**Báo cáo bài tập lớn giữa kì**

**Họ tên: Nguyễn Thành Tuấn**

**Mã SV: 21IT117**

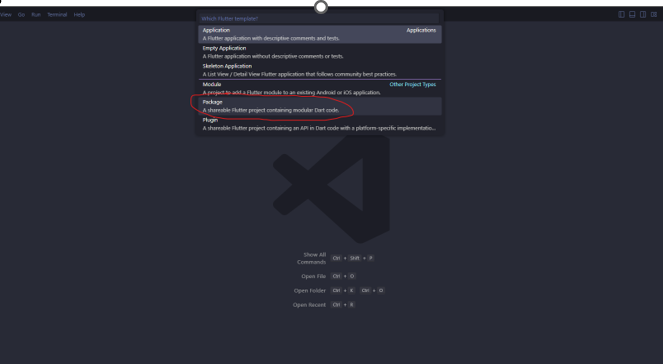
**HP: Lập trình đa nền tảng (6)**

**Bài tập giữa kì: Xây dựng app nhập và hiển thị, sửa, xoá sản phẩm ở phân hệ admin**

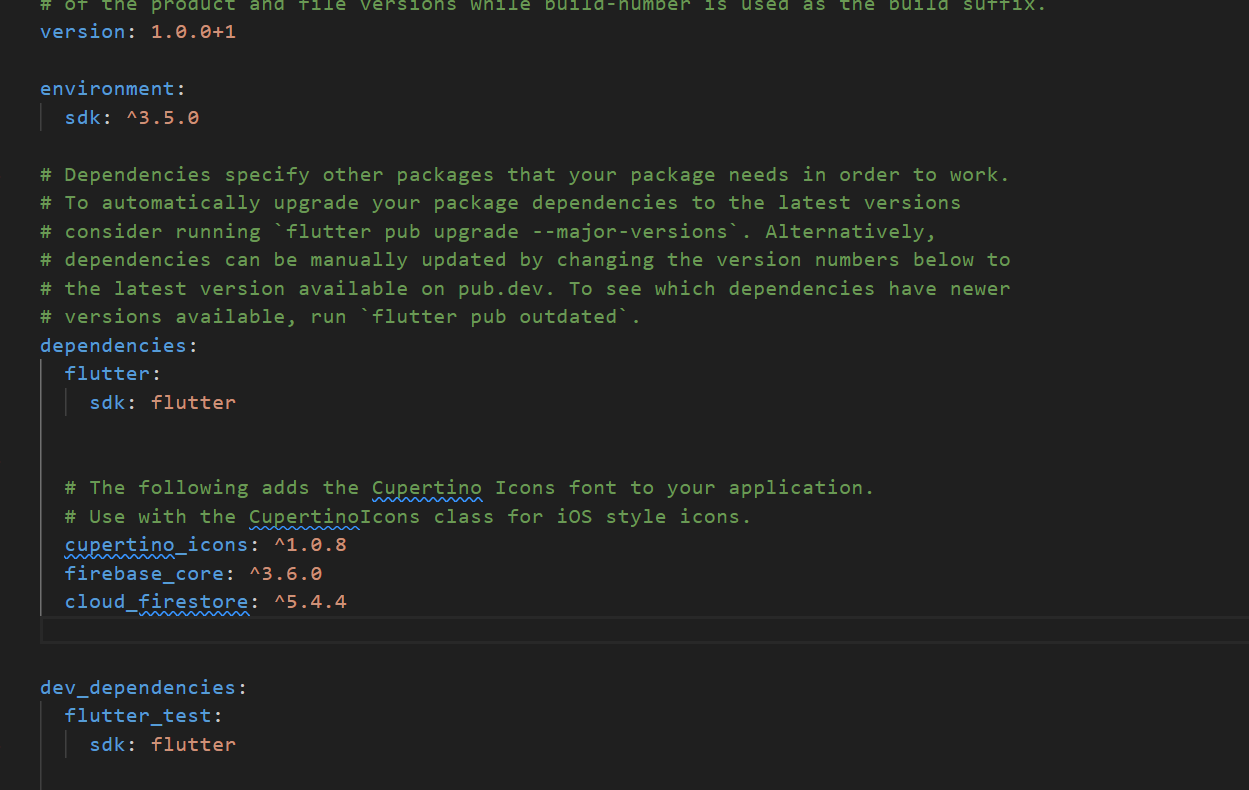
**Bài tập này sử dụng flutter + firebase.**

