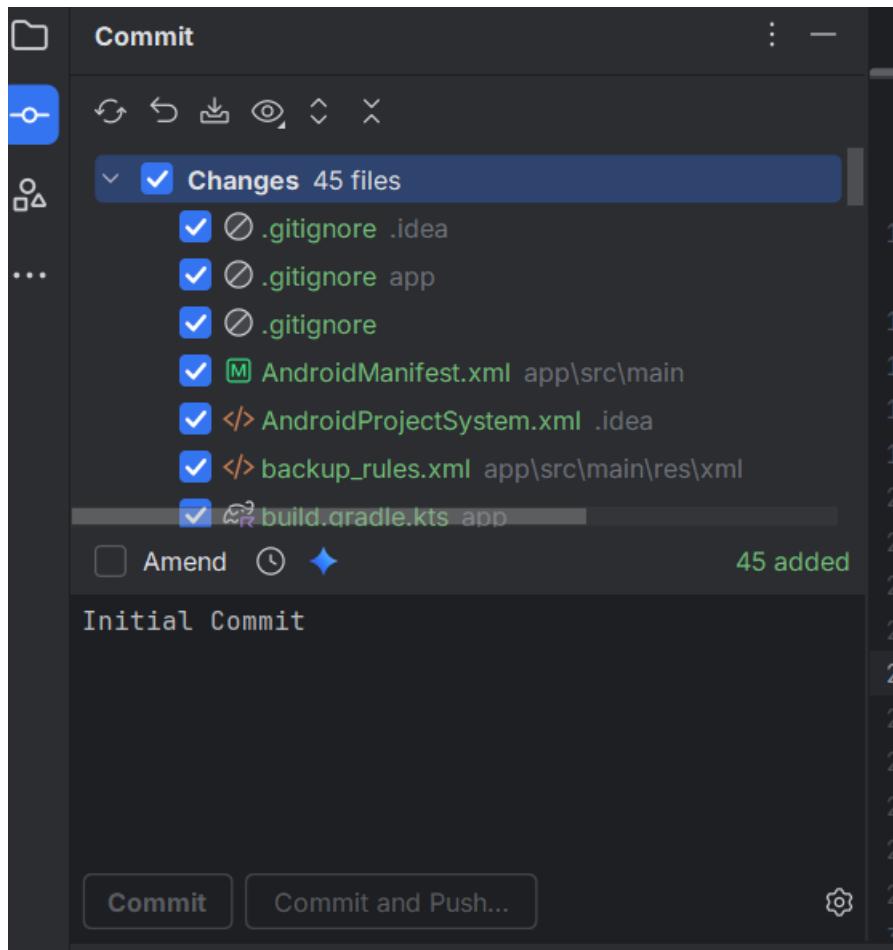


# Développement Mobile



**Create a new repository**

Repositories contain a project's files and version history. Have a project elsewhere? [Import a repository](#).  
Required fields are marked with an asterisk (\*).

**1 General**

Owner \* Repository name \*

iyedderbeli / SmartShop  
SmartShop is available.

Great repository names are short and memorable. How about [potential-octo-umbrella](#)?

Description

0 / 350 characters

**2 Configuration**

Choose visibility \* Public

Choose who can see and commit to this repository

Add README Off

READMEs can be used as longer descriptions. [About READMEs](#)

Add .gitignore No .gitignore

.gitignore tells git which files not to track. [About ignoring files](#)

Add license No license

Licenses explain how others can use your code. [About licenses](#)

**Create repository**

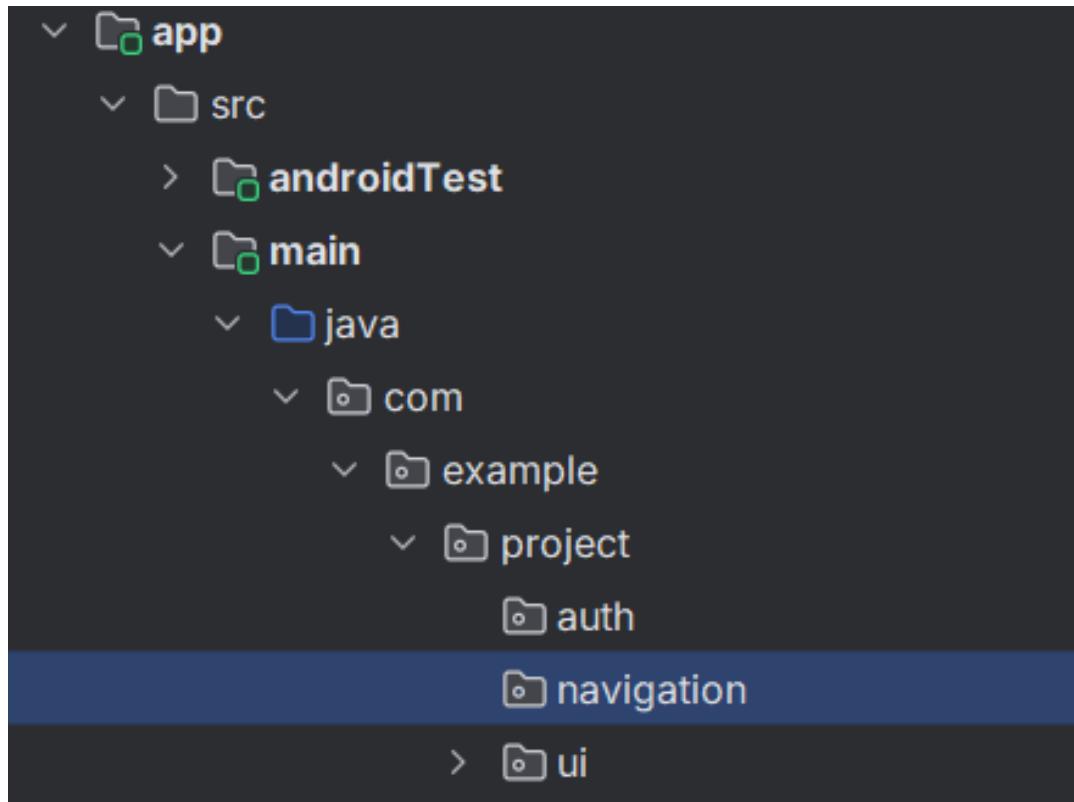
```
PS D:\Users\MSI\AndroidStudioProjects\Project> git remote add origin https://github.com/iyedderbeli/SmartShop.git
PS D:\Users\MSI\AndroidStudioProjects\Project> git branch -M main
PS D:\Users\MSI\AndroidStudioProjects\Project> git push -u origin main
Enumerating objects: 78, done.
Counting objects: 100% (78/78), done.
Delta compression using up to 12 threads
Compressing objects: 100% (63/63), done.
```

