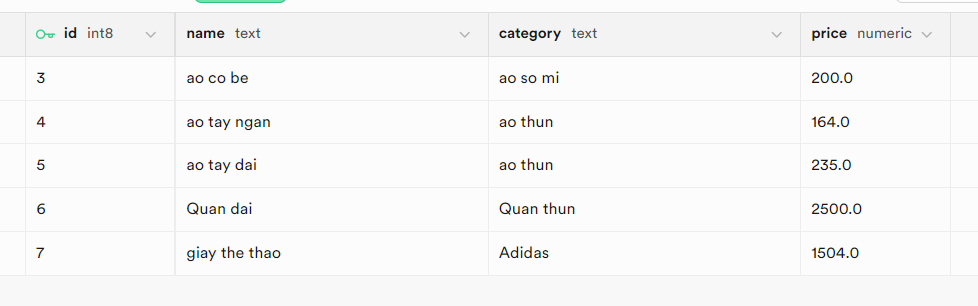
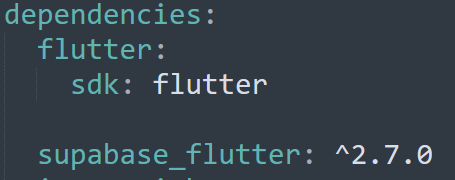
Các Bước Làm Dự Án Giữa Kỳ

**Bước 1: Tạo dự án supabase**

**Bước 2: Tạo bảng trong dự án supabase**

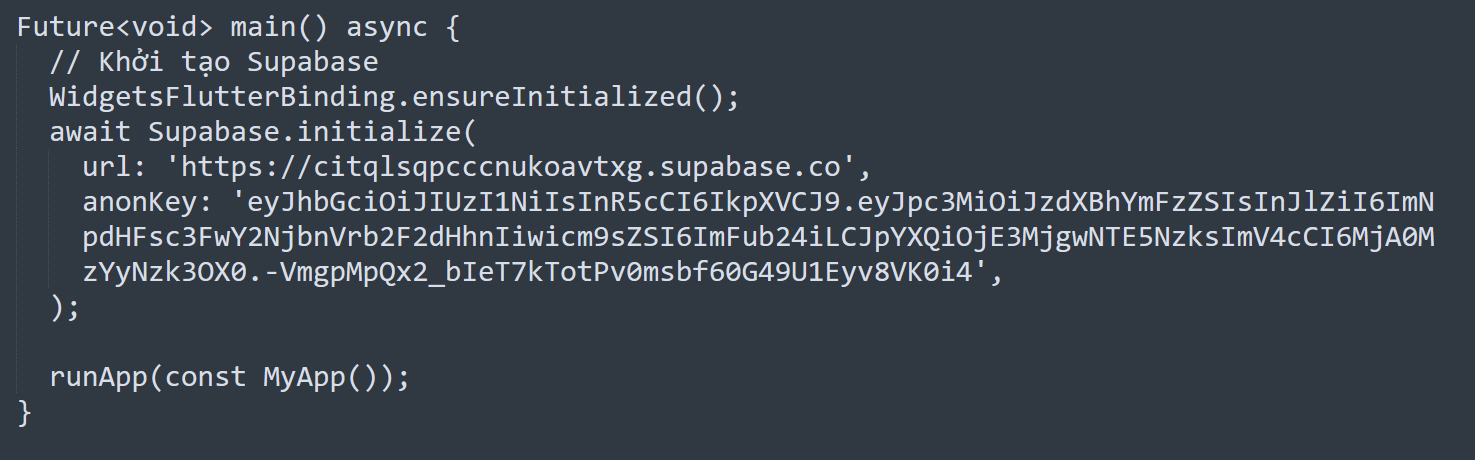


**Bước 3: Cài đặt gói supabase cho flutter**



Cmd: flutter pub get

**Bước 4: Kết Nối Supabase với project Flutter**



Khởi tạo Supabase với URL và key của dự án trên supabase

**Bước 5: Tạo các giao diện**

Giao diện chính

Widget build(BuildContext context) {

    return Scaffold(

      appBar: AppBar(

        backgroundColor: const Color.fromARGB(255, 98, 191, 234),

        title: Text('Quản Lý Sản Phẩm'),

        actions: [

          IconButton(

            icon: Icon(Icons.logout),

            onPressed: () async {

              await supabasehandler.signOut(); // Đăng xuất

              Navigator.pushReplacement(

                context,

                MaterialPageRoute(

                    builder: (context) => LoginPage()), // Quay lại trang login

              );

