

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**  
**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO THỰC TẬP CƠ SỞ**

**Nhóm 01**

**Đề tài: Xây dựng ứng dụng bán hàng  
mỹ phẩm và làm đẹp**

**Giảng viên hướng dẫn: Trịnh Thị Vân Anh**

**Sinh viên thực hiện: Nguyễn Thị Cẩm Ly**

**Mã sinh viên: B21DCCN506**

**Hà Nội 05/2024**

## LỜI CẢM ƠN

Để hoàn thành đề tài báo cáo thực tập cơ sở này trước tiên em xin gửi đến các quý thầy cô giáo của Học viện Công nghệ Bưu chính Viễn thông lời cảm ơn chân thành và sâu sắc nhất.

Đặc biệt, em xin gửi lời cảm ơn chân thành tới cô Trịnh Thị Vân Anh đã hướng dẫn và hỗ trợ cho em trong quá trình thực hiện bài tập lớn môn Thực tập cơ sở. Trong quá trình học tập và tìm hiểu em đã học được những kiến thức bổ ích cũng như những trải nghiệm thực tế.

Trong quá trình tìm hiểu và làm báo cáo, về kiến thức và kinh nghiệm còn hạn chế nên em rất mong nhận được những đóng góp từ cô, để hoàn thiện thêm kỹ năng và có rút ra được những kinh nghiệm bổ ích cho những bài báo cáo sắp tới.

Em xin chân thành cảm ơn ạ!

# Mục lục

<b>I.Giới thiệu đề tài và công nghệ sử dụng .....</b>	<b>1</b>
1.Giới thiệu đề tài.....	1
2.Các công cụ và công nghệ sử dụng.....	1
2.1.React Native .....	1
2.2.Firebase .....	5
2.3.Công cụ Visual Studio Code .....	7
<b>II.Phân tích và thiết kế ứng dụng .....</b>	<b>7</b>
1.Phân tích hệ thống .....	7
2.Biểu đồ UseCase toàn hệ thống .....	8
3.Biểu đồ UseCase cho những chức năng chính .....	8
4.Thiết kế cơ sở dữ liệu.....	10
<b>III.Cài đặt ứng dụng.....</b>	<b>13</b>
1.  Trang đăng ký, Đăng nhập.....	13
2.  Trang chủ .....	14
3.  Trang chi tiết sản phẩm.....	14
4.  Giỏ hàng và thanh toán .....	15
5.  Trang thêm địa chỉ.....	16
6.  Trang tìm kiếm sản phẩm .....	16
7.  Trang User và sửa thông tin user.....	17
8.  Trang Setting.....	17
9.  Trang chủ của Admin.....	19
10.  Trang sửa sản phẩm:.....	19
11.  Trang thêm sản phẩm .....	19
12.  Quản lý đơn hàng .....	20
13.  Xem thông tin khách hàng .....	20
<b>IV.Kết quả đạt được và hướng phát triển .....</b>	<b>21</b>

## **I.Giới thiệu đề tài và công nghệ sử dụng**

### **1.Giới thiệu đề tài**

Hiện nay trên thế giới với sự phát triển mạnh mẽ của khoa học công nghệ, cuộc sống của chúng ta đã thay đổi rất nhiều. Mọi công việc trong cuộc sống giờ đây đều có sự can thiệp của công nghệ. Một việc dễ thấy nhất là mua sắm, có thể nói ngày nay đang là một thời kì bùng nổ của mua sắm online với sự ra đời của rất nhiều ứng dụng mua sắm lớn như Shopee, Tiktok,... Mọi người giờ đây chỉ cần trong một vài thao tác trên các thiết bị di động là có thể dễ dàng mua cho mình những sản phẩm mong muốn và được giao đến tận nhà.

Mỹ phẩm và làm đẹp có lẽ là một trong những sản phẩm được quan tâm nhiều nhất. Chúng ta đều muốn chăm sóc bản thân, muốn bản thân đẹp lên mỗi ngày. Từ suy nghĩ đó nên em đã nghĩ đến làm một ứng dụng di động chuyên phân phối những mặt hàng mỹ phẩm chính hãng với mức giá hợp lý để cho mọi người có thể dễ dàng mua sắm những sản phẩm chất lượng nhất.

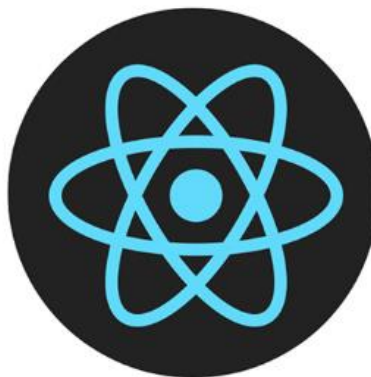
### **2.Các công cụ và công nghệ sử dụng**

#### **2.1.React Native**

React Native là một framework phát triển ứng dụng di động mã nguồn mở được Facebook tạo ra. React Native sử dụng ngôn ngữ lập trình JavaScript để tạo ứng dụng di động đa nền tảng trên cả Android và iOS.

Vì vậy các lập trình viên có thể tiết kiệm thời gian và công sức khi phát triển ứng dụng di động. Ra mắt năm 2015, React Native đã trở thành một trong những framework phổ biến nhất. Theo một khảo sát của StackOverflow năm 2021, có đến 58,08% các nhà phát triển yêu thích làm việc trên React Native trong năm qua và họ sẽ tiếp tục lựa chọn React Native trong tương lai.

Các hoạt động của React Native:



React Native hoạt động trên ba thread:

- Chuỗi giao diện người dùng(UI Thread): Chuỗi ứng dụng chính có quyền truy cập vào giao diện người dùng.
- Shadow Thread: Thread nền sử dụng thư viện React để tính toán bố cục ứng dụng.
- Chuỗi JavaScript (JavaScript Thread): Chuỗi thực thi có chứa mã React.

React Native sẽ hoạt động bằng cách tích hợp Main Thread và JS Thread cho một ứng dụng mobile. Main Thread sẽ có vai trò cập nhật giao diện người dùng (UI) và xử lý tương tác của người dùng. JS Thread là nơi thực thi và xử lý mã Javascript. Main Thread và JS Thread hoạt động độc lập vì vậy cần sử dụng một cầu nối (Bridge) để giúp 2 thread này tương tác với nhau. Cầu nối cho phép chúng chuyển đổi dữ liệu giữa thread này và thread kia, giúp chúng giao tiếp mà không phải phụ thuộc vào nhau.

### **Ưu điểm của React Native:**

- Khả năng tái sử dụng code – phát triển đa nền tảng

Với React Native, có thể sử dụng lại các code và các ứng dụng có thể chạy hiệu quả trên nhiều nền tảng. Đây là điều mà các CEO và các chủ sở hữu đánh giá rất cao. Họ có thể tích hợp 90% native framework để sử dụng lại code cho cả hai hệ thống hoạt động. Ưu điểm này giúp người dùng tiết kiệm được thời gian, giảm chi phí phát triển ứng dụng, cũng như tận dụng được nguồn nhân lực tốt hơn.