SmartShop Public

main 1 Branch 0 Tags

Go to file Add file Code

File	Commit	Time
.idea	Initial Commit	4 minutes ago
app	Initial Commit	4 minutes ago
gradle	Initial Commit	4 minutes ago
.gitignore	Initial Commit	4 minutes ago
build.gradle.kts	Initial Commit	4 minutes ago
gradle.properties	Initial Commit	4 minutes ago
gradlew	Initial Commit	4 minutes ago
gradlew.bat	Initial Commit	4 minutes ago
settings.gradle.kts	Initial Commit	4 minutes ago

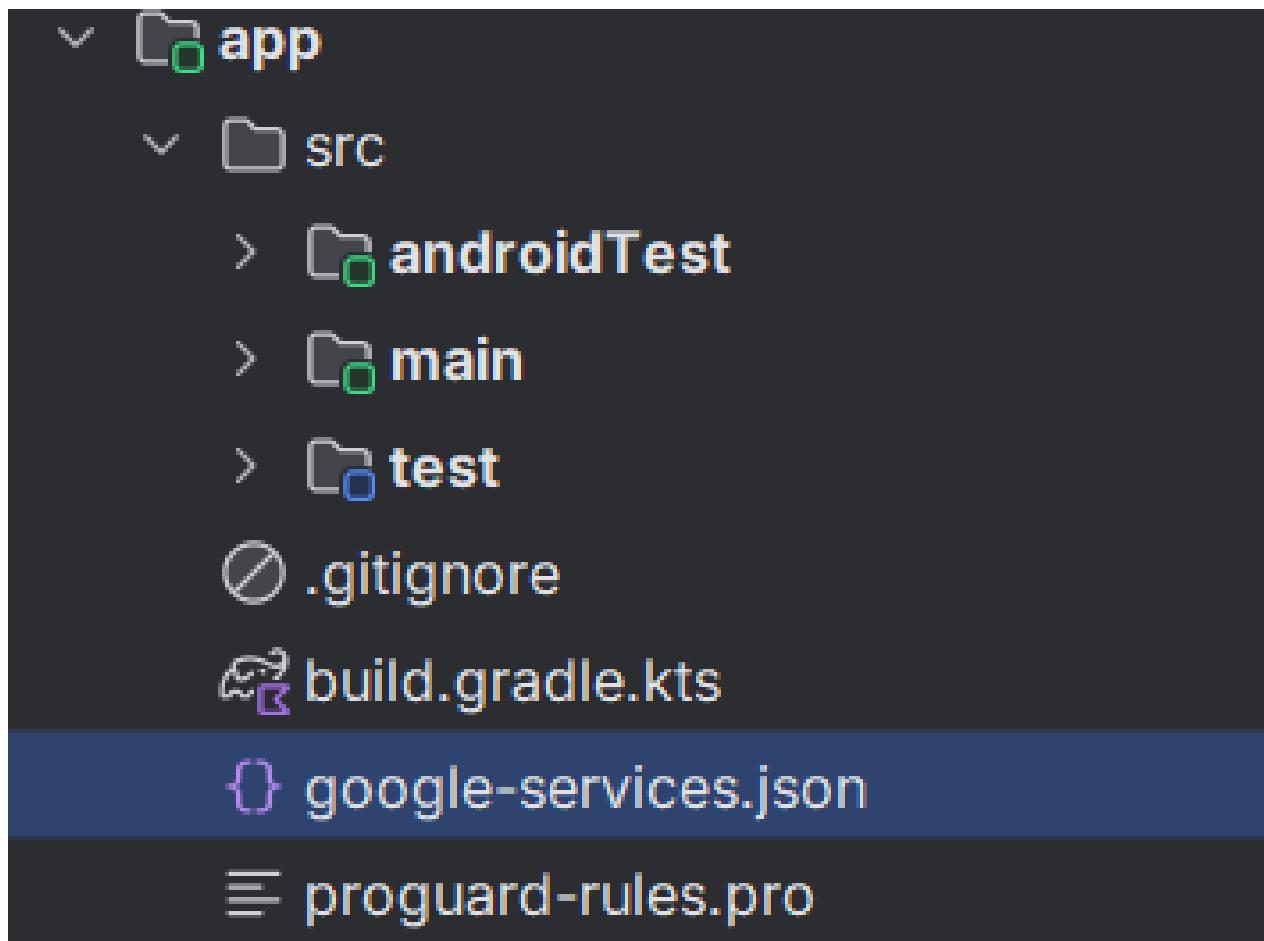


[+ Add app](#)

