

about_screen.dart

```
import 'package:flutter/material.dart';

class AboutScreen extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: Text('📄 My Resume')),
      body: Padding(
        padding: const EdgeInsets.all(16.0),
        child: SingleChildScrollView(
          child: Column(
            crossAxisAlignment: CrossAxisAlignment.start,
            children: [
              Text("Brunda G", style: TextStyle(fontSize: 26, fontWeight:
FontWeight.bold)),
              Text("Assistant Professor | Technology Mentor", style:
TextStyle(fontSize: 18)),
              SizedBox(height: 20),

              Text("💡 Professional Summary", style: TextStyle(fontSize: 20,
fontWeight: FontWeight.bold)),
              Text(
```

"I am an enthusiastic and innovative technology professional with expertise in empowering students and teams "

"through hands-on learning and practical projects. My work spans across mobile app development, cloud technologies, "

"DevOps automation, and AI-driven workflows. I believe in bridging the gap between academia and industry by "

"building real-world skills in learners.",

style: TextStyle(fontSize: 16),

),

SizedBox(height: 20),

Text("🧰 Skills & Expertise", style: TextStyle(fontSize: 20, fontWeight: FontWeight.bold)),

Text(

"• Flutter & Mobile App Development\n"

"• DevOps, CI/CD & Cloud Deployment\n"

"• AI & Agentic Workflows\n"

"• Data Visualization (Python, Power BI, Tableau)\n"

"• Git, GitHub, and Agile Project Management\n"

"• Mentoring and guiding student projects to success",

style: TextStyle(fontSize: 16),

),

SizedBox(height: 20),

Text("🏆 Achievements", style: TextStyle(fontSize: 20, fontWeight: FontWeight.bold)),

Text(

"• Guided numerous student projects leading to internships and job offers.\n"

"• Conducted Faculty Development Programs on AI, DevOps, and automation tools.\n"

"• Developed academic tools integrating Agile and automation principles.\n"

"• Successfully implemented cloud-based and on-premise DevOps pipelines.",

style: TextStyle(fontSize: 16),

),

SizedBox(height: 30),

```

        Center(
          child: ElevatedButton.icon(
            onPressed: () => Navigator.pop(context),
            icon: Icon(Icons.home),
            label: Text("Back to Home"),
          ),
        ),
      ],
    ),
  ),
);
}
}

```

contact_screen.dart

```

// lib/contact_screen.dart
import 'package:flutter/material.dart';
import 'package:cloud_firestore/cloud_firestore.dart';

class ContactScreen extends StatefulWidget {
  @override
  State<ContactScreen> createState() => _ContactScreenState();
}

class _ContactScreenState extends State<ContactScreen> {
  final _formKey = GlobalKey<FormState>();
  final _nameCtrl = TextEditingController();
  final _emailCtrl = TextEditingController();
  final _messageCtrl = TextEditingController();

  bool _submitting = false;

  @override
  void dispose() {

```

```

    _nameCtrl.dispose();
    _emailCtrl.dispose();
    _messageCtrl.dispose();
    super.dispose();
}

Future<void> _submit() async {
  if (!_formKey.currentState!.validate()) return;

  setState(() => _submitting = true);
  try {
    await FirebaseFirestore.instance.collection('contacts').add({
      'name': _nameCtrl.text.trim(),
      'email': _emailCtrl.text.trim(),
      'message': _messageCtrl.text.trim(),
      'createdAt': FieldValue.serverTimestamp(),
    });

    if (!mounted) return;
    ScaffoldMessenger.of(context).showSnackBar(
      const SnackBar(content: Text('Thanks! Your message was sent.')),
    );
    _formKey.currentState!.reset();
    _nameCtrl.clear();
    _emailCtrl.clear();
    _messageCtrl.clear();
  } catch (e) {
    if (!mounted) return;
    ScaffoldMessenger.of(context).showSnackBar(
      SnackBar(content: Text('Failed to submit: $e')),
    );
  } finally {
    if (mounted) setState(() => _submitting = false);
  }
}

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(title: const Text('📞 Contact Me')),

```

```

body: SafeArea(
  child: Padding(
    padding: const EdgeInsets.all(16.0),
    child: Form(
      key: _formKey,
      child: ListView(
        children: [
          const Text(
            "Get in Touch",
            style: TextStyle(fontSize: 24, fontWeight:
FontWeight.bold),
          ),
          const SizedBox(height: 16),

          // Name
          TextFormField(
            controller: _nameCtrl,
            decoration: const InputDecoration(
              labelText: 'Full Name',
              prefixIcon: Icon(Icons.person),
              border: OutlineInputBorder(),
            ),
            validator: (v) =>
              (v == null || v.trim().isEmpty) ? 'Enter your name' :
null,
          ),
          const SizedBox(height: 12),

          // Email
          TextFormField(
            controller: _emailCtrl,
            decoration: const InputDecoration(
              labelText: 'Email',
              prefixIcon: Icon(Icons.email),
              border: OutlineInputBorder(),
            ),
            keyboardType: TextInputType.emailAddress,
            validator: (v) {
              final text = v?.trim() ?? '';

