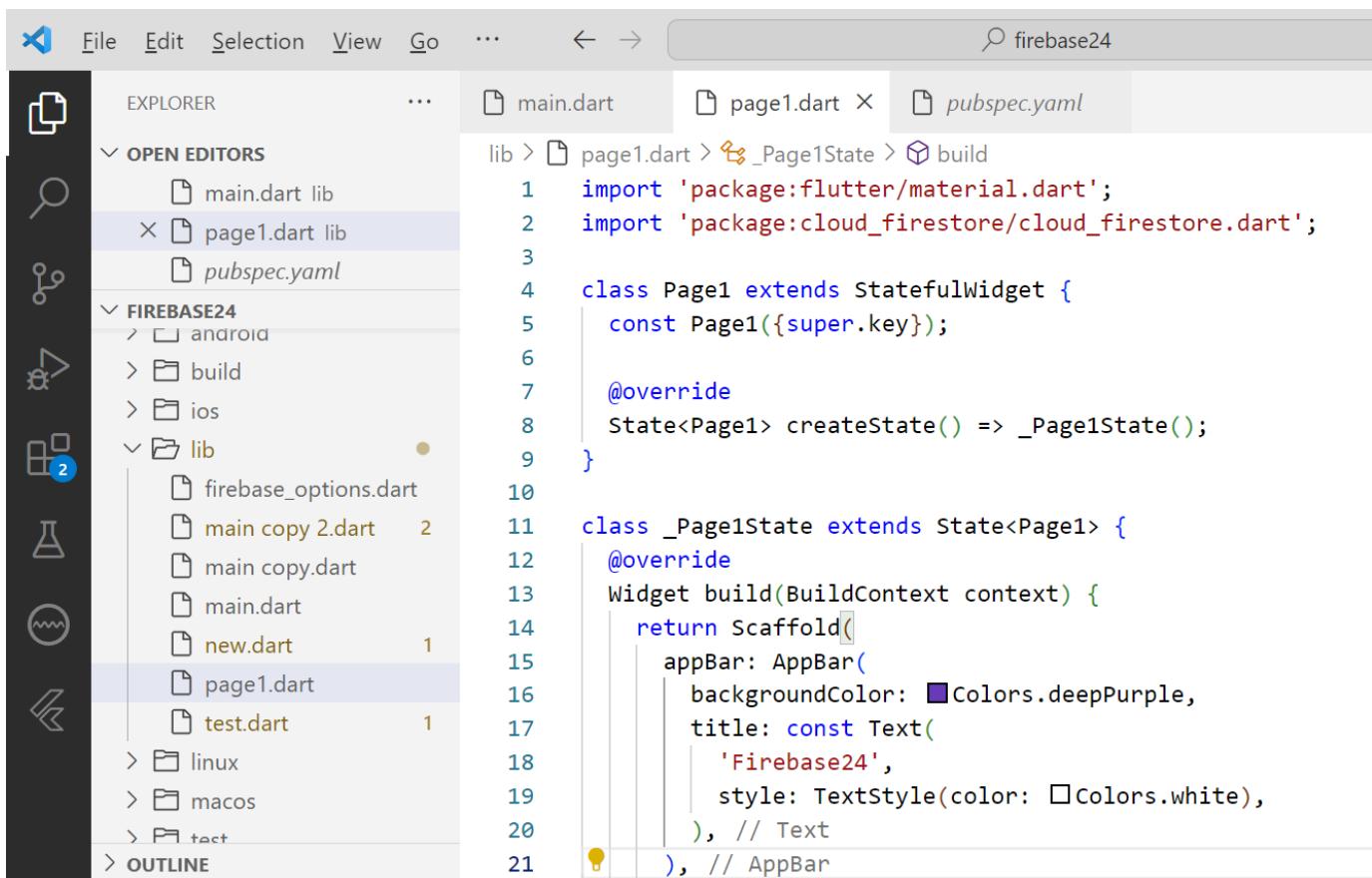
Database location: asia-southeast1

main.dart



```
lib > main.dart > ...
1 import 'page1.dart';
2 import 'package:flutter/material.dart';
3 import 'package:firebase_core/firebase_core.dart';
4 import 'firebase_options.dart';
5
6 void main() async {
7   WidgetsFlutterBinding.ensureInitialized();
8   await Firebase.initializeApp(
9     options: DefaultFirebaseOptions.currentPlatform,
10 );
11 runApp(const MyApp());
12 }
13
14 class MyApp extends StatelessWidget {
15   const MyApp({super.key});
16
17   // This widget is the root of your application.
18   @override
19   Widget build(BuildContext context) {
20     return MaterialApp(
21       debugShowCheckedModeBanner: false,
22       title: 'Firebase24',
23       theme: ThemeData(
24         colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),
25         useMaterial3: true,
26       ), // ThemeData
27       home: const Page1(),
28     ); // MaterialApp
29 }
30 }
```

page1.dart



```
lib > page1.dart > _Page1State > build
1 import 'package:flutter/material.dart';
2 import 'package:cloud_firestore/cloud_firestore.dart';
3
4 class Page1 extends StatefulWidget {
5   const Page1({super.key});
6
7   @override
8   State<Page1> createState() => _Page1State();
9 }
10
11 class _Page1State extends State<Page1> {
12   @override
13   Widget build(BuildContext context) {
14     return Scaffold(
15       appBar: AppBar(
16         backgroundColor: Colors.deepPurple,
17         title: const Text(
18           'Firebase24',
19           style: TextStyle(color: Colors.white),
20         ), // Text
21       ), // AppBar
22     );
23 }
```

```
22     backgroundColor: Colors.white,
23     body: StreamBuilder<QuerySnapshot>(
24       stream: FirebaseFirestore.instance.collection('userData').snapshots(),
25       builder: (BuildContext context, AsyncSnapshot<QuerySnapshot> snapshot) {
26         if (snapshot.hasError) {
27           return const Text('Something went wrong');
28         }
29
30         if (snapshot.connectionState == ConnectionState.waiting) {
31           return const Text('Loading');
32         }
33
34         return Padding(
35           padding: const EdgeInsets.all(8.0),
36           child: ListView(
37             children: snapshot.data!.docs.map(
38               (DocumentSnapshot document) {
39                 Map<String, dynamic> data =
40                   document.data() as Map<String, dynamic>;
41                 return Container(
42                   margin: const EdgeInsets.only(bottom: 10),
43                   child: ListTile(
44                     shape: RoundedRectangleBorder(
45                       borderRadius: BorderRadius.circular(20), // RoundedRectangleBorder
46                     title: Text(data['name']),
47                     subtitle: Text(data['job']),
48                     leading: const Icon(Icons.person),
49                     trailing: !data['admin']
50                     ? SizedBox(