- Cộng đồng người dùng lớn

React Native là một nền tảng open-source JavaScript, cho phép người dùng đóng góp kiến thức của mình vào sự phát triển của framework. Tất cả mọi người đều có thể truy cập miễn phí. Nhờ có cộng đồng người dùng rất lớn trên toàn thế giới, nên người dùng có thể tìm sự hỗ trợ nếu gặp khó khăn gì.

- Tiết kiệm chi phí

React Native có thể tái sử dụng code giúp giảm chi phí phát triển ứng dụng. Bên cạnh đó, React Native cũng không cần phải sử dụng các code riêng biệt cho cả hai nền tảng vì cả hai hệ điều hành đều có thể được mã hóa bằng một ngôn ngữ lập trình duy nhất. Vì thế, người dùng không cần phải thuê một nguồn nhân lực lớn các nhà phát triển khác nhau để hoàn thành công việc.

- Tính ổn định và tối ưu

Vì được phát triển bởi Facebook, nên React Native có hiệu năng ổn định khá cao.

- Trải nghiệm người dùng tốt hơn khi so sánh với ứng dụng Hybrids.

- Mã React Native giúp đơn giản hóa quá trình xử lý dữ liệu.
- Không cần đội ngũ phát triển ứng dụng lớn.

### Nhược điểm của React Native:

- Bảo mật kém

React Native là một thư viện JavaScript và open-source framework, điều này tạo ra một lỗ hổng lớn về bảo mật. Nếu người dùng cần tạo các ứng dụng tài chính và ngân hàng cần độ bảo mật cao thì React Native không phải là sự lựa chọn an toàn.

- Mất nhiều thời gian để khởi tạo

Vấn đề của React Native là mất rất nhiều thời gian để khởi tạo thời gian chạy, ngay cả đối với các thiết bị công nghệ cao. Điều này xảy ra là bởi vì thread JavaScript cần thời gian để khởi tạo.

- Gỡ lỗi và quản lý mã nguồn:

Khi có sự cố xảy ra, React Native tạo ra một tầng phức tạp trong quá trình gỡ lỗi và quản lý mã nguồn.. Điều này đặt ra thách thức đối với việc duy trì và mở rộng dự án.

### Các thư viện sử dụng:

1. Thư viện **@react-native-async-storage/async-storage** là một phần của React Native và được sử dụng để lưu trữ dữ liệu cục bộ (local storage) trong ứng dụng của bạn. Đây là một phương thức phổ biến để lưu trữ dữ liệu như cài đặt người dùng, thông tin phiên làm việc, dữ liệu cache, và nhiều hơn nữa. AsyncStorage cung cấp một API đơn giản để lưu trữ và truy xuất dữ liệu.
2. Các thư viện **@react-native-firebase/app**, **@react-native-firebase/auth**, và **@react-native-firebase/firestore** là các phần của React Native Firebase, một bộ công cụ được thiết kế để làm việc với Firebase trong ứng dụng React Native.
  - @react-native-firebase/app:** Thư viện này cung cấp các phương tiện để khởi tạo ứng dụng Firebase trong ứng dụng React Native.
  - @react-native-firebase/auth:** Thư viện này cung cấp các công cụ để quản lý xác thực người dùng trong Firebase Authentication.
  - @react-native-firebase/firestore:** Thư viện này cung cấp các công cụ để làm việc với Cloud Firestore, cơ sở dữ liệu NoSQL đám mây của Firebase.
3. **Firebase/firestore, Firebase/storage:**

Firebase/firestore cung cấp các phương thức để tương tác với cơ sở dữ liệu firestore như : doc, collection, getDoc, getDocs,...

Firebase/storage: cung cấp các phương thức để tương tác với storage như: `getStorage`, `ref`, `uploadBytes`,....

#### **4. React-navigation/native**

Thư viện này cung cấp các công cụ cơ bản cho việc quản lý điều hướng trong ứng dụng React Native của bạn. Nó cung cấp các thành phần và hook cho việc tạo và quản lý các màn hình và bộ điều hướng.

#### **5. React-navigation/stack**

Thư viện này cung cấp các công cụ cho việc quản lý điều hướng theo kiểu Stack Navigation trong ứng dụng React Native. Nó cho phép bạn xây dựng một ngăn xếp các màn hình có thể được đẩy và rút khỏi theo nguyên tắc "ngăn xếp".

#### **6. React-navigation/bottom-tabs**

Thư viện này cung cấp các công cụ cho việc tạo và quản lý Bottom Tab Navigation trong ứng dụng React Native của bạn. Bottom Tab Navigation cho phép bạn đặt các tab điều hướng ở dưới cùng của màn hình.

#### **7. React-navigation/native-stack**

Thư viện này cung cấp một cách tiếp cận khác cho việc quản lý điều hướng trong ứng dụng React Native, cung cấp một ngăn xếp điều hướng mới trong React Navigation.

#### **8. expo-location**

Thư viện này cung cấp các công cụ cho việc truy cập và quản lý vị trí địa lý trong ứng dụng của bạn. Nó cung cấp các phương thức cho phép bạn lấy vị trí hiện tại của thiết bị, theo dõi sự thay đổi vị trí, và thực hiện các tác vụ liên quan đến vị trí như định vị ngược, tính toán khoảng cách

#### **9. react-native-image-picker**

Thư viện này cung cấp các công cụ để chọn và chụp hình ảnh từ thư viện hình ảnh hoặc máy ảnh của thiết bị. Nó giúp bạn dễ dàng tích hợp chức năng chọn ảnh hoặc chụp ảnh trong ứng dụng của mình một cách thuận tiện.

#### **10. react-native-keyboard-aware-scroll-view**

Thư viện này cung cấp một ScrollView tự động điều chỉnh kích thước để tránh việc bị che khuất bởi bàn phím khi nhập liệu. Nó giúp bạn tạo ra các màn hình có thể cuộn một cách mượt mà mà không bị ảnh hưởng bởi bàn phím.

#### **11. react-native-modal-selector**

Thư viện này cung cấp một cửa sổ modal đơn giản cho việc chọn các tùy chọn từ danh sách. Nó cho phép bạn tạo ra các dropdown hoặc menu lựa chọn đơn giản trong ứng dụng của mình.

#### **12. react-native-uuid**

Thư viện "react-native-uuid" cung cấp một giao diện đơn giản để tạo và quản lý các UUID trong ứng dụng React Native. UUID là một chuỗi 128-bit được sử dụng để đại diện cho một đối tượng duy nhất mà không cần phụ thuộc vào bất kỳ ngữ cảnh nào.