```

```

        final emailRegex =
RegExp(r'^^[^@\s]+@[^@\s]+\.[^@\s]+\.$');
        if (text.isEmpty) return 'Enter your email';
        if (!emailRegex.hasMatch(text)) return 'Enter a valid
email';

        return null;
    },
),
const SizedBox(height: 12),

// Message
TextFormField(
  controller: _messageCtrl,
  decoration: const InputDecoration(
    labelText: 'Message',
    alignLabelWithHint: true,
    prefixIcon: Icon(Icons.message),
    border: OutlineInputBorder(),
  ),
  minLines: 4,
  maxLines: 6,
  validator: (v) =>
    (v == null || v.trim().isEmpty) ? 'Enter your
message' : null,
),
const SizedBox(height: 20),

// Submit Button
SizedBox(
  width: double.infinity,
  child: ElevatedButton.icon(
    onPressed: _submitting ? null : _submit,
    icon: _submitting
      ? const SizedBox(
        width: 18, height: 18, child:
CircularProgressIndicator(strokeWidth: 2),
      )
      : const Icon(Icons.send),
    label: Text(_submitting ? 'Sending...' : 'Send
Message'),

```

```

        ),
      ),
      const SizedBox(height: 12),
      OutlinedButton.icon(
        onPressed: () => Navigator.pop(context),
        icon: const Icon(Icons.home),
        label: const Text("Back to Home"),
      ),
    ],
  ),
),
),
),
),
);
}
}

```

home_screen.dart

```

import 'package:flutter/material.dart';

class HomeScreen extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: Text('🏠 Home')),
      body: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            CircleAvatar(
              radius: 50,
              child: Icon(Icons.person, size: 50, color: Colors.white),
              backgroundColor: Colors.teal,
            ),
            SizedBox(height: 20),

```

```

        Text(
          "Hi, I am Brunda 🙋",
          style: TextStyle(fontSize: 24, fontWeight: FontWeight.bold),
        ),
        SizedBox(height: 10),
        Text(
          "A passionate educator, developer, and technology mentor, "
          "helping students and professionals excel in modern tech
skills.",
          style: TextStyle(fontSize: 16),
          textAlign: TextAlign.center,
        ),
        SizedBox(height: 40),
        ElevatedButton.icon(
          onPressed: () => Navigator.pushNamed(context, '/about'),
          icon: Icon(Icons.person),
          label: Text("View My Resume"),
        ),
        SizedBox(height: 10),
        ElevatedButton.icon(
          onPressed: () => Navigator.pushNamed(context, '/contact'),
          icon: Icon(Icons.contact_mail),
          label: Text("Contact Me"),
        ),
      ],
    ),
  ),
);
}
}

```

main.dart

```

// lib/main.dart
import 'package:flutter/material.dart';
import 'package:firebase_core/firebase_core.dart';

```



```

import 'firebase_options.dart';

import 'home_screen.dart';
import 'about_screen.dart';
import 'contact_screen.dart';

void main() async {
  WidgetsFlutterBinding.ensureInitialized();
  await Firebase.initializeApp(
    options: DefaultFirebaseOptions.currentPlatform,
  );
  runApp(MyApp());
}


class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      title: 'Brunda - Flutter Demo',
      theme: ThemeData(primarySwatch: Colors.teal),
      initialRoute: '/',
      routes: {
        '/': (context) => HomeScreen(),
        '/about': (context) => AboutScreen(),
        '/contact': (context) => ContactScreen(),
      },
    );
  }
}

```

In vs code:

```
dart pub add firebase_core cloud_firestore
```

After executing the above command


 Firebase


BuildRunSolutionsPricingDocsCommunitySupport

Search /

English

BlogStudioGo to console





Introducing Firebase Studio

Prototype, build, and deploy full-stack, AI apps with our agentic development environment. Now with AI-optimized templates, seamless integration with Firebase services, and ability to fork workspaces, all designed to make AI-assisted development more powerful.


[Get Started](#)

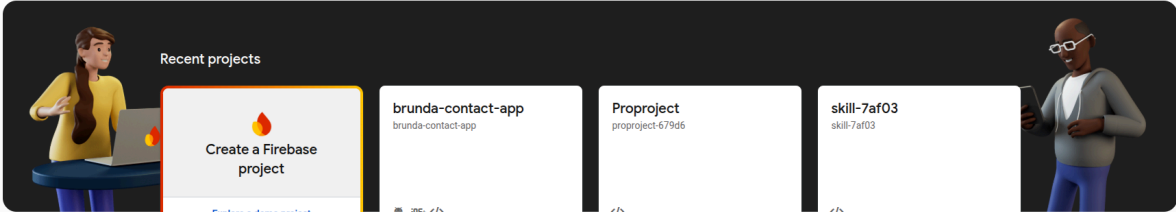
Make your app the best it can be with Firebase and generative AI

Prototype, build & run modern, AI-powered experiences users love with Firebase, a platform designed to support you throughout your app development lifecycle.

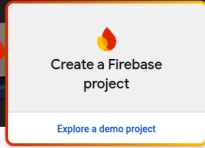
Backed by Google and trusted by millions of businesses around the world.

