

**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN  
KHOA MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG**



**BÁO CÁO ĐO ÁN CUỐI KỲ  
MÔN HỌC: LẬP TRÌNH ỨNG DỤNG WEB**

**Thiết kế và xây dựng Shop bán hàng online**

**Giáo viên hướng dẫn: Đỗ Thị Hương Lan  
Nhóm thực hiện:**

**20521585 - Phan Hữu Luân**

**20521560 - Phạm Ngọc Lợi**

**21522167 - Trần Đình Huy**

**21522138 - Đỗ Gia Huy**

**TP. HỒ CHÍ MINH, 31/12/2023**

## MENU

<b>Chương 1. : TỔNG QUAN, QUY TRÌNH KHẢO SÁT ĐỒ ÁN.....</b>	<b>5</b>
1.1.    Mục đích và mô tả tài liệu .....	5
1.2.    Phạm vi đồ án .....	5
1.3.    Quản lý đồ án với mô hình thác nước kết hợp với Scrum .....	5
1.4.    Các công cụ sử dụng trong đồ án.....	7
1.5.    Các công nghệ và thư viện sử dụng trong đồ án.....	7
1.5.1.    Frontend .....	7
1.5.2.    Backend .....	8
1.6.    Khảo sát .....	8
1.6.1.    Phương thức .....	8
1.6.2.    Thực hiện khảo sát .....	8
1.6.3.    Kết luận và rút kinh nghiệm .....	9
<b>Chương 2. ĐẶC TẢ HỆ THỐNG.....</b>	<b>10</b>
2.1.    Đặc tả .....	10
2.1.1.    Quản lý tài khoản.....	11
2.1.2.    Quản lý sản phẩm (điện thoại, macbook, ipad, watch).....	13
2.1.3.    Quản lý đơn hàng.....	15
2.1.4.    Người dùng mua hàng.....	16
<b>Chương 3. PHÂN TÍCH HỆ THỐNG .....</b>	<b>17</b>
Phân tích hệ thống.....	17
1. Sơ đồ use-case tổng quát.....	18
2. Danh sách các Actors.....	18

3. Danh sách các use-cases .....	19
4. Đặc tả use-case.....	20
4.1. Quản lý tài khoản.....	22
4.3. Quản lý khách hàng .....	24
4.4. Quản lý hóa đơn.....	25
<b>Chương 4. THIẾT KẾ HỆ THỐNG .....</b>	<b>25</b>
4.1.1. Kiến trúc hệ thống.....	25
4.1.2. Mô tả các thành phần .....	26
<b>Chương 5. THIẾT KẾ CƠ SỞ DỮ LIỆU.....</b>	<b>29</b>
<b>Chương 6. THIẾT KẾ GIAO DIỆN VÀ XỬ LÝ .....</b>	<b>32</b>
6.1. Danh sách các màn hình và sơ đồ liên kết các màn.....	32
6.1.1. Sơ đồ liên kết các màn hình .....	32
6.1.2. Danh sách các màn hình .....	32
6.2. Chi tiết màn hình.....	33
6.2.1. Màn hình Login .....	33
6.2.2. Màn hình Register.....	35
6.2.3. Màn hình Home .....	36
6.2.4. Màn hình Products .....	39
6.2.5. Màn hình DetailProduct.....	41
6.2.6. Màn hình Promotion .....	43
6.2.7. Màn hình Cart .....	44
6.2.8. Màn hình Orders Detail .....	45
6.2.9. Màn hình Address .....	46
6.2.10. Các màn hình Dashboard.....	48
<b>Chương 7. CÀI ĐẶT VÀ KIỂM THỦ - ĐỊNH HƯỚNG PHÁT TRIỂN.....</b>	<b>51</b>

7.1.	Các bước cài đặt, deploy đồ án .....	51
7.2.	Kiểm thử: .....	52
7.2.1.	Performance Testing .....	52
7.3.	Ưu nhược điểm đồ án.....	52
7.3.1.	Ưu điểm .....	52
7.3.2.	Nhược điểm.....	52
7.4.	Định hướng phát triển đồ án .....	53
<b>Chương 8.</b>	<b>KẾT LUẬN.....</b>	<b>53</b>

