

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA CÔNG NGHỆ PHẦN MỀM



LẬP TRÌNH TRÊN THIẾT BỊ DI ĐỘNG

ĐỀ TÀI: “XÂY DỰNG ỨNG DỤNG SHOPPING”

Sinh viên thực hiện:

- | | |
|------------------------|----------|
| 1. Nguyễn Hoàng Tuấn | 17521217 |
| 2. Trần Diệu Bảo Trân | 17521150 |
| 3. Nguyễn Du Phúc Ngân | 17520792 |

Tp. Hồ Chí Minh, ngày 02 tháng 07 năm 2021

Mục Lục

1. __ Giới thiệu	4
1.1. __ Đặt vấn đề	4
1.2. __ Mục đích của đề tài	4
2. __ Công nghệ sử dụng	5
2.1. __ JavaScript	5
2.2. __ Firebase	5
2.3. __ GetX	6
2.4. __ Flutter	10
2.5. __ MVVM	10
2.6. __ SQflite	12
3. __ Phân tích	14
3.1. __ Sơ đồ Usecase tổng quát	14
3.2. __ Đặc tả Usecase	19
4. __ Thiết kế	35
4.1. __ Thiết kế dữ liệu	35
4.3. __ Thiết kế giao diện	44
5. __ Triển khai ứng dụng	64
6. __ Bảng phân công công việc	65
7. __ Nhận xét và kết luận	66
7.1. __ Nhận xét	66
7.2. __ Hướng mở rộng	67
8. __ Tài liệu tham khảo	68

LỜI CẢM ƠN

Trong quá trình học tập, nghiên cứu và phát triển đề tài “*Xây dựng ứng dụng Shopping*” chúng em đã nhận được sự giúp đỡ, chỉ bảo nhiệt tình của thầy để hoàn thành đồ án này.

Chúng em xin bày tỏ sự biết ơn đặc biệt đến Thầy Huỳnh Tuấn Anh người đã trực tiếp hướng dẫn, giúp đỡ về kiến thức, tài liệu và phương pháp để chúng em hoàn thành đề tài này.

Chúng em xin chân thành cảm ơn vì những thời gian quý báu thầy đã dành ra để kiểm tra, đưa ra các ý kiến nhận xét và đóng góp rất nhiều vào những tính năng, hướng phát triển mới hơn cho đề tài.

Hi vọng với sự chỉ bảo và góp ý từ thầy, nhóm đề tài có thể tiếp tục phát triển để ứng dụng ngày càng được mở rộng và phục vụ được tốt hơn cho người dùng.

Mặc dù đã có nhiều cố gắng trong suốt quá trình thực hiện đề tài, song có thể còn có những mặt hạn chế, thiếu sót. Chúng em rất mong nhận được ý kiến đóng góp và sự chỉ dẫn của thầy và các bạn để đề tài của chúng em được hoàn thiện hơn.

Chúng em xin trân trọng cảm ơn!

TP. Hồ Chí Minh, ngày 02 tháng 07 năm 2021

1. Giới thiệu

1.1. Đặt vấn đề

- Ngày nay cùng với sự phát triển của internet, nhu cầu của con người tăng cao, việc tiết kiệm thời gian trong mua bán hàng hóa trở nên quan trọng, có thể mua hàng hóa mà không cần đến nơi sẽ rất tiện cho người tiêu dùng.
- Từ những vấn đề trên, nhóm thực hiện đề tài mong muốn xây dựng một ứng dụng e-commerce cung cấp một nơi mà mọi người có thể thỏa sức mua hàng hóa mà còn tiết kiệm rất nhiều thời gian. Ngoài ra nhóm muốn cung cấp một ứng dụng hoàn toàn miễn phí và dễ dàng tiếp cận cho mọi người.
- Từ những lý do trên, nhóm thực hiện đưa đến quyết định sẽ thực hiện:
“Ứng dụng Shopping”

1.2. Mục đích của đề tài

- Với bước khởi đầu là môn Lập trình trên thiết bị di động, nhóm thực hiện đề tài mong muốn sẽ nghiên cứu tìm ra các công nghệ có thể có thể áp dụng vào đề tài này.
- Từ các công nghệ đã tìm hiểu, nhóm sẽ chọn ra các công nghệ phù hợp, có thể mở rộng và nâng cấp thêm để ứng dụng sẽ hoàn thiện theo hướng mong muốn của nhóm.
- Trong phạm vi môn học, nhóm đặt ra các chức năng phải thực hiện gồm:
 - Xây dựng chức năng đăng nhập, đăng kí và quản lý thông tin người dùng.
 - Xây dựng chức năng liệt kê sản phẩm theo thể loại, công dụng
 - Xây dựng chức năng đặt hàng online

2. Công nghệ sử dụng

2.1. JavaScript

Giới thiệu:

- JavaScript theo phiên bản hiện tại là một ngôn ngữ thông dịch được phát triển bởi Brendan Eich tại hãng truyền thông Netscape. JavaScript xuất hiện lần đầu và tháng 5 năm 1995 với tên gọi ban đầu là Mocha, sau đó đổi tên thành LiveScript và cuối cùng là JavaScript.
- Phiên bản mới nhất của JavaScript là ECMAScript 7. ECMAScript là phiên bản chuẩn hóa của JavaScript, là quy chuẩn để sử dụng JavaScript tốt nhất cho các lập trình viên. Phiên bản ECMAScript 6 hiện đang là quy chuẩn sử dụng rộng rãi nhất đối với các lập trình viên sử dụng JavaScript dành cho phát triển sản phẩm và framework.

Lí do sử dụng trong đề án:

- JavaScript hiện tại là ngôn ngữ phổ biến đối với việc phát triển ứng dụng web, ứng dụng di động, hoặc làm việc với server, thông qua việc sử dụng các thư viện, framework mã nguồn mở.
- Vì JavaScript có thể sử dụng ở cả việc phát triển ứng dụng ở phía khách hàng và sử dụng để làm việc với server nên nhóm đề án có thể sử dụng chung một ngôn ngữ để cùng phát triển dự án và hỗ trợ nhau tốt hơn.

- JavaScript được dùng để phát triển rất nhiều thư viện, framework mã nguồn mở, vì vậy cộng đồng sử dụng JavaScript cũng rất lớn và nhóm thực đồ án cũng dễ dàng hơn trong việc phát triển đồ án.

2.2. Firebase

Giới thiệu:

- 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.
- 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. Bạn có thể tìm hiểu thêm về điện toán đám mây trong bài viết:
- Ưu điểm của Firebase:
 - + Tạo tài khoản và sử dụng dễ dàng
 - + Tốc độ phát triển nhanh
 - + Nhiều dịch vụ trong một nền tảng
 - + Được cung cấp bởi Google
 - + Tập trung vào phát triển giao diện người dùng
 - + Firebase không có máy chủ
 - + Học máy (Machine Learning)
 - + Tạo lưu lượng truy cập

- + Theo dõi lỗi
- + Sao lưu
- Hạn chế của Firebase
 - + Không phải là mã nguồn mở
 - + Người dùng không có quyền truy cập mã nguồn
 - + Firebase không hoạt động ở nhiều quốc gia
 - + Chỉ hoạt động với Cơ sở dữ liệu NoSQL
 - + Truy vấn chậm
 - + Không phải tất cả các dịch vụ Firebase đều miễn phí
 - + Firebase khá đắt và giá không ổn định
 - + Chỉ chạy trên Google Cloud
 - + Thiếu Dedicated Servers và hợp đồng doanh nghiệp
 - + Không cung cấp các API GraphQL

2.3. GetX

Giới thiệu:

- GetX trong Flutter là một thư viện rất tiện ích và được dùng rất nhiều trong lập trình Flutter bởi nhiều tính năng ưu việt, dễ sử dụng và nó trở thành trending trong lập trình Flutter hiện nay.
- GetX trong lập trình Flutter có rất nhiều tính năng tiện ích, dễ sử dụng. Trong đó có tính năng quản lý state- State Management.
- GetX có thể làm những việc như:
 - + State Manager : Quản lý state trong Flutter
 - + Navigation Manager: Quản lý việc điều hướng
 - + Dependencies Manager: Cung cấp giải pháp dependencies injection tuyệt vời
 - + Utils function: Các hàm tiện ích cực kỳ hữu ích trong lập trình Flutter

2.4. Flutter

Giới thiệu:

- Flutter là mobile UI framework của Google để tạo ra các giao diện chất lượng cao trên iOS và Android trong khoảng thời gian ngắn. Flutter hoạt động với những code sẵn có được sử dụng bởi các lập trình viên, các tổ chức.
- Flutter hoàn toàn miễn phí và cũng là mã nguồn mở.
- Mục tiêu cốt lõi của thư viện này là nhằm cung cấp hiệu suất làm việc cao nhất vì cần tìm hiểu 2 hệ sinh thái hoàn toàn khác biệt (iOS ,Android), cho phép các dev sử dụng JavaScript để làm mobile apps trên cả Android và iOS với cảm nhận và giao diện đơn giản.