**Các bước triển khai dự án:**

**Tạo project flutter trên visual studio code và chọn template Package**



**Cài đặt 1 số dependencies cho dự án ở thư mục pubspec.yaml**

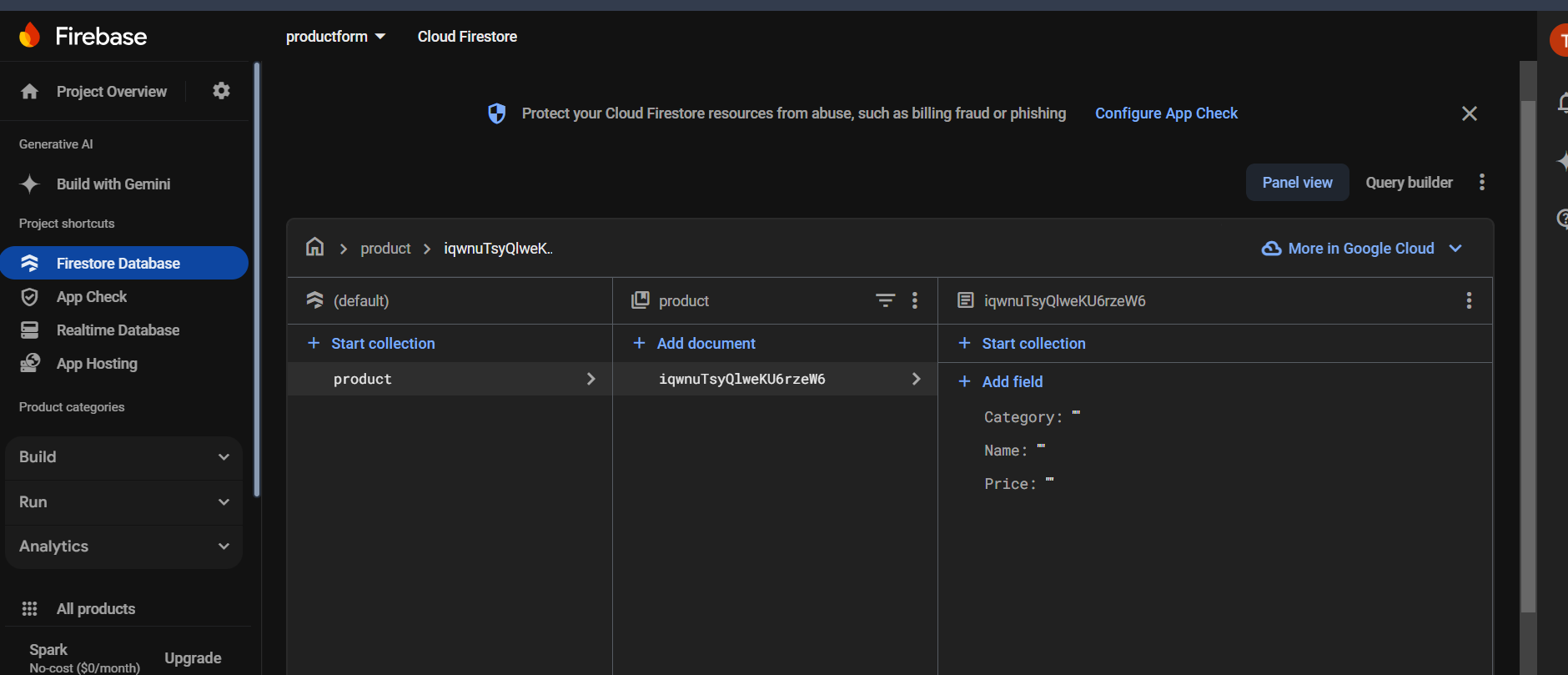


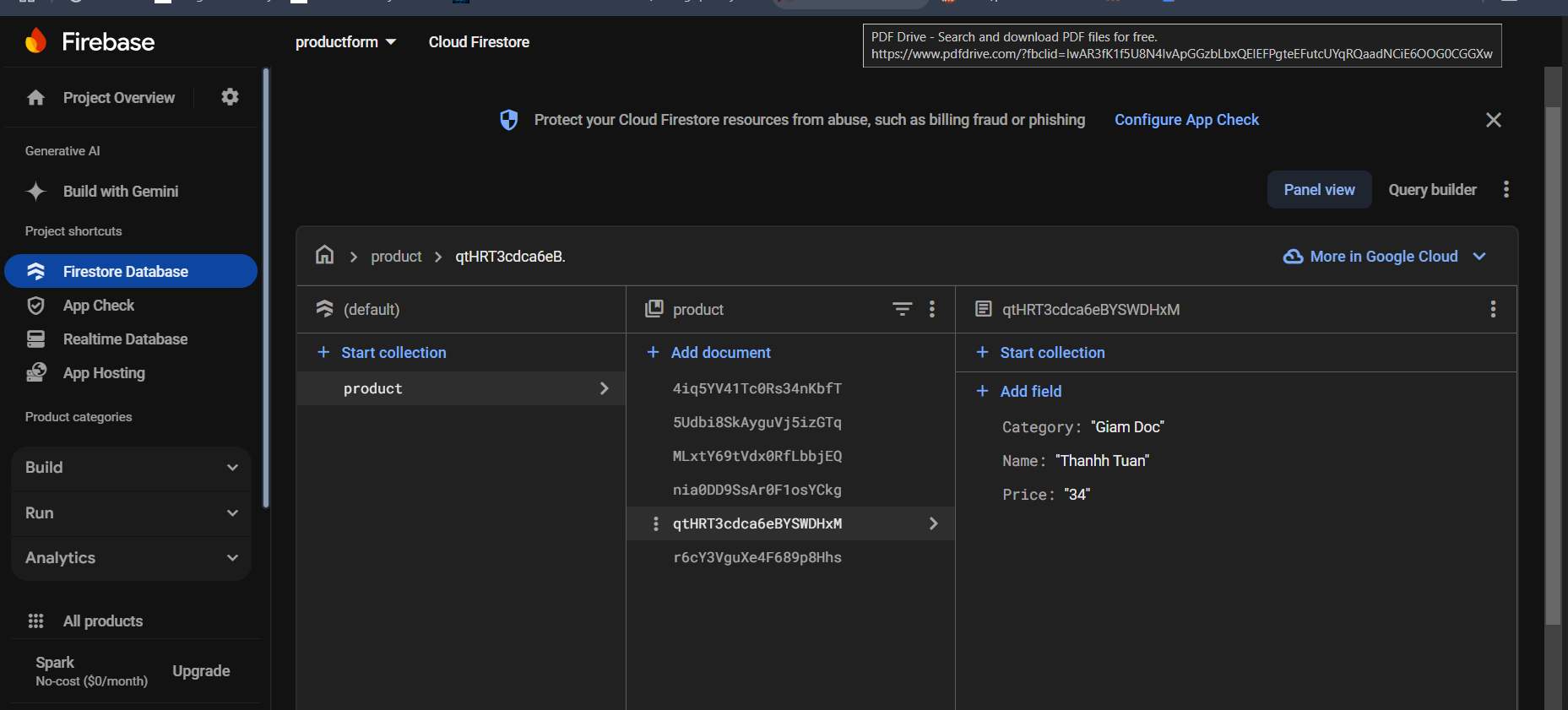
Truy cập vào Firebase Console, tạo một dự án Firebase mới và thêm ứng dụng Flutter vào Firebase.