## **Chương 1. : TỔNG QUAN, QUY TRÌNH KHẢO SÁT ĐỒ ÁN**

### **1.1. Mục đích và mô tả tài liệu**

Trong thời đại công nghệ hiện đại, nhu cầu mua sắm điện thoại di động ngày càng tăng cao và người tiêu dùng đang tìm kiếm một trang web cửa hàng điện thoại để đáp ứng nhu cầu này. Việc có một trang web chuyên nghiệp và đáng tin cậy giúp khách hàng dễ dàng tìm thấy và mua được điện thoại phù hợp với nhu cầu và ngân sách của họ. Trang web cửa hàng điện thoại cung cấp một loạt các sản phẩm từ các thương hiệu nổi tiếng, đảm bảo sự đa dạng và lựa chọn cho người mua. Bên cạnh đó, khách hàng cũng có thể tìm hiểu về các tính năng và đặc điểm của từng sản phẩm thông qua mô tả và đánh giá chi tiết. Trang web còn cung cấp dịch vụ tư vấn từ nhân viên chuyên nghiệp, giúp khách hàng hiểu rõ hơn về các sản phẩm và lựa chọn phù hợp. Ngoài ra, trang web cửa hàng điện thoại cung cấp các chính sách hậu mãi và bảo hành, mang lại sự an tâm và hỗ trợ sau khi mua hàng. Tổng quan, một trang web cửa hàng điện thoại đáng tin cậy không chỉ đáp ứng nhu cầu mua sắm hiện nay mà còn mang lại sự tiện lợi và tin tưởng cho khách hàng.

Khi thực hiện đề tài này, nhóm chúng em rất mong sẽ hiểu rõ hơn về các kiến thức về thiết kế và xây dựng web, biết cách vận dụng những kiến thức đã học và hiểu thêm về các công nghệ liên quan, đồng thời cũng hy vọng mô hình web sẽ phần nào hỗ trợ người dùng cảm thấy tiện lợi hơn trong việc tìm hiểu các thể loại thiết bị di động từ hàng phân phối và hỗ trợ được người quản lý những yêu cầu cơ bản trong việc điều hành một trang web.

### **1.2. Phạm vi đồ án**

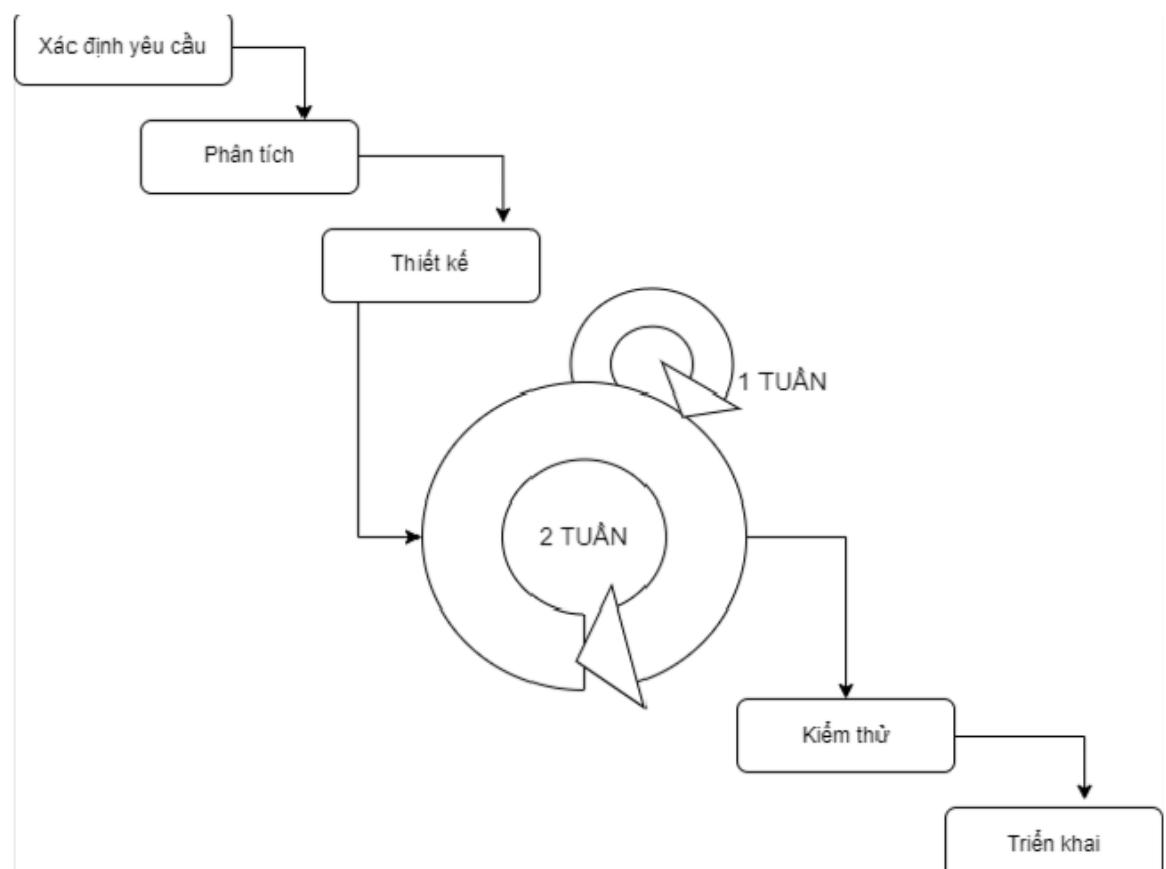
Phạm vi đồ án: xây dựng và vận hành được trang web sử dụng các công nghệ web như html, css, javascript, reactjs,... đối với frontend và xây dựng backend với nodejs và dùng csdl mongodb. Tuy nhiên chưa xem xét đến responsive trong giao diện.