            },

          ),

        ],

      ),

      body: FutureBuilder(

        future: supabasehandler.readData(),

        builder: (context, snapshot) {

          if (snapshot.connectionState == ConnectionState.waiting) {

            return const Center(child: CircularProgressIndicator());

          } else if (snapshot.hasError) {

            return Center(child: Text('Lỗi: ${snapshot.error}'));

          } else if (!snapshot.hasData || snapshot.data == null) {

            return const Center(child: Text('Không có dữ liệu.'));

          }

          // data thành danh sách

          final dataList = snapshot.data as List<dynamic>? ?? [];

          return ListView.builder(

            itemCount: dataList.length,

            itemBuilder: (context, index) {

              final item = dataList[index];

              return Container(

                height: 150.0,

                decoration: BoxDecoration(

                  color: Colors.white,

                  border: Border.all(

                      color: Colors.red, width: 1.0),

                  borderRadius: BorderRadius.circular(10.0),

                ),

                child: Row(

                  mainAxisAlignment: MainAxisAlignment.start,

                  children: [

                    Expanded(

                      child: Column(

                        mainAxisAlignment:

                            MainAxisAlignment.center,

                        crossAxisAlignment:

                            CrossAxisAlignment.start,

                        children: [

                          Text('Sản Phẩm: ${item['name'] ?? 'Không tên'}'),

                          Text('Loại: ${item['category'] ?? 'Không thể loại'}'),

                          Text(

                              'Giá: ${item['price']?.toStringAsFixed(3) ?? '0.000'} VND'),

                        ],

                      ),

                    ),

                    Column(

                      // Tạo một Column cho các nút

                      mainAxisAlignment:

                          MainAxisAlignment.center,

                      children: [

                        IconButton(

                          onPressed: () {

                            Navigator.push(

                              context,

                              MaterialPageRoute(

                                builder: (context) => EditPage(

                                  id: item['id'],

                                  currentName: item['name'],

                                  currentCategory: item['category'],

                                  currentPrice: item['price'].toDouble(),

                                ),

                              ),

                            );

                          },

                          icon: const Icon(Icons.edit),

                        ),

                        IconButton(

                          onPressed: () async {

                            await supabasehandler.deleteData(item['id']);

                            setState(() {});

                          },

                          icon: const Icon(Icons.delete),

                        ),

                      ],

                    ),

                  ],

                ),

              );

            },

          );

        },

      ),

      floatingActionButton: Padding(

        padding: const EdgeInsets.all(8.0),

        child: Container(

          width: MediaQuery.of(context).size.width \* 0.9,

          child: Row(

            mainAxisAlignment: MainAxisAlignment.spaceEvenly,

            children: [

              FloatingActionButton(

                onPressed: () {

                  Navigator.push(

                    context,

                    MaterialPageRoute(

                      builder: (context) => const AddProductsPage(),

                    ),

                  ).then((value) =>

                      setState(() {})); // Làm mới

                },

                child: const Icon(Icons.add),

              ),

              FloatingActionButton(

                onPressed: () {

                  setState(() {});