## 2.2.Firebase

\*Giới thiệu:

**Firestore** là một nền tảng phát triển ứng dụng di động và website do Google cung cấp. Nền tảng này cung cấp bộ công cụ toàn diện với các API đơn giản và mạnh mẽ, giúp các nhà phát triển xây dựng ứng dụng mà không cần lo lắng về việc quản lý server hay cơ sở hạ tầng backend phía sau.

Firestore không chỉ giúp tiết kiệm thời gian triển khai ứng dụng mà còn hỗ trợ mở rộng quy mô một cách dễ dàng. Nền tảng này cung cấp dịch vụ cơ sở dữ liệu đám mây, được vận hành trên hệ thống máy chủ mạnh mẽ của Google, giúp việc lập trình trở nên đơn giản hơn nhờ tự động hóa nhiều thao tác liên quan đến cơ sở dữ liệu.

Firestore còn cung cấp các giao diện lập trình ứng dụng (API) thân thiện, giúp thu hút nhiều người dùng hơn và tăng khả năng sinh lời cho các nhà phát triển. Đặc biệt, Firestore đảm bảo tính đa năng và bảo mật cao, hỗ trợ cả hai hệ điều hành phổ biến là Android và iOS.

### \*Firestore Authentication

Firestore Authentication một trong những tính năng nổi bật của Firestore là hệ thống xác thực người dùng, hỗ trợ nhiều phương thức đăng nhập khác nhau như Email, Facebook, Twitter, GitHub, và Google.

Firestore còn cung cấp khả năng xác thực ẩn danh, rất hữu ích cho các ứng dụng muốn cho phép người dùng trải nghiệm mà không cần đăng nhập ngay lập tức. Tính năng xác thực của Firestore giúp bảo vệ thông tin cá nhân của người dùng, đảm bảo an toàn và ngăn ngừa rủi ro bị đánh cắp tài khoản.

### \*Firestore Database :

Firestore là một Database linh hoạt và dễ mở rộng cho mobile, web. Firestore giúp cho việc đồng bộ dữ liệu giữa các ứng dụng phía client một cách nhanh chóng (Realtime) và hỗ trợ lưu offline data trong ứng dụng của bạn.

Firestore là một cloud-hosted, NoSQL database mà các ứng dụng phía client có thể trực tiếp truy cập thông qua native SDKs. Nó lưu dữ liệu theo mô hình dữ liệu NoSQL. Dữ liệu được lưu trữ trong các file tài liệu chứa các trường được ánh xạ vào các giá trị. Các file tài liệu này được lưu trữ trong các tập hợp chúng có thể sử dụng nó để tổ chức dữ liệu và truy vấn dữ liệu. Cloud Firestore hỗ trợ rất nhiều kiểu dữ liệu từ đơn giản như String, Integer hay những kiểu dữ liệu phức tạp như các nested object.



Firestore cho phép đồng bộ hóa, lưu trữ và truy xuất dữ liệu theo thời gian thực đồng thời cung cấp chức năng mạnh mẽ, bao gồm hỗ trợ ngoại tuyến, tổ chức dữ liệu phân cấp và bộ khả năng truy vấn toàn diện.

### \* **Firestore Storage**

Storage là dịch vụ được xây dựng cho mục đích lưu trữ và quản lý các nội dung mà người dùng ứng dụng tạo ra như ảnh, videos hay dữ liệu dạng file.

Firestore Storage cung cấp các API cho việc uploads và download các file từ app của bạn một cách bảo mật và bạn không cần quan tâm đến chất lượng đường truyền mạng.

Ưu điểm của Firestore:

- Firestore giúp tiết kiệm thời gian và chi phí phát triển ứng dụng, bằng cách cung cấp các dịch vụ và công cụ sẵn có và dễ sử dụng.
- Firestore cho phép các nhà phát triển tập trung vào việc xây dựng giao diện và tính năng của ứng dụng, mà không cần lo lắng về việc xử lý và lưu trữ dữ liệu ở phía máy chủ.
- Firestore hỗ trợ nhiều nền tảng khác nhau, như web, Android, iOS, Unity, C++,...
- Firestore có khả năng mở rộng tự động theo quy mô của ứng dụng, không cần phải quản lý cơ sở hạ tầng hay máy chủ.
- Firestore được hưởng lợi từ hệ thống máy chủ mạnh mẽ và an toàn của Google, đảm bảo hiệu suất cao và độ tin cậy cao cho các ứng dụng.
- Firestore hỗ trợ nhiều ngôn ngữ lập trình và nền tảng khác nhau, cho phép phát triển các ứng dụng đa nền tảng một cách hiệu quả và dễ dàng.
- Firestore có nhiều tính năng hữu ích cho việc phát triển ứng dụng như xác thực người dùng, lưu trữ tệp tin, tích hợp thanh toán, thực hiện kiểm thử phân mềm và gỡ lỗi.

Nhược điểm của Firestore:

- Firestore có giới hạn về dung lượng lưu trữ và số lượng kết nối trong phiên bản miễn phí. Để sử dụng các tính năng nâng cao hoặc quy mô lớn hơn, các nhà phát triển phải trả phí theo mức sử dụng.
- Firestore có thể không đáp ứng được một số yêu cầu đặc biệt hoặc phức tạp của các ứng dụng. Ví dụ, Firestore Realtime Database chỉ hỗ trợ cơ sở dữ liệu NoSQL, không hỗ trợ các truy vấn phức tạp hay các thao tác với dữ liệu liên quan.
- Firestore có thể gặp khó khăn trong việc tích hợp với một số công nghệ hoặc dịch vụ khác. Ví dụ, Firestore không hỗ trợ GraphQL hay Apollo Client, hai công nghệ phổ biến trong việc xây dựng API hiện đại.

### 2.3.Công cụ Visual Studio Code

**Visual Studio Code (VS Code)** là một trình soạn thảo mã nguồn mở được phát triển bởi Microsoft, ra mắt lần đầu vào tháng 4 năm 2015. Đây là một công cụ mạnh mẽ và linh hoạt, được thiết kế để hỗ trợ lập trình viên trong việc viết và quản lý mã nguồn một cách hiệu quả.

**\*Đặc điểm chính:**

- **Miễn phí và Mã nguồn mở:** VS Code là một công cụ miễn phí và mã nguồn mở, có sẵn trên GitHub, cho phép cộng đồng đóng góp và tùy chỉnh.
- **Đa nền tảng:** Hỗ trợ trên các hệ điều hành Windows, macOS và Linux, giúp lập trình viên có thể làm việc trên nhiều môi trường khác nhau.
- **Giao diện người dùng thân thiện:** Giao diện trực quan và dễ sử dụng, với các tính năng mạnh mẽ được tổ chức một cách hợp lý.
- **Hỗ trợ đa ngôn ngữ:** VS Code hỗ trợ nhiều ngôn ngữ lập trình thông qua các tiện ích mở rộng (extensions), từ các ngôn ngữ phổ biến như JavaScript, Python, Java, C++, đến các ngôn ngữ chuyên biệt.
- **Tiện ích mở rộng phong phú:** Cửa hàng tiện ích mở rộng (Marketplace) của VS Code cung cấp hàng ngàn tiện ích mở rộng để tăng cường chức năng, từ hỗ trợ ngôn ngữ, công cụ gỡ lỗi, đến tích hợp với các dịch vụ đám mây và hệ thống quản lý mã nguồn.