Phạm vi và đối tượng hệ thống hướng đến tất cả những người có nhu cầu tìm hiểu, mua sắm điện thoại

### **1.3. Quản lý đồ án với mô hình thác nước kết hợp với Scrum**

Nhóm lựa chọn mô hình thác nước để quản lý các bước trong đồ án bởi lý do đây là mô hình đơn giản, dễ tiếp cận đối với các thành viên trong nhóm, việc chia

từng giai đón rõ ràng cũng giúp nhóm có thể phân chia công việc rõ ràng và đều hơn. Ngoài ra nhóm còn kết hợp với mô hình Scrum ở giai đoạn phát triển vì lý do nhóm cần quản lý tốt tiến độ của các thành viên để nhanh chóng có thể giúp đỡ, giải quyết các khó khăn các thành viên gặp phải, cụ thể mô hình quản lý của nhóm như sau:



Hình 1 Mô hình quản lí đồ án

Xác định yêu cầu: thực hiện khảo sát và xem xét các yêu cầu của môn học, đưa ra loạt các yêu cầu tối thiểu và các yêu cầu có thể bổ sung cho đồ án.

Phân tích: sau khi xác định các yêu cầu cần thực hiện, nhóm xem xét khả năng, thời gian để lên kế hoạch phù hợp cho bước tiếp theo, đưa ra bảng kế hoạch phát triển dựa trên yêu cầu đồ án và khả năng của nhóm.

Thiết kế: dựa trên các phân tích đã có nhóm bắt đầu thiết kế cơ sở dữ liệu, api đối với backend và thiết kế giao diện, tương tác đối với front end.

Phát triển kết hợp Scrum: Tiến hành lập trình từng thành phần nhỏ của trang web, logic của hệ thống và các chức năng đã được làm rõ ở các bước trước đó. Với mỗi sprint, nhóm sẽ tiến hành thiết kế chi tiết cho từng thành phần, cài đặt và kiểm thử đơn vị cho từng phần đã thực hiện.

Kiểm thử: sau khi đã phát triển xong dựa trên các phân tích, nhóm tiến hành kiểm thử toàn bộ những phần code, xác định lỗi và thực hiện chỉnh sửa ngay.

Triển khai: sau bước kiểm thử đã đạt được những yêu cầu đã đặt ra, nhóm tiến hành deploy trang web và viết báo cáo.

#### 1.4. Các công cụ sử dụng trong đồ án

Google Meet: Google Meet là công cụ nhóm dùng trong quá trình thực hiện đồ án với mục đích trao đổi thông tin, tổ chức các cuộc họp hàng tuần cũng như các cuộc họp đột xuất, lưu trữ lại thông tin cuộc họp.

Google Drive: Google Drive là nơi được nhóm dùng để lưu trữ toàn bộ tài liệu liên quan tới đồ án và chủ yếu là hình ảnh sử dụng trong đồ án, tài liệu về đồ án, các dữ liệu được sử dụng và các báo cáo của đồ án.