Lí do sử dụng trong đồ án:

- Flutter sử dụng Dart, một ngôn ngữ nhanh, hướng đối tượng với nhiều tính năng hữu ích như mixin, generic, isolate, và static type.
- Flutter có các thành phần UI của riêng nó, cùng với một cơ chế để kết xuất chúng trên nền tảng Android và iOS. Hầu hết các thành phần giao diện người dùng, đều sẵn dùng, phù hợp với các nguyên tắc của Material Design.
- Các ứng dụng Flutter có thể được phát triển bằng cách sử dụng IntelliJ IDEA, một IDE rất giống với Android Studio.

2.5. MVVM

Giới thiệu:

- MVVM (Model – View – ViewModel) là mô hình hỗ trợ two-way data binding (ràng buộc dữ liệu) giữa View và ViewModel.
- Mô hình này bao gồm các thành phần chính sau:
 - + Model: Giống như mô hình MVP và MVC, Model trong MVVM là bộ phận đại diện cho các tầng business logic, định nghĩa business rules cho dữ liệu.
 - + View: Đây là nơi cho phép người dùng tương tác, đưa ra yêu cầu.
 - + View Model: Đây là lớp trung gian giữa View và Model và xem như thành phần thay thế cho Controller trong mô hình MVC.
- Ở mô hình MVVM, ViewModel là nơi thực hiện hầu hết tất cả các business logic, quản lý methods, properties khác để duy trì viewstate. Điều này đồng nghĩa với việc View trở thành một ảnh xạ hiển thị trạng thái của ViewModel tại mọi thời điểm.
- Ưu điểm của MVVM: người dùng có thể thực hiện unit testing dễ dàng do không phải phụ thuộc vào View. Khi test, bạn không cần phải tạo mockup như mô hình MVP mà chỉ cần xác nhận biến observable thích hợp.
- Nhược điểm của MVVM: có nhược điểm về khả năng duy trì. Khi gán biến và biểu thức vào View, các logic rải rác sẽ tăng dần theo thời gian. Điều này khiến việc kiểm soát và thêm code dần trở nên khó khăn hơn.

2.6. SQflite

Giới thiệu:

- SQflite là 1 ứng dụng của SQLite cho flutter. Nó cho phép bạn hoàn toàn kiểm soát database, queries, relationships và bất kể những gì bạn có thể nghĩ tới.
- Ưu điểm:
 - + Kiểm soát toàn bộ dữ liệu.
 - + Cơ sở dữ liệu SQLite rất hiệu quả cho ứng dụng của bạn (hãy đảm bảo bạn đã có kiến thức về SQLite từ trước đó)
 - + Dễ dàng sử dụng ứng dụng của bên thứ ba để đọc và mở nội dung của SQLite (vì vậy bạn có thể mở cơ sở dữ liệu trên máy tính khác và xem được nội dung bên trong như thế nào)
- Nhược điểm:
 - + Mất nhiều thời gian vì tất cả các truy vấn đều phải viết bằng tay.
 - + Dữ liệu trả về từ Database không được nhập tốt ngay từ ban đầu.
 - + Khó để xử lý việc thay đổi trên thiết bị (ví dụ: khi thay đổi dữ liệu giữa các bản cập nhật phiên bản)
 - + Không có hỗ trợ cho web.

3. Phân tích

3.1. Sơ đồ Usecase tổng quát



3.2. Đặc tả Usecase

3.2.1. Usecase đăng nhập

Tiêu đề	Nội dung
Tên usecase	Đăng nhập
Mô tả	Usecase cho phép người dùng đăng nhập vào ứng dụng để thao tác các chức năng
Sự kiện kích hoạt	Người dùng truy cập vào ứng dụng
Tiền điều kiện	Người dùng phải có tài khoản
Hậu điều kiện	Người dùng đăng nhập thành công vào ứng dụng
Luồng sự kiện chính	<ol style="list-style-type: none">1. Đăng nhập<ol style="list-style-type: none">1.1. Đăng nhập với tài khoản được tạo trên ứng dụng<ul style="list-style-type: none">• Ứng dụng yêu cầu người dùng nhập tên đăng nhập và mật khẩu.• Người dùng nhập tên đăng nhập và mật khẩu.• Hệ thống kiểm tra thông tin đăng nhập.• Đăng nhập thành công và chuyển người dùng vào màn hình sử dụng ứng dụng.• Kết thúc usecase.1.2. Đăng nhập với tài khoản Google<ul style="list-style-type: none">• Người dùng chọn Đăng nhập với Google• Ứng dụng yêu cầu người dùng nhập tài khoản Google.• Hệ thống kiểm tra tài khoản Google có hiệu lực.• Đăng nhập thành công và chuyển người dùng vào .màn hình sử dụng ứng dụng.

	<ul style="list-style-type: none"> • Kết thúc usecase. <p>1.3. Đăng nhập với tài khoản Facebook</p> <ul style="list-style-type: none"> • Người dùng chọn Đăng nhập với Google. • Ứng dụng yêu cầu người dùng nhập tài khoản Google. • Hệ thống kiểm tra tài khoản Google có hiệu lực. • Đăng nhập thành công và chuyển người dùng vào màn hình sử dụng ứng dụng. • Kết thúc usecase. <p>2. Tạo tài khoản</p> <ul style="list-style-type: none"> • Người dùng chọn Đăng ký. • Hệ thống yêu cầu người dùng nhập tên, email và mật khẩu. • Người dùng cung cấp thông tin cần thiết. • Hệ thống cập nhật thông tin vào database. • Tạo thành công và cho phép người dùng đăng nhập với tài khoản này. • Kết thúc usecase. <p>3. Lấy lại mật khẩu</p> <ul style="list-style-type: none"> • Người dùng chọn Quên mật khẩu. • Hệ thống yêu cầu người dùng email. • Người dùng cung cấp thông tin cần thiết. • Hệ thống kiểm tra cập email có tồn tại trong database. • Ứng dụng yêu cầu người dùng nhập mật khẩu mới. • Cập nhật thành công và cho phép người dùng đăng nhập với mật khẩu này. • Kết thúc usecase.
--	--

	<p>4. Đăng xuất</p> <ul style="list-style-type: none"> • Người dùng chọn đăng xuất • Hệ thống xóa bỏ các token và dữ liệu lưu tại máy người dùng. • Hệ thống thông báo đăng xuất thành công và chuyển Acotr ra màn hình đăng nhập
Luồng sự kiện phụ	<p>Khi người dùng nhập sai tên đăng nhập hoặc mật khẩu:</p> <ul style="list-style-type: none"> • Hệ thống thông báo tên đăng nhập hoặc mật khẩu sai • Yêu cầu người dùng đăng nhập lại

3.2.2. Usecase quản lý thông tin

Tiêu đề	Nội dung
Tên usecase	Cập nhật thông tin
Mô tả	Usecase cho phép người dùng cập nhật và lưu lại thông tin
Sự kiện kích hoạt	Người dùng chọn Cập nhật thông tin
Tiền điều kiện	Người dùng đăng nhập
Hậu điều kiện	Cập nhật thông tin thành công
Luồng sự kiện chính	<ul style="list-style-type: none">• Người dùng cập nhật các thông tin cần thiết• Hệ thống cập nhật thông tin người dùng vào database• Hệ thống hiển thị thông tin vừa cập nhật• Kết thúc usecase
Luồng sự kiện phụ	<ul style="list-style-type: none">• Người dùng nhập thiếu thông tin cần thiết như tên, email.• Hệ thống thông báo yêu cầu người dùng nhập đầy đủ thông tin cần thiết.

3.2.3. Usecase tìm kiếm sản phẩm

Tiêu đề	Nội dung
Tên usecase	Tìm kiếm
Mô tả	Usecase cho phép người dùng tìm kiếm thông tin sản phẩm theo từ khóa
Sự kiện kích hoạt	Người dùng chọn tìm kiếm
Tiền điều kiện	Người dùng đăng nhập
Hậu điều kiện	Hiển thị thông tin liên quan tới từ khóa
Luồng sự kiện chính	<ul style="list-style-type: none">• Người dùng nhập keyword vào ô tìm kiếm.• Hệ thống kiểm tra database và xuất ra thông tin cần tìm.• Người dùng chọn sản phẩm muốn tìm kiếm và xem thông tin sản phẩm đó.• Usecase kết thúc.
Luồng sự kiện phụ	<ul style="list-style-type: none">• Người dùng nhập keyword không liên quan tới sản phẩm trong database.• Hệ thống sẽ hiện thông báo “Không tìm thấy sản phẩm”.