# Hello, Iyed

## Welcome to your Firebase project!

Gemini can give you suggestions for Firebase services and solutions for your app



## Authentication

Users    Sign-in method    Templates    Usage    Settings    Extensions

Sign-in providers

[Add new provider](#)

Provider	Status
Email/Password	<input checked="" type="checkbox"/> Enabled

## Authentication

Users Sign-in method Templates Usage Settings Extensions

The following Authentication features will stop working when Firebase Dynamic Links shuts down soon: email link authentication for mobile apps, as well as Cordova OAuth support for web apps.

Search by email address, phone number, or user UID [Add user](#) [C](#) [⋮](#)

Identifier	Providers	Created	Signed In	User UID
derbaliyed@gmail.com	✉	Dec 4, 2025		YqHmROCcZ7VrEYo37MlxvAr...

Rows per page: 50 ▾ 1 – 1 of 1 < >

## LoginScreen.kt :

```
package com.example.project.auth

import androidx.compose.foundation.layout.Column
import androidx.compose.foundation.layout.Spacer
import androidx.compose.foundation.layout.fillMaxSize
import androidx.compose.foundation.layout.fillMaxWidth
import androidx.compose.foundation.layout.height
import androidx.compose.foundation.layout.padding
import androidx.compose.foundation.layout.Arrangement
import androidx.compose.material3.Button
import androidx.compose.material3.Text
import androidx.compose.material3.TextField
import androidx.compose.material3.MaterialTheme
import androidx.compose.runtime.Composable
import androidx.compose.runtime.collectAsState
import androidx.compose.ui.Modifier
import androidx.compose.ui.graphics.Color
import androidx.compose.ui.text.input.PasswordVisualTransformation
import androidx.compose.ui.unit.dp

@Composable
fun LoginScreen(
    viewModel: LoginViewModel,
    onLoginSuccess: () -> Unit
) {
    val emailState = viewModel.email.collectAsState()
    val passwordState = viewModel.password.collectAsState()
    val errorState = viewModel.error.collectAsState()
    val loadingState = viewModel.loading.collectAsState()

    Column(
        modifier = Modifier
            .fillMaxSize()
            .padding(16.dp),
        verticalArrangement = Arrangement.Center
    ) {

        TextField(
            value = emailState.value,
            onValueChange = { viewModel.updateEmail(it) },
            label = { Text("Email") },
            colors = MaterialTheme.colors
        )

        Spacer(modifier = Modifier.height(16.dp))

        Button(
            onClick = { onLoginSuccess() },
            modifier = Modifier
                .fillMaxWidth()
                .padding(16.dp)
        ) {
            Text("Log In")
        }
    }
}
```

```

        modifier = Modifier.fillMaxWidth()
    )

    Spacer(modifier = Modifier.height(16.dp))

    TextField(
        value = passwordState.value,
        onValueChange = { viewModel.updatePassword(it) },
        label = { Text("Password") },
        visualTransformation = PasswordVisualTransformation(),
        modifier = Modifier.fillMaxWidth()
    )

    Spacer(modifier = Modifier.height(16.dp))

    Button(
        onClick = { viewModel.login(onLoginSuccess) },
        enabled = !loadingState.value,
        modifier = Modifier.fillMaxWidth()
    ) {
        Text(if (loadingState.value) "Loading..." else "Login")
    }

    errorState.value?.let { errorMsg ->
        Spacer(modifier = Modifier.height(8.dp))
        Text(text = errorMsg, color = Color.Red, style =
MaterialTheme.typography.bodyMedium)
    }
}
}

```

## LoginViewModel.kt :

```

package com.example.project.auth

import androidx.lifecycle.ViewModel
import com.google.firebase.auth.FirebaseAuth
import kotlinx.coroutines.flow.MutableStateFlow
import kotlinx.coroutines.flow.StateFlow

class LoginViewModel : ViewModel() {

    private val auth: FirebaseAuth = FirebaseAuth.getInstance()

    private val _email = MutableStateFlow("")
    val email: StateFlow<String> = _email

    private val _password = MutableStateFlow("")
    val password: StateFlow<String> = _password

    private val _error = MutableStateFlow<String?>(null)
    val error: StateFlow<String?> = _error

    private val _loading = MutableStateFlow(false)
    val loading: StateFlow<Boolean> = _loading

    fun updateEmail(value: String) {
        _email.value = value
    }

    fun updatePassword(value: String) {
        _password.value = value
    }

    fun login(onSuccess: () -> Unit) {

```

```

        _loading.value = true
        _error.value = null

        auth.signInWithEmailAndPassword(_email.value, _password.value)
            .addOnCompleteListener { task ->
                _loading.value = false
                if (task.isSuccessful) {
                    onSuccess()
                } else {
                    _error.value = task.exception?.message ?: "Login failed"
                }
            }
    }
}

```

## AppNav.kt :

```

package com.example.project.navigation

import androidx.compose.runtime.Composable
import androidx.compose.material3.Text
import androidx.navigation.compose.NavHost
import androidx.navigation.compose.composable
import androidx.navigation.compose.rememberNavController
import com.example.project.auth.LoginScreen
import com.example.project.auth.LoginViewModel

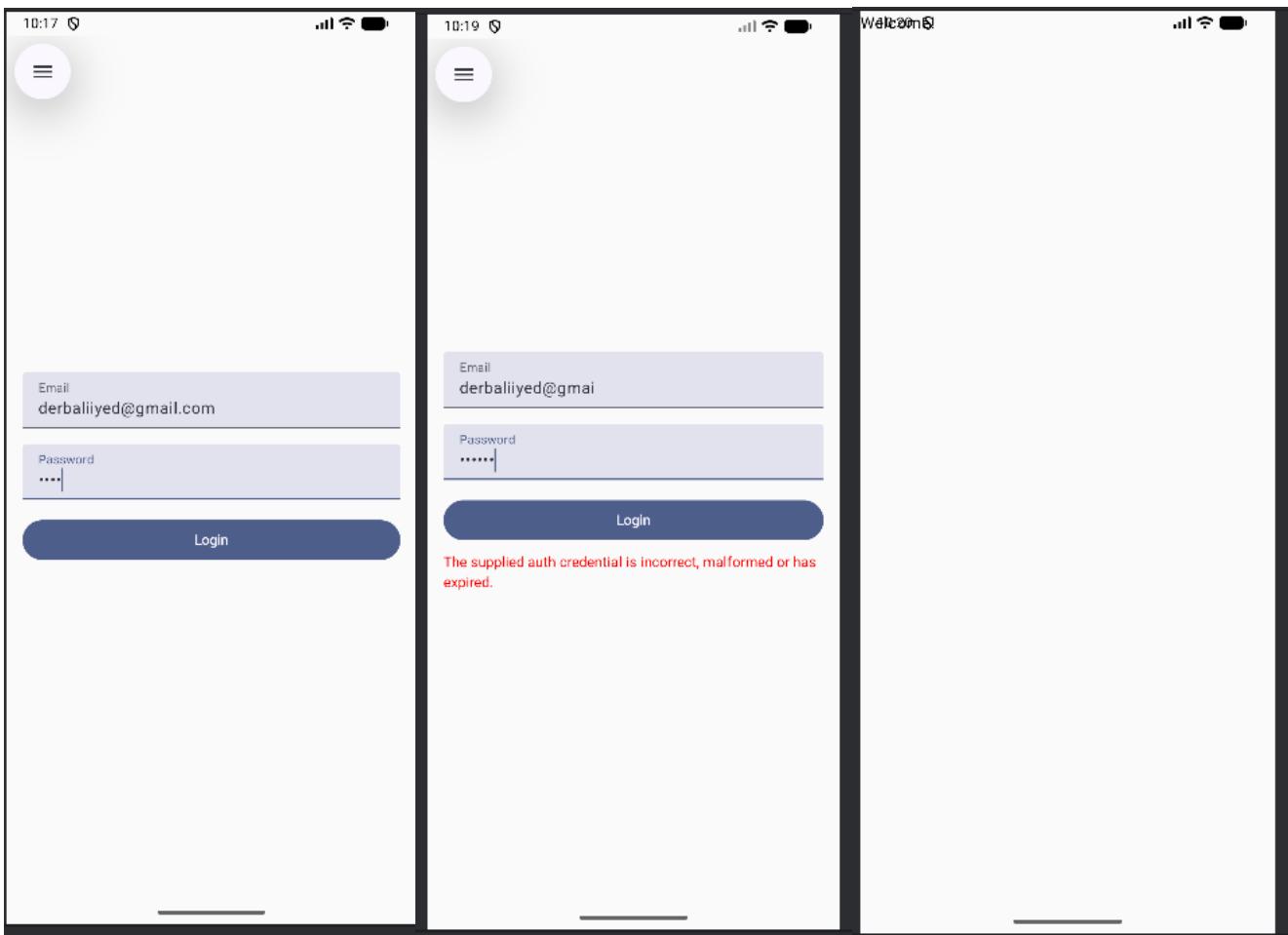
@Composable
fun AppNav() {
    val navController = rememberNavController()

    NavHost(
        navController = navController,
        startDestination = "login"
    ) {

        composable("login") {
            LoginScreen(
                viewModel = LoginViewModel(),
                onLoginSuccess = { navController.navigate("home") }
            )
        }

        composable("home") {
            Text("Welcome!")
        }
    }
}

```



```
PS D:\Users\MSI\AndroidStudioProjects\Project> git add .
warning: in the working copy of 'app/build.gradle.kts', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'app/src/main/java/com/example/project/MainActivity.kt', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'build.gradle.kts', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of '.kotlin/errors/errors-1764881807704.log', LF will be replaced by CRLF the next time Git touches it
PS D:\Users\MSI\AndroidStudioProjects\Project> git commit -m "Login + Navigation"
[main a02be2b] Login + Navigation
8 files changed, 192 insertions(+), 22 deletions(-)
create mode 100644 .kotlin/errors/errors-1764881807704.log
create mode 100644 app/google-services.json
create mode 100644 app/src/main/java/com/example/project/auth/LoginScreen.kt
create mode 100644 app/src/main/java/com/example/project/auth/LoginViewModel.kt
create mode 100644 app/src/main/java/com/example/project/navigation/AppNav.kt
PS D:\Users\MSI\AndroidStudioProjects\Project> git push
Enumerating objects: 31, done.
Counting objects: 100% (31/31), done.
Delta compression using up to 12 threads
Compressing objects: 100% (16/16), done.
Writing objects: 100% (21/21), 3.65 KiB | 746.00 KiB/s, done.
Total 21 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
remote: This repository moved. Please use the new location:
remote: https://github.com/Iyedderbeli/SmartShop.git
To https://github.com/Iyedderbeli/SmartShop.git
 b9f03b3..a02be2b main -> main
```

## PART 2:

# Cloud Firestore

Realtime updates, powerful queries, automatic scaling, and MongoDB compatibility

[Create database](#) [Ask Gemini](#)



### Create a database

3 Configure

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.

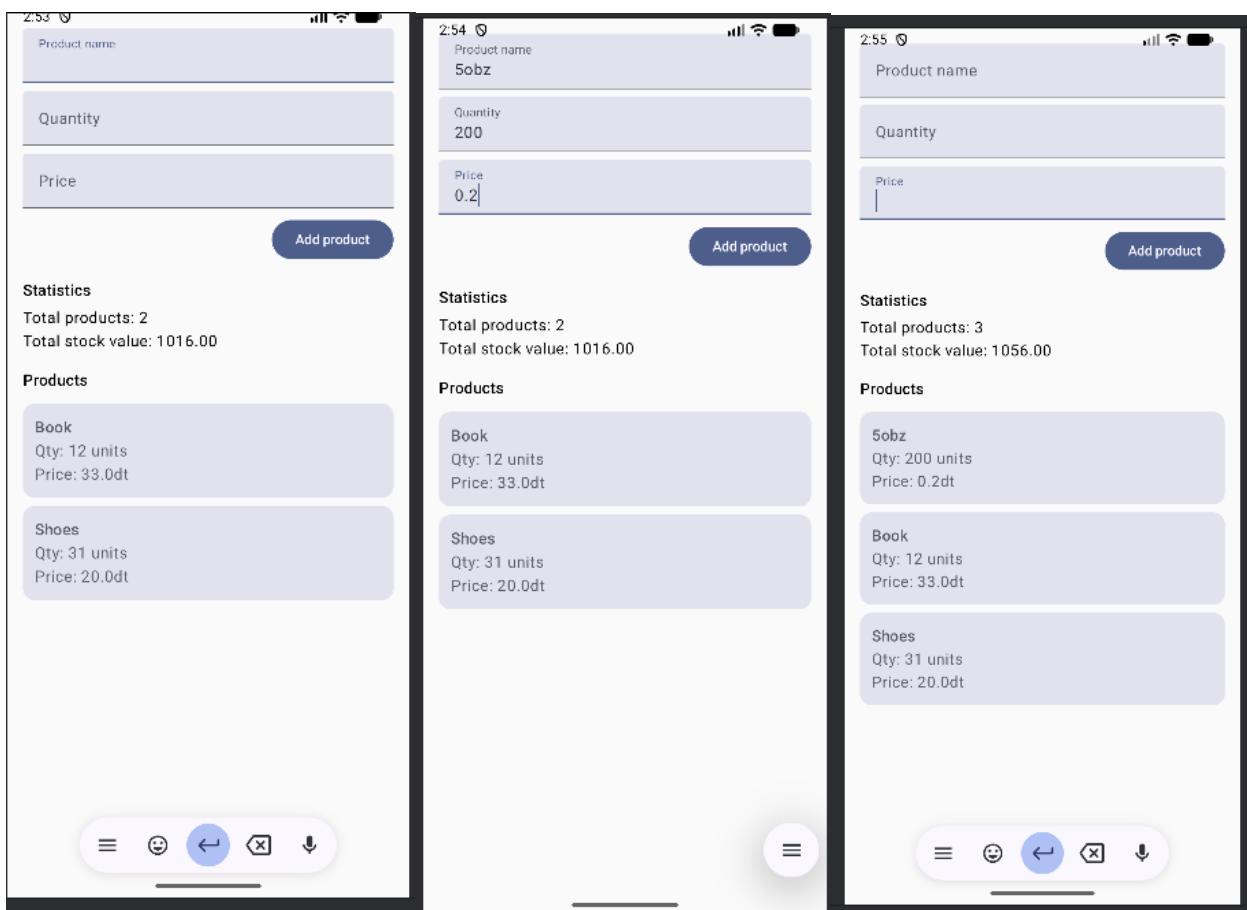
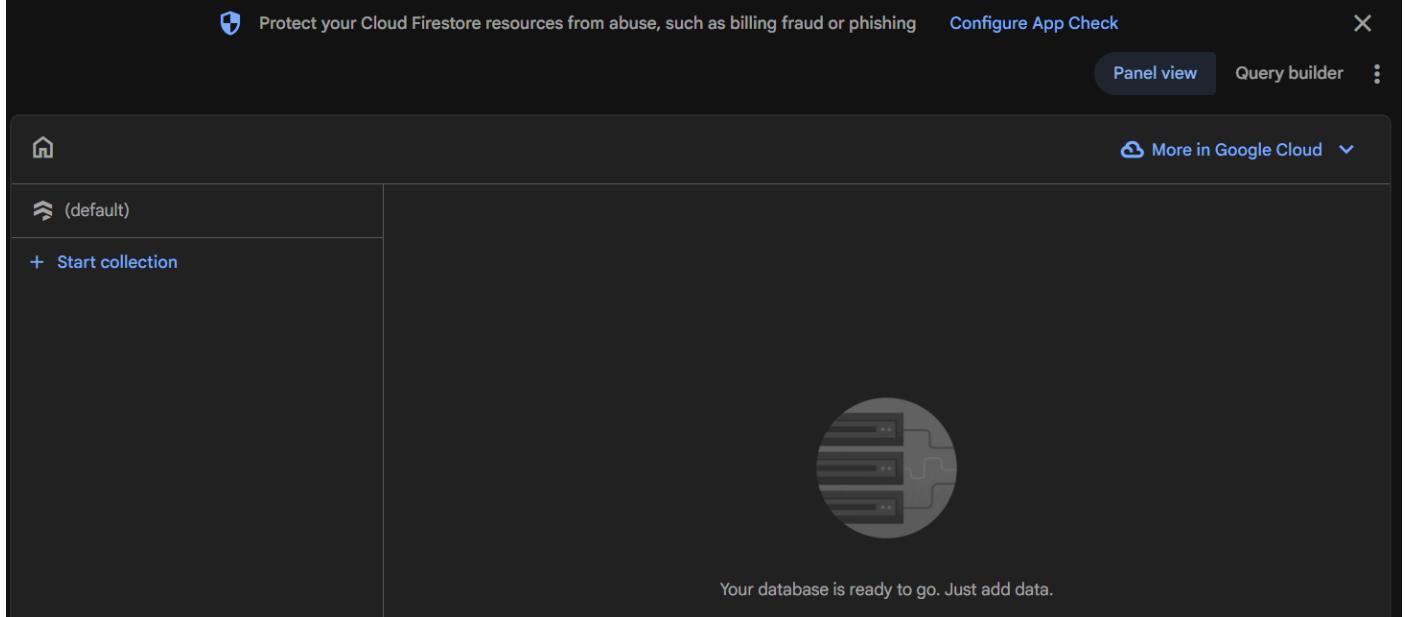
Start in test mode  
Your data is open by default to enable quick setup. However, you must update your security rules within 30 days to enable long-term client read/write access.

```
rules_version = '2';

service cloud.firestore {
    match /databases/{database}/documents {
        match /{document=**} {
            allow read, write: if
                request.time < timestamp.date(2026, 1, 4);
        }
    }
}
```