 Sau khi thêm Firebase vào dự án, tải xuống các tệp cấu hình (google-services.json cho Android và GoogleService-Info.plist cho iOS) và thêm vào thư mục tương ứng trong dự án Flutter.

Thêm các thư viện Firebase vào tệp pubspec.yaml để cài đặt Firebase cho Flutter:

**Firebase**

****

****

**Khởi tạo Firebase trong ứng dụng**

Trong file firebase\_options.dart, khởi tạo Firebase trong ứng dụng. Việc khởi tạo này là cần thiết để sử dụng các dịch vụ của Firebase như Firebase Realtime Database và Firebase Storage.

// File generated by FlutterFire CLI.

// ignore\_for\_file: type=lint

import 'package:firebase\_core/firebase\_core.dart' show FirebaseOptions;

import 'package:flutter/foundation.dart'

    show defaultTargetPlatform, kIsWeb, TargetPlatform;

/// Default [FirebaseOptions] for use with your Firebase apps.

///

/// Example:

/// ```dart

/// import 'firebase\_options.dart';

/// // ...

/// await Firebase.initializeApp(

///   options: DefaultFirebaseOptions.currentPlatform,

/// );

/// ```

class DefaultFirebaseOptions {

  static FirebaseOptions get currentPlatform {

    if (kIsWeb) {

      return web;

    }

    switch (defaultTargetPlatform) {

      case TargetPlatform.android:

        return android;

      case TargetPlatform.iOS:

        return ios;

      case TargetPlatform.macOS:

        return macos;

      case TargetPlatform.windows:

        return windows;

      case TargetPlatform.linux:

        throw UnsupportedError(

          'DefaultFirebaseOptions have not been configured for linux - '

          'you can reconfigure this by running the FlutterFire CLI again.',

        );

      default:

        throw UnsupportedError(

          'DefaultFirebaseOptions are not supported for this platform.',

        );

    }

  }

  static const FirebaseOptions web = FirebaseOptions(

    apiKey: 'AIzaSyAwhdonjDLA0YUL-Hx8kUh45ASJzlHwGK8',

    appId: '1:907615055976:web:5c59c5b168bfc9950fcbdd',

    messagingSenderId: '907615055976',

    projectId: 'productform-1da60',

    authDomain: 'productform-1da60.firebaseapp.com',

    storageBucket: 'productform-1da60.appspot.com',

    measurementId: 'G-6D316SD0T7',

  );

  static const FirebaseOptions android = FirebaseOptions(

    apiKey: 'AIzaSyC5OFPJjy7l5TckECidXJMzSTfcWTiAA-0',

    appId: '1:907615055976:android:9ddec9068457f26f0fcbdd',

    messagingSenderId: '907615055976',

    projectId: 'productform-1da60',

    storageBucket: 'productform-1da60.appspot.com',

  );

  static const FirebaseOptions ios = FirebaseOptions(

    apiKey: 'AIzaSyA8SKyBpX0iWUI8GANQuyuMaSLgONW4JBQ',

    appId: '1:907615055976:ios:c064584c7e923b800fcbdd',

    messagingSenderId: '907615055976',

    projectId: 'productform-1da60',

    storageBucket: 'productform-1da60.appspot.com',

    iosBundleId: 'com.example.productform',

  );

  static const FirebaseOptions macos = FirebaseOptions(

    apiKey: 'AIzaSyA8SKyBpX0iWUI8GANQuyuMaSLgONW4JBQ',

    appId: '1:907615055976:ios:c064584c7e923b800fcbdd',

    messagingSenderId: '907615055976',

    projectId: 'productform-1da60',

    storageBucket: 'productform-1da60.appspot.com',

    iosBundleId: 'com.example.productform',

  );