Create a project in flutter
Login to flutter
Then click on go to console

 Firebase



Recent projects

[Explore a demo project](#)

brunda-contact-app
brunda-contact-app

iOS- </>

Proproject
proproject-679d6

</>

skill-7af03
skill-7af03

</>

Test
test-253bd

</>

food
food-362d8

All Firebase projects

brunda-contact-app
brunda-contact-app

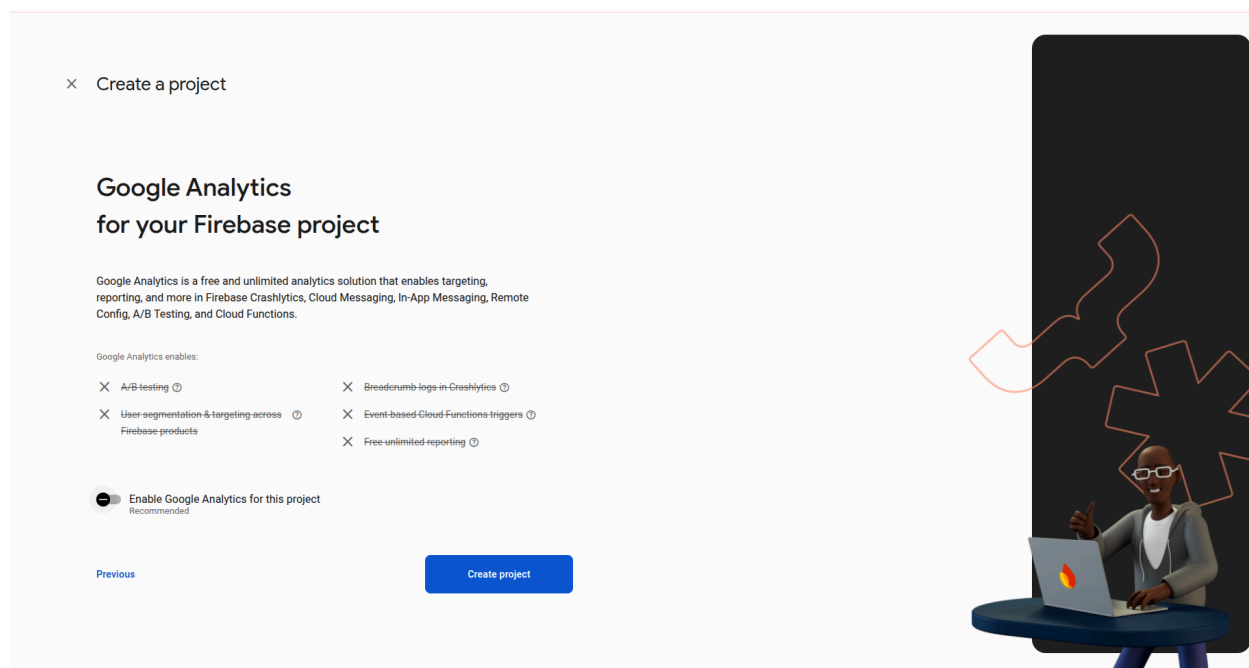
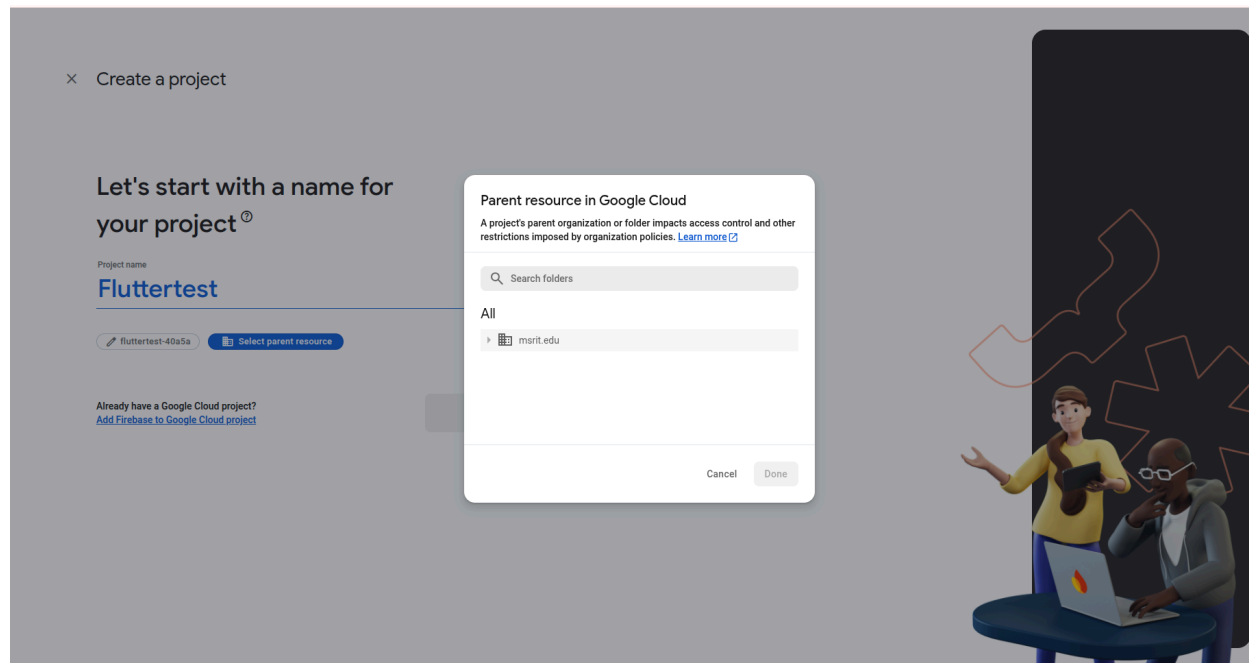
iOS- </>