## II.Phân tích và thiết kế ứng dụng

### 1.Phân tích hệ thống

**Phân tích hệ thống:** Ứng dụng được xây dựng để phục vụ hai đối tượng là khách hàng và quản trị viên

a, Khách hàng

+ **Đăng ký:** Khách hàng muốn mua hàng thì cần tạo tài khoản bằng gmail và mật khẩu.

+ **Đăng nhập:** Sau khi đăng ký thì khách hàng có thể xem và lựa chọn sản phẩm mình muốn mua.

+ **Xem các thông tin về sản phẩm:** Khách hàng lựa chọn xem thông tin mô tả chi tiết của sản phẩm để cân nhắc có mua hay không.

+ **Tìm kiếm sản phẩm theo yêu cầu:** Khách hàng tìm kiếm những sản phẩm mình muốn mua.

+ **Đặt mua sản phẩm:** Khách hàng bấm thêm sản phẩm vào giỏ hàng và vào giỏ hàng bấm thanh toán để thanh toán sản phẩm.

+Chỉnh sửa thông tin cá nhân: Khách hàng có thể sửa thông tin cá nhân của mình: tên, giới tính, ngày sinh, ...

+Hủy đơn hàng: Những đơn hàng nếu chưa được vận chuyển thì khách hàng có thể hủy đơn hàng.

+Xác nhận đã nhận hàng thành công: Khi nhận được hàng thì khách hàng có quyền xác nhận hoàn thành đơn.

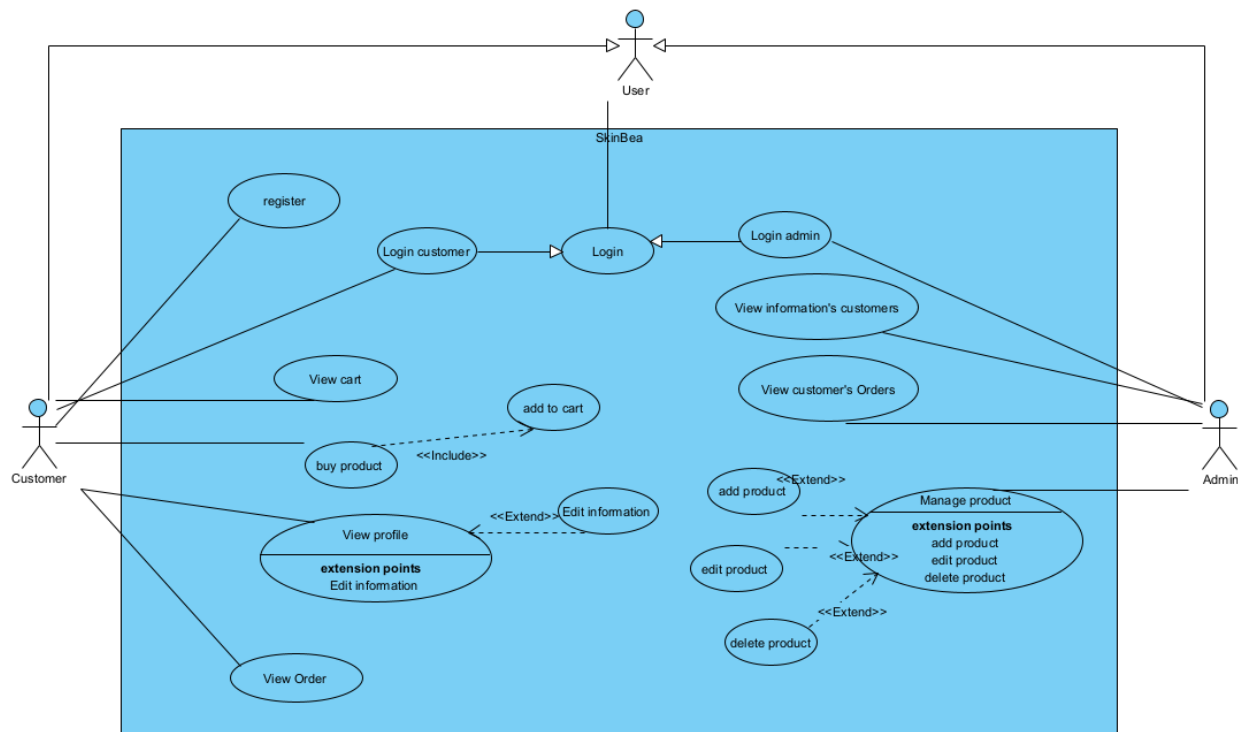
b, Quản trị viên

+Quản lý sản phẩm: Quản trị viên có quyền thêm sản phẩm để bán, có thể xóa hoặc sửa thông tin sản phẩm theo mong muốn.

+Quản lý đơn hàng: Quản trị viên có thể xem trạng thái của những đơn hàng được khách hàng đặt, sau khi đơn hàng đã được bàn giao cho đơn vị vận chuyển thì bấm xác nhận để chuyển đơn hàng sang trạng thái đang vận chuyển.

+Xem thông tin về khách hàng: Quản trị viên có quyền xem thông tin của những khách hàng của mình.

