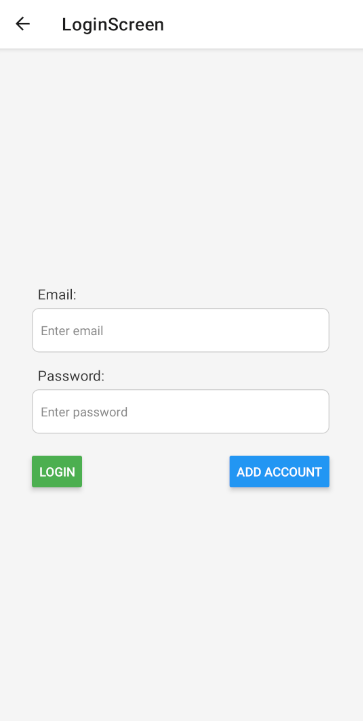
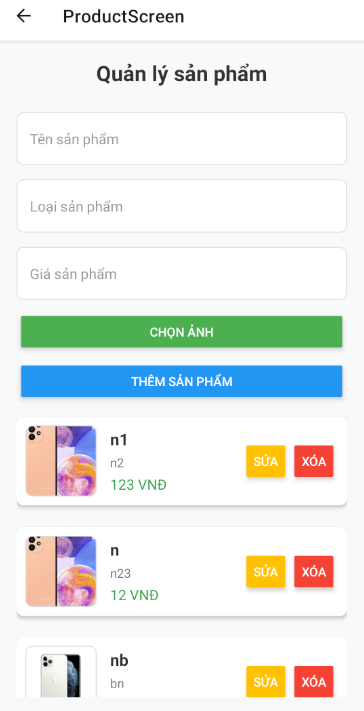
**BÁO CÁO MÔ TẢ PROJECT GIỮA KỲ**

|  |  |
| --- | --- |
| Họ và tên sinh viên: TĂNG NGỌC NHÂN  Chuyên ngành: Kỹ thuật phần mềm  Điện thoại: 0949922971 | MSSV: 21IT296  Lớp: 21SE2  Email: nhantn.21it@vku.udn.vn |
| Giảng viên hướng dẫn: ThS. Ngô Lê Quân |  |