**!** The default security rules for test mode allow anyone with your database reference to view, edit and delete all data in your database for the next 30 days

[Cancel](#) [Create](#)

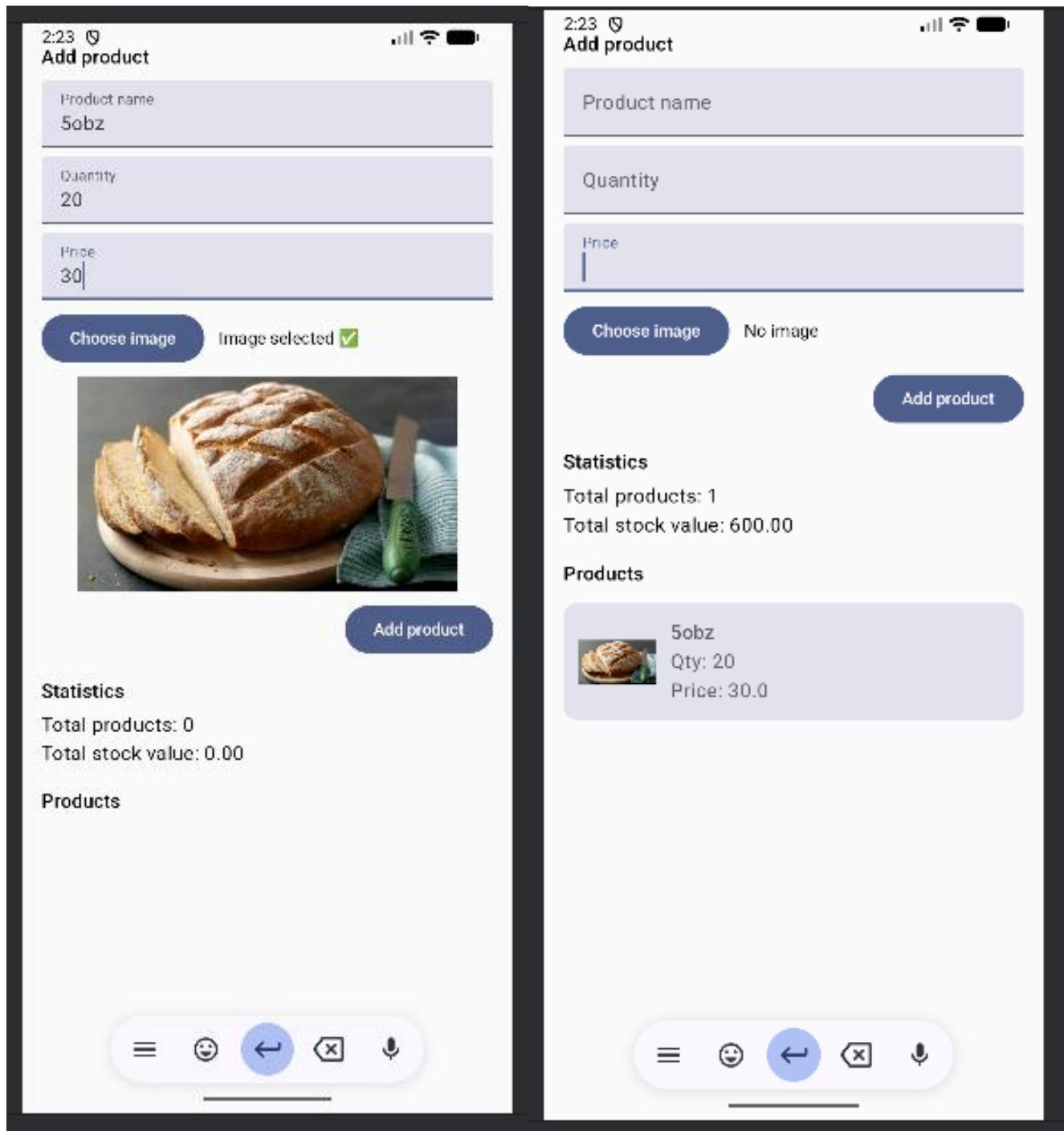
[Add database](#)[Ask Gemini about the core concepts to use Firestore](#)[Data](#)   [Rules](#)   [Indexes](#)   [Disaster Recovery](#)   [Usage](#)   [Extensions](#)