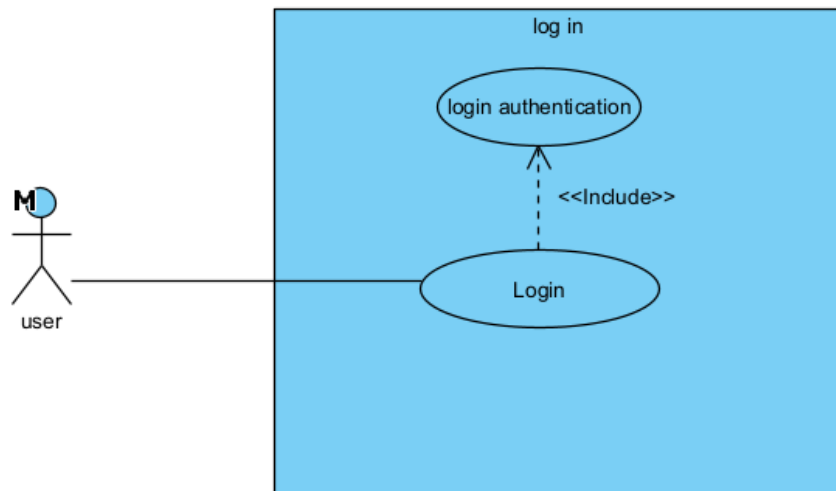
## 2. Biểu đồ UseCase toàn hệ thống



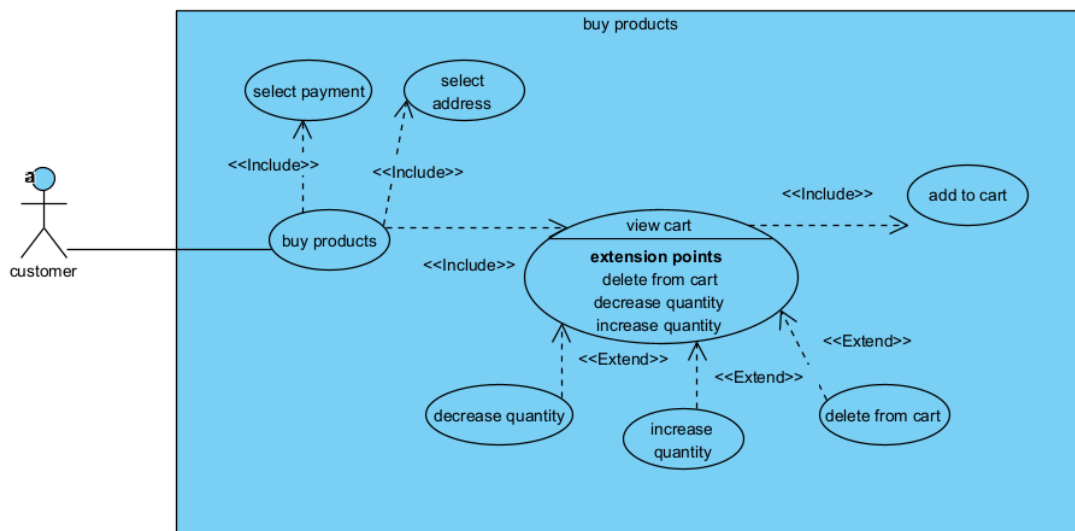
## 3. Biểu đồ UseCase cho những chức năng chính

a, Đăng nhập

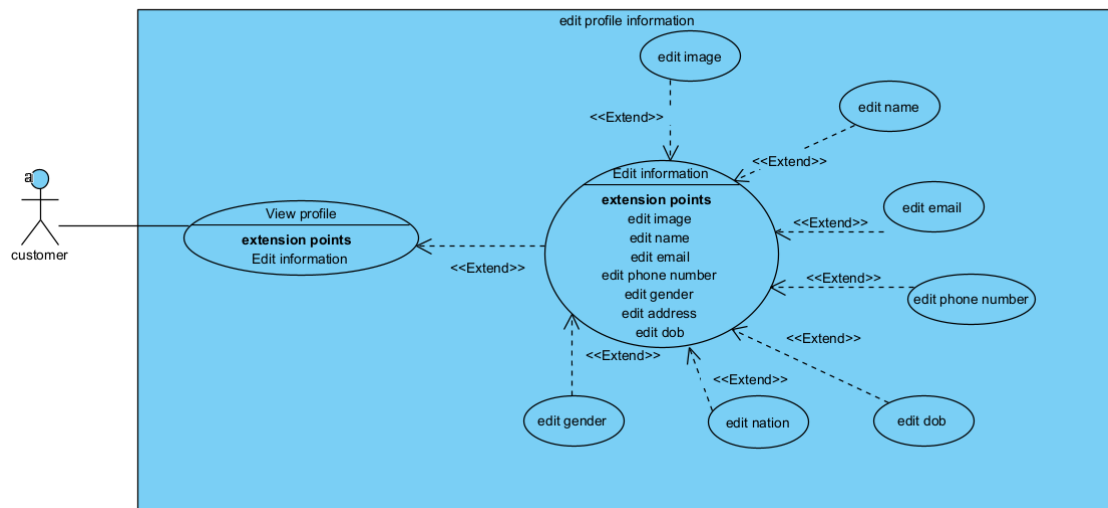
Biểu đồ UseCase chi tiết:



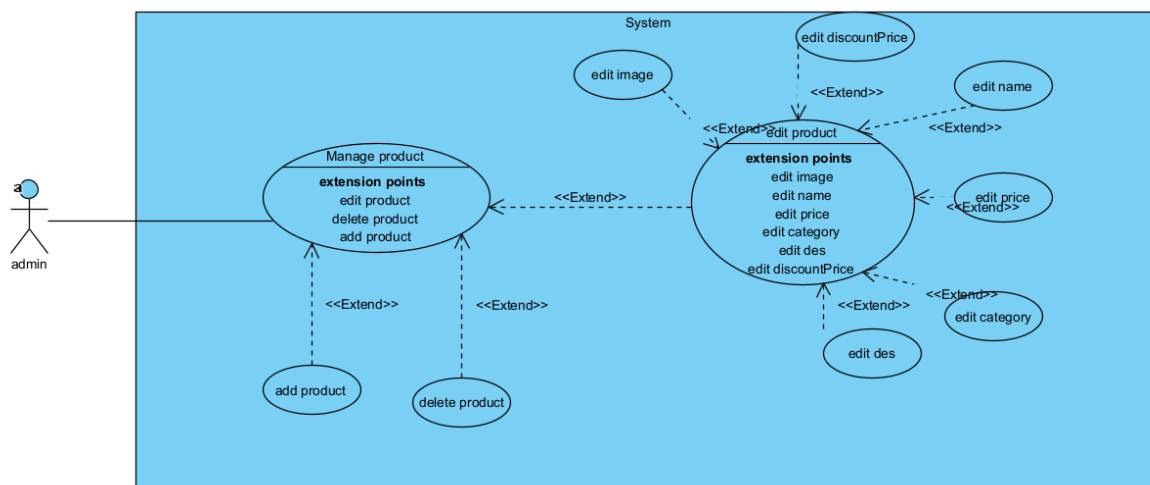
b, Mua hàng



c, Sửa thông tin cá nhân



#### d, Quản lý sản phẩm



### 4. Thiết kế cơ sở dữ liệu

Trích dẫn các lớp thực thể:

+ User (thông tin tài khoản người dùng)

Thuộc tính	Kiểu dữ liệu	Mô tả
userId	String	Mã người dùng
email	String	Email người dùng
dob	String	Ngày sinh

gender	String	Giới tính
name	String	Họ tên
nationality	String	Quốc gia
phone	String	Số điện thoại
position	String	Vai trò của người dùng(0: admin, 1:customer)
uri	String	Link hình ảnh
address	Array[]	Chứa các địa chỉ đã được thiết lập bởi người dùng.
cart	Array[product]	Chức các sản phẩm trong giỏ hàng
orders	Array[orders]	Thông tin các đơn hàng của khách hàng.

+Category(Thông tin danh mục).