3.2.4. Usecase thêm sản phẩm vào giỏ hàng

Tiêu đề	Nội dung
Tên usecase	Thêm sản phẩm vào giỏ hàng
Mô tả	Usecase cho phép người dùng chọn sản phẩm họ muốn mua và đưa vào giỏ hàng để thanh toán
Sự kiện kích hoạt	Người dùng chọn Thêm sản phẩm vào giỏ hàng
Tiền điều kiện	Người dùng đăng nhập, người dùng chọn xem thông tin sản phẩm
Hậu điều kiện	Thêm sản phẩm vào giỏ hàng thành công
Luồng sự kiện chính	Người dùng chọn sản phẩm mình muốn mua trong danh sách sản phẩm. Hệ thống cập nhật sản phẩm vào giỏ hàng. Usecase kết thúc.
Luồng sự kiện phụ	N/A

3.2.5. Usecase cập nhật giỏ hàng

Tiêu đề	Nội dung
Tên usecase	Cập nhật giỏ hàng
Mô tả	Usecase cho phép người dùng cập nhật số lượng sản phẩm đã chọn mua hoặc xóa sản phẩm khỏi giỏ hàng
Sự kiện kích hoạt	Người dùng chọn Xem giỏ hàng
Tiền điều kiện	Người dùng đăng nhập
Hậu điều kiện	Cập nhật giỏ hàng thành công
Luồng sự kiện chính	<ul style="list-style-type: none">• Người dùng tăng giảm số lượng sản phẩm muốn mua• Hệ thống xác cập nhật số lượng.• Usecase kết thúc.
Luồng sự kiện phụ	Nếu như người dùng trước đó không có sản phẩm trong giỏ hàng, <ul style="list-style-type: none">• Ứng dụng hiển thị Không có sản phẩm nào trong giỏ hàng.

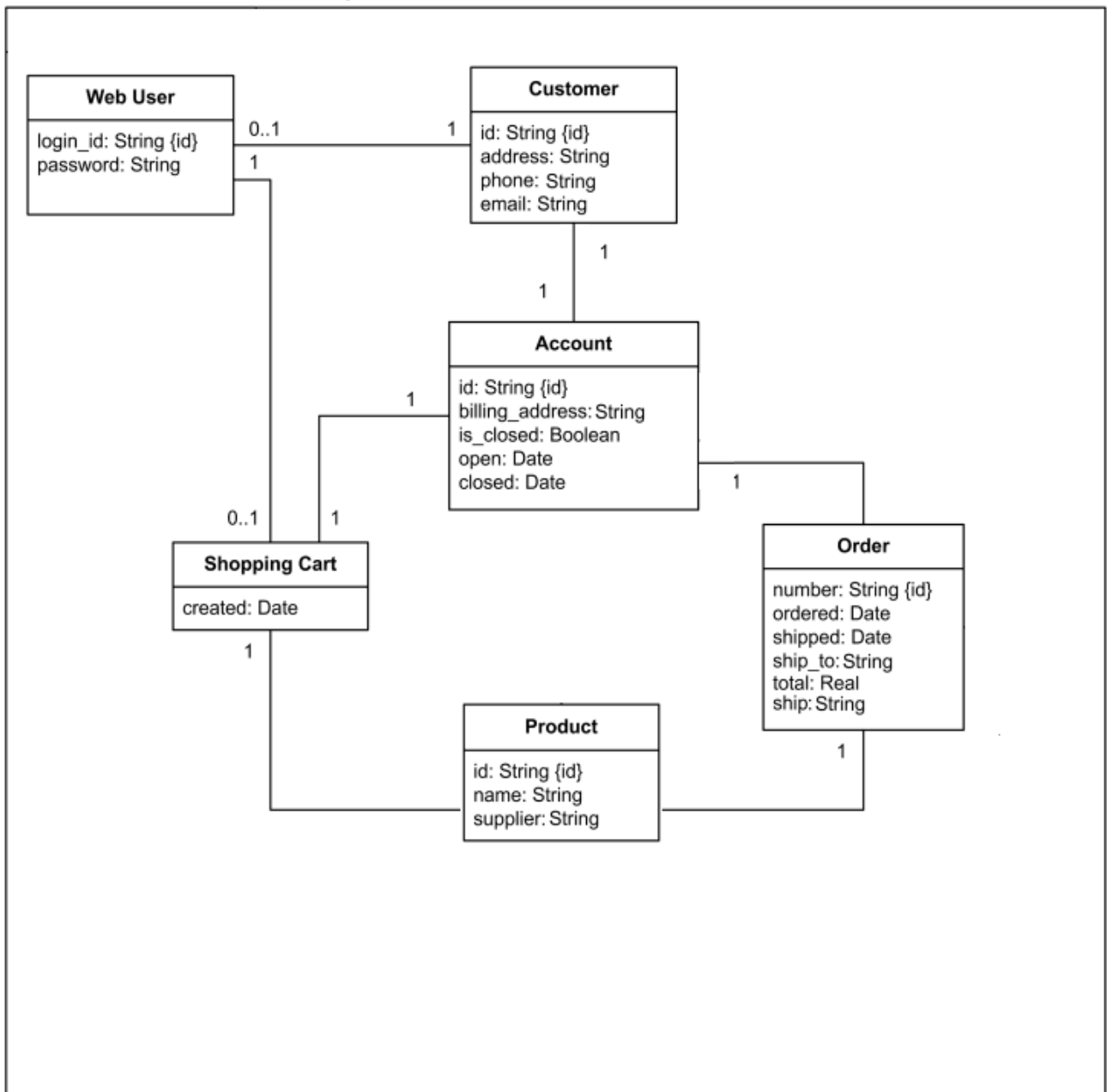
3.2.6. Usecase thanh toán

Tiêu đề	Nội dung
Tên usecase	Thanh toán
Mô tả	Usecase cho phép người dùng thanh toán
Sự kiện kích hoạt	Người dùng chọn Thanh toán
Tiền điều kiện	Người dùng thêm sản phẩm vào giỏ hàng
Hậu điều kiện	Xác nhận đơn hàng thành công, tiến hành giao hàng
Luồng sự kiện chính	<ul style="list-style-type: none">• Người dùng chọn mua sản phẩm• Ứng dụng hiển thị màn hình yêu cầu người dùng nhập địa chỉ giao hàng.• Người dùng nhập đầy đủ thông tin cần thiết.• Hệ thống kiểm tra đầy đủ thông tin.• Hoàn thành đơn hàng.• Usecase kết thúc.
Luồng sự kiện phụ	<ul style="list-style-type: none">• Người dùng nhập thiếu thông tin địa chỉ• Hệ thống thông báo yêu cầu người dùng nhập đầy đủ thông tin.

4. Thiết kế

4.1. Thiết kế dữ liệu

4.1.1. Cơ sở dữ liệu tổng quan



4.1.2. Chi tiết thông tin lưu trữ tại cơ sở dữ liệu

a. Customer

Customer
id: String {id} address: String phone: String email: String

Tên trường	Kiểu dữ liệu	Ý nghĩa	Ghi chú
Id	String	Id của người dùng	Id là duy nhất
address	String	Địa chỉ của người dùng	Địa chỉ của người dùng
phone	String	Số điện thoại của người dùng	
email	String	Email của người dùng	Email là duy nhất

b. Web User

Web User
login_id: String {id} password: String

Tên trường	Kiểu dữ liệu	Ý nghĩa	Ghi chú
Login_Id	String	Id login thông qua (facebook)	Id là duy nhất
password	String	Password của account	

c. Account

Account
id: String {id} billing_address: String is_closed: Boolean open: Date closed: Date

Tên trường	Kiểu dữ liệu	Ý nghĩa	Ghi chú
Id	String	Id của account	Id là duy nhất
Billing_address	String	Địa chỉ thanh toán	
phone	String	Số điện thoại của người dùng	

Is_closed	Boolean	Xác định trạng thái của account	Mở/Đóng
open	Date	Ngày tạo account	
closed	Date	Ngày hủy account	

d. Order

Order
number: String {id} ordered: Date shipped: Date ship_to: String total: Real ship: String