                },

                child: const Icon(Icons.refresh),

              ),

            ],

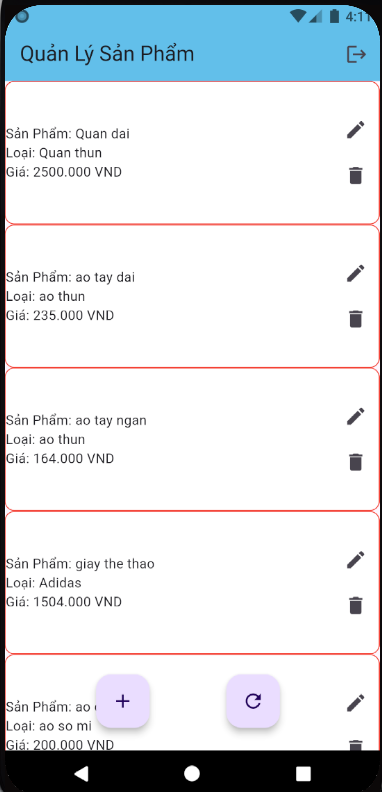
          ),

        ),

      ),

    );

  }



Giao diện thêm

Widget build(BuildContext context) {

    return Scaffold(

      appBar: AppBar(

        title: const Text('Thêm Sản Phẩm'),

      ),

      body: Padding(

        padding: const EdgeInsets.all(16.0),

        child: Column(

          children: [

            TextField(

              controller: \_nameController,

              decoration: const InputDecoration(

                labelText: 'Nhập Tên Sản Phẩm',

                border: OutlineInputBorder(),

              ),

            ),

            const SizedBox(height: 16.0),

            TextField(

              controller: \_categoryController,

              decoration: const InputDecoration(

                labelText: 'Nhập Thể Loại',

                border: OutlineInputBorder(),

              ),

            ),

            const SizedBox(height: 16.0),

            TextField(

              controller: \_priceController,

              decoration: const InputDecoration(

                labelText: 'Nhập Giá',

                border: OutlineInputBorder(),

              ),

              keyboardType: TextInputType.number,

            ),

            const SizedBox(height: 20.0),

            ElevatedButton(

              onPressed: () async {

                String nameValue = \_nameController.text;

                String categoryValue = \_categoryController.text;

                double priceValue = double.tryParse(\_priceController.text) ?? 0.0;

                if (nameValue.isNotEmpty && categoryValue.isNotEmpty && priceValue > 0) {

                  await supabasehandler.addData(nameValue, categoryValue, priceValue);

                  \_nameController.clear();

                  \_categoryController.clear();

                  \_priceController.clear();

                  Navigator.pop(context); // Quay lại trang home

                } else {

                  ScaffoldMessenger.of(context).showSnackBar(

                    const SnackBar(

                      content: Text('Vui lòng nhập tên, thể loại và giá hợp lệ'),

                    ),

                  );

                }

              },

              child: const Text('Thêm Sản Phẩm'),

            ),

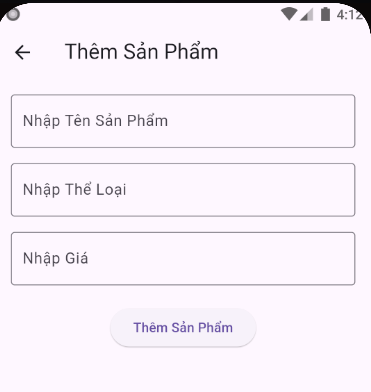
          ],

        ),

      ),

    );

  }



Giao diện sửa

Widget build(BuildContext context) {

    return Scaffold(

      appBar: AppBar(

        title: const Text('Cập Nhật Sản Phẩm'),

      ),

      body: Padding(

        padding: const EdgeInsets.all(16.0),

        child: Column(

          children: [

            TextField(

              controller: \_nameController,

              decoration: const InputDecoration(

                labelText: 'Tên Sản Phẩm',

                border: OutlineInputBorder(),

              ),

            ),

            const SizedBox(height: 16.0),

            TextField(

              controller: \_categoryController,

              decoration: const InputDecoration(

                labelText: 'Thể Loại',

                border: OutlineInputBorder(),

              ),

            ),

            const SizedBox(height: 16.0),

            TextField(

              controller: \_priceController,

              decoration: const InputDecoration(

                labelText: 'Giá',

                border: OutlineInputBorder(),

              ),

              keyboardType: TextInputType.number,

            ),

            const SizedBox(height: 20.0),

            ElevatedButton(

              onPressed: () async {

                String nameValue = \_nameController.text;

                String categoryValue = \_categoryController.text;

                double priceValue =

                    double.tryParse(\_priceController.text) ?? 0.0;

                if (nameValue.isNotEmpty &&

                    categoryValue.isNotEmpty &&

                    priceValue > 0) {

                  await supabasehandler.updateData(

                    widget.id,

                    nameValue,

                    categoryValue,

                    priceValue,

                  );

                  Navigator.pop(context); // Quay về trang home

                } else {

                  ScaffoldMessenger.of(context).showSnackBar(

                    const SnackBar(content: Text('Vui lòng nhập dữ liệu hợp lệ')),

                  );

                }

              },

              child: const Text('Cập Nhật'),

            ),

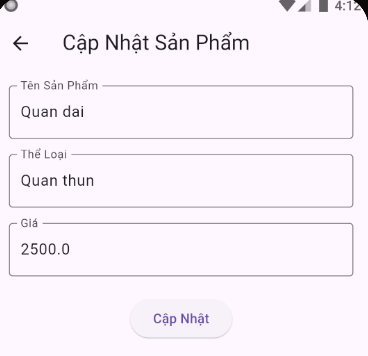
          ],

        ),

      ),

    );

  }



Giao diện đăng nhập

Widget build(BuildContext context) {

    return Scaffold(

      appBar: AppBar(title: Text('Đăng nhập')),

      body: Padding(

        padding: EdgeInsets.all(16.0),

        child: Column(

          mainAxisAlignment: MainAxisAlignment.center,

          children: [

            TextField(

              controller: \_emailController,

              decoration: const InputDecoration(

                labelText: 'Email',

                border: OutlineInputBorder(),

              ),

            ),

            SizedBox(height: 10.0,),

            TextField(

              controller: \_passwordController,

              decoration: const InputDecoration(

                labelText: 'Mật khẩu',

                border: OutlineInputBorder(),

              ),

              obscureText: true,

            ),

            SizedBox(height: 16.0),

            ElevatedButton(

              onPressed: \_login,

              child: Text('Đăng Nhập'),

            ),

          ],

        ),

      ),

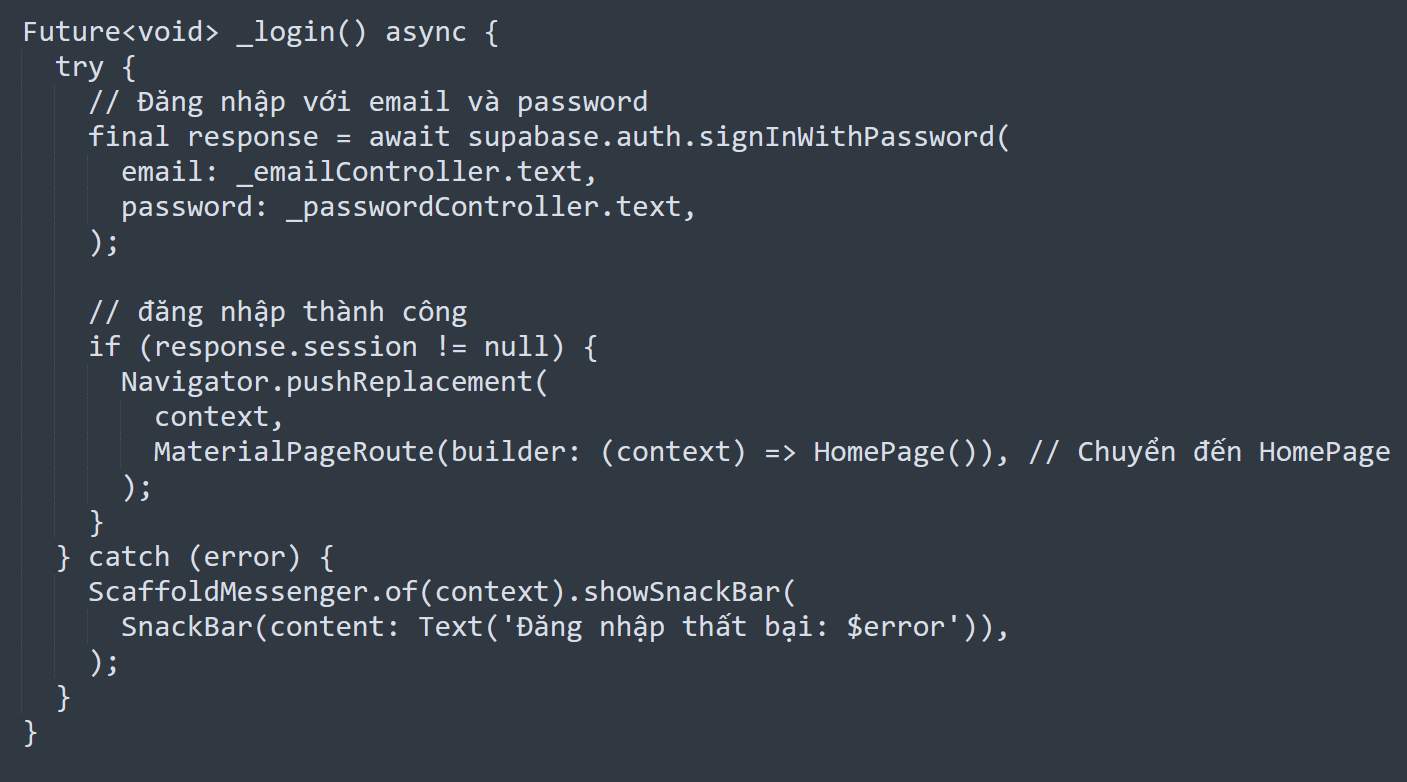
    );