Thuộc tính	Kiểu dữ liệu	Mô tả
id	String	Mã danh mục
image	String	Link ảnh tương ứng của danh mục
name	String	Tên của danh mục

+Products(thông tin về sản phẩm)

Thuộc tính	Kiểu dữ liệu	Mô tả
id	String	Mã sản phẩm
name	String	Tên sản phẩm
discountPrice	number	Giá khuyến mãi
price	number	Giá niêm yết

decription	String	Mô tả sản phẩm
image	String	Link ảnh sản phẩm
category	String	Tên danh mục sản phẩm
quality	number	Số lượng sản phẩm(mặc định là 0, cái này để tính số lượng sản phẩm trong giỏ hàng)

#### +Orders(Thông về đơn hàng)

Thuộc tính	Kiểu dữ liệu	Mô tả
id	String	Mã đơn hàng
address	String	Địa chỉ giao hàng
createdAt	String	Thời gian tạo đơn
items	Array[product]	Những sản phẩm kèm số lượng được đặt
orderBy	String	Email của người đặt
orderTotal	number	Tổng tiền của đơn hàng
paymentMethod	String	Phương thức thanh toán
userId	String	Mã người đặt hàng
userMobile	String	Số điện thoại người đặt
confirm	String	Chuỗi xác nhận xem đơn hàng đã được quản trị viên chấp nhận hay chưa.
finish	String	Chuỗi xác nhận xem đơn hàng đã được khách hàng nhận hay chưa.

Thiết kế cơ sở dữ liệu:

User
userId: String (requier)
email: String (unique)
dob: String
gender: String
name: String
nationality: String
phone: String
position: String
uri: String
address[{ addressId: String city: String mobile: String street: String }]
cart[product]
orders [orders]

Product
id: String (requier)
name: String
discountPrice: number
price: number
decription: String
image: String
category: String(categoryName)
quality: number


Category
id: String (requier)
name: String
image: String

Orders
id : String (requier)
address: String
createAt: String
items [product]
orderBy: String (userName)
orderTotal: number
paymentMethod: String
userId : String
userMobile: String
confirm: String
finish: String

### III.Cài đặt ứng dụng

#### 1. Trang đăng ký, Đăng nhập

00:51



**Register**


Username:

Password:

Accuracy:

You have an account? [Log In](#)

00:51



**Log In**

Username:

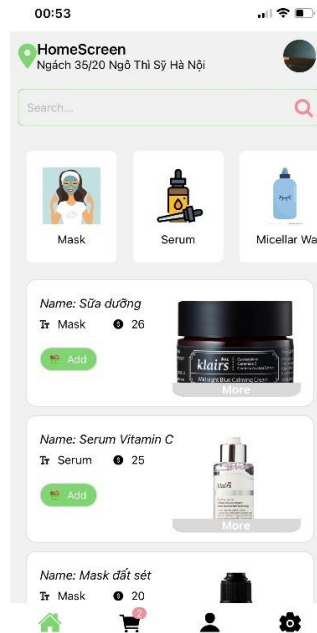
Password:

You don't have an account? [Register](#)



## 2. Trang chủ

Sau khi đăng nhập thì trang chủ Home hiện ra: với thanh tìm kiếm, những danh mục sản phẩm, và sản phẩm. Bên dưới còn có một thanh Bottom cho phép chuyển đổi giữa các trang với nhau một cách dễ dàng. Chúng ta có thể nhấn Add để thêm sản phẩm vào giỏ hàng.



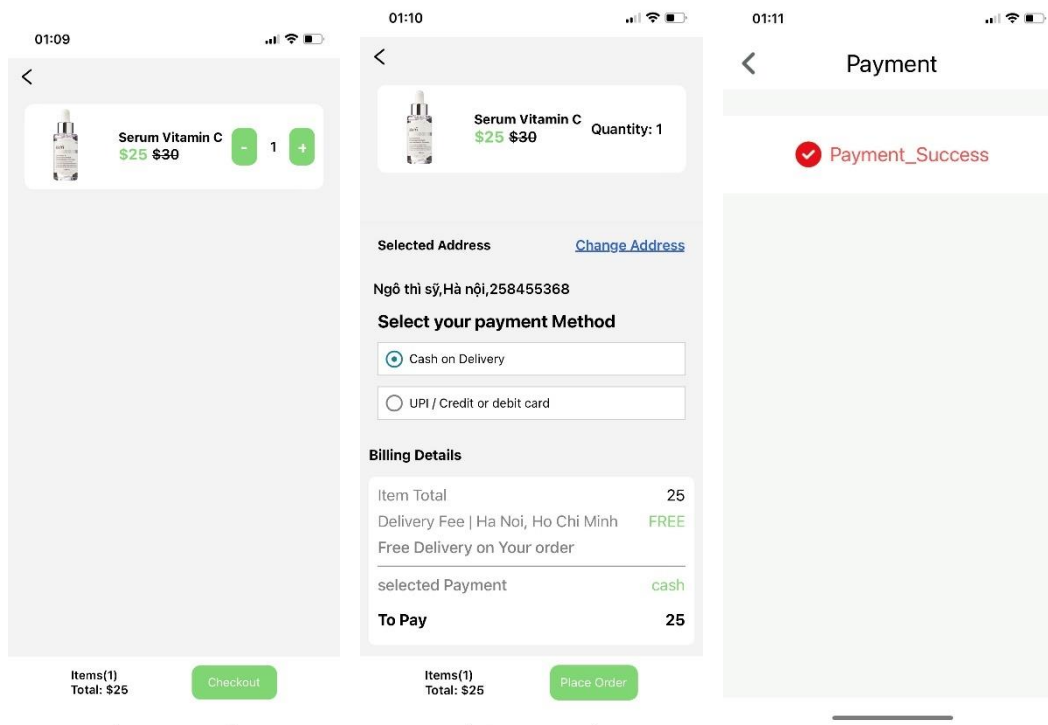
## 3. Trang chi tiết sản phẩm

Khi bấm vào chữ “more” ở dưới ảnh sản phẩm thì ta sẽ đi vào trang xem chi tiết sản phẩm như sau: chúng ta có thể nhấn vào giỏ hàng với dấu cộng phía dưới để thêm sản phẩm vào giỏ hàng.



#### 4. Giỏ hàng và thanh toán

Khi bấm vào biểu tượng giỏ hàng kèm số lượng sản phẩm (hình trên ở góc trên bên phải) hoặc trên thanh Bottom phía dưới sẽ đi đến trang giỏ hàng: có thể tăng giảm số lượng sản phẩm. Sau đó nhấn “check out” để đến trang thanh toán.