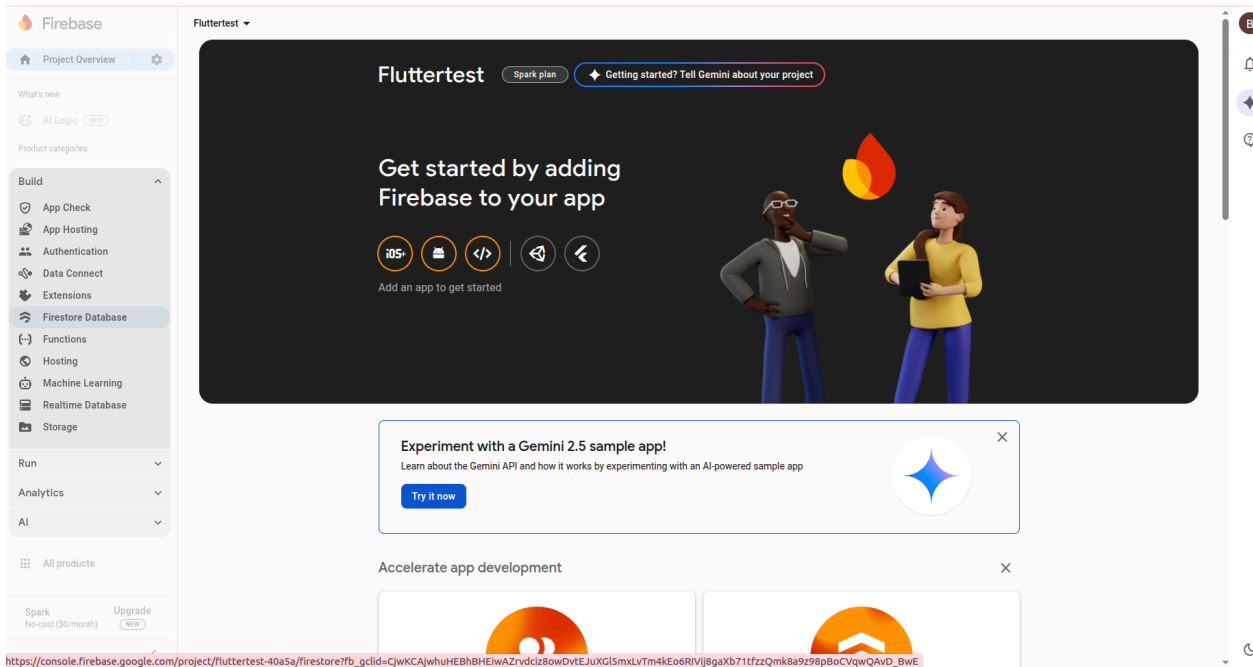
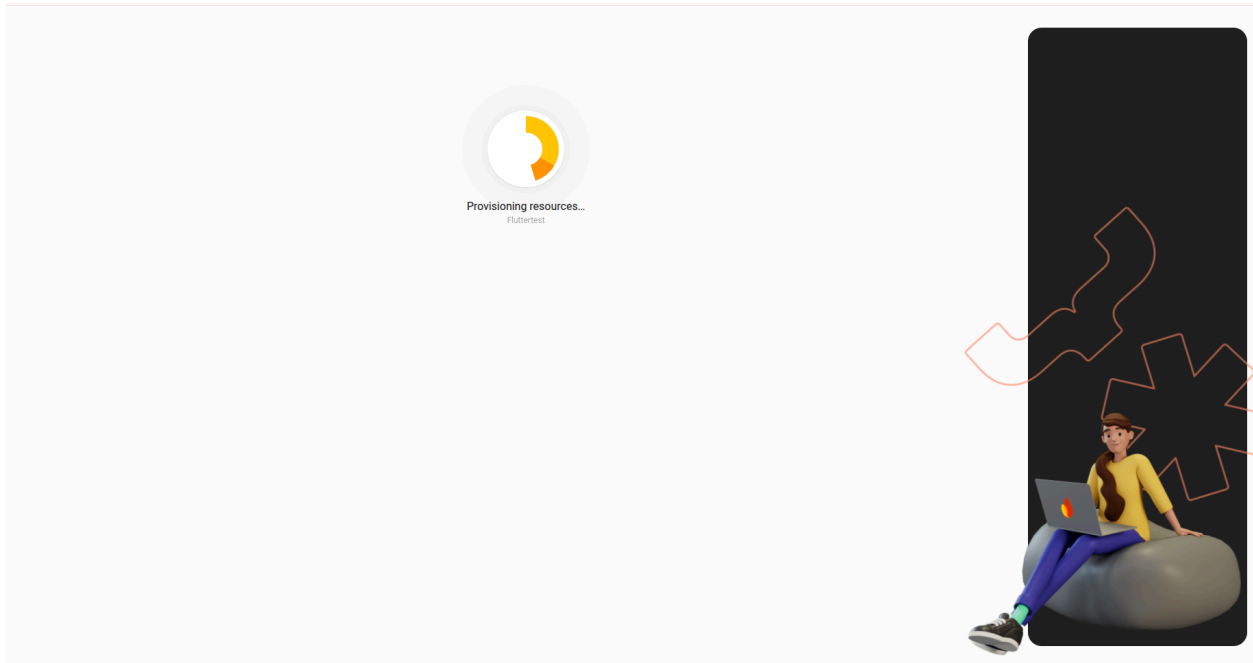
brunda
brunda-d68d1

food
food-362d8

Proproject
proproject-679d6

</>





In the above select the firebase database

Now come back to vs code in terminal, execute the below command:

install once

npm i -g firebase-tools

dart pub global activate flutterfire_cli

```
# log in
firebase login
```

```
# from your Flutter project root
flutterfire configure
```