Tên trường	Kiểu dữ liệu	Ý nghĩa	Ghi chú
number	String	Số của đơn đặt	Số là duy nhất
ordered	Date	Ngày đặt hàng	
shipped	Date	Ngày giao hàng	
Shipped_to	String	Địa chỉ giao hàng của khách hàng	
total	Real	Tổng số tiền	

ship	String	Đơn vị giao hàng	
------	--------	------------------	--

e. Product

Product
id: String {id} name: String supplier: String

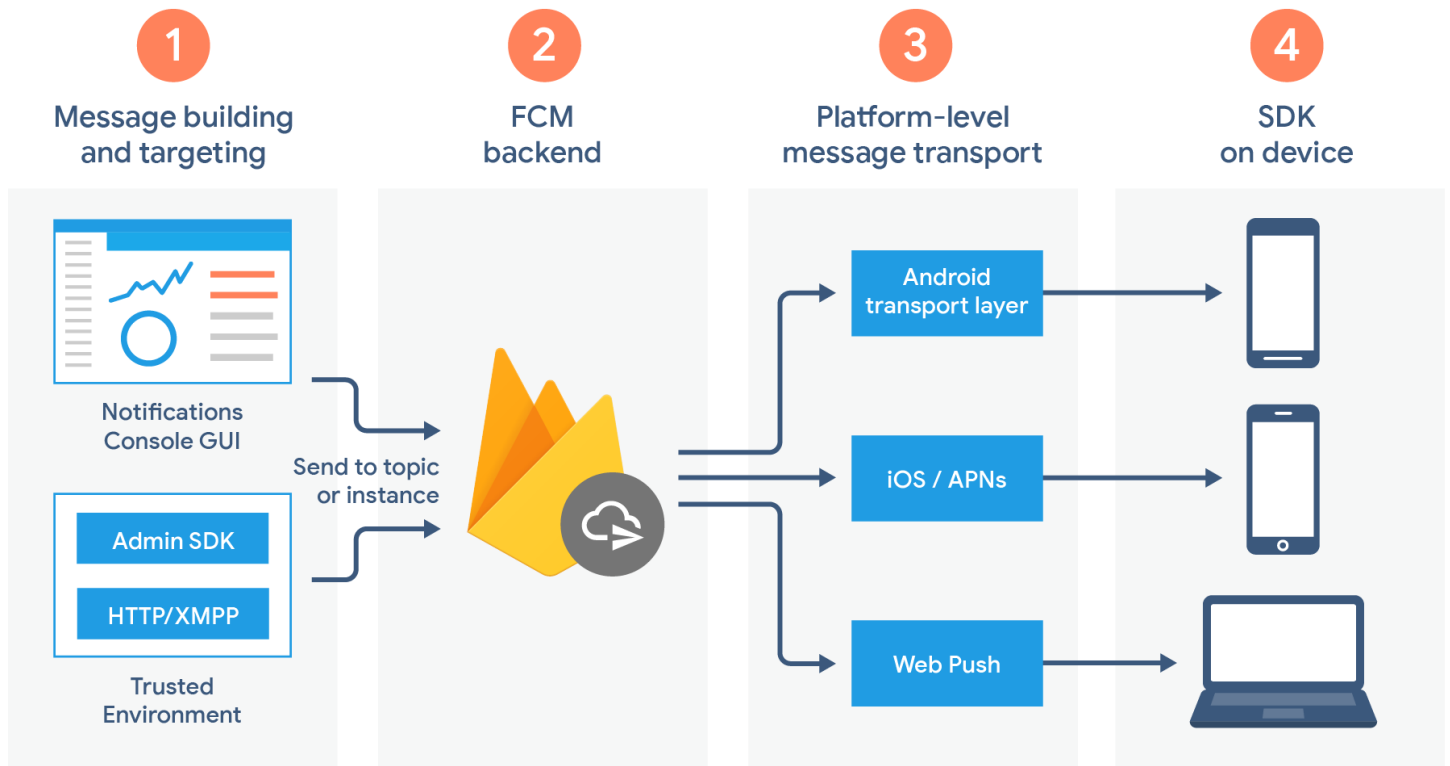
Tên trường	Kiểu dữ liệu	Ý nghĩa	Ghi chú
id	String	Id sản phẩm	Id là duy nhất
name	String	Tên sản phẩm	
supplier	String	Nguồn giao	

f. Shopping Cart

Shopping Cart
created: Date

Tên trường	Kiểu dữ liệu	Ý nghĩa	Ghi chú
id	String	Id sản phẩm	Id là duy nhất
name	String	Tên sản phẩm	

4.2. Thiết kế kiến trúc hệ thống



Phần mềm bọn em sử dụng Firebase real time database có thể trực tiếp lấy dữ liệu ngay trên web. Dữ liệu nhận được dưới dạng JSON. Đồng thời nó cũng luôn được đồng bộ thời gian thực đến mọi kết nối client.


4.3. Thiết kế giao diện







4.3.1. Danh sách các màn hình

STT	Tên màn hình
1	Màn hình đăng nhập
2	Màn hình đăng ký
3	Màn hình trang chủ
4	Màn hình tài khoản
5	Màn hình giỏ hàng
6	Màn hình chi tiết sản phẩm
7	Màn hình chọn gói vận chuyển
8	Màn hình chọn địa chỉ
9	Màn hình đặt hàng
10	Màn hình track order (chưa hoàn thành)
11	Màn hình time line order (chưa hoàn thành)
12	Màn hình địa chỉ nhận hàng (chưa hoàn thành)
13	Màn hình category (chưa hoàn thành)

4.3.2. Mô tả chi tiết các màn hình

4.3.2.1. Màn hình đăng nhập

12:07 

     80% 

Welcome

Sign up

Sign in to continue

Email


nguyenhoangtuan@gmail.com


Password




Forgot Password?


SIGN IN

-OR-


 Sign In with Facebook

 Sign In with Google



facebook



Hãy đăng nhập vào tài khoản
Facebook của bạn để kết nối với
test

Đăng nhập


[Quên mật khẩu?](#)

[Tạo tài khoản](#)




[Lúc khác](#)

[Trung tâm trợ giúp](#)

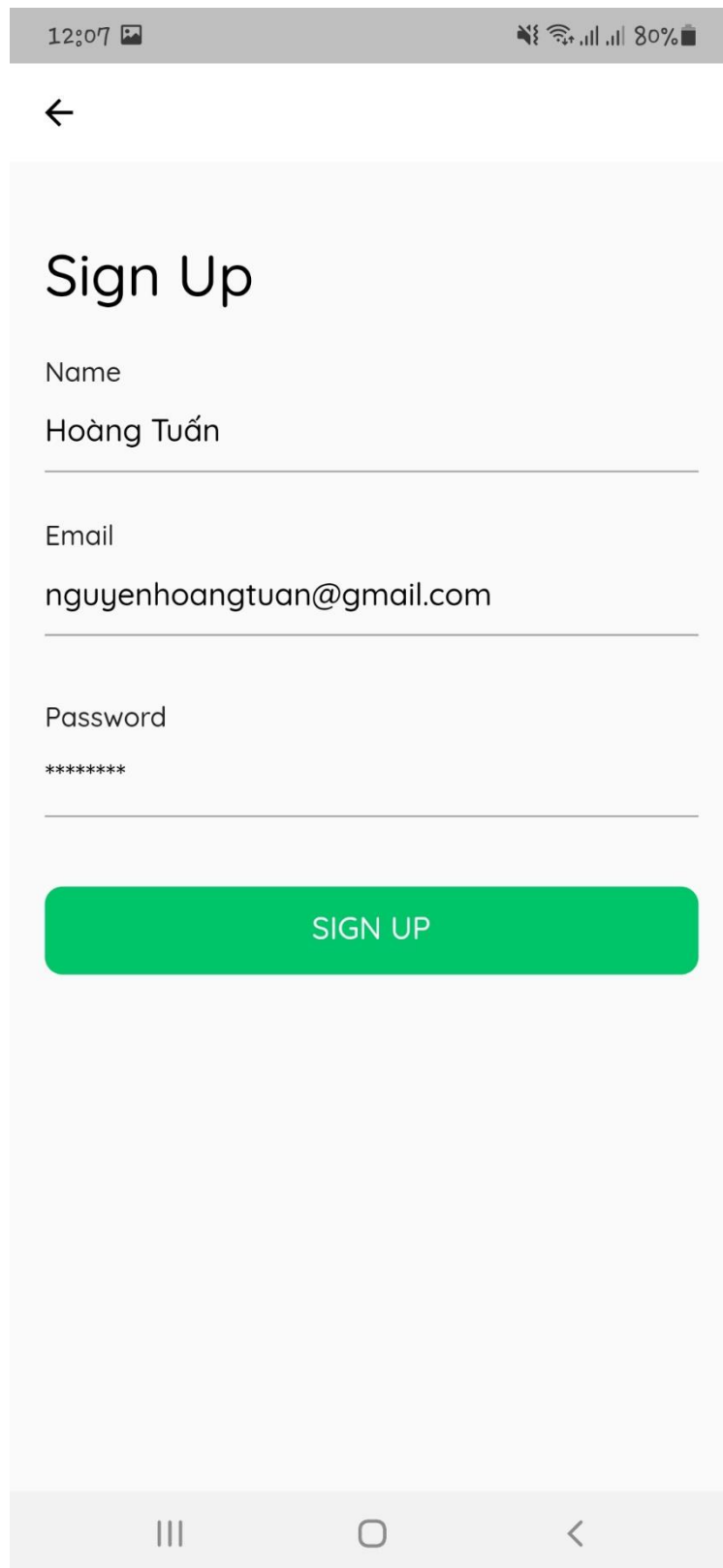
Tiếng Việt
中文(台灣)
Español
Français (France)