  }



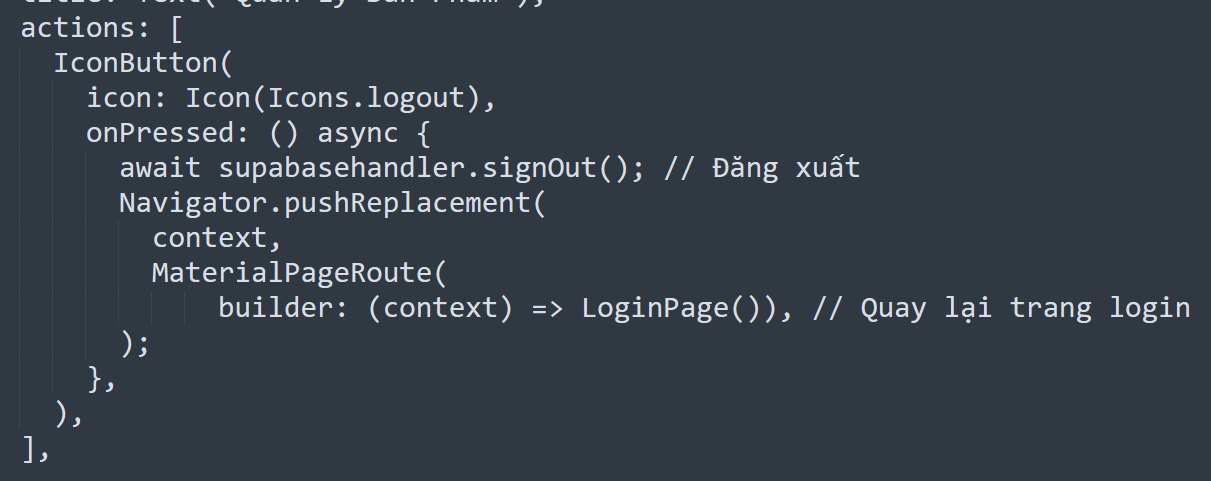
**Bước 6: Xây dựng các chức năng**

Chức năng đăng nhập



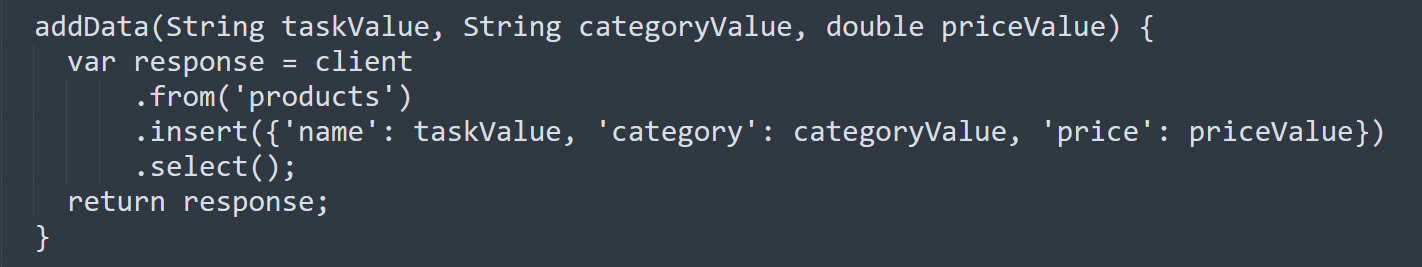
Đăng nhập thành công chuyển đến homepage

Chức năng đăng xuất

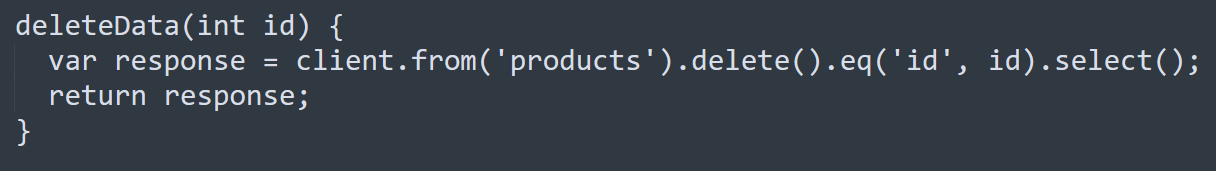