51                       width: 100,
52                       child: Row(
53                         children: <Widget>[
54                           IconButton(
55                             icon: const Icon(Icons.edit),
56                             onPressed: () {
57                               doEdit(document, data);
58                             },
59                           ), // IconButton
60                           IconButton(
61                             icon: const Icon(Icons.delete),
62                             onPressed: () {
63                               doDel(document.id);
64                             },
65                           ), // IconButton
66                         ], // <Widget>[]
67                       ), // Row
68                     ) // SizedBox
69                     : null,
70                     tileColor: Colors.amberAccent,
71                   ), // ListTile
```



```
122           mainCollection.add({
123             'name': nameController.text,
124             'job': jobController.text,
125             'admin': false,
126           });
127
128           Navigator.pop(context);
129         },
130         child: const Text('Add'),
131       ),
132     ],
133   ],
134 );
135 },
136 );
137 }
138
139 void doEdit(DocumentSnapshot document, Map<String, dynamic> data) {
140   //modal input
141   TextEditingController nameController = TextEditingController();
142   TextEditingController jobController = TextEditingController();
143
144   nameController.text = data['name'];
145   jobController.text = data['job'];
146
147   showDialog(
148     context: context,
149     builder: (BuildContext context) {
150       return AlertDialog(
151         title: const Text('Edit Data'),
152         content: Column(
153           children: <Widget>[
154             TextField(
155               controller: nameController,
156               decoration: const InputDecoration(labelText: 'Name'),
157             ),
158             TextField(
159               controller: jobController,
160               decoration: const InputDecoration(labelText: 'Job'),
161             ),
162           ],
163         ),
164         actions: <Widget>[
165           TextButton(
166             onPressed: () {
167               Navigator.pop(context);
168             },
169             child: const Text('Cancel'),
170           ),
171           TextButton(
172             onPressed: () {
173               Navigator.pop(context);
174             },
175             child: const Text('Save'),
176           ),
177         ],
178       );
179     }
180   );
181 }
```

```
172     onPressed: () async {
173         //confirm modal
174         showDialog(
175             context: context,
176             builder: (BuildContext context) {
177                 return AlertDialog(
178                     title: const Text('Save Data'),
179                     content: const Text('Are you sure?'),