English (UK)
한국어
Português (Brasil)


Facebook, Inc.



4.3.2.2. Màn hình đăng ký



The image shows a mobile application interface for a sign-up screen. At the top, there is a status bar with the time 12:07, a camera icon, and various system icons including signal strength, Wi-Fi, and a battery level at 80%. Below the status bar is a white header with a back arrow icon. The main content area has a light gray background. The title 'Sign Up' is displayed in a large, bold, black font. Below the title are three input fields: 'Name' with the value 'Hoàng Tuấn', 'Email' with the value 'nguyenhoangtuan@gmail.com', and 'Password' with the value '*****'. Each input field has a thin gray underline. At the bottom of the form is a large, rounded green button with the text 'SIGN UP' in white, uppercase letters. The bottom of the screen features a white navigation bar with three icons: a hamburger menu (three horizontal lines), a home icon (a circle), and a back arrow.

12:07

←

Sign Up

Name

Hoàng Tuấn

Email

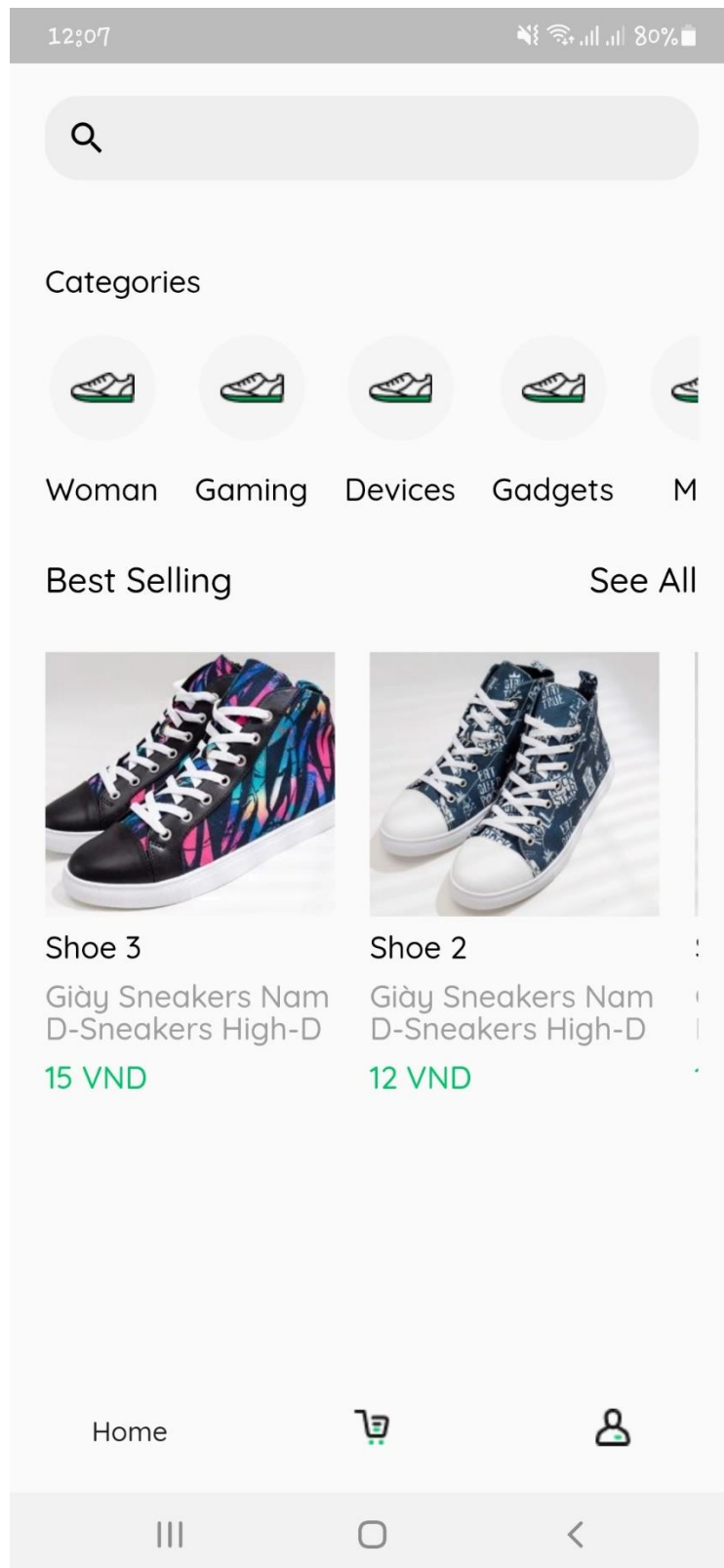
nguyenhoangtuan@gmail.com

Password

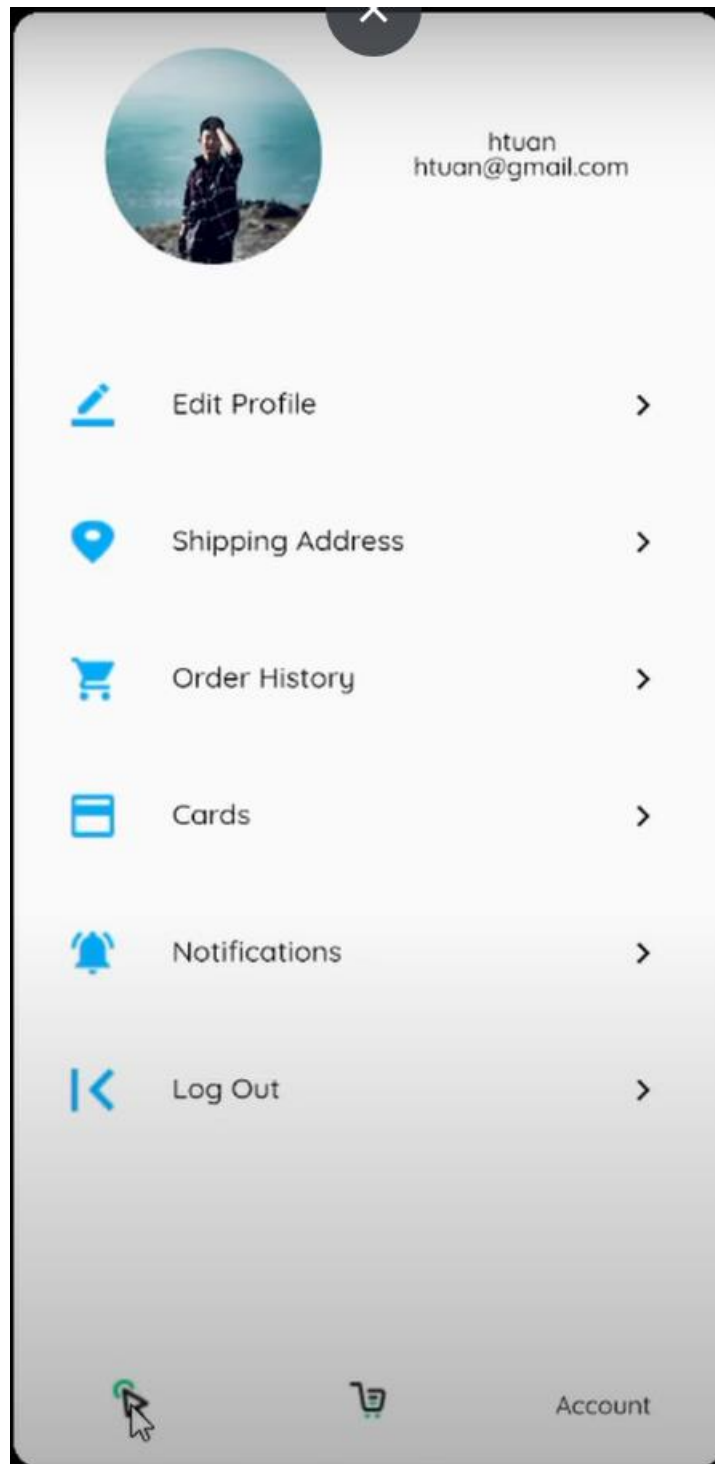
SIGN UP

≡ ○ <

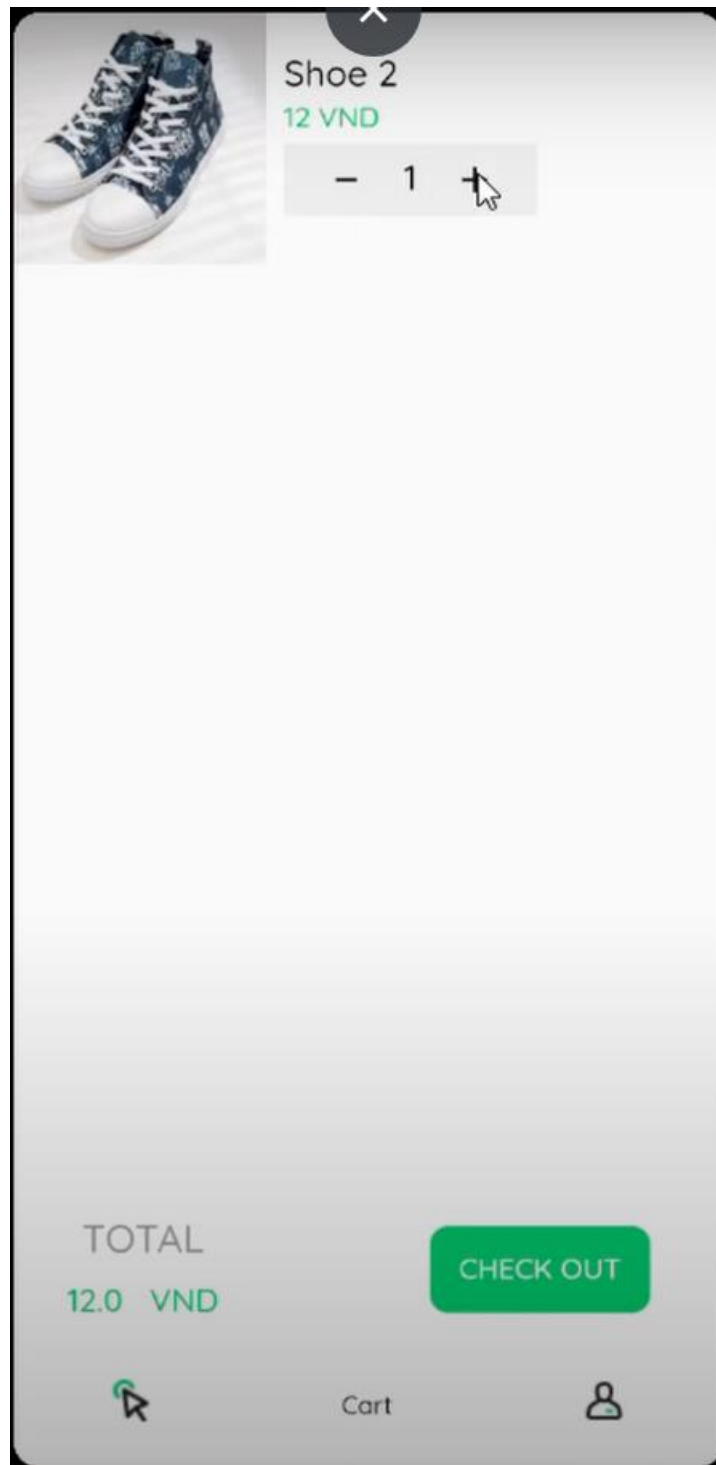
4.3.2.3. Màn hình trang chủ



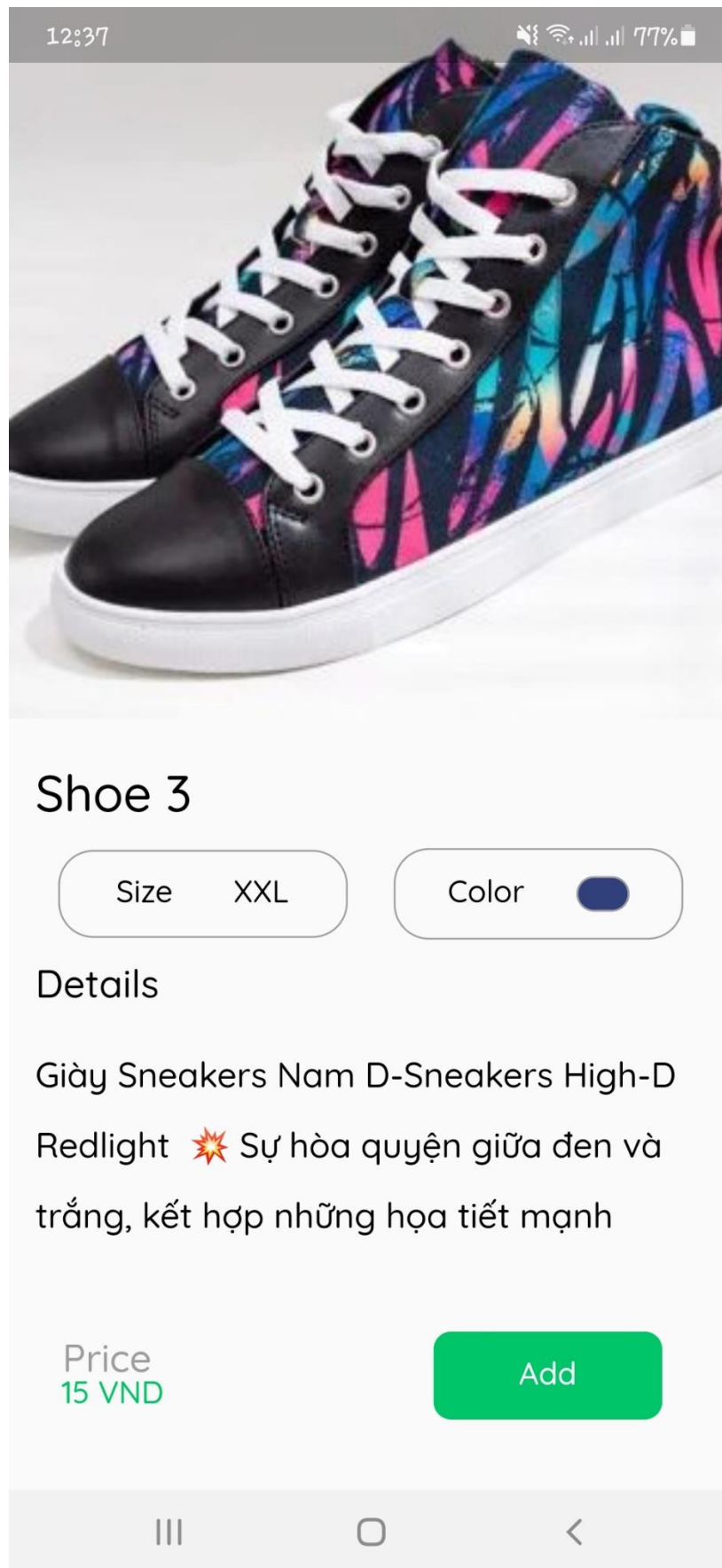