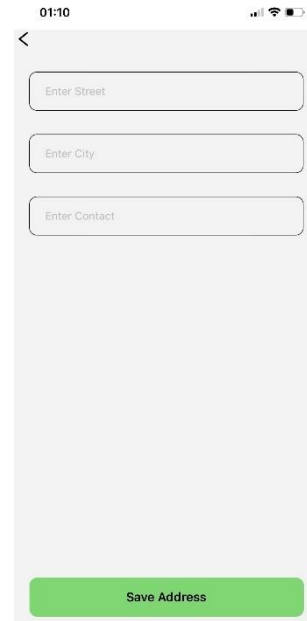
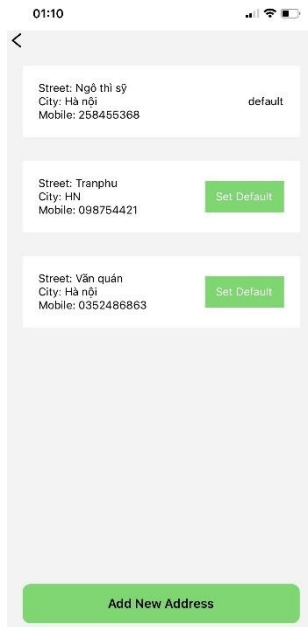
Trang giỏ hàng

Trang thanh toán

Trang thanh toán thành công

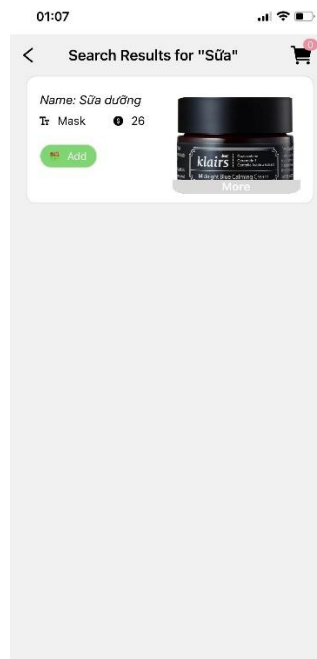
## 5. Trang thêm địa chỉ

Trên trang thanh toán, ta có thể chọn hoặc thêm địa chỉ giao hàng theo ý muốn:

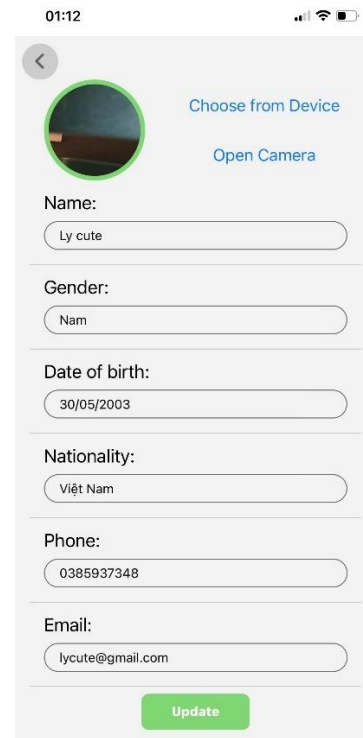
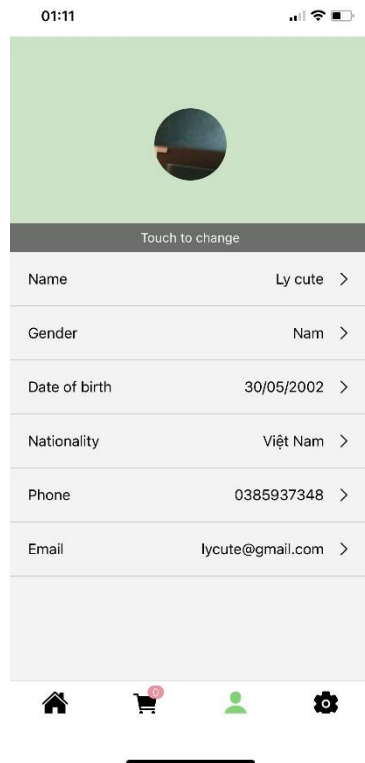


## 6. Trang tìm kiếm sản phẩm

Khi bấm tìm kiếm ở trang chủ thì sẽ đi đến trang tìm kiếm sản phẩm:

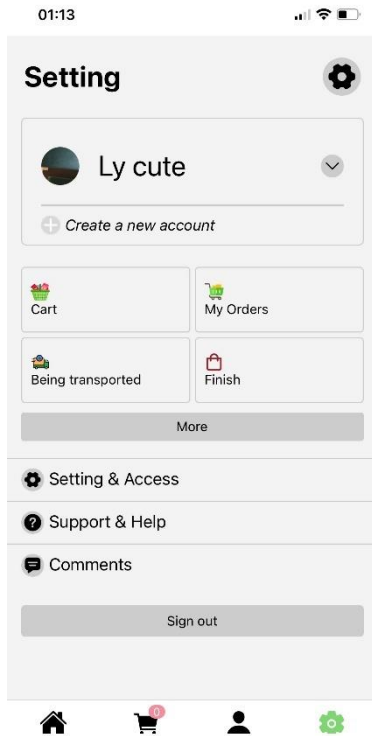


## 7. Trang User và sửa thông tin user

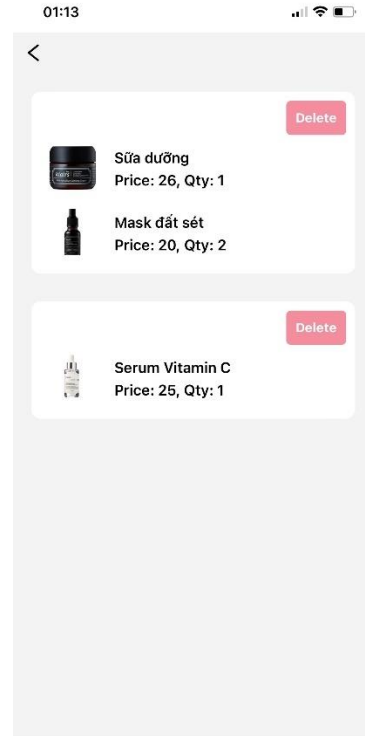


## 8. Trang Setting

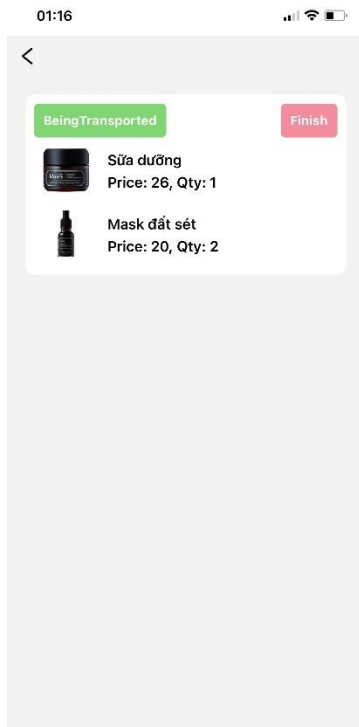
Trang cài đặt, bạn có thể có những tùy chọn như xem giỏ hàng, xem đơn hàng đã đặt, xem đơn hàng đang trong quá trình vận chuyển, xem đơn hàng đã hoàn thành, đăng xuất.