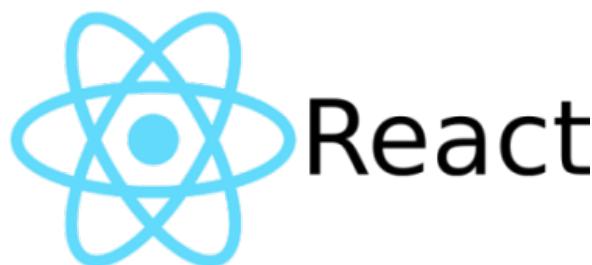
Visual Studio Code: Visual Studio Code là text editor được nhóm lựa chọn vì nó miễn phí, đa nền tảng, có kho tiện ích mở rộng đa dạng giúp quá trình code được thuận tiện và nhanh hơn, hỗ trợ nhiều ngôn ngữ lập trình.

Git – GitHub: Git là một hệ thống quản lý phiên bản phân tán (Distributed Version Control System – DVCS), nó là một trong những hệ thống quản lý phiên bản phân tán phổ biến nhất hiện nay. Git cung cấp cho mỗi lập trình viên kho lưu trữ (repository) riêng chứa toàn bộ lịch sử thay đổi. Toàn bộ source code của đồ án sẽ được quản lý bằng Github.

#### 1.5. Các công nghệ và thư viện sử dụng trong đồ án

##### 1.5.1. Frontend

**ReactJS:** ReactJS là một opensource được phát triển bởi Facebook, ra mắt vào năm 2013, bản thân nó là một thư viện Javascript được dùng để xây dựng các tương tác với các thành phần trên website. Một trong những điểm nổi bật nhất của ReactJS đó là việc render dữ liệu không chỉ thực hiện được trên tầng Server mà còn ở dưới Client nữa. ReactJS là một thư viện JavaScript chuyên giúp các nhà phát triển xây dựng giao diện người dùng hay UI.



### 1.5.2. Backend

**Node:** NodeJS là một môi trường runtime chạy JavaScript đa nền tảng và có mã nguồn mở, được sử dụng để chạy các ứng dụng web bên ngoài trình duyệt của client. Nền tảng này được phát triển vào năm 2009, được xem là một giải pháp hoàn hảo cho các ứng dụng sử dụng nhiều dữ liệu nhờ vào mô hình hướng sự kiện (event-driven) không đồng bộ



**MongoDB:** MongoDB là một hệ quản trị cơ sở dữ liệu mã nguồn mở, là CSDL thuộc NoSql và được hàng triệu người sử dụng. MongoDB là một database hướng tài liệu (document), các dữ liệu được lưu trữ trong document kiểu JSON thay vì dạng bảng như CSDL quan hệ.