  static const FirebaseOptions windows = FirebaseOptions(

    apiKey: 'AIzaSyAwhdonjDLA0YUL-Hx8kUh45ASJzlHwGK8',

    appId: '1:907615055976:web:e754ea2bc1728d610fcbdd',

    messagingSenderId: '907615055976',

    projectId: 'productform-1da60',

    authDomain: 'productform-1da60.firebaseapp.com',

    storageBucket: 'productform-1da60.appspot.com',

    measurementId: 'G-277LBCFC1Z',

  );

}

**Tạo trang Home\_screen**

Đây là một trang chính của ứng dụng Flutter, cho phép người dùng xem danh sách sản phẩm được lưu trữ trong Firebase Firestore, cũng như thực hiện các thao tác thêm, chỉnh sửa và xóa sản phẩm.

import 'package:cloud\_firestore/cloud\_firestore.dart';

import 'package:flutter/material.dart';

class HomeScreen extends StatefulWidget {

  const HomeScreen({super.key});

  @override

  State<HomeScreen> createState() => \_HomeScreenState();

}

class \_HomeScreenState extends State<HomeScreen> {

  final TextEditingController \_nameController = TextEditingController();

  final TextEditingController \_categoryController = TextEditingController();

  final TextEditingController \_priceController = TextEditingController();

  CollectionReference products =

      FirebaseFirestore.instance.collection('product');

  void \_addProduct() {

    products.add({

      'Name': \_nameController.text,

      'Category': \_categoryController.text,

      'Price': \_priceController.text,

    });

    \_nameController.clear();

    \_categoryController.clear();

    \_priceController.clear();

  }

  void \_deleteProduct(String productId) {

    products.doc(productId).delete();

  }

  void \_editProduct(DocumentSnapshot product) {

    \_nameController.text = product['Name'];

    \_categoryController.text = product['Category'];

    \_priceController.text = product['Price'];

    showDialog(

        context: context,

        builder: (context) {

          return AlertDialog(

            title: Text('Edit product'),

            content: Column(

              mainAxisSize: MainAxisSize.min,

              children: [

                TextFormField(

                  controller: \_nameController,

                  decoration: InputDecoration(labelText: 'Tên NV'),

                ),

                SizedBox(height: 8),

                TextFormField(

                  controller: \_categoryController,

                  decoration: InputDecoration(labelText: 'Vai Trò'),

                ),

                SizedBox(height: 8),

                TextFormField(

                  controller: \_priceController,

                  decoration: InputDecoration(labelText: 'Tuổi'),

                ),

              ],

            ),

            actions: [

              TextButton(

                  onPressed: () {

                    Navigator.pop(context);

                  },

                  child: Text('Cancel')),

              ElevatedButton(

                  onPressed: () {

                    \_updateProduct(product.id);

                  },

                  child: Text('Update')),

            ],

          );

        });

  }

  void \_updateProduct(String productId) {

    products.doc(productId).update({

      'Name': \_nameController.text,

      'Category': \_categoryController.text,

      'Price': \_priceController.text,

    });

  }

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      appBar: AppBar(

        title: Text("DANH SÁCH NHÂN SỰ"),

      ),

      body: Padding(

          padding: const EdgeInsets.all(16.0),

          child: Column(

            children: [

              TextField(

                controller: \_nameController,

                decoration: InputDecoration(labelText: 'Tên nhân sự'),

              ),

              SizedBox(height: 16),

              TextField(

                controller: \_categoryController,

                decoration: InputDecoration(labelText: 'Vị trí làm việc'),

              ),

              SizedBox(height: 16),

              TextField(

                controller: \_priceController,

                keyboardType: TextInputType.number,

                decoration: InputDecoration(labelText: 'Tuổi'),

              ),

              SizedBox(height: 16),

              ElevatedButton(

                onPressed: () {

                  \_addProduct();