The screenshot shows the Google Cloud Firestore interface. On the left, there's a sidebar with a 'Database' tab selected. It shows a 'products' collection with a single document named 'zm4LiWDH9Vk2IzKMI5TD'. This document contains three fields: 'name' with value '5obz', 'price' with value 0.2, and 'quantity' with value 200.

(default)	products	zm4LiWDH9Vk2IzKMI5TD
+ Start collection	+ Add document	+ Start collection
products	9Sqgb04pDAdTBbpW5D6A	+ Add field
	zm4LiWDH9Vk2IzKMI5TD	name: "5obz"
		price: 0.2
		quantity: 200

```
PS D:\Users\MSI\AndroidStudioProjects\Project> git push
Enumerating objects: 40, done.
Counting objects: 100% (40/40), done.
Delta compression using up to 12 threads
Compressing objects: 100% (21/21), done.
Writing objects: 100% (26/26), 5.25 KiB | 1.05 MiB/s, done.
Total 26 (delta 7), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (7/7), completed with 7 local objects.
remote: This repository moved. Please use the new location:
remote:   https://github.com/Iyedderbeli/SmartShop.git
To https://github.com/iyedderbeli/SmartShop.git
  a02be2b..f718573 main -> main
PS D:\Users\MSI\AndroidStudioProjects\Project>
```

## Adding Images to products

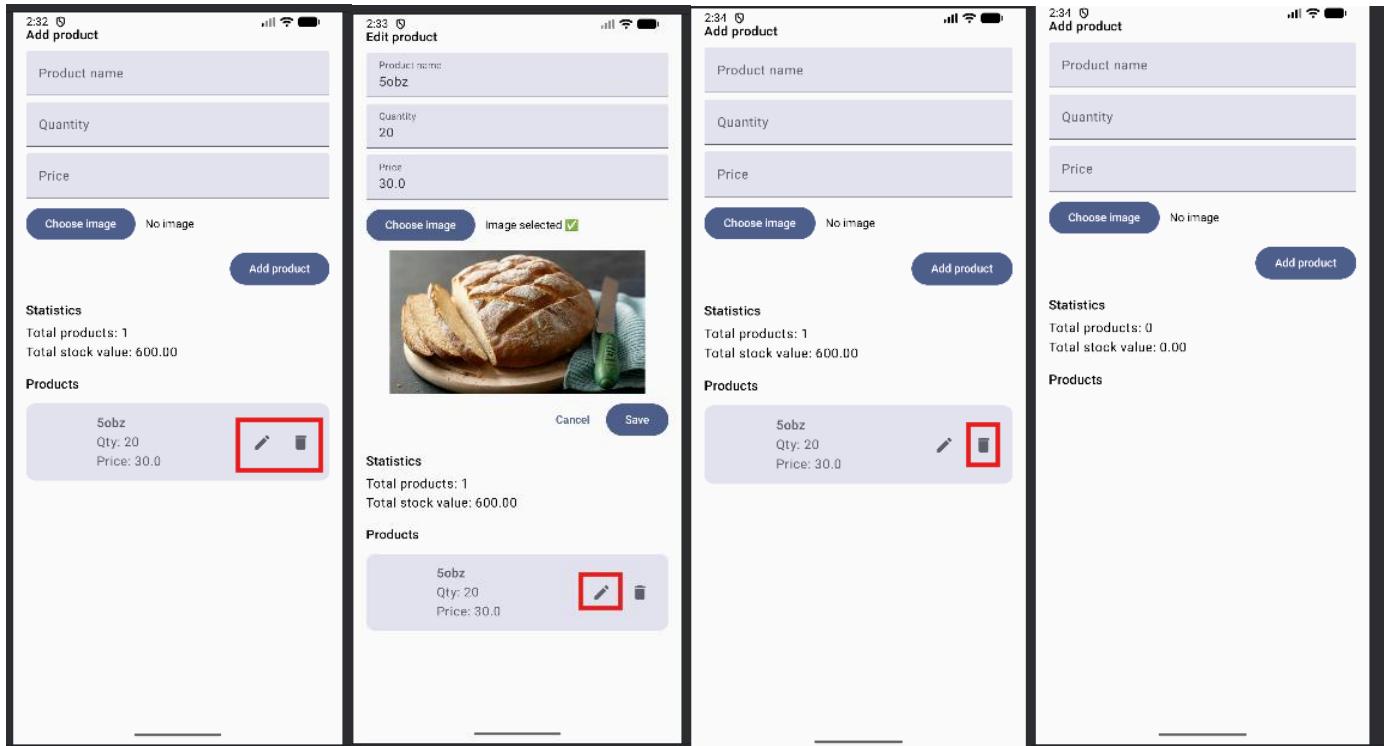


```

PS D:\Users\MSI\AndroidStudioProjects\Project> git add .
warning: in the working copy of 'app/build.gradle.kts', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'app/src/main/java/com/example/project/MainActivity.kt', LF will be replaced by CRLF the next time Git touches it
PS D:\Users\MSI\AndroidStudioProjects\Project> git commit -m "Added Images"
[main 30cf98b] Added Images
 9 files changed, 112 insertions(+), 91 deletions(-)
PS D:\Users\MSI\AndroidStudioProjects\Project> git push
Enumerating objects: 43, done.
Counting objects: 100% (43/43), done.
Delta compression using up to 12 threads
Compressing objects: 100% (17/17), done.
Writing objects: 100% (22/22), 2.87 KiB | 734.00 KiB/s, done.
Total 22 (delta 11), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (11/11), completed with 11 local objects.
remote: This repository moved. Please use the new location:
remote:   https://github.com/Iyedderbeli/SmartShop.git
To https://github.com/Iyedderbeli/SmartShop.git
 f718573..30cf98b main -> main
PS D:\Users\MSI\AndroidStudioProjects\Project>

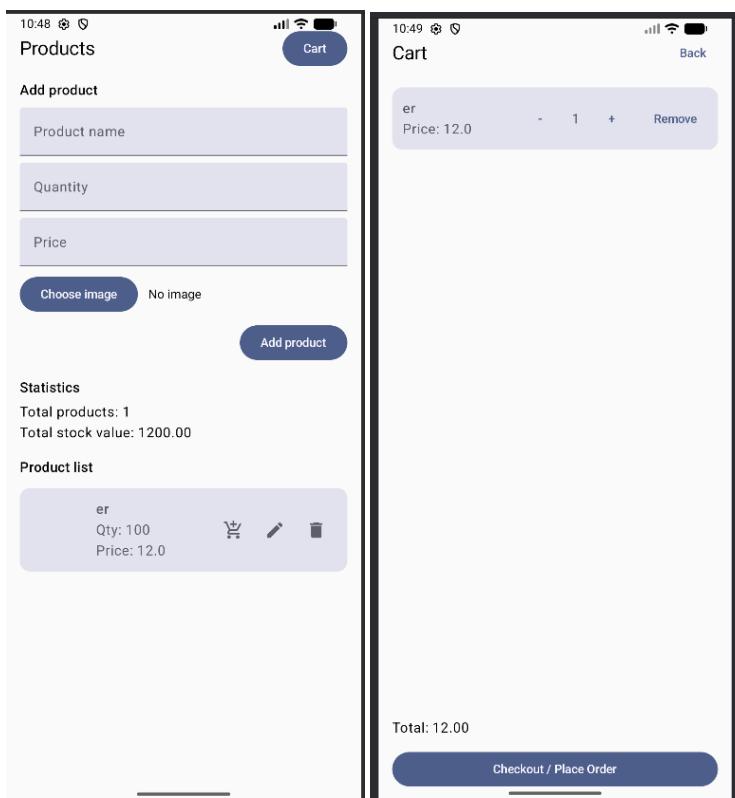
```