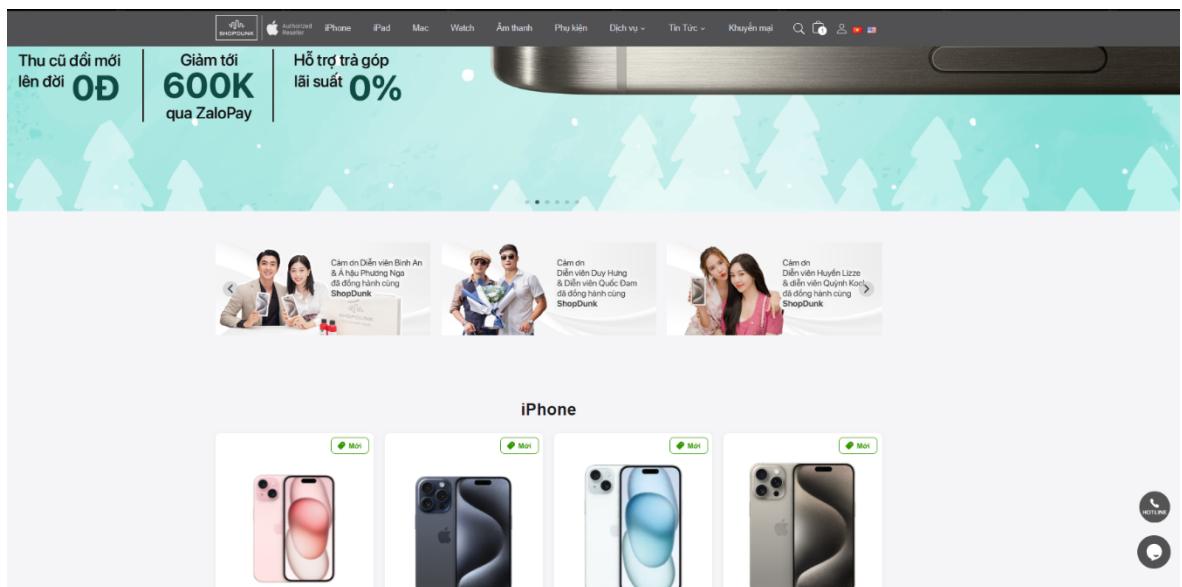
## 1.6. Khảo sát

### 1.6.1. Phương thức

Khảo sát các trang web bán điện thoại tương tự khác để tham khảo giao diện, các chức năng, luồng hoạt động của các trang web đó, đồng thời rút ra các ưu nhược điểm của các trang web để làm nền tảng phát triển đồ án của nhóm

### 1.6.2. Thực hiện khảo sát

Khảo sát trang web <https://shopdunk.com/>



Hình 2 Hình ảnh website Shopdunk

**Giao diện:** gồm các thành phần cơ bản đối với một trang web trình bày sản phẩm như trang chủ, trang trình bày tất cả sản phẩm, trang chi tiết sản phẩm, tin tức và liên hệ. Trong đó nổi bật là trang chủ với thiết kế chia từng phần rất rõ ràng. Nhóm chúng em quyết định thực hiện theo trang chủ này, và xây dựng cấu trúc các trang tương tự như trang web shopdunk.

**Chức năng:** hiển thị, tìm kiếm sản phẩm, có thông tin chi tiết của sản phẩm, trình bày, cho phép đọc các tin tức, phân trang các trang có nhiều thành phần, cho phép người dùng liên hệ với chăm sóc khách hàng thông qua hotline hay messenger trên trang web.

**Ưu điểm:** Có đầy đủ chức năng, dễ sử dụng, có thể xem như là đại diện cho một trang web bán điện thoại cơ bản

**Nhược điểm:** Thiết kế phức tạp.

### 1.6.3. Kết luận và rút kinh nghiệm

**Về giao diện:** nhóm học hỏi và xem xét đồ án của nhóm sẽ có các trang cơ bản đối với trang web shopdunk như trang chủ, trang giới thiệu, trang danh sách sản phẩm, trang tin tức, trang dành cho user, trang dành cho quản lý, hệ thống trang cho đăng nhập đăng ký, trang tìm kiếm, các header footer phù hợp với từng trang

**Về Chức năng:** xem xét các chức năng cơ bản từ trang web đã khảo sát để áp dụng vào đồ án môn học như lấy dữ liệu từ csdl, trình bày sản phẩm theo loại, trình bày sản phẩm theo kết quả tìm kiếm và thực hiện đánh giá khả năng của nhóm để bổ sung các chức năng trình bày thông tin người dùng bao gồm các thông tin cá

nhân và các đơn người dùng đã lập, hệ thống quản lý sản phẩm, quản lý nhân viên, khách hàng, quản lý đơn hàng, quản lý hóa đơn, quản lý tin tức.

## **Chương 2. ĐẶC TẢ HỆ THỐNG**

### **2.1. Đặc tả**

Chương này giúp chúng ta đặc tả chi tiết hơn yêu cầu của hệ thống với các yêu cầu được mô tả trong danh sách các yêu cầu chức năng dưới đây.

#### **❖ Yêu cầu chức năng**

##### **Quản lý tài khoản**

- Đăng ký
- Đăng nhập
- Đăng xuất

##### **Quản lý sản phẩm**

- Thêm mới sản phẩm.
- Xem thông tin sản phẩm.
- Sửa thông tin sản phẩm.
- Xóa sản phẩm.
- Mua sản phẩm.

##### **Quản lý đơn hàng**

- Xem chi tiết đơn hàng
- Thêm đơn hàng

##### **Người dùng mua hàng**

- Thêm vào giỏ hàng
- Sửa/xoá số lượng

Với mỗi yêu cầu sẽ gồm nhiều yêu cầu nhỏ thuộc chức năng cần tìm hiểu, chúng sẽ được đặc tả có cấu trúc dựa trên biểu mẫu (form) và thể hiện dưới dạng bảng các thông tin chung của chức năng đó. Bảng thông tin chung bao gồm định nghĩa chức năng, mô tả đầu vào, mô tả đầu ra, thông tin cần thiết cho tính toán và thực thể khác được sử dụng, mô tả hành động sẽ được thực hiện, điều kiện trước và sau (nếu phù hợp) và các tác dụng phụ (nếu có).