                },

                child: Text("Thêm nhân sự"),

              ),

              SizedBox(height: 16),

              Expanded(

                  child: StreamBuilder(

                stream: products.snapshots(),

                builder: (context, snapshot) {

                  if (!snapshot.hasData) {

                    return CircularProgressIndicator();

                  }

                  return ListView.builder(

                    itemCount: snapshot.data!.docs.length,

                    itemBuilder: (context, index) {

                      var product = snapshot.data!.docs[index];

                      return Dismissible(

                        key: Key(product.id),

                        background: Container(

                            color: Color.fromARGB(255, 82, 255, 252),

                            alignment: Alignment.centerRight,

                            padding: EdgeInsets.only(right: 16),

                            child: Icon(

                              Icons.delete,

                              color: Colors.white,

                            )),

                        onDismissed: (direction) {

                          \_deleteProduct(product.id);

                        },

                        direction: DismissDirection.endToStart,

                        child: Card(

                          elevation: 4,

                          margin: EdgeInsets.symmetric(vertical: 8),

                          child: ListTile(

                            title: Text('Tên ns: ${product['Name']}'),

                            subtitle: Column(

                              crossAxisAlignment: CrossAxisAlignment.start,

                              children: [

                                Text('Vai trò ns: ${product['Category']}'),

                                Text('Tuổi ns: ${product['Price']}'),

                              ],

                            ),

                            trailing: Column(

                              children: [

                                IconButton(

                                  onPressed: () {

                                    \_editProduct(product);

                                  },

                                  icon: Icon(Icons.edit),

                                ),

                              ],

                            ),

                          ),

                        ),

                      );

                    },

                  );

                },

              )),

            ],

          )),

    );