4.3.2.4. Màn hình tài khoản



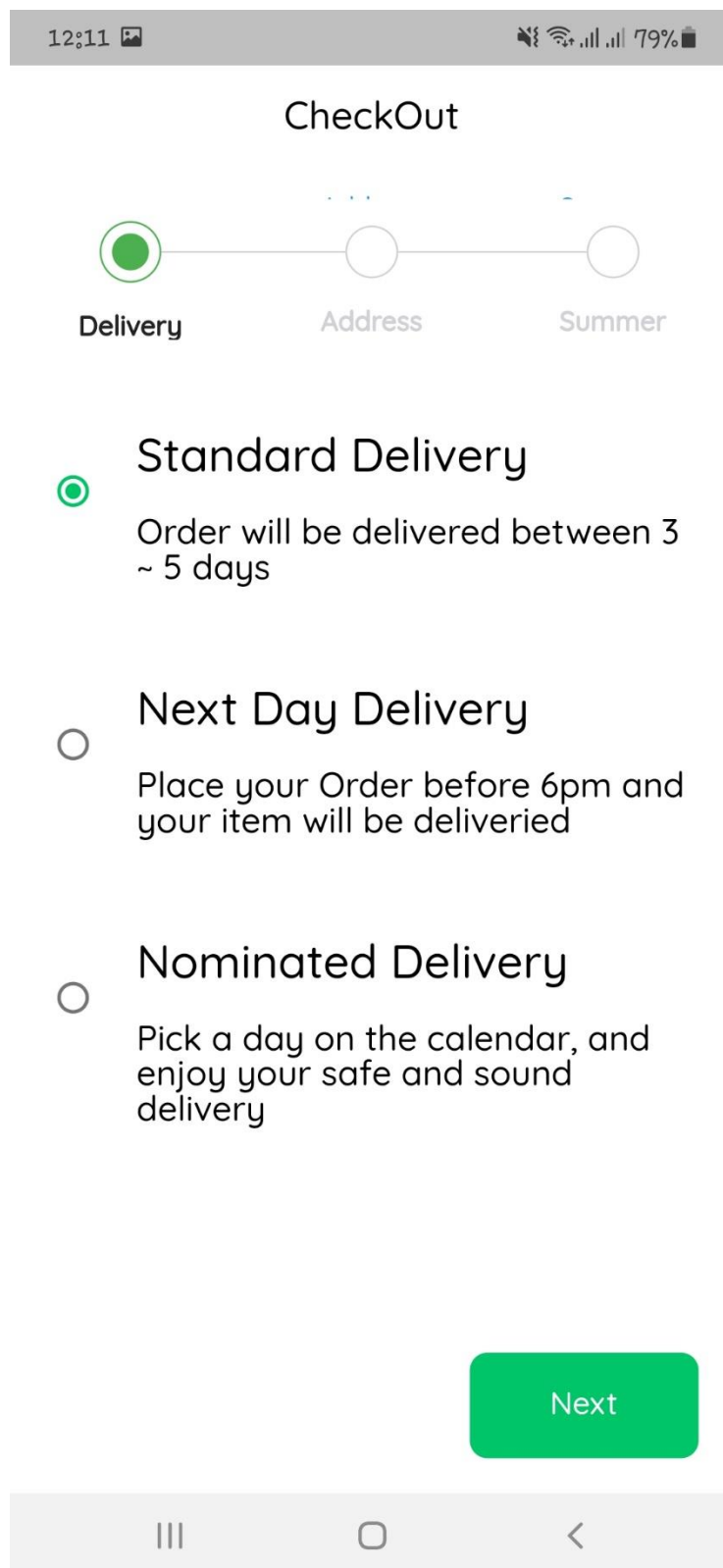
4.3.2.5. Màn hình giỏ hàng



4.3.2.6. Màn hình chi tiết sản phẩm



4.3.2.7. Màn hình chọn gói vận chuyển



4.3.2.8. Màn hình chọn địa chỉ

12:12

79%

CheckOut

Delivery

Address

Summer

Billing address is the same as delivery address

Steet 1

16, Ngo Quyen

Steet 2

Floor 5, Room 2, Sunshine apartment

City

Bien Hoa city

State

Dong Nai

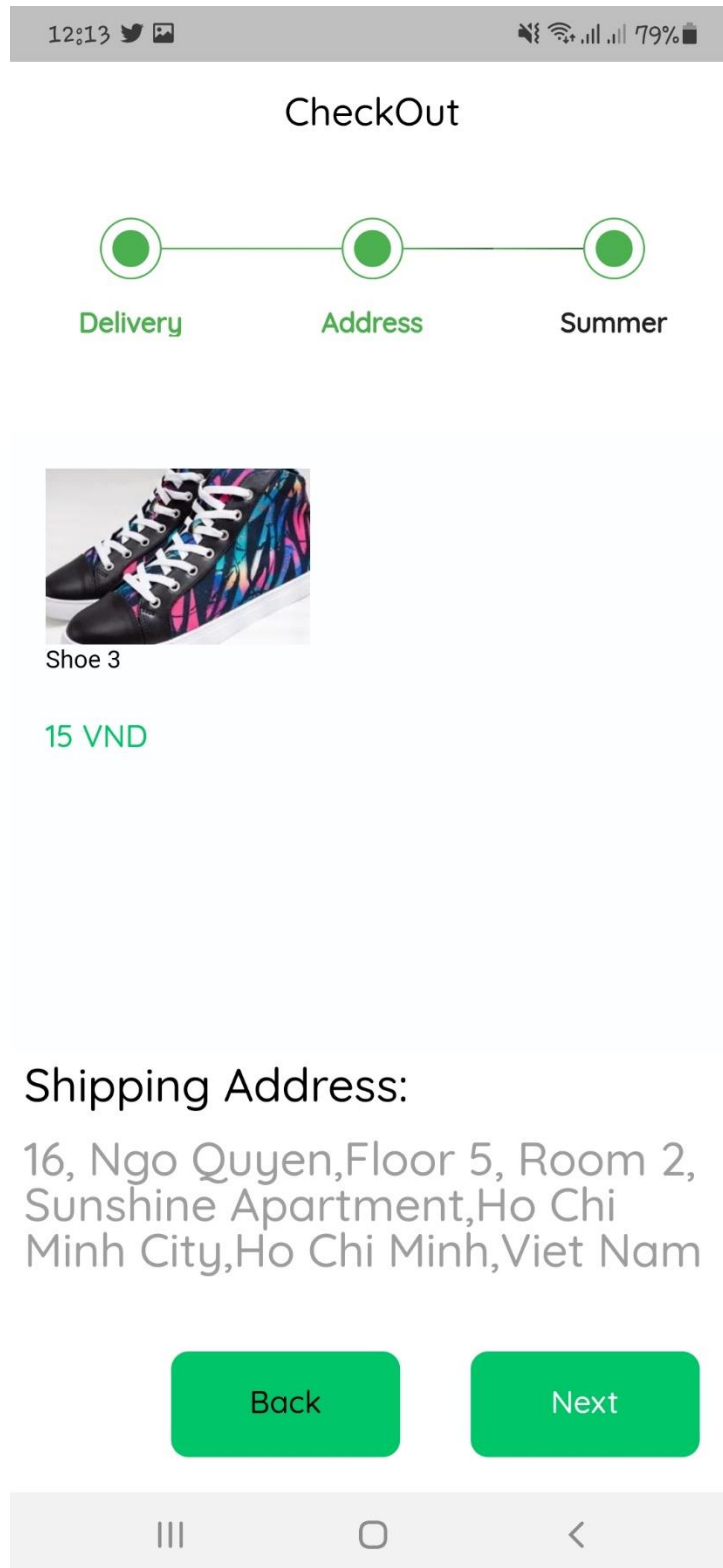
Country

Viet Nam

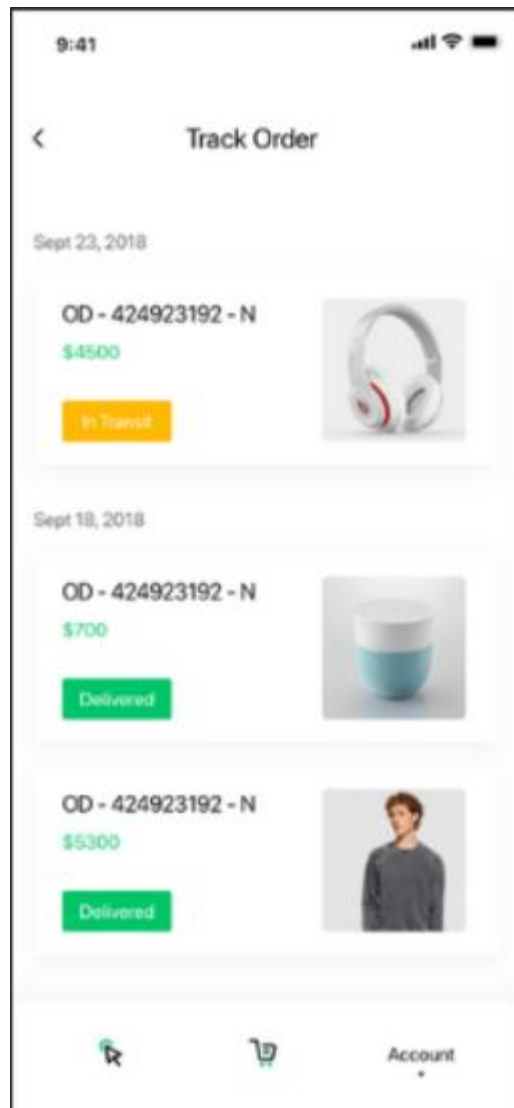
Back

Next

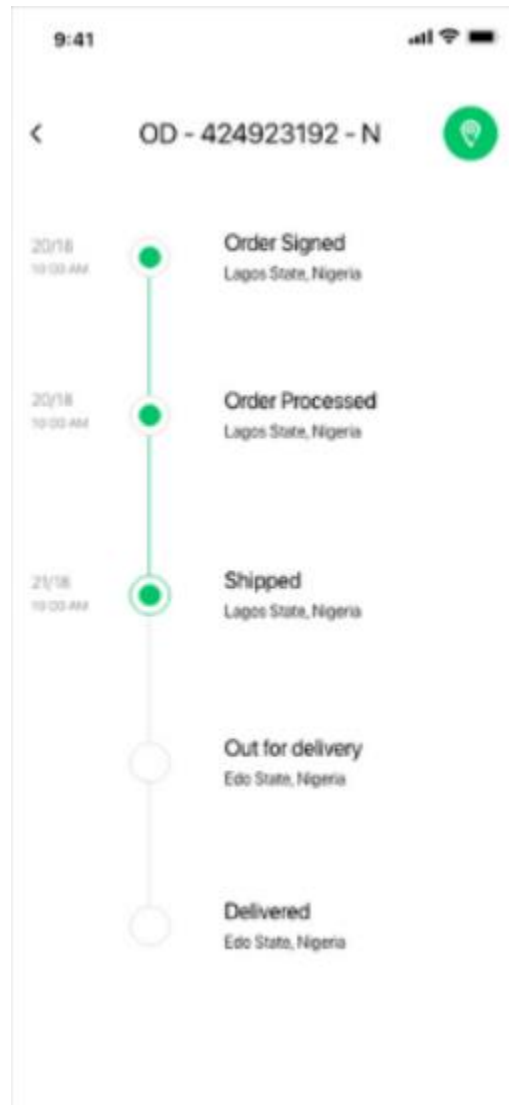
4.3.2.9. Màn hình đặt hàng



4.3.2.10. Màn hình track order



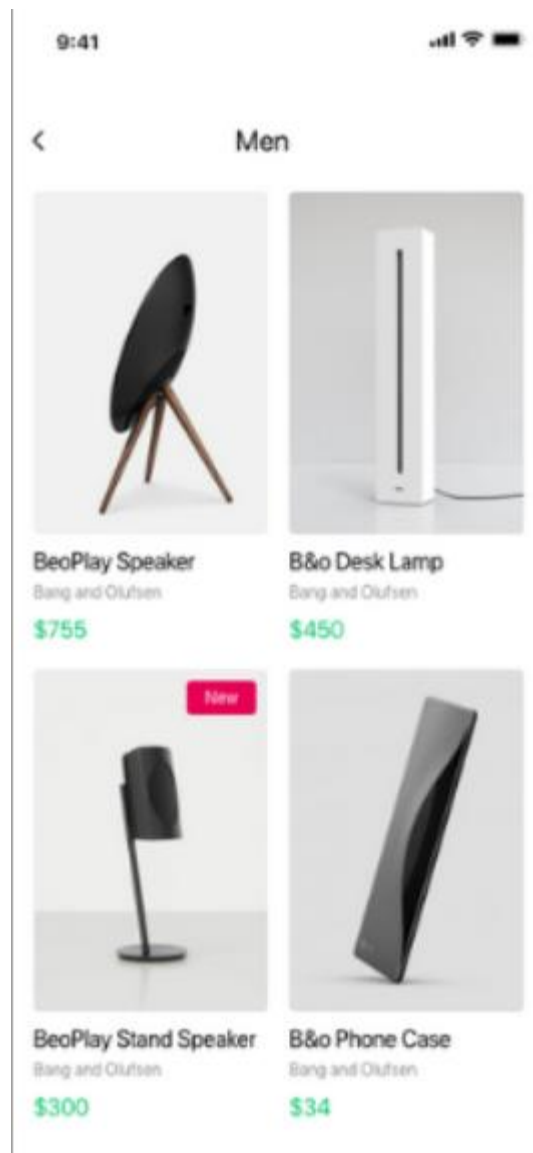
4.3.2.11. Màn hình time line order



4.3.2.12. Màn hình địa chỉ nhận hàng



4.3.2.13. Màn hình category



5. Triển khai ứng dụng

- Ứng dụng được triển khai demo (file apk)
- Cơ sở dữ liệu: Firebase & SQflite
- Quản lý source code: Github
- Đường dẫn của file apk demo (chưa hoàn thành đầy đủ các feature):
https://drive.google.com/file/d/175Az8C_RBsmqxdAIddekC8jWTnCgpg5/view?usp=sharing
- Đường dẫn của source-code:
https://github.com/HoangTuan0611/Shopping-App_Flutter

6. Bảng phân công công việc

Tên và MSSV	Công việc thực hiện
Nguyễn Hoàng Tuấn - 17521217	<ul style="list-style-type: none">- Phân tích chức năng của ứng dụng- Thiết kế giao diện- Thiết kế Database và kiến trúc hệ thống- Code và triển khai App
Trần Diệu Bảo Trân - 17521150	<ul style="list-style-type: none">- Phân tích chức năng của ứng dụng- Viết đặc tả usecase- Vẽ sơ đồ use case tổng quan
Nguyễn Du Phúc Ngân - 17520792	<ul style="list-style-type: none">- Phân tích chức năng của ứng dụng- Thiết kế Database và kiến trúc hệ thống- Code và triển khai App

7. Nhận xét và kết luận

7.1. Nhận xét

7.1.1. Ưu điểm

- Hệ thống thực hiện được cơ bản các chức năng của một ứng dụng shopping
- Hệ thống sử dụng các công nghệ mới được nhiều lập trình viên, công ty tin tưởng và sử dụng.
- Hệ thống sử dụng giao thức HTTPS giúp bảo mật thông tin người dùng tốt hơn.
- Thời gian phản hồi của ứng dụng tương đối nhanh
- Giao diện phù hợp với xu hướng UI/UX của năm
- Hệ thống dễ dàng mở rộng thông qua việc phát triển các module nhỏ

7.1.2. Khuyết điểm

- Việc tích hợp các thư viện mã nguồn mở mới khiến ứng dụng không hoàn toàn ổn định và cần update theo thời gian khi các thư viện mã nguồn mở được mở rộng hơn.
- Các chức năng mới lạ và độc đáo không có nhiều, khó thu hút người dùng.