### Nội dung chương này gồm có:

- Quản lý tài khoản
- Quản lý sản phẩm
- Quản lý đơn hàng
- Gợi ý sản phẩm
- Yêu cầu phi chức năng

#### 2.1.1. Quản lý tài khoản

##### 2.1.1.1. Đăng ký

Chức năng	Đăng kí
Mô tả	Đăng kí thành viên cho hệ thống
Mô tả đầu vào nguồn	Người dùng chọn tính năng đăng ký trên giao diện. Nhập các thông tin để đăng ký tài khoản.
Mô tả đầu ra và đích đến	Đăng kí thành công. Đích đến là trang chủ home
Mô tả hành động	Để trở thành khách hàng của trang thì người dùng phải đăng ký tài khoản bằng cách cung cấp các thông tin như: Email, tên và mật khẩu để hệ thống cung cấp tài khoản tương ứng. Các bước nhập được xác thực về các yếu tố đảm bảo an toàn cho tài khoản.
Điều kiện trước	Tài khoản chưa tồn tại trong cơ sở dữ liệu.
Điều kiện sau	Để có thể truy cập vào lần tiếp theo, người dùng phải đăng nhập để có thể sử dụng tài khoản đã được đăng ký (không đăng ký lại)

Tác dụng phụ	Thông báo không thành công nếu điều kiện trước không được đảm bảo.
--------------	--

*Bảng 1: Đặc tả có cấu trúc cho yêu cầu đăng ký*

#### 2.1.1.2. Đăng nhập

Chức năng	Đăng nhập
Mô tả	Cho phép đăng nhập và sử dụng trang web với tư cách là thành viên khách hàng.
Mô tả đầu vào nguồn	Người dùng chọn nút đăng nhập trên giao diện
Mô tả đầu ra và đích đến	Đăng nhập/đăng kí thành công và dẫn đến giao diện màn hình trang chủ của hệ thống.
Mô tả hành động	Người dùng sử dụng tài khoản đã được đăng kí trước đó để đăng nhập vào trang web để tiến hành mua hàng
Điều kiện trước	Người dùng phải dùng tài khoản đã được đăng kí
Điều kiện sau	Người dùng được xem tất cả các thông tin hiển thị công khai trên web (trừ trang quản lý, trang quản lý chỉ nhân viên với admin mới có thể truy cập vào)
Tác dụng phụ	Nếu tài khoản không tồn tại hiển thị “Đăng nhập thất bại”

*Bảng 1: Đặc tả có cấu trúc cho yêu cầu đăng ký*

#### 2.1.1.3. Đăng xuất

Chức năng	Đăng xuất
Mô tả	Người dùng có nhu cầu đăng xuất khỏi trang web.
Mô tả đầu vào nguồn	Người dùng nhấp vào nút "Tài khoản" trên thanh menu và sau đó thực hiện thao tác đăng xuất khỏi tài khoản cá nhân.

Mô tả đầu ra và đích đến	Đăng xuất thành công. Người dùng sẽ được chuyển hướng về trang chủ chính của trang web.
Mô tả hành động	Người dùng muốn đăng xuất ra khỏi hệ thống để có thể sử dụng tài khoản khác hoặc tạo tài khoản mới.
Điều kiện trước	Trước đó, người dùng phải đăng nhập vào hệ thống.
Điều kiện sau	Trở về màn hình trang chủ chính
Tác dụng phụ	Không

Bảng 1: Đặc tả có cấu trúc cho yêu cầu đăng ký

### 2.1.2. Quản lý sản phẩm (điện thoại, macbook, ipad, watch)

#### 2.1.2.1. Thêm mới sản phẩm

Chức năng	Thêm mới sản phẩm
Mô tả	Thêm dữ liệu sản phẩm mới vào cơ sở dữ liệu.
Mô tả đầu vào nguồn	Người dùng chọn tính năng thêm sản phẩm trong giao diện quản lý sản phẩm.
Mô tả đầu ra và đích đến	Thêm mới thành công. Người dùng sẽ được chuyển đến màn hình thêm mới sản phẩm sau khi quá trình thêm mới hoàn tất.
Mô tả hành động	Người dùng cần cung cấp đầy đủ thông tin của sản phẩm để thực hiện thêm mới.
Điều kiện trước	Người dùng phải có quyền truy cập, tức là là admin hoặc nhân viên.
Điều kiện sau	Dữ liệu mới được thêm vào cơ sở dữ liệu và ứng dụng.
Tác dụng phụ	Nếu thông tin cung cấp không đúng hoặc không đủ, hệ thống sẽ hiển thị thông báo thêm dữ liệu thất bại để thông báo cho người dùng.