  }

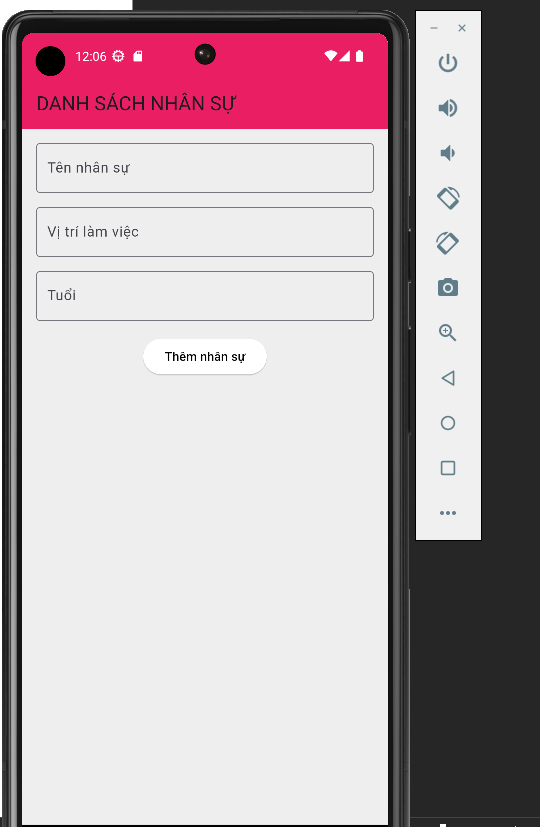
}

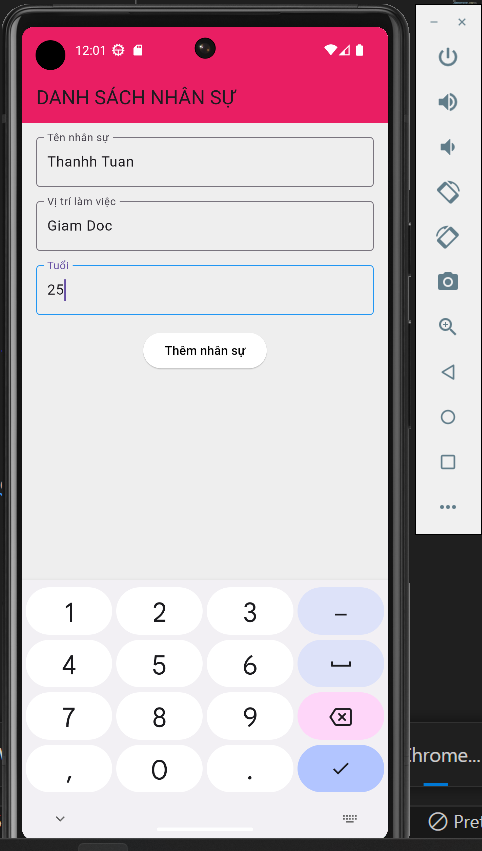
extension on Object {

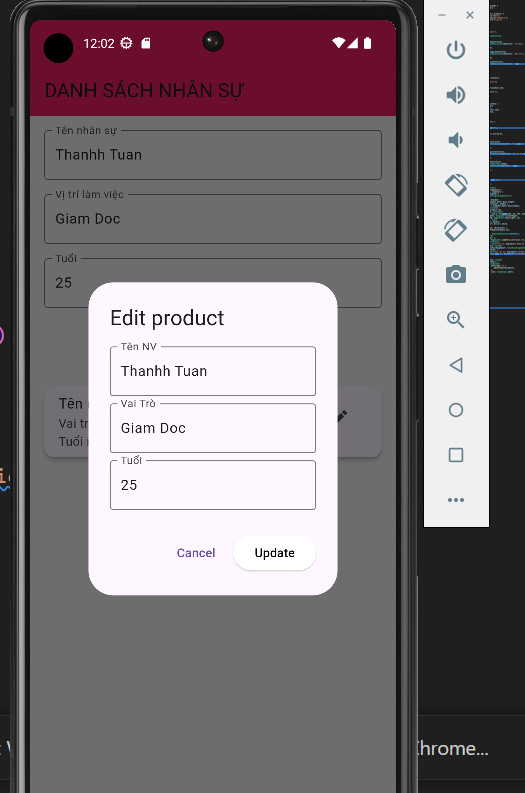
  get docs => null;

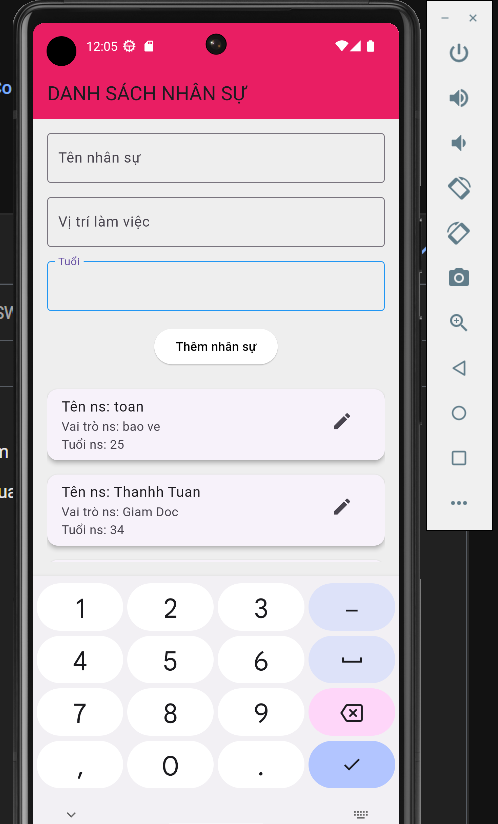
}

**Demo dự án**

****

****

****

****

**3. Kết luận**

Dự án này đã hoàn thành việc tích hợp Firebase với ứng dụng Flutter và cung cấp các chức năng CRUD cho quản lý sản phẩm. Thông qua ứng dụng, người dùng có thể thêm, xem, chỉnh sửa và xóa sản phẩm.