Đăng xuất thành công thì quay lại trang login

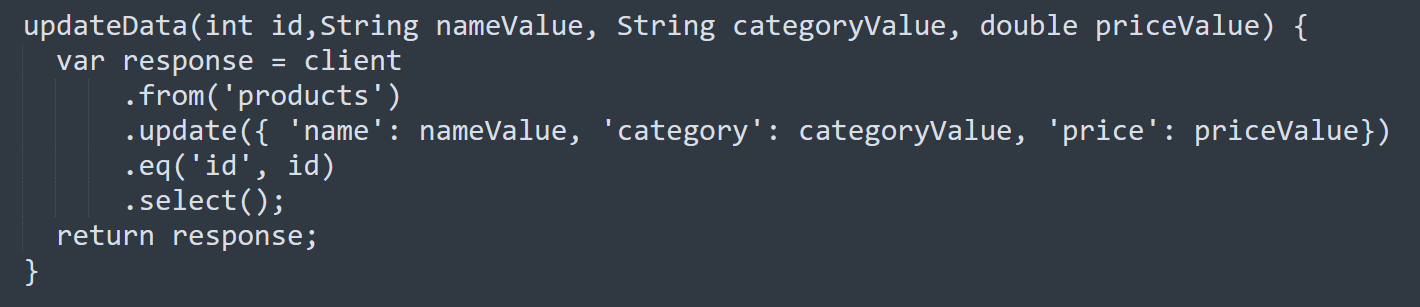
Chức năng thêm



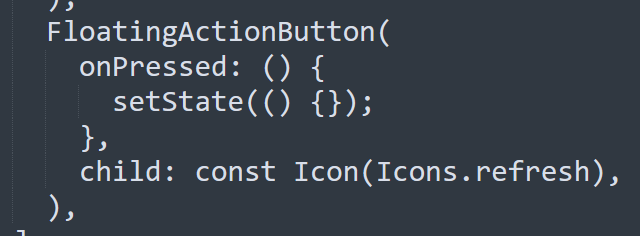
Chức năng xóa



Chức năng sửa



Chức năng reset giao diện chính



**Bước 7: Chạy project**