# Modifying + Deleting Products



```
PS D:\Users\MSI\AndroidStudioProjects\Project> git add .
PS D:\Users\MSI\AndroidStudioProjects\Project> git commit -m "Modify + Delete"
[main 0a1b9ce] Modify + Delete
 3 files changed, 96 insertions(+), 19 deletions(-)
PS D:\Users\MSI\AndroidstudioProjects\Project> git push
Enumerating objects: 25, done.
Counting objects: 100% (25/25), done.
Delta compression using up to 12 threads
Compressing objects: 100% (10/10), done.
Writing objects: 100% (13/13), 1.86 KiB | 953.00 KiB/s, done.
Total 13 (delta 6), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (6/6), completed with 6 local objects.
remote: This repository moved. Please use the new location:
remote:   https://github.com/Iyedderbeli/SmartShop.git
To https://github.com/Iyedderbeli/SmartShop.git
 30cf98b..0a1b9ce main -> main
```

# Cart + Checkout



```
PS D:\Users\MSI\AndroidStudioProjects\Project> git add .
warning: in the working copy of 'app/build.gradle.kts', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of '.idea/inspectionProfiles/Project_Default.xml', LF will be replaced by CRLF the next time Git touches it
PS D:\Users\MSI\AndroidStudioProjects\Project> git commit -m "Cart + Checkout"
[main 5422909] Cart + Checkout
14 files changed, 434 insertions(+), 9 deletions(-)
create mode 100644 .idea/inspectionProfiles/Project_Default.xml
create mode 100644 Report.pdf
create mode 100644 app/src/main/java/com/example/project/data/local/CartDao.kt
create mode 100644 app/src/main/java/com/example/project/data/local/CartItemEntity.kt
create mode 100644 app/src/main/java/com/example/project/data/local/OrderDao.kt
create mode 100644 app/src/main/java/com/example/project/data/local/OrderEntity.kt
create mode 100644 app/src/main/java/com/example/project/data/local/OrderItemEntity.kt
create mode 100644 app/src/main/java/com/example/project/orders/CartOrderRepository.kt
create mode 100644 app/src/main/java/com/example/project/orders/CartScreen.kt
create mode 100644 app/src/main/java/com/example/project/orders/CartViewModel.kt
PS D:\Users\MSI\AndroidStudioProjects\Project> git push
```