180                     actions: <Widget>[
181                         TextButton(
182                             onPressed: () {
183                                 Navigator.pop(context);
184                             },
185                             child: const Text('Cancel'),
186                         ), // TextButton
187                         TextButton(
188                             onPressed: () async {
189                                 Navigator.pop(context);
190                                 final FirebaseFirestore firestore =
191                                     FirebaseFirestore.instance;
192                                 final CollectionReference mainCollection =
193                                     firestore.collection('userData');

194                                     await mainCollection.doc(document.id).update({
195                                         'name': nameController.text,
196                                         'job': jobController.text,
197                                         });
198                                         },
199                                         child: const Text('Save'),
200                                         ), // TextButton
201                                         ],
202                                         // <Widget>[]
203                                         ); // AlertDialog
204                                         },
205                                         ).then((value) => Navigator.pop(context));
206                                         },
207                                         child: const Text('Save'),
208                                         ), // TextButton
209                                         ],
210                                         // <Widget>[]
211                                         ); // AlertDialog
212                                         },
213                                         );
214                                         }

215     void doDel(String id) {
216         //modal dialog
217         showDialog(
218             context: context,
219             builder: (BuildContext context) {
220                 return AlertDialog(
221                     title: const Text('Delete Data'),
```

```
222 content: const Text('Are you sure?'),
223 actions: <Widget>[
224   TextButton(
225     onPressed: () {
226       Navigator.pop(context);
227     },
228     child: const Text('Cancel'),
229   ), // TextButton
230   TextButton(
231     onPressed: () async {
232       Navigator.pop(context);
233       final FirebaseFirestore firestore = FirebaseFirestore.instance;
234       final CollectionReference mainCollection =
235         firestore.collection('userData');
236
237       await mainCollection.doc(id).delete();
238     },
239     child: const Text('Delete'),
240   ), // TextButton
241 ], // <Widget>[]
242 ); // AlertDialog
243 },
244 );
245 }
246 }
```

X 0 ⚠ 4 ⓘ 10 ⚡ Java: Ready SonarLint focus: overall code CRLF {} Dart ⚡ Go Live Pixel_3a_API_34_

Test Application

The screenshot shows the Firebase Cloud Firestore console on the left and a mobile application on the right, both displaying user data.

Firebase Cloud Firestore Console (Left):

- Project Overview:** Firebase
- Firestore Database:** Selected
- Extensions:** (NEW)
- Release Monit...** (NEW)
- Product categories:** Build, Release & Monitor, Analytics, Engage, All products
- Customize your nav!** You can now focus your console
- Spark:** No-cost \$0/month Upgrade

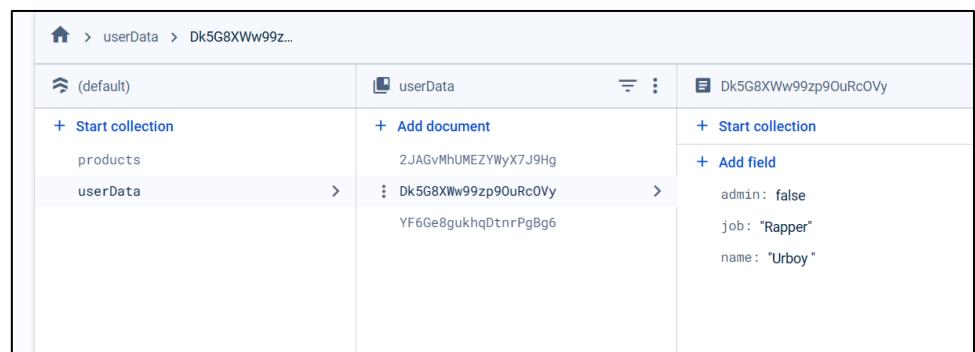
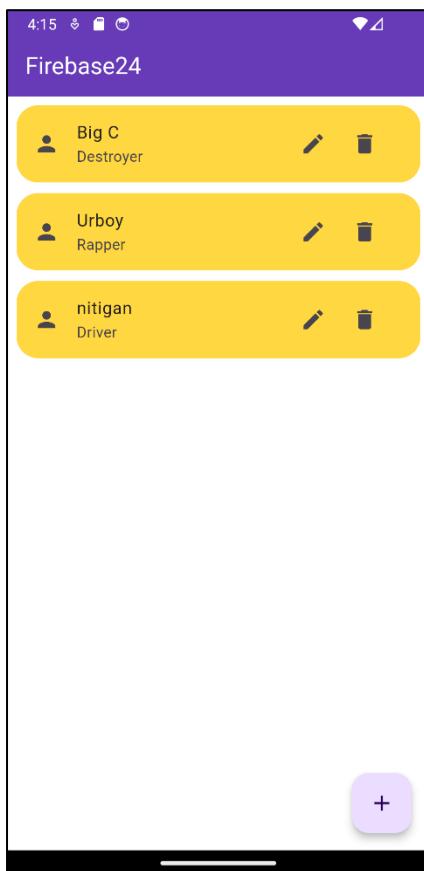
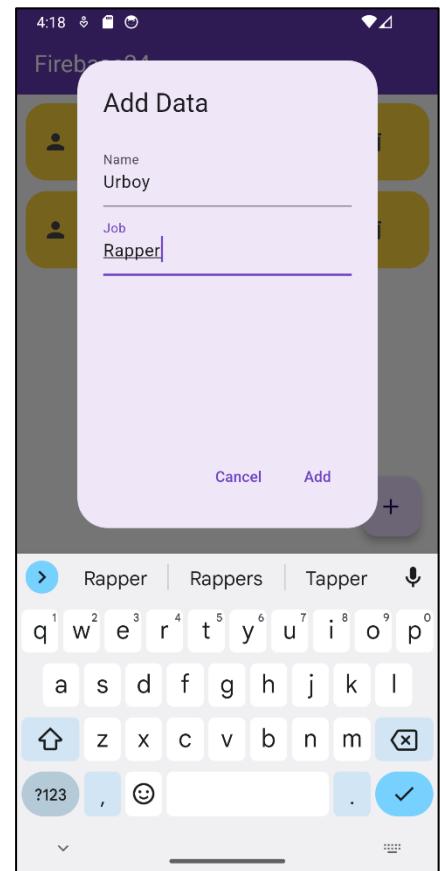
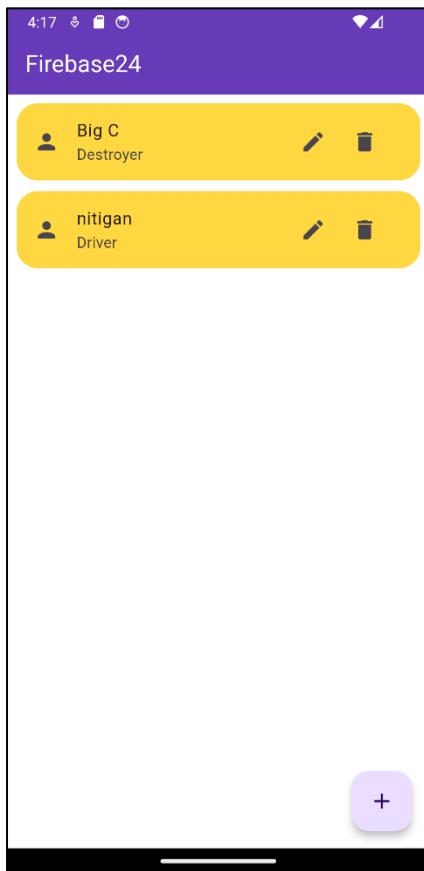
Cloud Firestore Data View:

- Collection:** userData (Default)
- Documents:** 2JAGvMhUMEZ... (Start collection), 2JAGvMhUMEZYWyX7J9Hg (Add document), YF6Ge8gukhqDtnrPgBg6 (Add document)
- Fields:** admin: false, job: "Destroyer", name: "Big C"

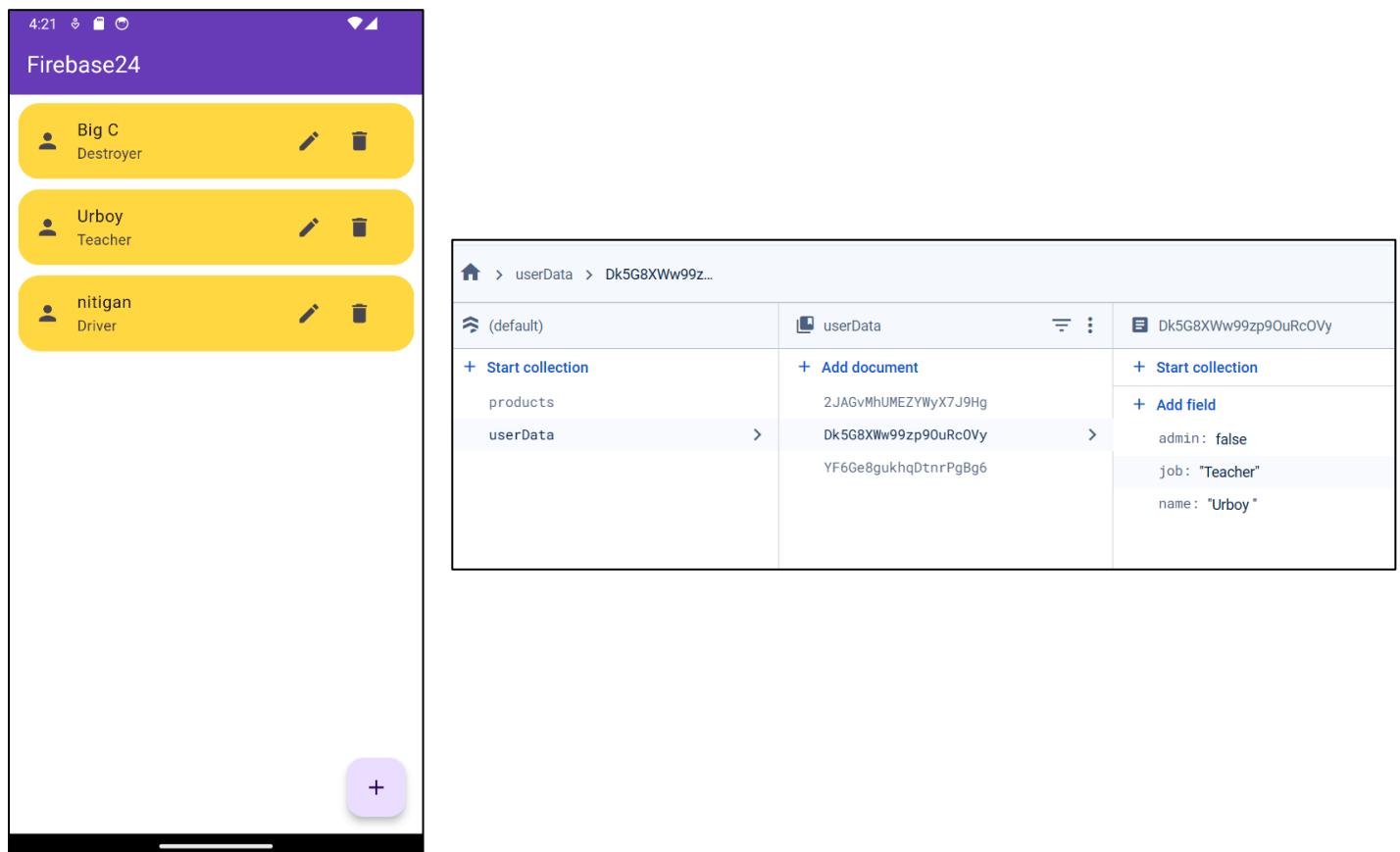
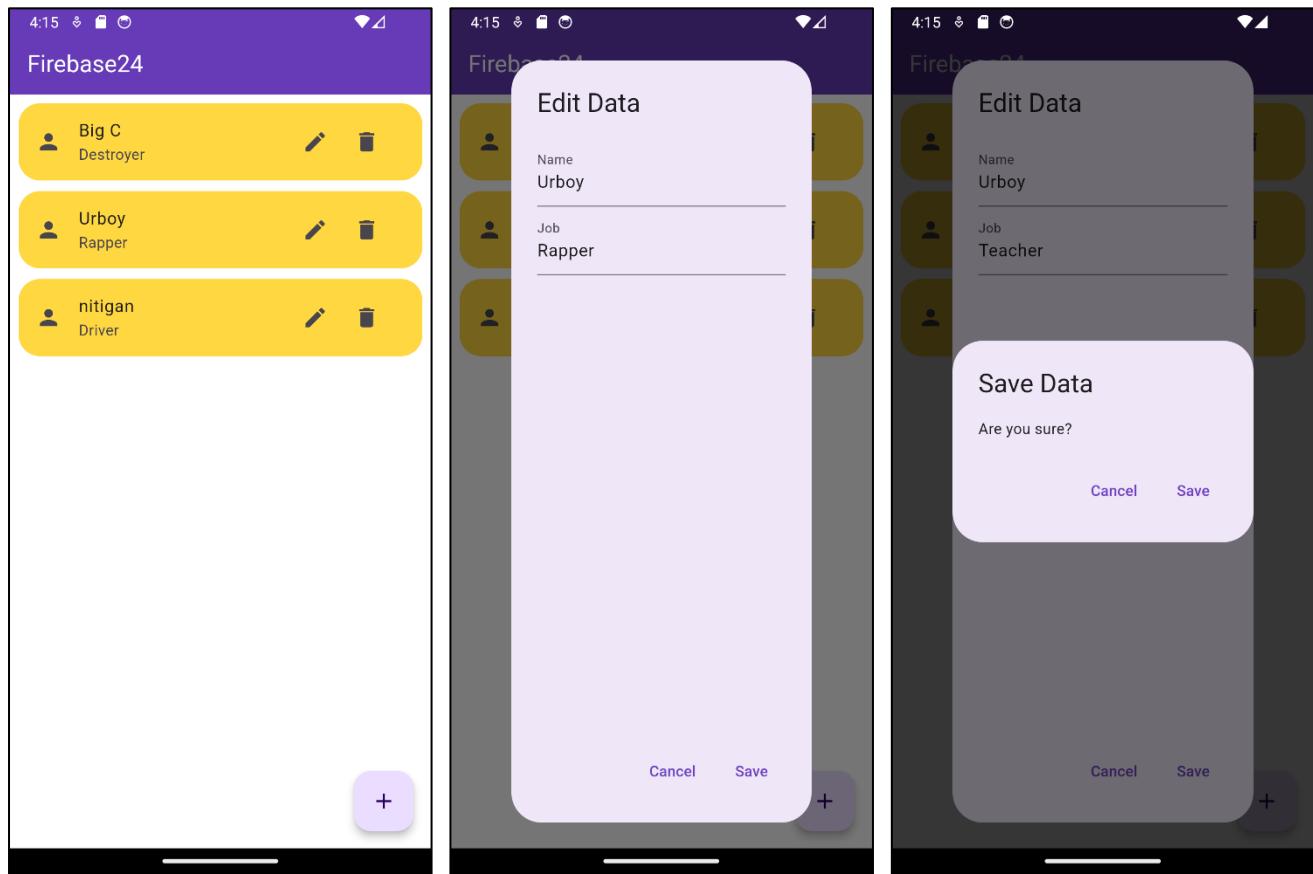
Mobile Application (Right):

- Screen Title:** Firebase24
- UI Components:** List of users with edit and delete icons.
- Users:**
 - Big C (Destroyer)
 - nitigan (Driver)

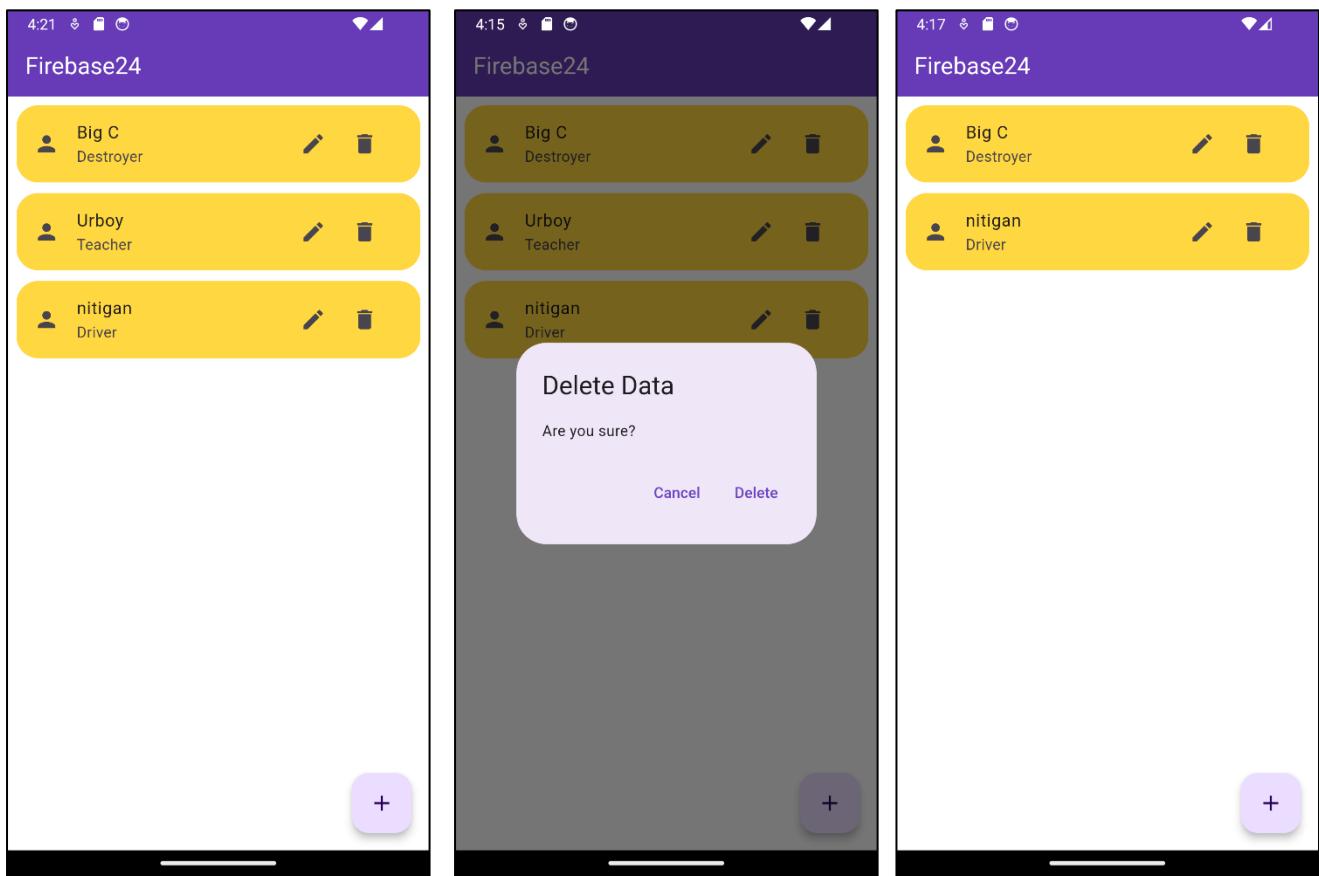
Add Data



Edit Data



Delete Data



The image shows the Firebase Cloud Firestore console interface.

- Left Sidebar:** Shows "Project Overview" and "Firestore Database" (selected).
- Header:** "Cloud Firestore" with tabs: Data, Rules, Indexes, Usage, Extensions.
- Central Area:** A document view for "userData" with ID "Dk5G8XWw99z...". It shows a table with a single document "userData" containing fields "products" and "userData".
- Right Side:** A message: "This document does not exist, it will not appear in queries or snapshots." with a "Learn more" link.