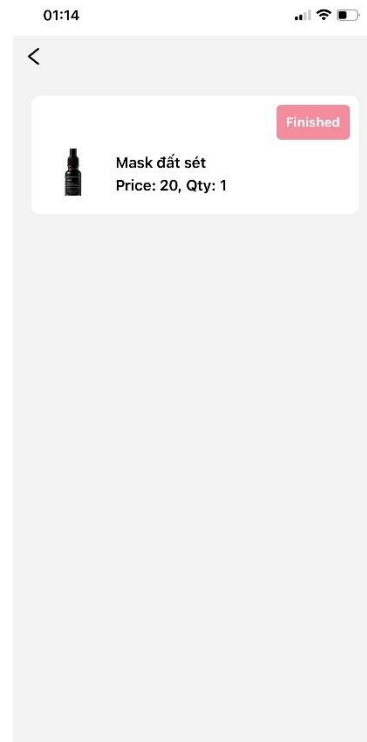
Trang setting



Trang My Orders



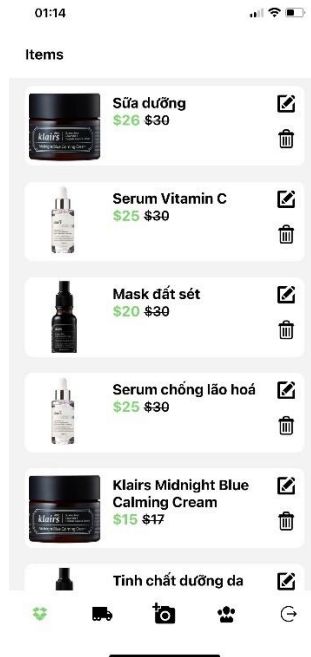
Trang Being Transported



Trang Finish

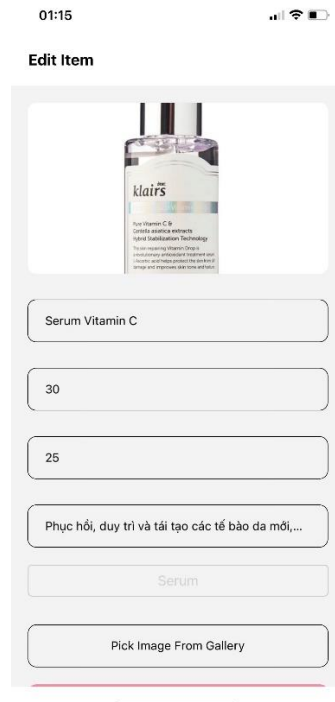
## 9. Trang chủ của Admin

Mỗi sản phẩm sẽ hiện thông tin trên một dòng, khi muốn xóa sản phẩm thì nhấn biểu tượng thùng rác, còn khi muốn sửa thì nhấn biểu tượng sửa, màn hình sẽ chuyển đến trang sửa sản phẩm tương ứng.



## 10. Trang sửa sản phẩm:

Sau khi sửa xong nhập update mà hình sẽ quay lại trang chủ.



## 11. Trang thêm sản phẩm

Nhập các trường tương ứng cho sản phẩm, số đó nhấn upload, màn hình sẽ trở về trang chủ.

01:14

**Add Item**

Pick Image From Gallery

Enter Item Name

Enter Item Price

Enter Item Discount Price

Enter Item Description

Select a category

Upload Item

Icons: Home, Cart, Add Item, Profile, Search

## 12. Quản lý đơn hàng

Quản trị viên có quyền chuyển đơn hàng từ trạng thái Confirm sang Transport.

01:14

**Orders**

OrderBy: lycute@gmail.com **Finished**

Mask đất sét  
Price: 20, Qty: 1

OrderBy: lycute@gmail.com **Transport**

Sữa dưỡng  
Price: 26, Qty: undefined

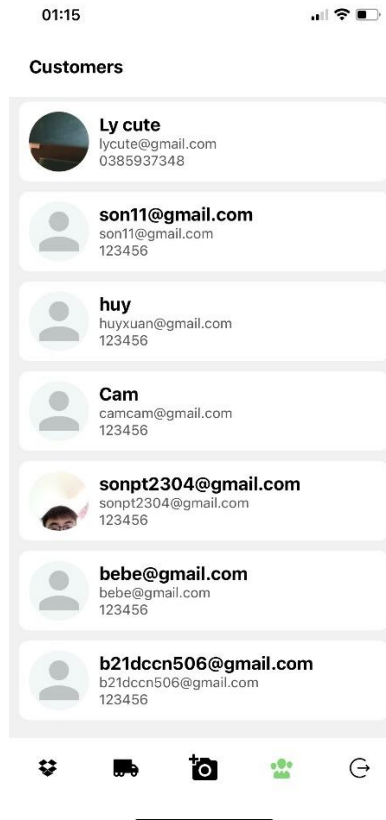
Mask đất sét  
Price: 20, Qty: undefined

OrderBy: lycute@gmail.com **Confirm**

Serum Vitamin C  
Price: 25, Qty: undefined

Icons: Home, Cart, Add Item, Profile, Search

## 13. Xem thông tin khách hàng



#### IV. Kết quả đạt được và hướng phát triển

- Mục tiêu: Xây dựng một ứng dụng bán hàng mỹ phẩm cho phép khách hàng có thể dễ dàng lựa chọn mua sắm với các chức năng chính như mua hàng, xem giỏ hàng, xem thông tin sản phẩm.
- Kết quả đạt được:
  - + Tạo được giao diện ứng dụng
  - + Hoàn thiện những chức năng đã được đề ra ở phần II.
  - + Tuy nhiên còn một số hạn chế như: tốc độ truy cập chậm, giao diện ứng dụng chưa thực sự bắt mắt thân thiện với người dùng, chưa xử lý được ngoại lệ khi nhập vào dữ liệu.
- Hướng phát triển:
  - Hoàn thiện giao diện thân thiện với người dùng hơn.
  - Thêm những tính năng như:
    - Bình luận về sản phẩm đã mua
    - Xem chi tiết đơn hàng
    - Thêm liên kết thẻ ngân hàng (các ứng dụng thanh toán trực tuyến),
    - Chức năng thêm sản phẩm vào danh sách yêu thích



- Liên kết với tài khoản Google, Facebook,...
- Phần cài đặt ngôn ngữ
- Thử nghiệm trên các thiết bị khác nhau để đảm bảo sự tương thích.
- Thu thập phản hồi từ người dùng và cải thiện ứng dụng.

**Kết luận:** Dự án ứng dụng bán hàng mỹ phẩm này là một trải nghiệm học tập và rèn luyện bổ ích cho bản thân. Qua dự án, bản thân đã học được nhiều kiến thức và kỹ năng về lập trình, quản lý dự án và làm việc nhóm. Ứng dụng có tiềm năng trở thành một công cụ hữu ích cho người dùng trong việc mua sắm online.