The screenshot shows an IDE with a project explorer on the left and a terminal window at the bottom. The project explorer displays the following structure:

- EXPLORER
 - OPEN EDITORS
 - Untitled-1
 - about_screen.dar...
 - contact_screen.d...
 - main.dart
 - home_screen.dar...
 - FLUTTER_APPLICATION_1
 - .dart_tool
 - .idea
 - android
 - ios
 - lib
 - about_screen.dart
 - contact_screen.dart
 - firebase_options.dart
 - home_screen.dart
 - main.dart
 - linux
 - macos
 - test
 - web
 - OUTLINE
 - TIMELINE
 - DEPENDENCIES

The terminal window shows the following output:

```

android 1:1050275396451:android:f537e8a1203aa6bb56b156
ios      1:1050275396451:ios:7ea0f94a7cf3bacd56b156
macos    1:1050275396451:ios:7ea0f94a7cf3bacd56b156
windows  1:1050275396451:web:1f6708380bc543b856b156

Learn more about using this file and next steps from the documentation:
> https://firebase.google.com/docs/flutter/setup

o cselab3@cselab3:~/flutter_firebase/flutter_application_1$ # install once
npm i -g firebase-tools
dart pub global activate flutterfire cli

# log in
firebase login

# from your Flutter project root
flutterfire configure
  
```

The screenshot shows an IDE interface with a dark theme. On the left, a sidebar displays the project structure, including files like 'home_screen.dart', 'main.dart', and folders for 'linux', 'macos', 'test', 'web', 'OUTLINE', 'TIMELINE', and 'DEPENDENCIES'. The main editor area shows a dialog titled 'Select a Firebase project to configure your Flutter application with'. This dialog lists several Firebase projects: 'brunda-contact-app', 'brunda-d68d1', 'fluttertest-40a5a', 'food-362d8', 'proproject-679d6', 'skill-7af03', and 'test-253bd'. A green link '<create a new project>' is also visible. The bottom status bar indicates the current line and column as 'Ln 48, Col 1' and the operating system as 'Linux (linux-x64)'.

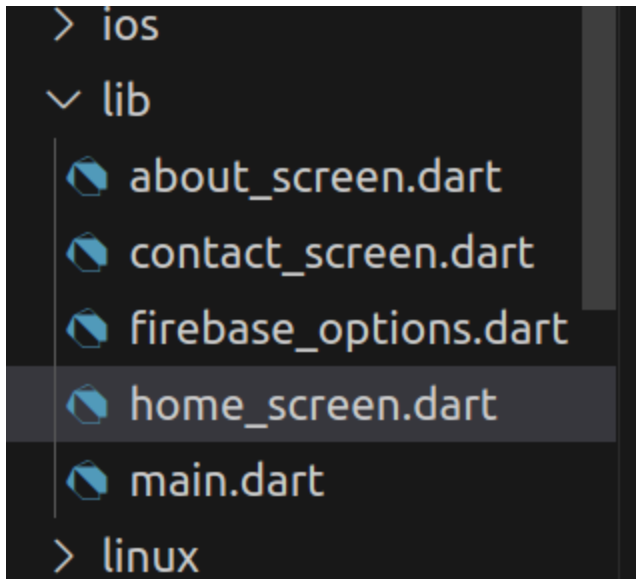
By using arrows, select the project you have created

The screenshot shows an IDE interface with the following elements:

- Left Sidebar:** A file explorer showing a project structure. The file `main.dart` is selected. Below it are folders for `linux`, `macos`, `test`, `web`, `android`, `ios`, `macos`, `web`, `windows`, `OUTLINE`, `TIMELINE`, and `DEPENDENCIES`.
- Main Editor:** Displays the Flutter CLI command `flutter test` and the prompt `Which platforms should your configuration support (use arrow keys & space to select)?`. Below the prompt, a list of platforms is shown with checkboxes:
 - ☒ `android`
 - ☒ `ios`
 - ☒ `macos`
 - ☒ `web`
 - ☒ `windows`
- Bottom Status Bar:** Shows the current line and column as `Ln 48, Col 1` and the operating system as `Linux (linux-x64)`.