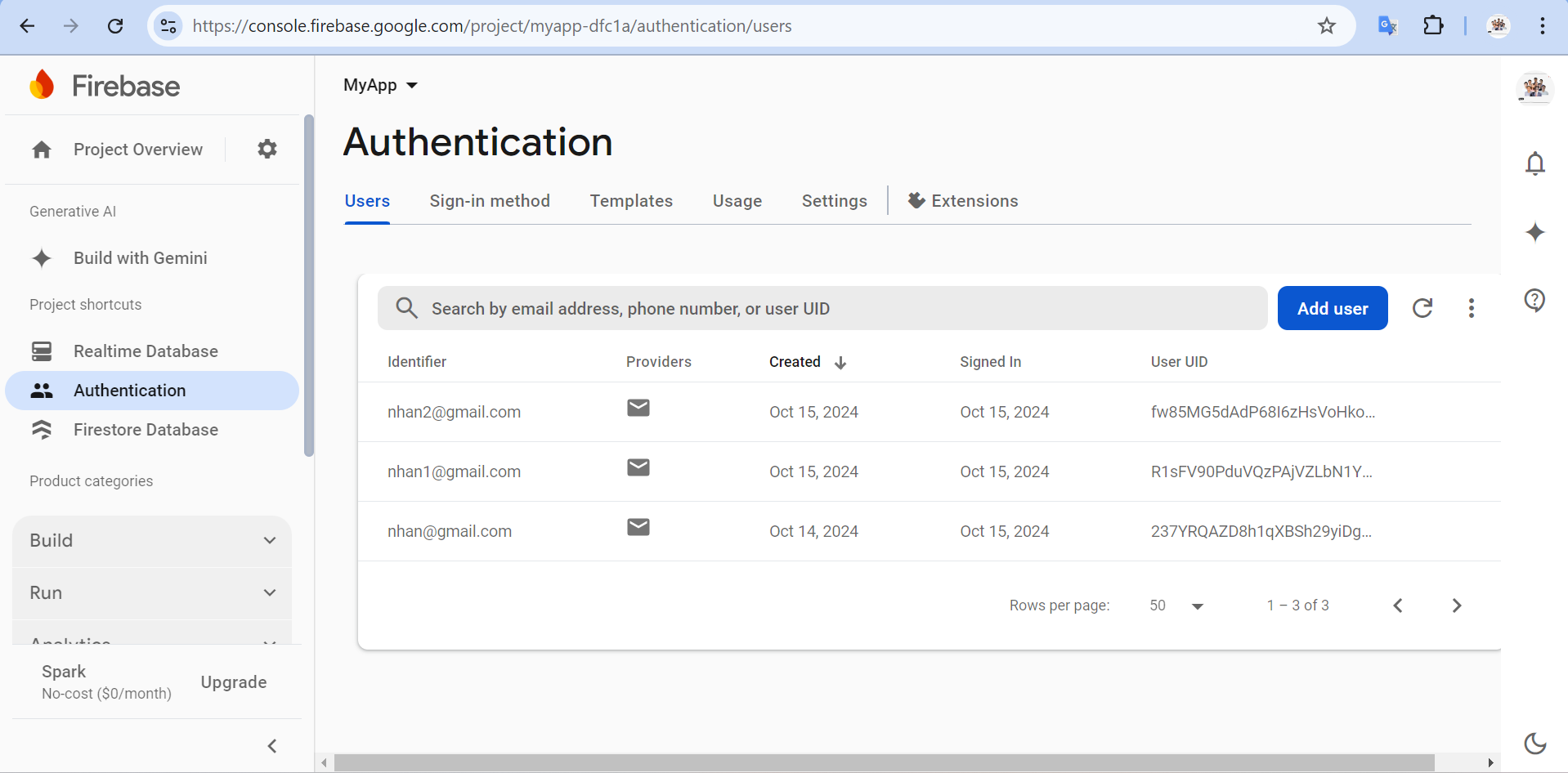
# Các tính năng làm được:

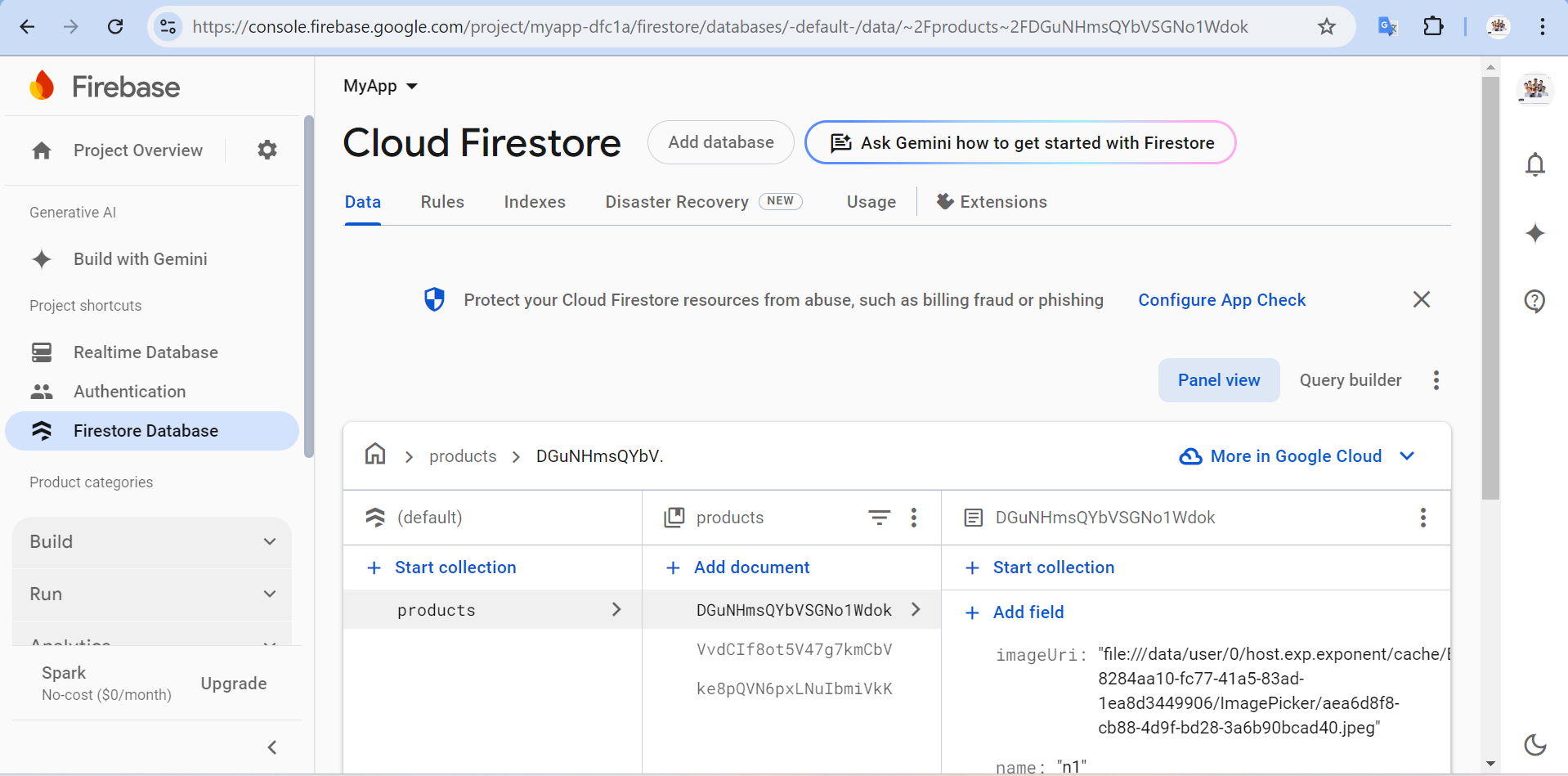
* Đăng nhập, thêm account
* Thêm sản phẩm: Người dùng có thể nhập để thêm thông tin sản phẩm tên sản phẩm, loại sản phẩm, giá và thêm hình ảnh cho sản phẩm đó
* Hiển thị danh sách sản phẩm đã thêm: Lấy và hiển thị danh sách thông tin các sản phẩm đã được thêm.
* Sửa sản phẩm: Hiển thị thông tin sản phẩm được chọn để sửa trên các ô input và cho phép sửa thông tin bao gồm cả hình ảnh
* Xóa sản phẩm: Người dùng có thể xóa sản phẩm khỏi cơ sở dữ liệu.

# Các công nghệ được sử dụng

* Cơ sở dữ liệu firebase (Sử dụng Authentication trong thêm account và đăng nhập, Firestore Database trong lưu trữ và quản lý thông tin sản phẩm)





* Framework React Native (Expo): Xây dựng ứng dụng di động đa nền tảng kết nối với firebase để tương tác với dữ liệu sản phẩm.
* Một số thư viện chính: react, react-native, firebase, expo, expo-image-picker

# Quy trình xây dựng ứng dụng:

## Cài đặt các thư viện cần thiết:

Sử dụng lệnh để cài đặt:

npm install firebase

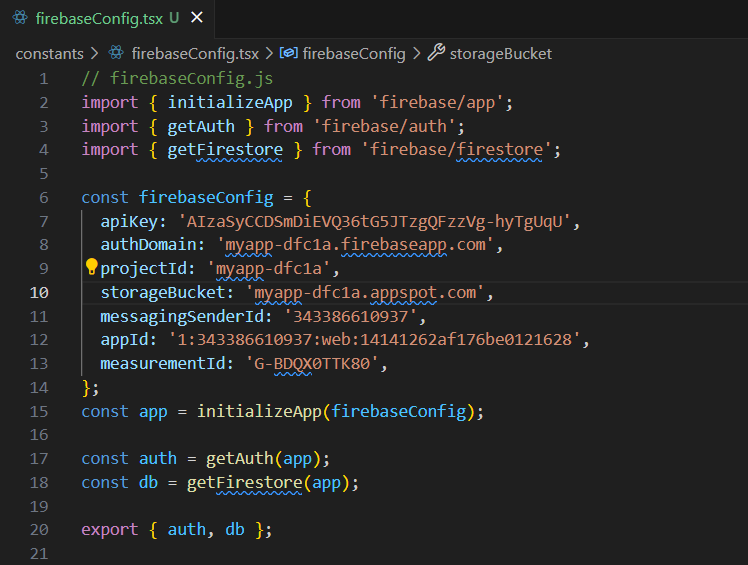
npm install @react-navigation/native

npm install expo-image-picker

## Cấu hình để kết nối Firebase cho dự án:

Tạo dự án trên Firebase Console và sao chép cấu hình.

Tạo file cấu hình Firebase (firebaseConfig.js) và khởi tạo Firestore và Authentication.



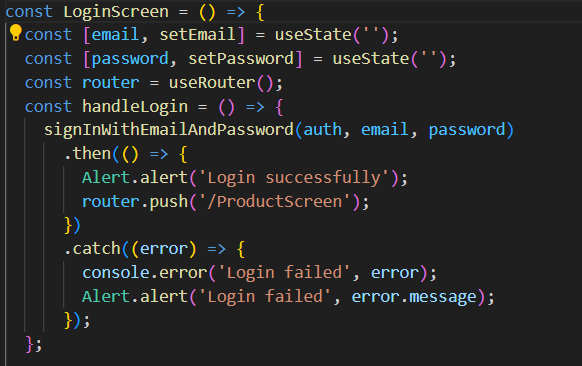
## Tạo Màn hình Đăng nhập (LoginScreen)

Sử dụng các hook (useState) để lưu trữ giá trị email và mật khẩu.

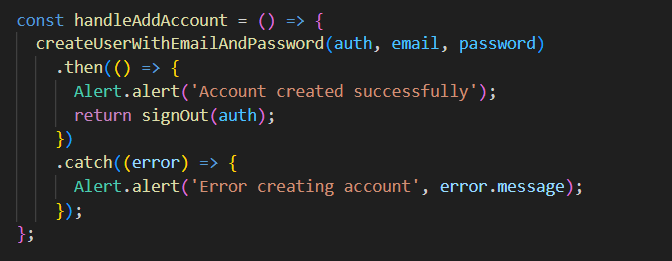
Sử dụng signInWithEmailAndPassword và createUserWithEmailAndPassword để xác thực người dùng.

Hiển thị thông báo thành công hoặc thất bại qua Alert.

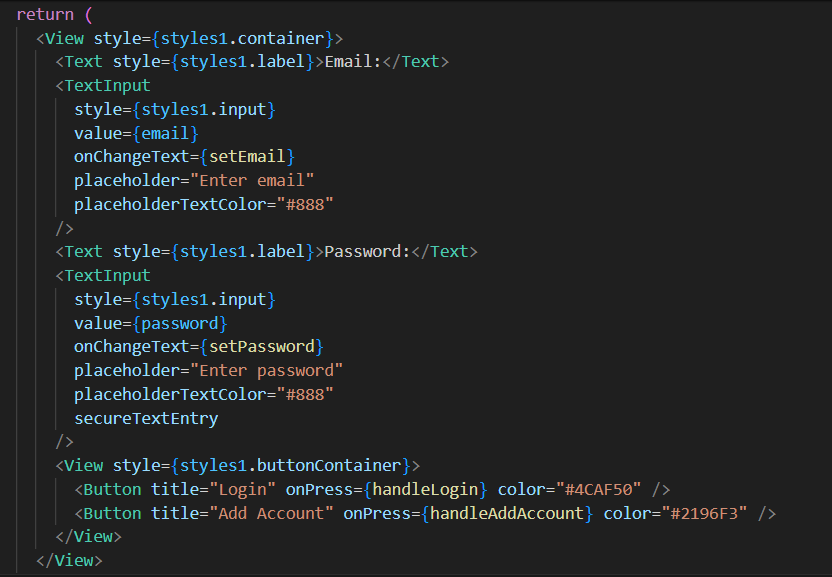
Hàm xử lý đăng nhập:



Hàm xử lý thêm account



Render giao diện:



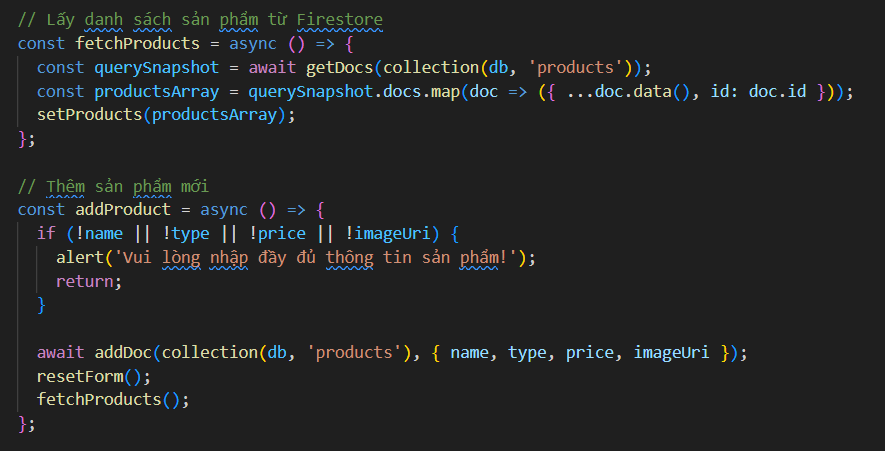
## Tạo Màn hình Quản lý Sản phẩm (ProductScreen)

Sử dụng Firestore để lấy danh sách sản phẩm và lưu trữ chúng trong state.

Sử dụng addDoc, updateDoc, và deleteDoc để quản lý sản phẩm.

Cho phép người dùng chọn hình ảnh từ thư viện và lưu trữ URL của hình ảnh.

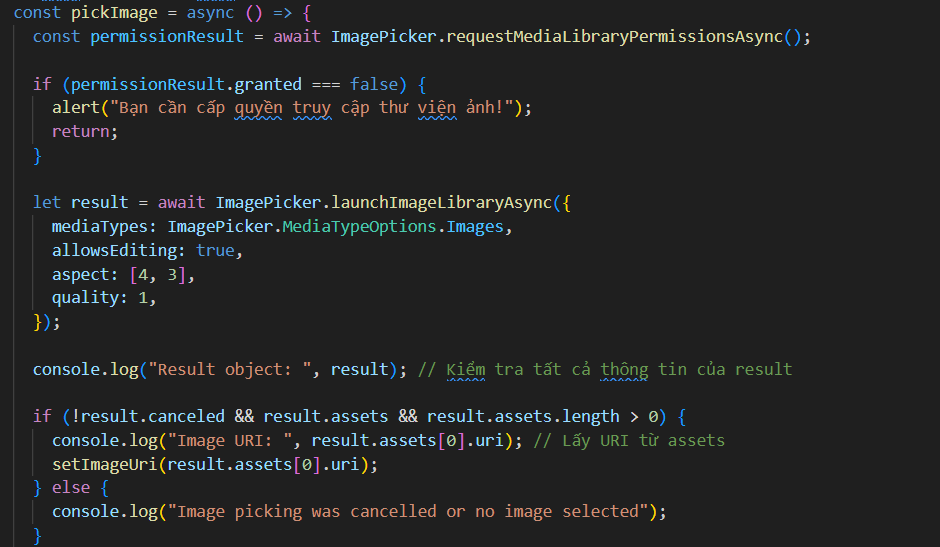
Hàm lấy dữ liệu hiển thị và thêm sản phẩm:



Hàm sửa và xoá sản phẩm



Hàm xử lý chọn ảnh:



Render ra giao diện:

