TRƯỜNG ĐẠI HỌC NHA TRANG

**KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỀ TÀI**

**XÂY DỰNG ỨNG DỤNG BÁN GIÀY**

**Học phần: Lập trình thiết bị di động**

**Giáo viên: Huỳnh Tuấn Anh**

**Thành viên nhóm:**

* Nguyễn Trọng Tín 62132235
* Nguyễn Tiến Dũng 62130312
* Đoàn Duy Thiện 62132003
* Từ Công Thắng 62131885

KHÁNH HÒA, 2022

**MỤC LỤC**

[LỜI MỞ ĐẦU 1](#_Toc136182153)

[CHƯƠNG I: GIỚI THIỆU ĐỀ TÀI 2](#_Toc136182154)

[1. Lý do chọn đề tài 2](#_Toc136182155)

[2. Mục tiêu của đề tài 2](#_Toc136182156)

[CHƯƠNG II: CƠ SỞ LÝ THUYẾT 3](#_Toc136182157)

[1. Hệ điều hành Android: 3](#_Toc136182158)

[Ưu điểm: 4](#_Toc136182159)

[Nhược điểm: 4](#_Toc136182160)

[*2.* Ngôn ngữ Dart, Flutter framework: 5](#_Toc136182161)

[a. Dart 5](#_Toc136182162)

[b. Flutter Framework 6](#_Toc136182163)

[3. Android Studio 7](#_Toc136182164)

[4. Firebase: 8](#_Toc136182165)

[a. Firebase là gì? 8](#_Toc136182166)

[b. Lịch sử phát triển Firebase 9](#_Toc136182167)

[c. Cách thức hoạt động của Firebase là gì? 9](#_Toc136182168)

[d. Ưu điểm khi dùng Firebase là gì? 10](#_Toc136182169)

[e. Nhược điểm của Firebase là gì? 10](#_Toc136182170)

[Chương III : Phân tích và thiết kế hệ thống 11](#_Toc136182171)

[1. Phân tích các chức năng chính của ứng dụng 11](#_Toc136182172)

[a. Chức năng đăng ký, đăng nhập: 11](#_Toc136182173)

[b. Chức năng đặt hàng: 11](#_Toc136182174)

[c. Chức năng theo dõi đơn hàng 11](#_Toc136182175)

[d. Chức năng yêu thích sản phẩm 11](#_Toc136182176)

[2. Phân tích dữ liệu trên FireBase 11](#_Toc136182177)

[a. Danh sách sản phẩm (Product) 11](#_Toc136182178)

[b. Danh sách sản phẩm yêu thích (Favourites) 12](#_Toc136182179)

[c. Hóa đơn (Orders) 13](#_Toc136182180)

[d. Lưu trữ ảnh 14](#_Toc136182181)

[3. Mã lệnh giao diện 14](#_Toc136182182)

[**a.** Giao diện đăng nhập (page\_login.dart) 14](#_Toc136182183)

[**b.** Giao diện trang hiển thị sản phẩm (product\_page.dart) 15](#_Toc136182184)

[**c.** Giao diện giỏ hàng (cart\_page.dart) 16](#_Toc136182185)

[**d.** Đặt hàng (cart\_page.dart) 17](#_Toc136182186)

[**e.** Giao diện yêu thích (favourite\_page.dart) 18](#_Toc136182187)

[**f.** Giao diện quản lý sản phẩm (product\_adminpage.dart) 19](#_Toc136182188)

[4. Giao diện hệ thống 22](#_Toc136182189)

[a. Giao diện đăng nhập 22](#_Toc136182190)

[b. Giao diện đăng ký 26](#_Toc136182191)

[c. Giao diện trang chủ 28](#_Toc136182192)

[d. Giao diện đơn hàng 30](#_Toc136182193)

[e. Giao diện profile 33](#_Toc136182194)

[f. Giao diện yêu thích 36](#_Toc136182195)

[g. Giao diện chính (quản lý) 37](#_Toc136182196)

[h. Giao diện quản lý sản phẩm 38](#_Toc136182197)

[i. Giao diện cập nhật đơn hàng 43](#_Toc136182198)

# LỜI MỞ ĐẦU

Trong sự phát triển không ngừng của xã hội thì nhu cầu của con người về cuộc sống ngày càng cao. Từ ý nghĩa thực tiễn đó mà đã sinh ra nhiều loại dịch vụ dể dáp ứng nhu cầu thiết yêu của con người. các loại dịch vụ đa dạng về nhiều mặt với nhiều cách đáp ứng khác nhau. Loại hình dịch vụ khá phổ biển trong đó là bán hàng qua mạng. Khi vào dịch vụ mua bán qua mạng, người ta thấy được sự tiện lợi mà nhiều dịch vụ khác không có được, tạo cho con người có sự tin cậy. Mua bán được hoạt động với nhiều hình thức khác nhau do đó chúng em thực hiện đề tài “xây dựng ứng dụng bán hàng” để nói lên sự tiên lợi của một hệ thống mua bán sản phẩm giầy dép qua mạng. Nhiệm vụ chính của chúng em là xây dựng nên một ứng dụng bán hàng qua mạng với các tiện ích như đặt hàng, coi thông tin sản phẩm,… với mục đích củng cổ và kiểm tra lại toàn bộ kiến thức đã học và thực hành.

# CHƯƠNG I: GIỚI THIỆU ĐỀ TÀI

## Lý do chọn đề tài

Với sự phát triển không ngừng của công nghệ thông tin, các hệ điều hành trên thiết bị di dộng trở nên rất được quan tâm. Trong vài năm trở lại đây, sự xuất hiện của thuật ngữ “đa nền tảng” đã xuất hiện rất nhiều trên các ứng dụng như game, web, hoặc thậm chí là các ứng dụng giải trí, … phục vụ nhu cầu con người. Thuật ngữ “đa nền tảng” có thể hiểu là các phương thức điện toán và các khái niệm được thực thi đầy đủ và vận hành cùng nhau trên nhiều thiết bị. Vì thế các công ty phần mềm đã mở rộng ra sản xuất ứng dụng trên thiết bị di động cùng với hướng đi mới trong việc lựa chọn nghề nghiệp, tương lai bản thân. Lập trình cho thiết bị Android là một trong số đó, chúng em sẽ sử dụng phần mền này để xây dựng đề tài “Xây dựng ứng dụng bán giày” của chúng em để làm rõ công dụng và lý do sử dụng rộng rãi của hệ điều hành Android.

## Mục tiêu của đề tài

* + Tìm hiểu về hệ điều hành android, nắm bắt được cách thức xây dựng.
* Một ứng dụng trên hệ điều hành Android.
* Phân tích thiết kế ứng dụng bán hàng.
* Xây dựng ứng dụng bán hàng.

# CHƯƠNG II: CƠ SỞ LÝ THUYẾT

## Hệ điều hành Android:

Android là một hệ điều hành có mã nguồn mở dựa trên nền tảng Linux được thiết kế dành cho các thiết bị di động có màn hình cảm ứng như điện thoại thông minh và máy tính bảng.

Ban đầu, Android được phát triển bởi Tổng công ty Android, với sự hỗ trợ tài chính từ Google, sau này được chính Google mua lại vào năm 2005 và hệ điều hành Android đã ra mắt vào năm 2007. Chiếc điện thoại đầu tiên chạy Android là HTC Dream được bán vào ngày 22 tháng 10 năm 2008.

Chính mã nguồn mở cùng với giấy phép không có nhiều ràng buộc đã cho phép các nhà phát triển thiết bị, mạng di động và các lập trình viên được điều chỉnh và phân phối Android một cách tự do. Những yếu tố này đã giúp Android trở thành nền tảng điện thoại thông minh phổ biến nhất thế giới.

Giao diện người dùng của Android dựa trên nguyên tắc tác động trực tiếp, sử dụng cảm ứng chạm tương tự như những động tác ngoài đời thực như vuốt, chạm, kéo dãn và thu lại để xử lý các đối tượng trên màn hình. Các thiết bị Android sau khi khởi động sẽ hiển thị màn hình chính, điểm khởi đầu với các thông tin chính trên thiết bị, tương tự như khái niệm desktop trên máy tính để bàn. Màn hính chính Android thường gồm nhiều biểu tượng (icon) và tiện ích (widget). Giao diện màn hình chính của Android có thể tùy chỉnh ở mức cao, cho phép người dùng tự do sắp đặt hình dáng cũng như hành vi của thiết bị theo sở thích. Những ứng dụng do các hãng thứ ba có trên Google Play và các kho ứng dụng khác còn cho phép người dùng thay đổi "chủ đề" của màn hình chính, thậm chí bắt chước hình dáng của hệ điều hành khác như Windows Phone hay iOS. Phần lớn những nhà sản xuất, và một số nhà mạng, thực hiện thay đổi hình dáng và hành vi của các thiết bị Android của họ để phân biệt với các hãng cạnh tranh. Android Google đưa ra các bản cập nhật lớn cho Android theo chu kỳ từ 6 đến 9 tháng, mà phần lớn thiết bị đều có thể nhận được qua sóng không dây. Bản cập nhật lớn mới nhất là Android 12.0.

### **Ưu điểm**:

* Là hệ điều hành có mã nguồn mở nên khả năng tuỳ biến cao, có thể tùy ý chỉnh sửa mà không có sự can thiệp hay cấm cản từ Google.
* Đa dạng sản phẩm, rất nhiều hãng điện thoại, thiết bị công nghệ đã ưu ái chọn Android cho thiết bị của họ, giá cả thì hợp lý từ bình dân đến cao cấp.
* Kho ứng dụng Google Play Store phong phú.
* Thân thiện và dễ sử dụng.
* Khả năng đa nhiệm, chạy cùng lúc nhiều ứng dụng cao.

### Nhược điểm:

* Dễ nhiễm phần mềm độc hại và virus. Do tính chất mã nguồn mở, nhiều phần mềm không được kiểm soát có chất lượng không tốt hoặc lỗi bảo mật vẫn được sử dụng.
* Kho ứng dụng quá nhiều dẫn đến khó kiểm soát chất lượng, thiếu các ứng dụng thật sự tốt.
* Sự phân mảnh lớn. Trong khi một số thiết bị Android xuất sắc đã trình làng như Galaxy S5, Galaxy Note 4, Xperia Z3…, vẫn còn rất nhiều sản phẩm giá rẻ bình thường khác.
* Cập nhật không tự động với tất cả thiết bị. Khi một phiên bản hệ điều hành mới ra mắt, không phải tất cả sản phẩm đều được cập nhật, thậm chí nếu muốn trải nghiệm bạn thường xuyên phải mua mới thiết bị.

## **Ngôn ngữ Dart, Flutter framework**:

### Dart

Dart là một ngôn ngữ lập trình đa nền tảng, được phát triển bởi Google. Nó được sử dụng chủ yếu trong việc phát triển ứng dụng di động, web, và máy chủ. Dart kết hợp các tính năng từ nhiều ngôn ngữ khác nhau như JavaScript, Java và C#, và nó có một cú pháp dễ đọc và dễ hiểu.

Dart có những đặc điểm chính sau:

* + Dart là ngôn ngữ kiểu tĩnh (statically typed), có nghĩa là bạn cần khai báo kiểu của biến trước khi sử dụng nó.
  + Hỗ trợ đa luồng (concurrency) thông qua việc sử dụng hàm async/await và Future để xử lý các tác vụ không đồng bộ.
  + Dart hỗ trợ tính kế thừa (inheritance), đa hình (polymorphism), và đóng gói (encapsulation), giúp xây dựng các lớp đối tượng mạnh mẽ.
  + Dart có cú pháp gần giống với các ngôn ngữ lập trình khác như JavaScript, Java hoặc C#, giúp người lập trình dễ dàng học và chuyển đổi giữa các ngôn ngữ.

### Flutter Framework

Flutter là một framework mã nguồn mở được phát triển bởi Google, cho phép bạn xây dựng ứng dụng di động đẹp và chạy trên nhiều nền tảng như Android, iOS, web và desktop từ một mã nguồn duy nhất. Flutter sử dụng ngôn ngữ Dart để viết mã.

* + Các điểm nổi bật của Flutter bao gồm:
  + Flutter sử dụng cấu trúc giao diện người dùng linh hoạt (UI framework) để xây dựng giao diện đẹp và tương tác.
  + Nó cung cấp một tập hợp các thành phần giao diện người dùng được tạo sẵn (widgets) để xây dựng giao diện nhanh chóng và dễ dàng.
  + Flutter sử dụng kỹ thuật vẽ giao diện (UI rendering) riêng biệt, giúp ứng dụng chạy mượt mà và có hiệu suất cao.
  + Flutter cung cấp tích hợp sẵn với các API và dịch vụ như camera, location, network, và nhiều hơn nữa, giúp bạn dễ dàng truy cập và sử dụng các chức năng hệ thống.
  + Flutter hỗ trợ hot reload, cho phép bạn thực hiện các thay đổi trong mã nguồn và xem kết quả ngay lập tức mà không cần khởi động lại ứng dụng.

## Android Studio

Chức năng của Android Studio là cung cấp giao diện để tạo các ứng dụng và xử lý phần lớn các công cụ quản lý file phức tạp đằng sau hậu trường. Ngôn ngữ lập trình được sử dụng ở đây là Java và được cài đặt riêng trên thiết bị của bạn. Android Studio rất đơn giản, bạn chỉ cần viết, chỉnh sửa và lưu các dự án của mình và các file trong dự án đó. Đồng thời, Android Studio sẽ cấp quyền truy cập vào Android SDK.

Hãy coi đây là đuôi cho code Java cho phép nó chạy trơn tru trên các thiết bị Android và tận dụng lợi thế của phần cứng gốc. Bạn cần sử dụng ngôn ngữ lập trình Java để viết các chương trình, Android SDK có nhiệm vụ kết nối các phần này lại với nhau. Cùng lúc đó Android Studio kích hoạt để chạy code, thông qua trình giả lập hoặc qua một phần cứng kết nối với thiết bị. Sau đó, bạn cũng có thể “gỡ rối” chương trình khi nó chạy và nhận phản hồi giải thích sự cố, v.v… để bạn có thể nhanh chóng giải quyết vấn đề.

Google đã nỗ lực rất nhiều để làm cho Android Studio trở nên mạnh mẽ và hữu ích nhất có thể. Nó cung cấp những gợi ý trực tiếp trong khi viết code và thường đề xuất những thay đổi cần thiết để sửa lỗi hoặc làm code hiệu quả hơn. Ví dụ, nếu không sử dụng biến, biến đó sẽ được tô đậm bằng màu xám. Và khi bắt đầu gõ một dòng code, Android Studio sẽ cung cấp danh sách gợi ý tự hoàn thành để giúp bạn hoàn thiện dòng code đó. Chức năng này rất hữu ích khi bạn không nhớ được chính xác cú pháp hoặc để tiết kiệm thời gian.

## **Firebase**:

1. Firebase là gì?

Firebase là một nền tảng để phát triển ứng dụng di động và trang web, bao gồm các API đơn giản và mạnh mẽ mà không cần backend hay server. Lợi ích của Firebase là gì? Firebase còn giúp các lập trình viên rút ngắn thời gian triển khai và mở rộng quy mô của ứng dụng mà họ đang phát triển.

Firebase là gì? Firebase giúp bạn lập trình ứng dụng mà không cần quan tâm đến phần Backend. Firebase giúp bạn lập trình ứng dụng mà không cần quan tâm đến phần backend. Firebase là dịch vụ cơ sở dữ liệu hoạt động trên nền tảng đám mây – cloud. Kèm theo đó là hệ thống máy chủ cực kỳ mạnh mẽ của Google. Chức năng chính là giúp người dùng lập trình ứng dụng bằng cách đơn giản hóa các thao tác với cơ sở dữ liệu. Cụ thể là những giao diện lập trình ứng dụng API đơn giản. Mục đích nhằm tăng số lượng người dùng và thu lại nhiều lợi nhuận hơn.

Đặc biệt, còn là dịch vụ đa năng và bảo mật cực tốt. Firebase hỗ trợ cả hai nền tảng Android và IOS. Không có gì khó hiểu khi nhiều lập trình viên chọn Firebase làm nền tảng đầu tiên để xây dựng ứng dụng cho hàng triệu người dùng trên toàn thế giới.

1. Lịch sử phát triển Firebase



Gần một thập niên trước, Firebase ra đời với tiền thân là Envolve. Đây là một nền tảng đơn giản chuyên cung cấp những API cần thiết để tích hợp tính năng chat vào trang Web. Bên cạnh ứng dụng nhắn tin trực tuyến, Envolve còn được người dùng sử dụng để truyền và đồng bộ hóa dữ liệu cho những ứng dụng khác như các trò chơi trực tuyến,… Do đó, các nhà sáng lập đã tách biệt hệ thống nhắn tin trực tuyến và đồng bộ dữ liệu thời gian thực thành hai phần riêng biệt.

Trên cơ sở đó, năm 2012, Firebase ra đời với sản phẩm cung cấp là dịch vụ Backend-as-a-Service. Tiếp đến, vào năm 2014, Google mua lại Firebase và phát triển nó thành một dịch vụ đa chức năng được hàng triệu người sử dụng cho đến hiện nay.

1. Cách thức hoạt động của Firebase là gì?

**Firebase Realtime Database là gì?**

Khi đăng ký một tài khoản trên Firebase để tạo ứng dụng, bạn đã có một cơ sở dữ liệu thời gian thực. Dữ liệu bạn nhận được dưới dạng JSON. Đồng thời nó cũng luôn được đồng bộ thời gian thực đến mọi kết nối client.

Đối với các ứng dụng đa nền tảng, tất cả các client đều sử dụng cùng một cơ sở dữ liệu. Nó được tự động cập nhật dữ liệu mới nhất bất cứ khi nào các lập trình viên phát triển ứng dụng. Cuối cùng, tất cả các dữ liệu này được truyền qua kết nối an toàn SSL có bảo mật với chứng nhận 2048 bit.

Trong trường hợp bị mất mạng, dữ liệu được lưu lại ở local, vì thế khi có mọi sự thay đổi nào đều được tự động cập nhật lên Server của Firebase. Bên cạnh đó, đối với các dữ liệu ở local cũ hơn với Server thì cũng tự động cập nhật để được dữ liệu mới nhất.

**Firebase Authentication là gì?**

Hoạt động nổi bật của Firebase là xây dựng các bước xác thực người dùng bằng Nhóm email, Facebook, Twitter, GitHub, Google. Đồng thời cũng xác thực nặc danh cho các ứng dụng. Hoạt động xác thực có thể giúp thông tin cá nhân của người sử dụng được an toàn và đảm bảo không bị đánh cắp tài khoản.

**Firebase Hosting là gì?**

Cách thức hoạt động cuối cùng của Firebase được đề cập trong bài viết này là cung cấp các hosting. Hosting được phân phối qua tiêu chuẩn công nghệ bảo mật SSL từ mạng CDN. Hoạt động này giúp lập trình viên tiết kiệm thời gian thiết kế, xây dựng và phát triển ứng dụng.

1. Ưu điểm khi dùng Firebase là gì?

**Các lợi ích này bao gồm:**

* + - Triển khai ứng dụng nhanh chóng
    - Bảo mật
    - Sự ổn định

1. Nhược điểm của Firebase là gì?

“Firebase” chính là phần Realtime Database, đúng hơn chỉ là phần Database. Cơ sở dữ liệu của họ được tổ chức theo kiểu trees, parent-children, không phải là kiểu Table nên những ai đang quen với SQL có thể sẽ gặp khó khăn từ mức đôi chút tới khá nhiều.

# Chương III : Phân tích và thiết kế hệ thống

## Phân tích các chức năng chính của ứng dụng

### Chức năng đăng ký, đăng nhập:

* + Người dùng có thể đăng ký tài khoản mới.
  + Người dùng phải đăng nhập tài khoản mới được sử dụng hệ thống.

### Chức năng đặt hàng:

* + Cho phép người dùng thêm sản phẩm vào giỏ hàng và đặt hàng.

### Chức năng theo dõi đơn hàng

* + Cho phép người dùng kiểm tra tình trạng đơn hàng.

### Chức năng yêu thích sản phẩm

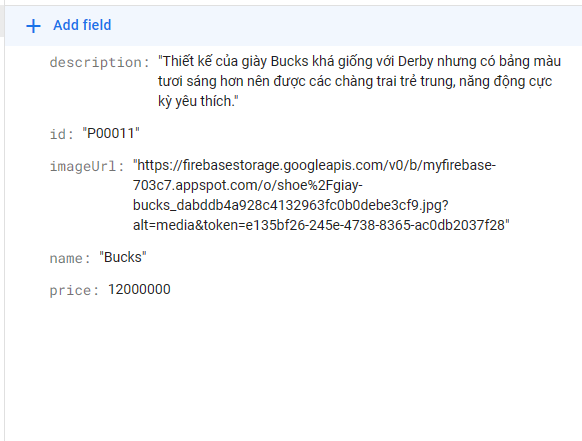
* + Người dùng có thể lưu sản phẩm vào danh sách yêu thích để dễ dàng tìm để đặt hàng.

1. Chức năng quản lý sản phẩm (admin)
   * Quản lý có thể thêm, sửa, xóa sản phẩm
2. Chức năng cập nhật đơn hàng (admin)
   * Quản lý có thể cập nhật trạng thái đơn hàng

## Phân tích dữ liệu trên FireBase

### Danh sách sản phẩm (Product)

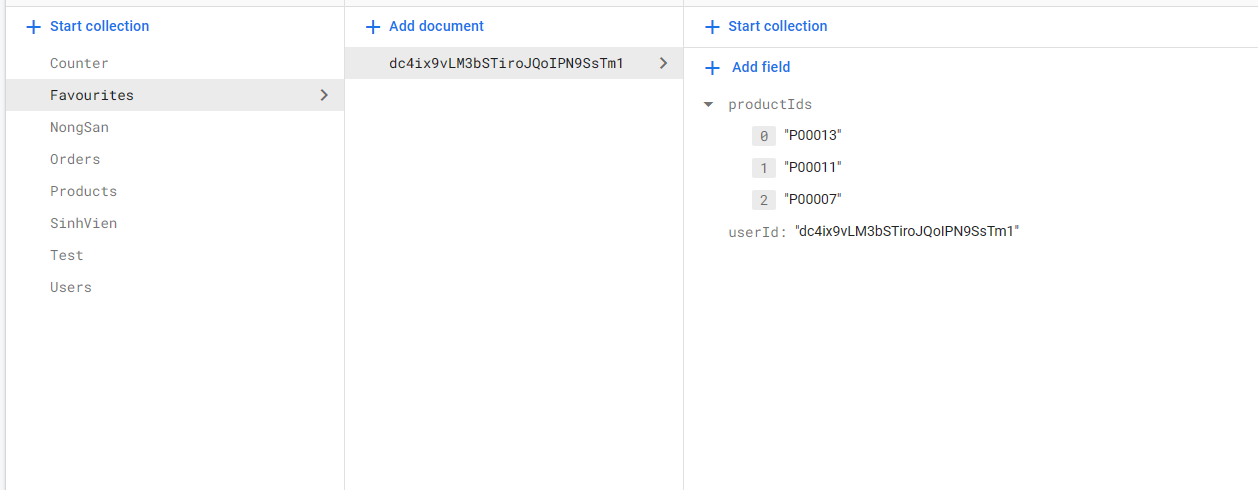
* + Chứa nhiều Document, mỗi Document chứa 4 trường thuộc tính: Mã sản phẩm(id), Tên sản phẩm(name), Giá sản phẩm(price), Ghi Chú(description), Url của ảnh (imageUrl).



Hình 1: Ví dụ Document sản phẩm

### Danh sách sản phẩm yêu thích (Favourites)

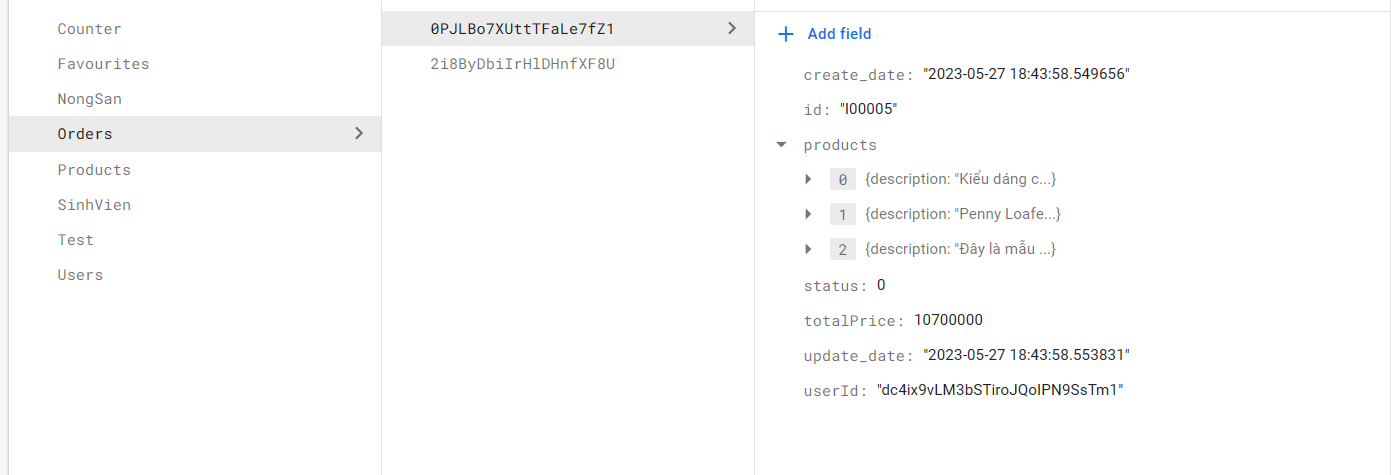
* + Khi click vào chức năng yêu thích sản phẩm, Collection Favourite sẽ khởi tạo 1 Document với Document id là id của user hiện tại chứa 2 trường thuộc tính: Mảng mã sản phẩm(productIds) và mã user hiện tại (userId).



Hình 2: Yêu thích sản phẩm

### Hóa đơn (Orders)

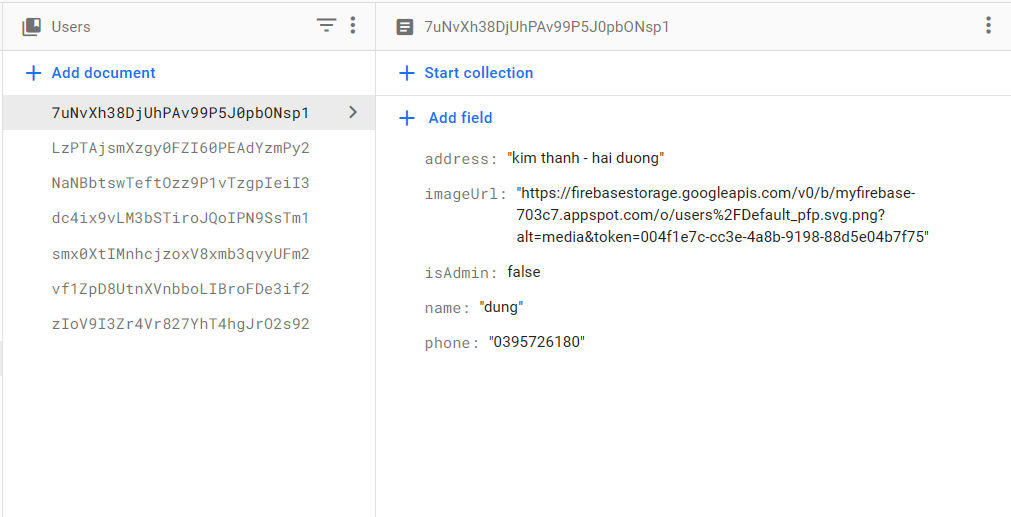
* + Chứa nhiều Document, mỗi Document có Document id là id của user hiện tại chứa 6 trường thuộc tính: Mã hóa đơn(id), Mảng sản phẩm (products), trạng thái(status), tổng tiền(totalPrice), thời gian tạo hóa đơn (create\_date), thời gian cập nhật hóa đơn (update\_date).



Hình 3: Hóa đơn

### Lưu trữ thông tin khách hàng (Users)

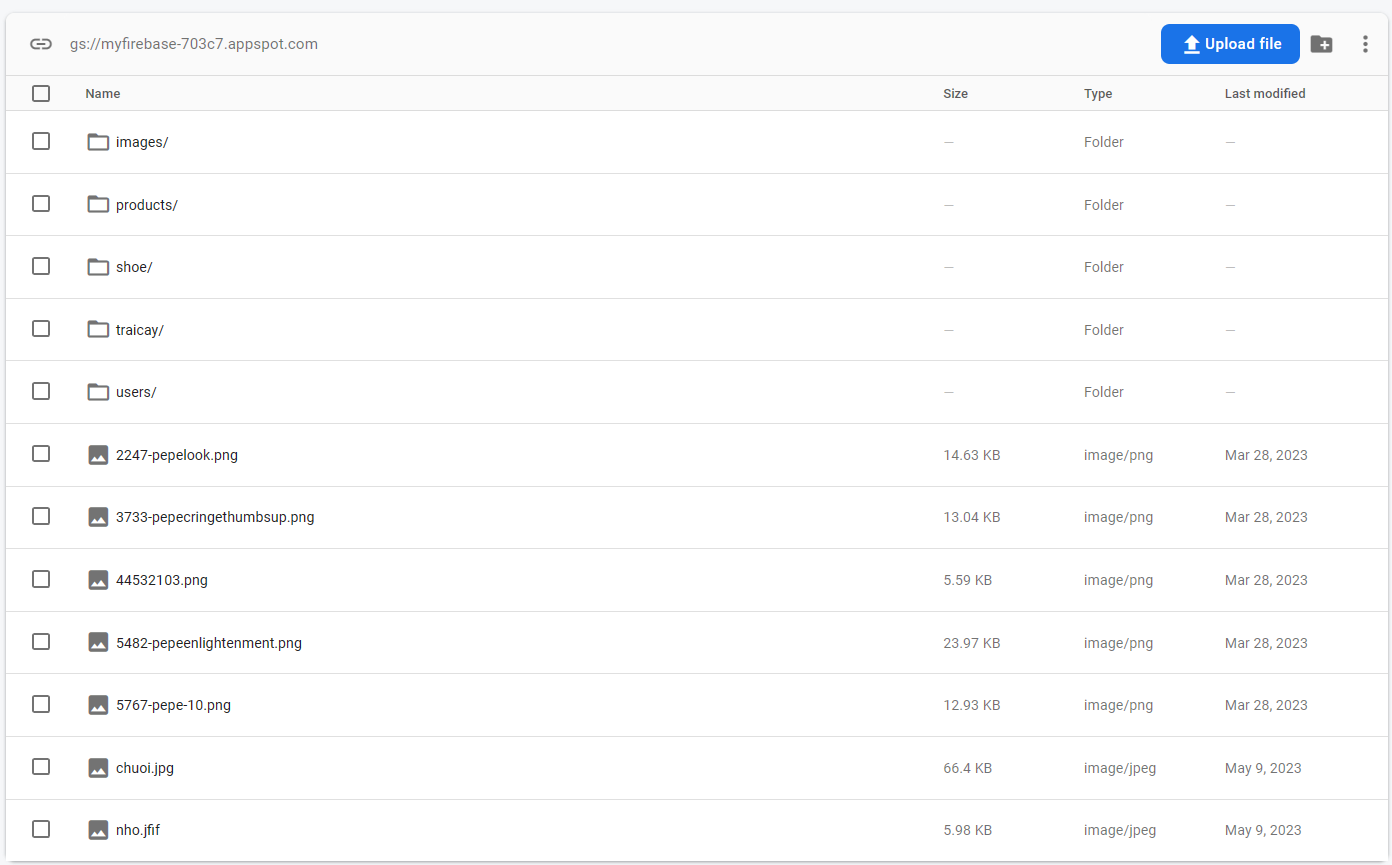
* + Chứa nhiều Document, mỗi Document có Document id là id của user hiện tại chứa 6 trường thuộc tính: Tên(name), Số điện thoại (phone), url ảnh(imageUrl), Địa chỉ(address), Là Admin?(isAdmin).



Hình 4: Người dùng

### Lưu trữ ảnh

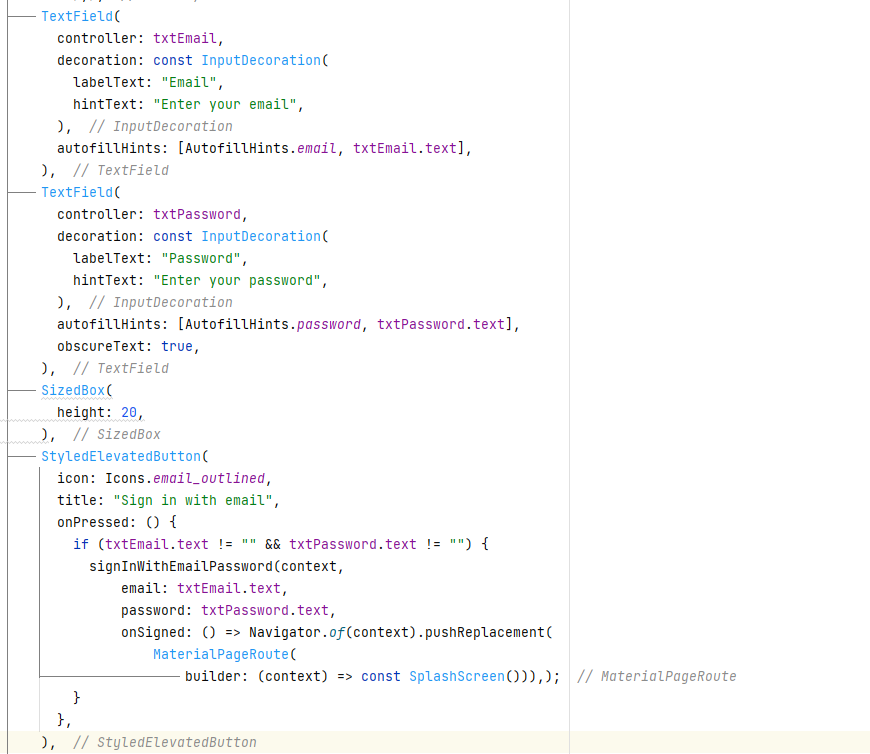
* + Ảnh sản phẩm được lưu trữ ở Storage FireBase.



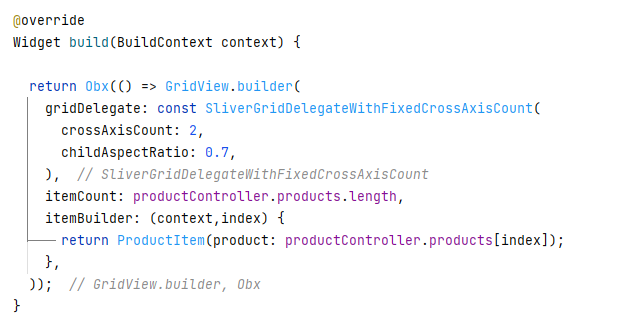
Hình 5: Lưu trữ ảnh sản phẩm

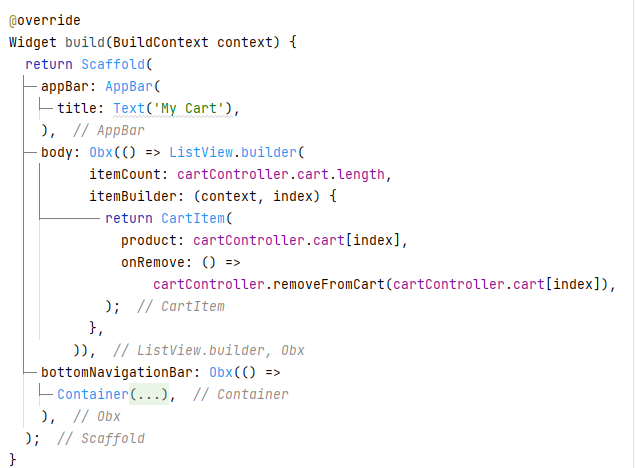
## Mã lệnh giao diện

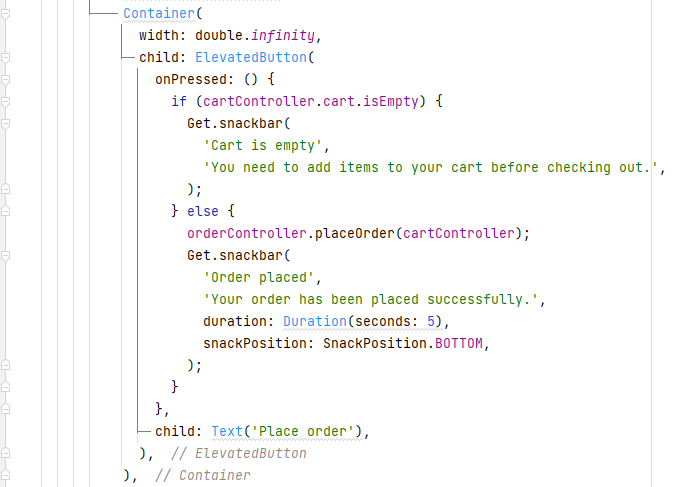
* 1. Giao diện đăng nhập (page\_login.dart)



* 1. Giao diện trang hiển thị sản phẩm (product\_page.dart)



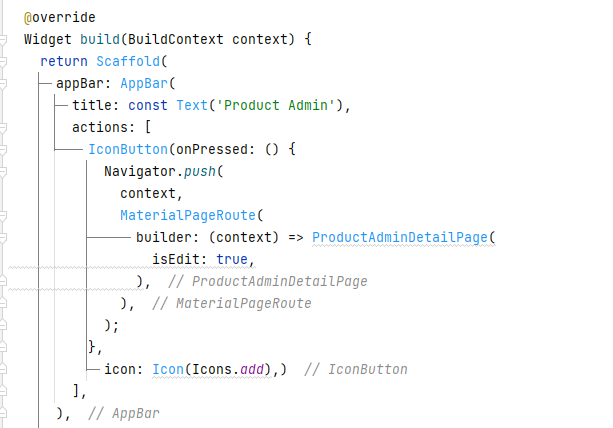
* 1. Giao diện giỏ hàng (cart\_page.dart)
  2. Đặt hàng (cart\_page.dart)

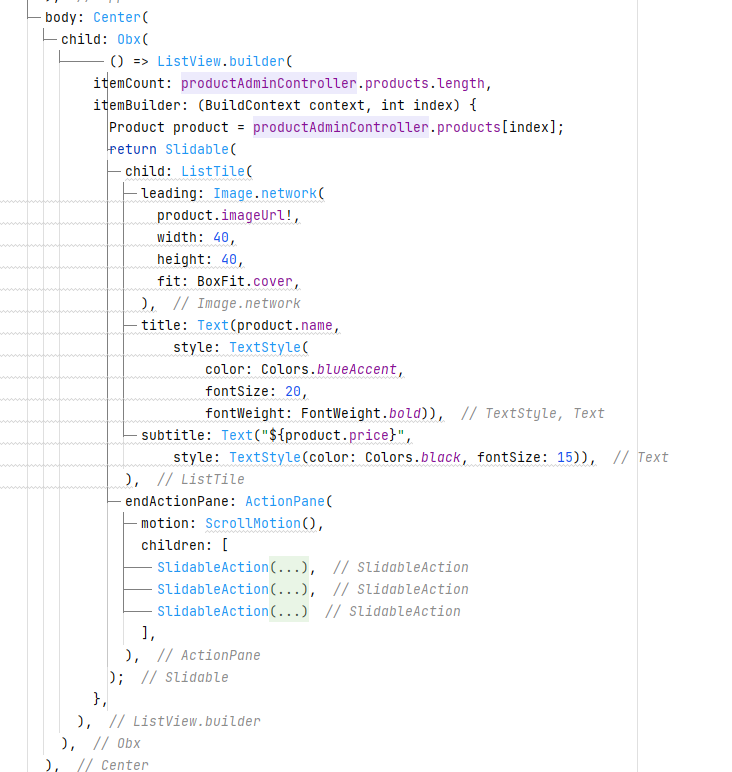


* 1. Giao diện yêu thích (favourite\_page.dart)

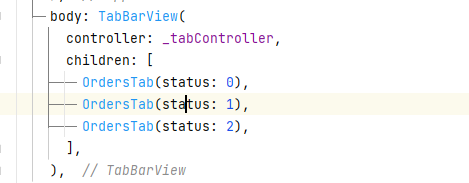


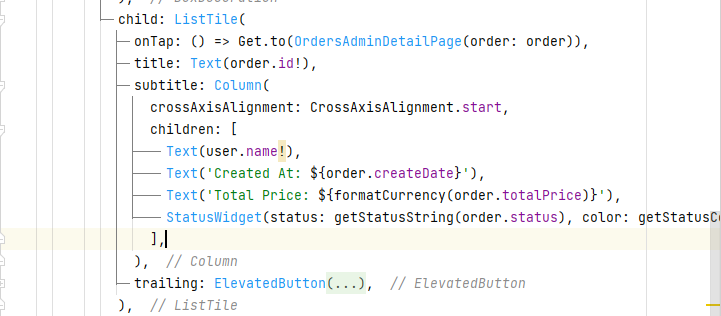
* 1. Giao diện quản lý sản phẩm (product\_adminpage.dart)





* 1. Giao diện hiển thị danh sách đơn hàng (orders\_adminpage.dart)

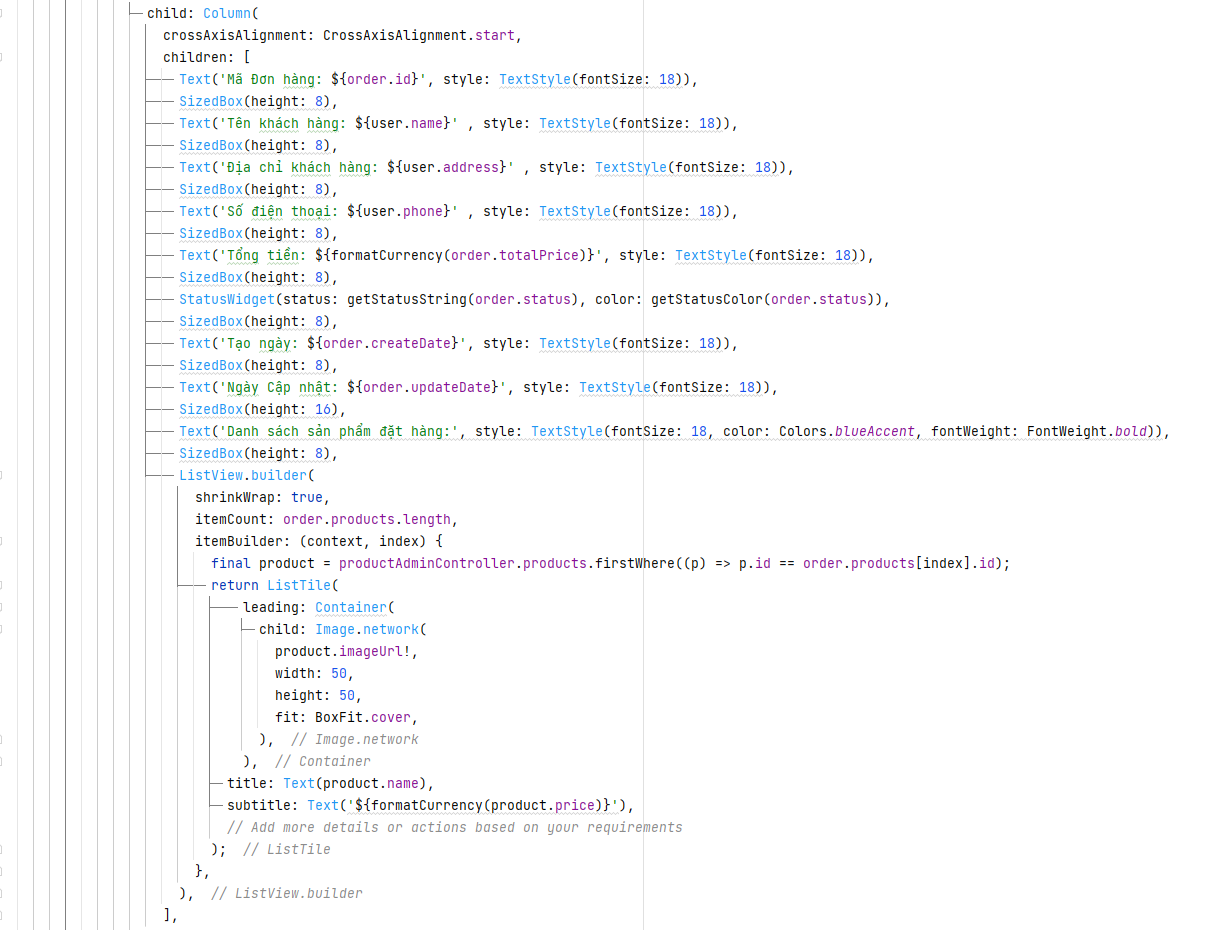




* 1. Giao diện cập nhật trạng thái đơn hàng (order\_adminpage.dart)



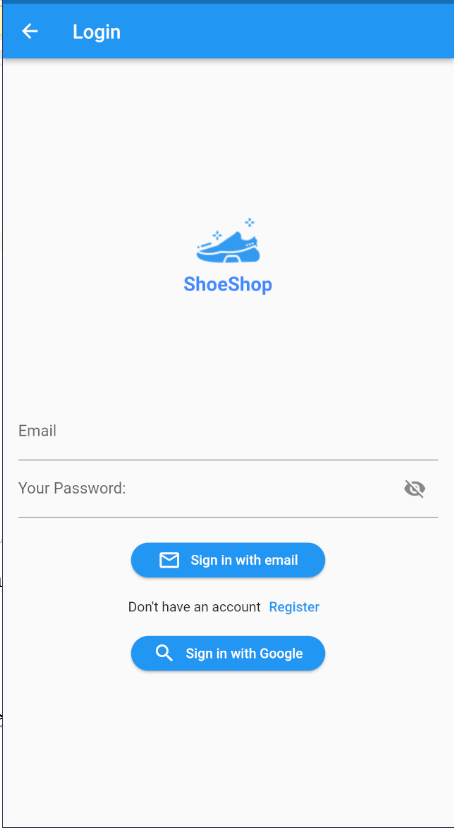
* 1. Giao diện hiển thị chi tiết đơn hàng (order\_admin\_detailpage.dart)



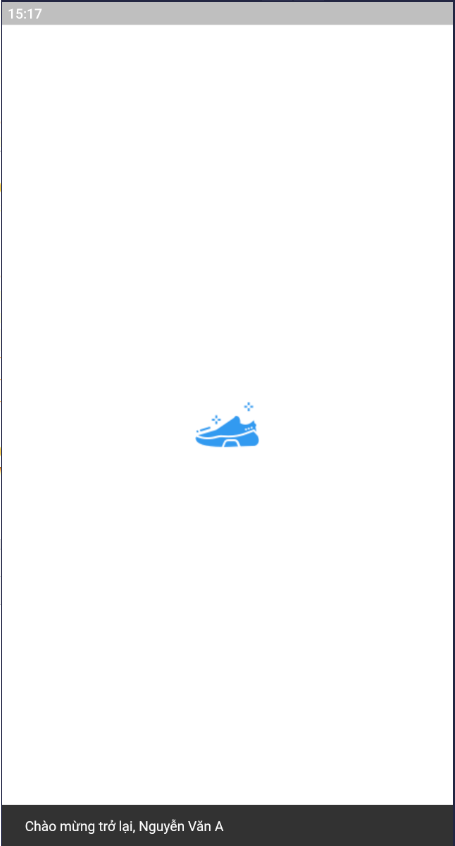
## Giao diện hệ thống

### Giao diện đăng nhập

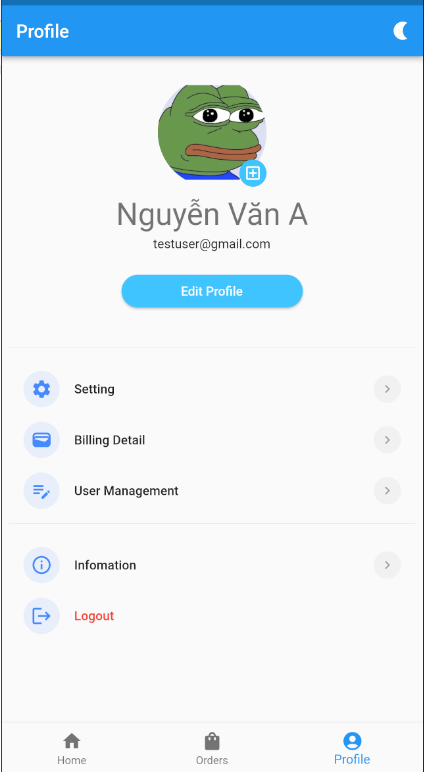
* + Hệ thống đăng nhập tài khoản qua hai cách: đăng nhập bằng email và mật khẩu qua giao diện đăng ký và đăng nhập bằng gmail. Sau khi đăng nhập thành công sẽ hiện thông báo “Chào mừng trở lại, Tên user!”.
  + Nếu đăng nhập bằng gmail lần đầu tiên thì sẽ yêu cầu nhập thông tin để bổ sung vào thông tin tài khoản
  + Nếu đăng nhập bằng tài khoản admin   
    (tài khoản: [admin@gmail.com](mailto:admin@gmail.com), mật khẩu admin@) thì sẽ tự động chuyển hướng đến giao diện chính quản lý



Hình 5: Giao diện để đăng nhập hệ thống



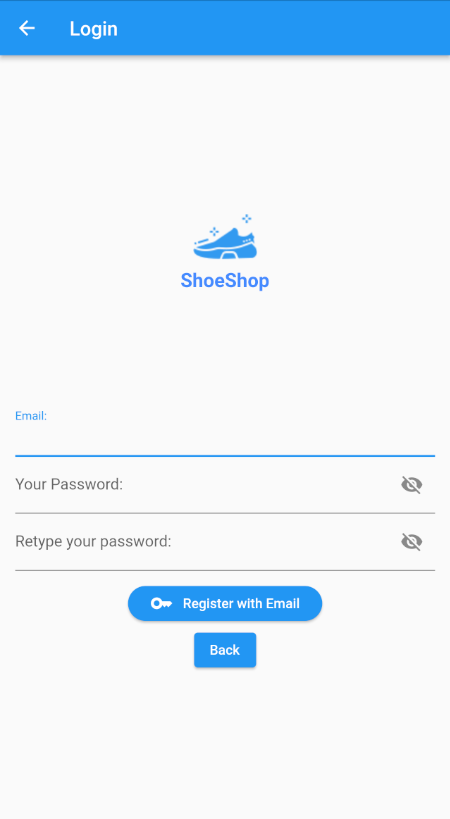
Hình 6: SplashScreen để chuyển hướng



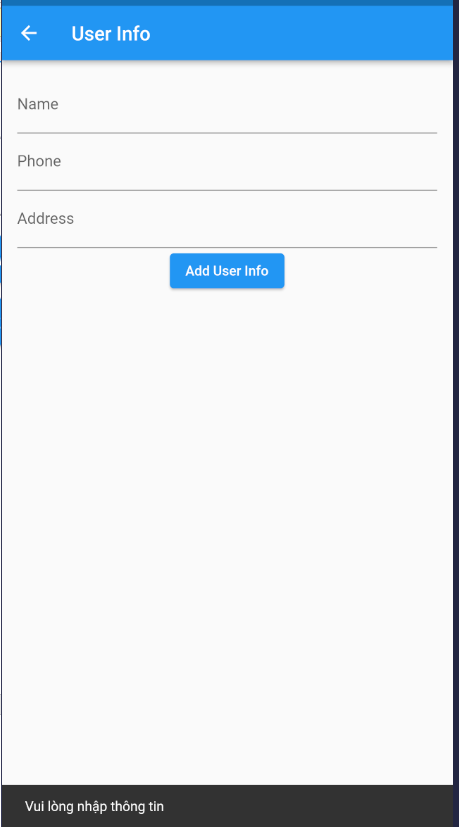
Hình 7: Giao diện profile khi đăng nhập thành công

### Giao diện đăng ký

* + Nhập email và mật khẩu để đăng ký, sau đó nhập thêm thông tin cơ bản vào tài khoản



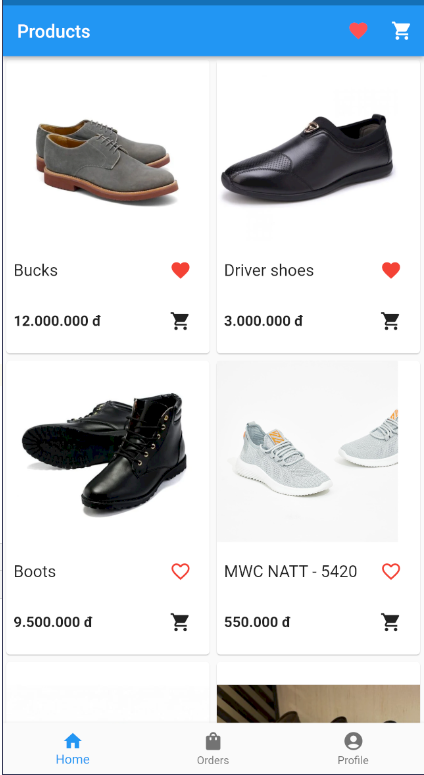
Hình 8: Giao diện đăng ký bằng email, mật khẩu



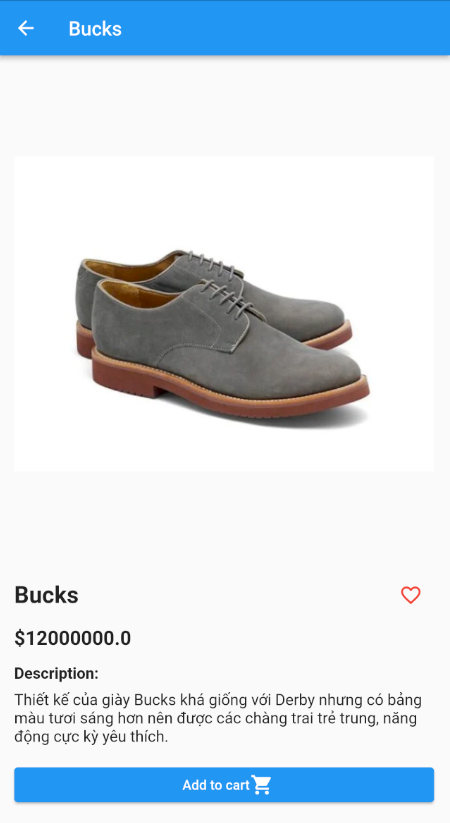
Hình 9: Giao diện thêm thông tin cơ bản

### Giao diện trang chủ

* + Hiển thị BottomNavigationBar để chuyển trang Home, Orders và Profile, Appbar có 2 nút để xem danh mục yêu thích và danh mục cart.
  + Khi nhấn vào ảnh từng sản phẩm sẽ chuyển hướng đến màn hình chi tiết sản phẩm
  + Nút hình giỏ hàng để thêm sản phẩm vào giỏ hàng



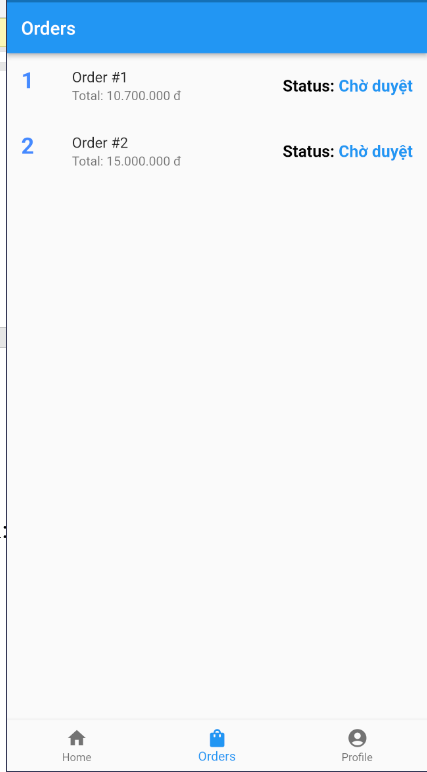
Hình 10: Giao diện trang chủ (hiển thị sản phẩm)



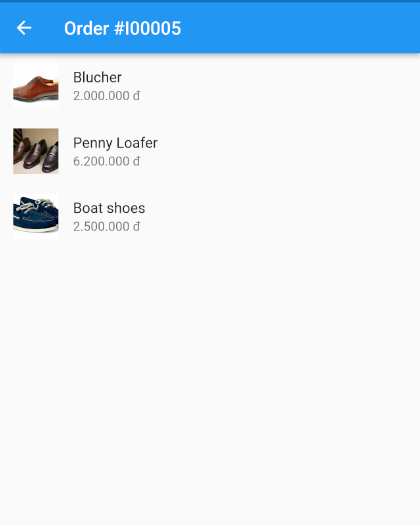
Hình 11: Giao diện chi tiết sản phẩm

### Giao diện đơn hàng

* + Hiển thị danh sách đơn hàng đã đặt và trạng thái của đơn hàng
  + Khi nhấn vào sẽ mở ra danh sách sản phẩm trong đơn hàng đó.



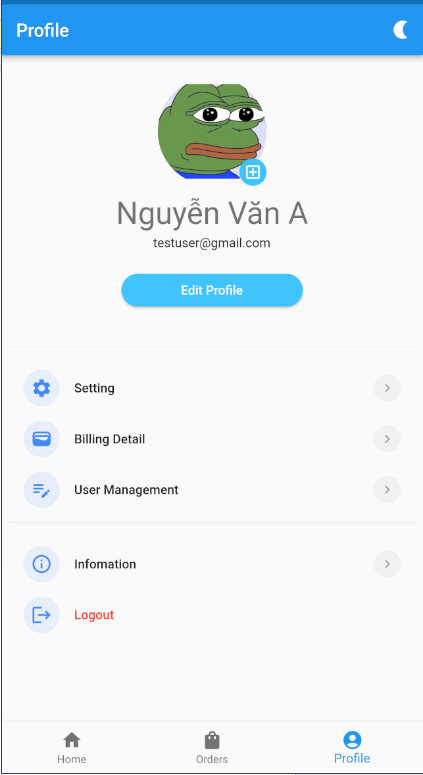
Hình 12: Giao diện danh sách đơn hàng



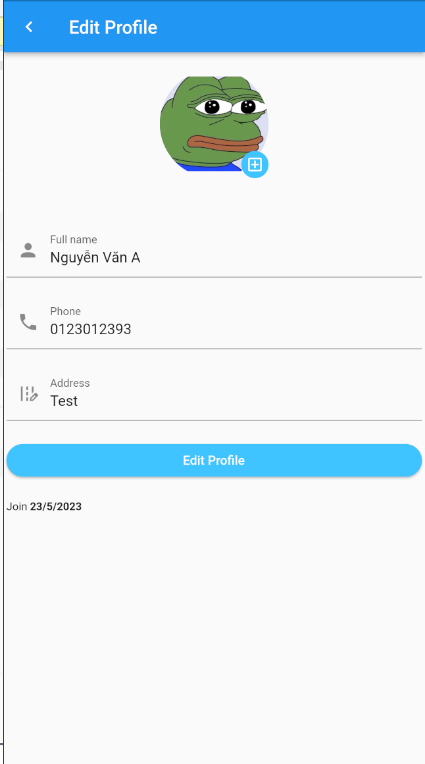
Hình 14: Giao diện chi tiết đơn hàng

### Giao diện profile

* + Hiển thị tên người dùng và nút edit profile để chỉnh sửa thông tin tài khoản, nút logout để đăng xuất, các nút còn lại đang phát triển.



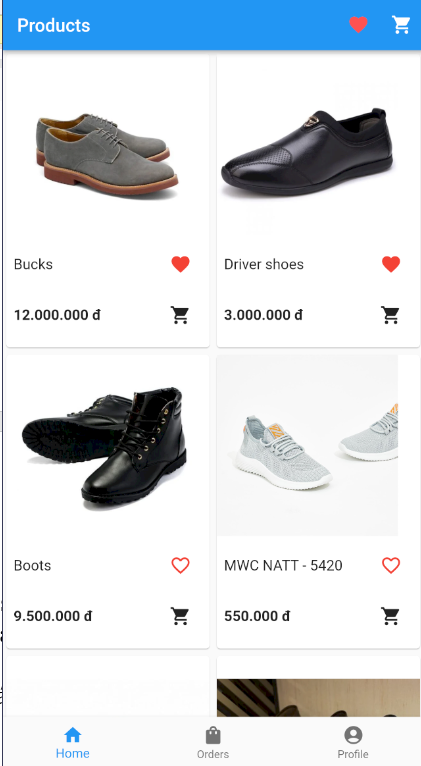
Hình 15: Giao diện Profile đã đăng nhập



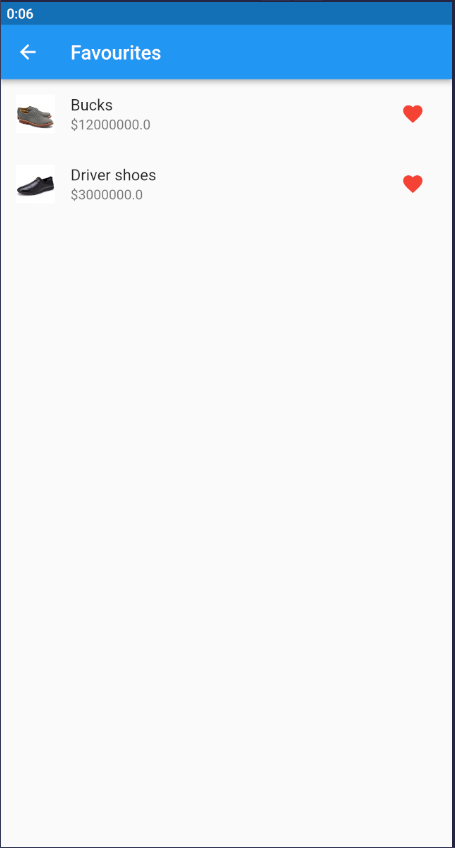
Hình 16: Giao diện chỉnh sửa profile

### Giao diện yêu thích

* + Khi ta nhấn nút yêu thích trên màn hình hiển thị sản phẩm hoặc chi tiết sản phẩm, ta có thể thêm sản phẩm vào danh mục yêu thích
  + Ta có thể xóa sản phẩm yêu thích ra khỏi danh sách yêu thích



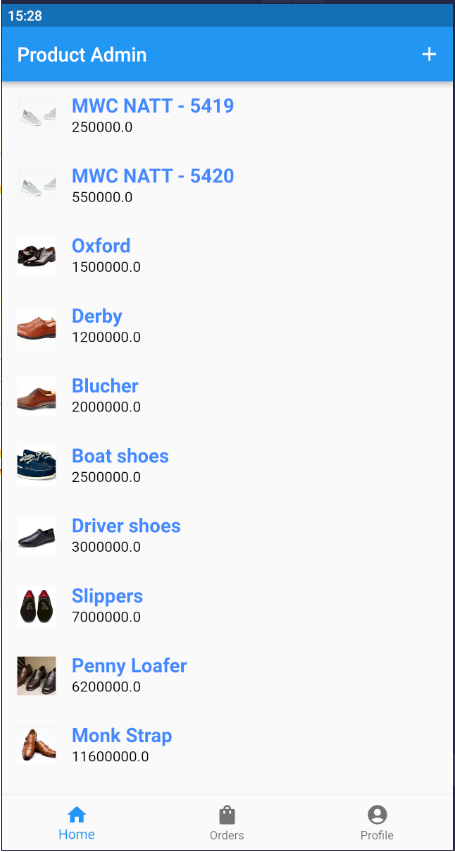
Hình 17: Giao diện có nút yêu thích sản phẩm



Hình 18: Giao diện danh mục sản phẩm yêu thích

### Giao diện chính (quản lý)

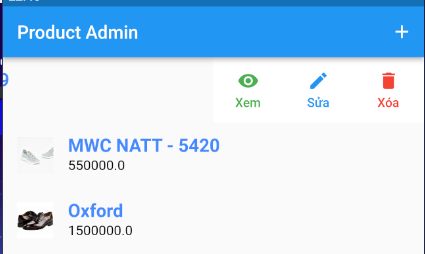
* + BottomNavigationBar hiển thị để chuyển trang: Products(Quản lý sản phẩm), Orders (Cập nhật trạng thái đơn hàng), Profile (profile của quản lý, đăng xuất)



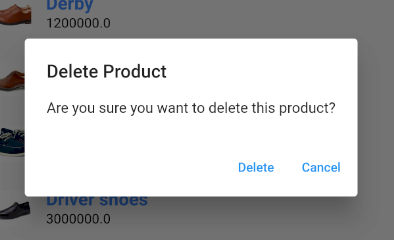
Hình 19: Giao diện quản lý sản phẩm

### Giao diện quản lý sản phẩm

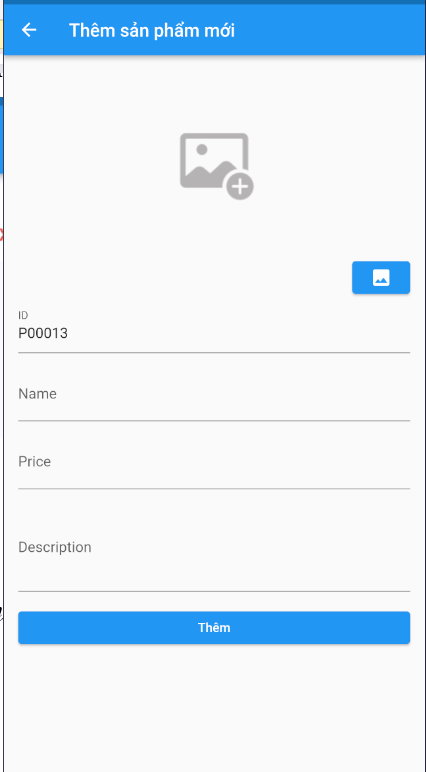
* + Nút + trên appbar để truy cập thêm sản phẩm.
  + Kéo slide từng sản phẩm để truy cập sửa, xóa sản phẩm.



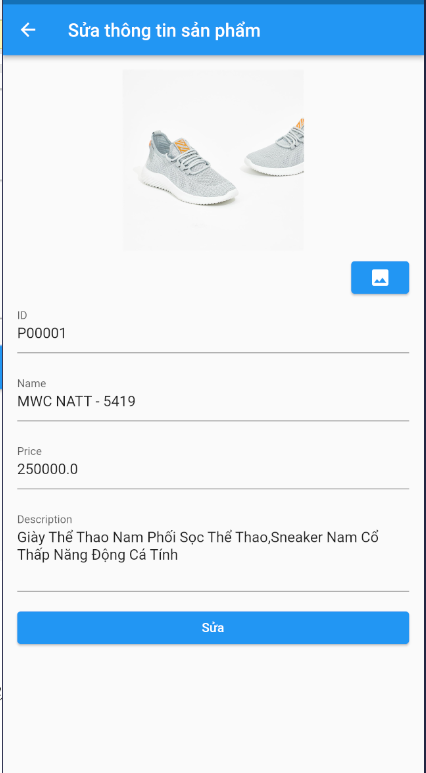
Hình 20: Giao diện quản lý sản phẩm



Hình 21: Giao diện xóa sản phẩm



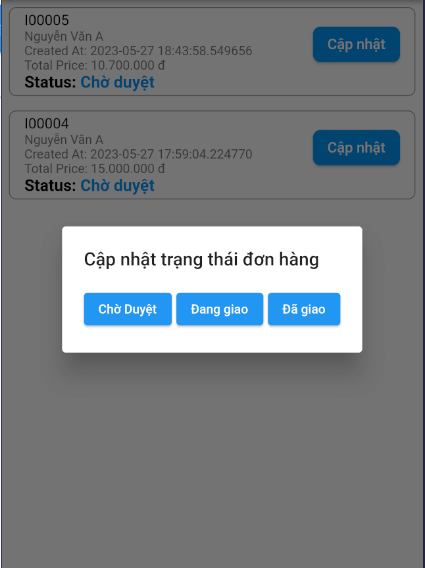
Hình 22: Giao diện thêm sản phẩm



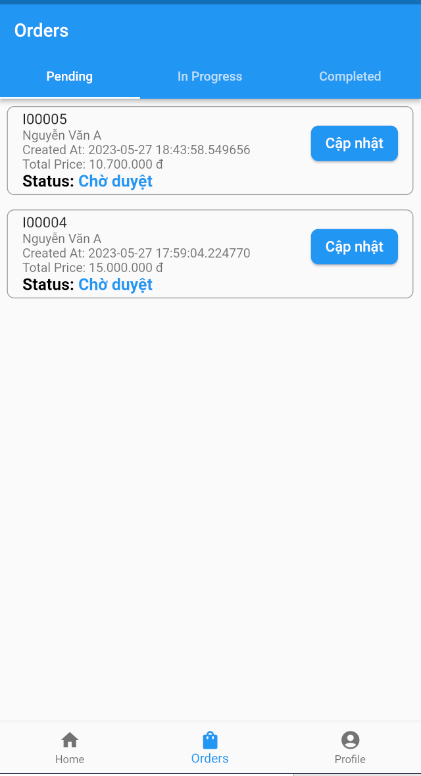
Hình 23: Giao diện sửa sản phẩm

### Giao diện cập nhật đơn hàng

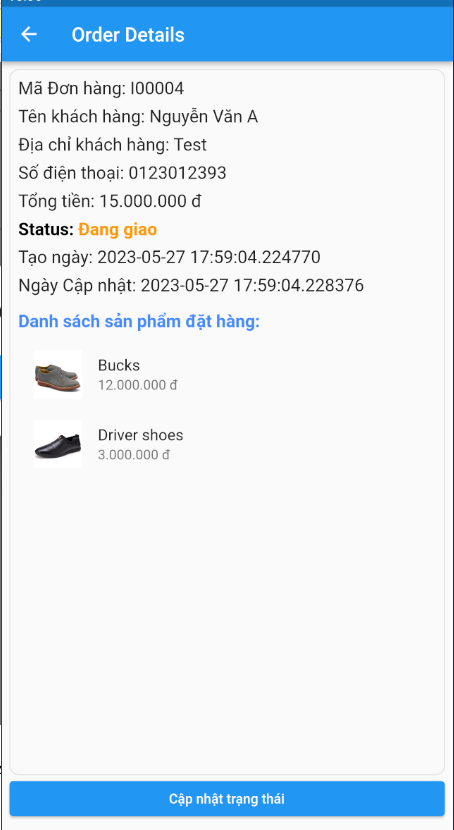
* + Hiển thị danh sách các đơn hàng theo từng trạng thái và cập nhật trạng thái các đơn hàng.
  + Hiển thị chi tiết đơn hàng khi nhấn vào từng đơn hàng



Hình 24: Giao diện cập nhật trạng thái đơn hàng



Hình 25: Giao diện danh sách đơn hàng



Hình 26: Giao diện chi tiết đơn hàng