In the above select android and click enter

After this, it will create a `firebase_options.dart` file



```
flutterfire --platform ios
i Registered a new Firebase ios app on Firebase project fluttertest-40a5a.
i Firebase macos app com.example.flutterApplication1 registered.
i Firebase web app flutter_application_1 (web) is not registered on Firebase project fluttertest-40a5a.
i Registered a new Firebase web app on Firebase project fluttertest-40a5a.
i Firebase windows app flutter_application_1 (windows) is not registered on Firebase project fluttertest-40a5a.
i Registered a new Firebase windows app on Firebase project fluttertest-40a5a.

Firebase configuration file lib/firebase_options.dart generated successfully with the following Firebase apps:

Platform  Firebase App Id
web       1:928880383388:web:3f0555f1a5eafccf59a459
android   1:928880383388:android:15353c9e5152991859a459
ios       1:928880383388:ios:c6ae207dd7dbd51259a459
macos     1:928880383388:ios:c6ae207dd7dbd51259a459
windows   1:928880383388:web:1920902e47b2f51d59a459

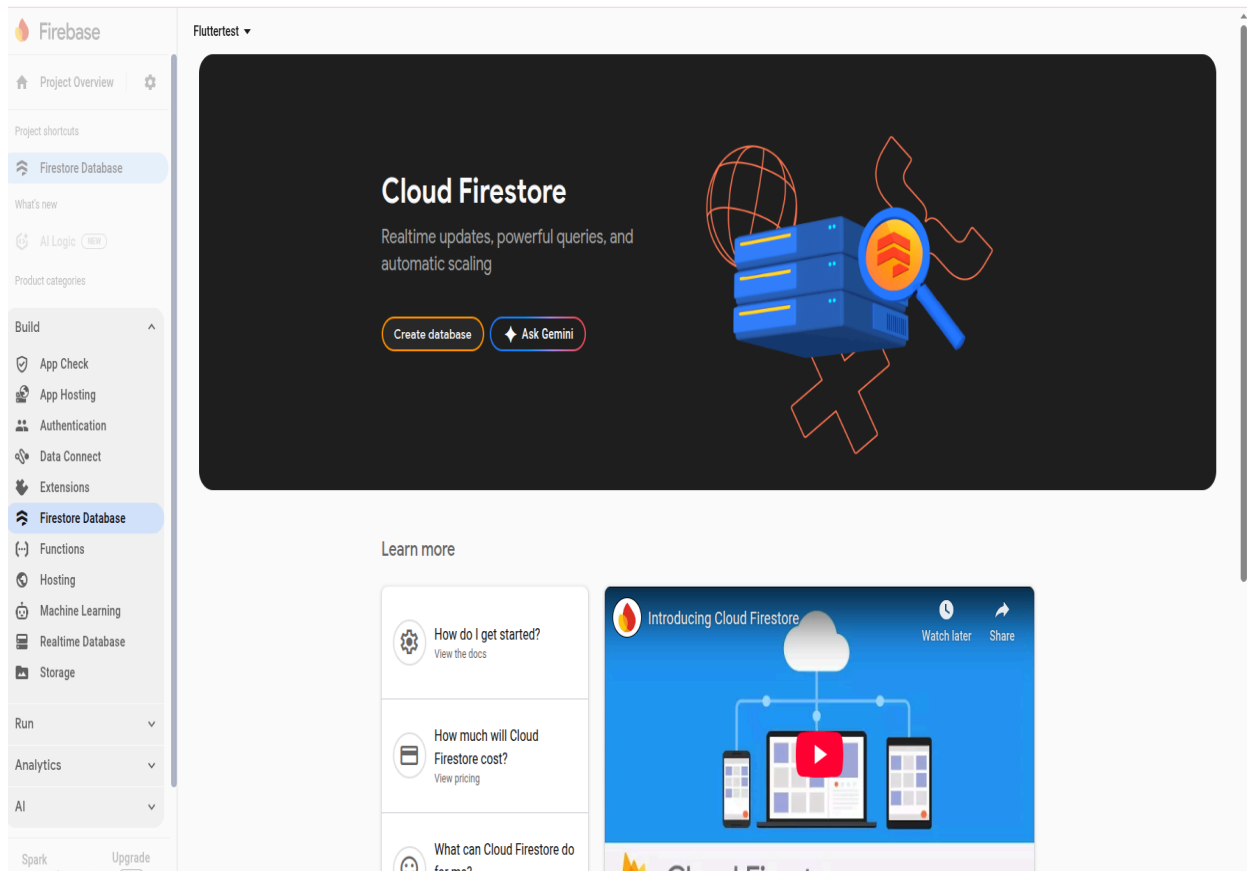
Learn more about using this file and next steps from the documentation:
> https://firebase.google.com/docs/flutter/setup
```

```
cselab3@cselab3:~/flutter_firebase/flutter_application_1$ flutterfire --version
1.3.1
cselab3@cselab3:~/flutter_firebase/flutter_application_1$
```

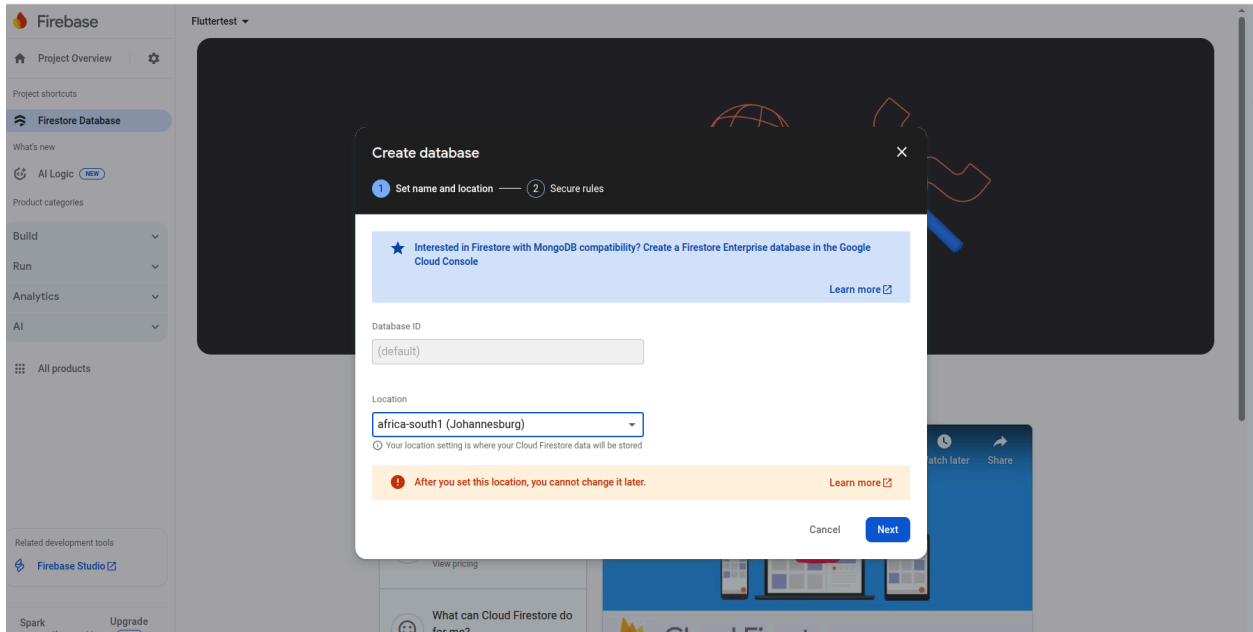
Now go to firebase login

In Firebase Console → left menu → **Firestore Database** → **Create database**

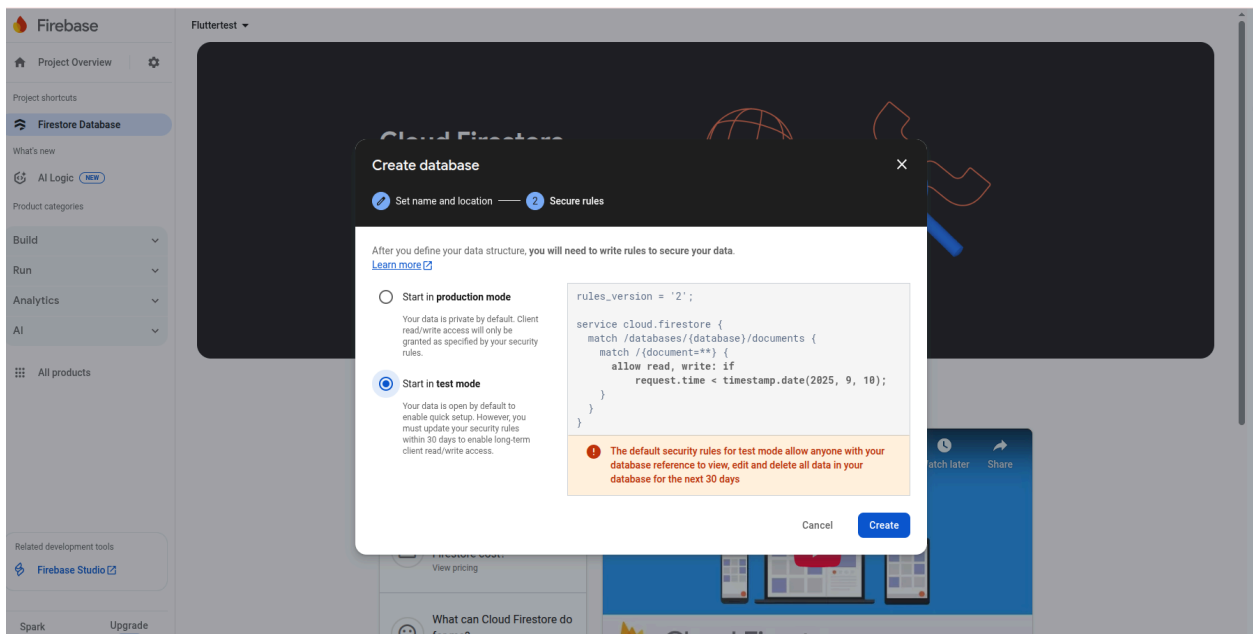
Choose **Start in test mode** (dev only) → pick region (e.g., **asia-south1**) → **Enable**.
Reply “done firestore” if that’s already done.

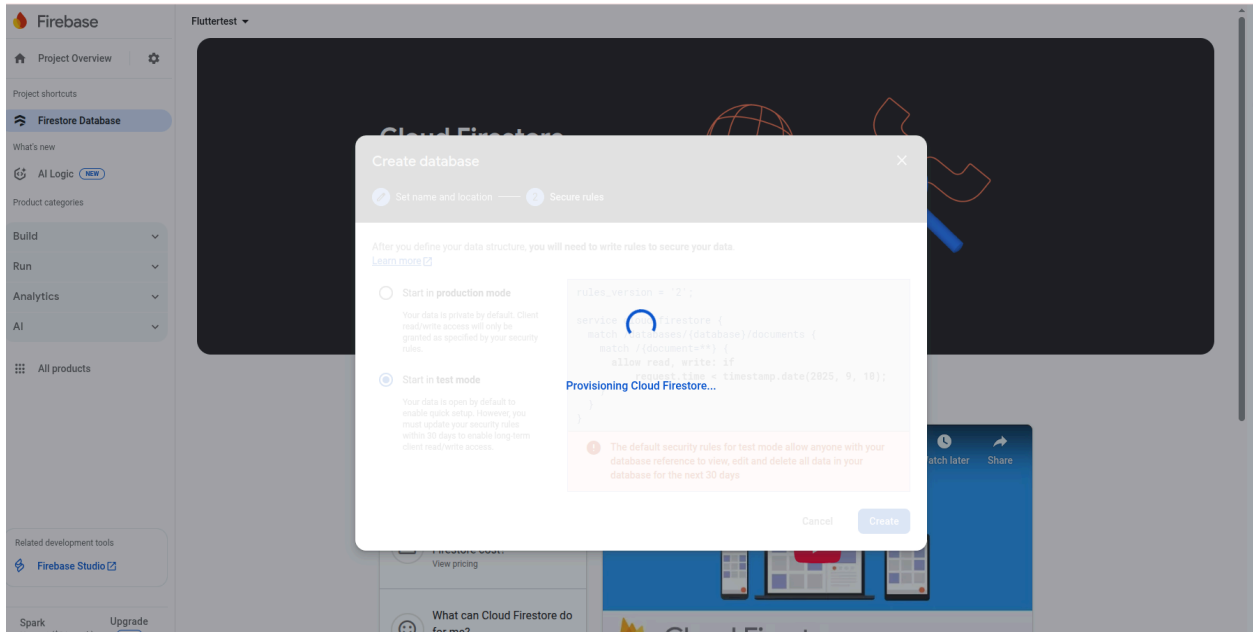


Choose **Start in test mode** (dev only) → pick region (e.g., **asia-south1**) → **Enable**.
Reply “done firestore” if that’s already done.



Select the start in test mode



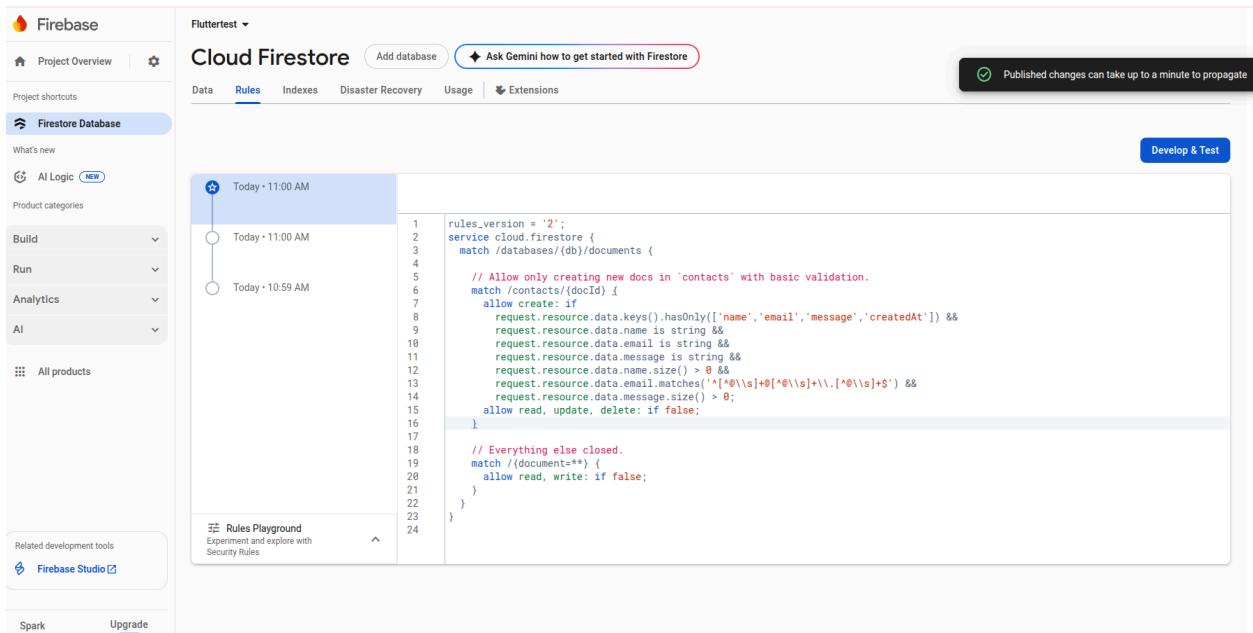


For development you can keep “test mode”. For a safer setup, use rules like this (Console → Firestore → Rules):

```
rules_version = '2';
service cloud.firestore {
  match /databases/{db}/documents {

    // Allow only creating new docs in `contacts` with basic validation.
    match /contacts/{docId} {
      allow create: if
        request.resource.data.keys().hasOnly(['name','email','message','createdAt']) &&
        request.resource.data.name is string &&
        request.resource.data.email is string &&
        request.resource.data.message is string &&
        request.resource.data.name.size() > 0 &&
        request.resource.data.email.matches('^[^@\\s]+@[^@\\s]+\\.([^@\\s]+)$') &&
        request.resource.data.message.size() > 0;
      allow read, update, delete: if false;
    }

    // Everything else closed.
    match /{document=**} {
      allow read, write: if false;
    }
  }
}
```



firebase init

flutter pub add firebase_core

flutter pub add cloud_firestore # for Firestore

or

flutter pub add firebase_database # for Realtime Database

flutter run -d chrome