- Cơ sở dữ liệu thiết kế rời rạc, dễ dàng cho việc mở rộng nhưng sẽ gây cản trở hơn cho việc thống kê tổng hợp thông tin.
- Chưa hoàn thành được những tính năng cần thiết.

7.2. Hướng mở rộng

- Mở rộng ứng dụng theo hướng phát triển mà nhóm đã đề ra.
- Tích hợp thêm các chức năng thanh toán online.
- Tích hợp thêm google map của goole.

8. Tài liệu tham khảo

- Diễn đàn công nghệ của các lập trình viên: <https://stackoverflow.com/>
- Bách khoa toàn thư mở: <https://vi.wikipedia.org/wiki/Wikipedia>
- Trang web chính thức của flutter: <https://flutter.dev/>
- Trang web chính thức tổng hợp các thư viện mã nguồn mở của các lập trình viên: <https://www.npmjs.com/>
- Tài liệu hướng dẫn và cài đặt cơ sở dữ liệu sqflite:
<https://pub.dev/packages/sqflite>
- Diễn đàn công nghệ của lập trình viên Việt Nam: <https://viblo.asia/>
- Trang web hướng dẫn lập trình web của tổ chức W3C – tổ chức World Wide Web: <https://www.w3schools.com/>
- Trang web chính thức của Firebase: <https://firebase.google.com/>
- Tài liệu và nhiều thư viện khác:
<https://pub.dev/packages/get>
https://pub.dev/packages/firebase_core
https://pub.dev/packages/shared_preferences
<https://pub.dev/packages/path>
https://pub.dev/packages/status_change
https://pub.dev/packages/google_sign_in
https://pub.dev/packages/cupertino_icons