#### 2.1.2.2. Xem thông tin chi tiết sản phẩm

Chức năng	Xem thông tin chi tiết sản phẩm
Mô tả	Người dùng muốn xem thông tin sản phẩm

Mô tả đầu vào nguồn	Người dùng nhập vào nút xem.
Mô tả đầu ra và đích đến	Mở ra màn hình chi tiết cửa hàng.
Mô tả hành động	Người dùng xem các thông tin của sản phẩm, số lượng còn lại trong kho
Điều kiện trước	Người dùng phải có quyền truy cập, tức là là admin hoặc nhân viên.
Điều kiện sau	Người dùng được xem toàn bộ thông tin sản phẩm
Tác dụng phụ	

#### 2.1.2.3. Cập nhật thông tin sản phẩm

Chức năng	Cập nhật thông tin sản phẩm
Mô tả	Người dùng muốn sửa thông tin sản phẩm.
Mô tả đầu vào nguồn	Người dùng nhập vào biểu tượng sửa trong trang quản lý sản phẩm.
Mô tả đầu ra và đích đến	Sửa thông tin thành công, chi tiết sản phẩm sẽ được thay đổi ở giao diện của người dùng web
Mô tả hành động	Người dùng nhập các thông tin mới.
Điều kiện trước	Người dùng phải có quyền truy cập, tức là là admin hoặc nhân viên.
Điều kiện sau	Thông tin phải được thay đổi trên cơ sở dữ liệu và ứng dụng hiển thị thông tin mới.
Tác dụng phụ	

#### 2.1.2.4. Xóa sản phẩm

Chức năng	Xóa sản phẩm
Mô tả	Người dùng muốn xóa sản phẩm.
Mô tả đầu vào nguồn	Người dùng nhập vào biểu tượng xóa trên giao diện quản lý sản phẩm.
Mô tả đầu ra và đích đến	Mở ra màn hình thông báo xóa.
Mô tả hành động	Người dùng chọn xóa để xác nhận quyết định xóa sản phẩm.
Điều kiện trước	Người dùng phải có quyền truy cập, tức là là admin hoặc nhân viên.

Điều kiện sau	Dữ liệu phải bị xóa trên cơ sở dữ liệu và ứng dụng.
Tác dụng phụ	Nếu người dùng hủy bỏ thao tác xóa, hệ thống có thể quay lại trạng thái trước đó mà không thực hiện xóa dữ liệu sản phẩm.

### 2.1.3. Quản lý đơn hàng

#### 2.1.3.1. Thêm đơn hàng

Chức năng	Thêm đơn hàng
Mô tả	Người dùng thêm một đơn hàng mới.
Mô tả đầu vào nguồn	Người dùng sau khi đặt hàng chọn tùy chọn "Thanh toán"
Mô tả đầu ra và đích đến	Đơn hàng được đặt thành công và in ra hoá đơn
Mô tả hành động	Người dùng sau khi đặt hàng thực hiện thao tác chọn tùy chọn "Thanh toán".
Điều kiện trước	Người dùng đã xác nhận đặt hàng.
Điều kiện sau	Đơn hàng của người dùng được thêm vào hệ thống.
Tác dụng phụ	Thông báo xác nhận việc thêm đơn hàng thành công.

#### 2.1.3.2. Xem thông tin đơn hàng

Chức năng	Xem thông tin đơn hàng
Mô tả	Người dùng muốn xem thông tin chi tiết của một đơn hàng.
Mô tả đầu vào nguồn	Người dùng sau khi đặt hàng sẽ được xem hoá đơn của đơn hàng
Mô tả đầu ra và đích đến	Người dùng được chuyển đến trang chi tiết đơn hàng và có thể xem được thông tin chi tiết của đơn hàng đó.
Mô tả hành động	Người dùng vào trang lịch sử đặt hàng trong trang người dùng và thực hiện thao tác chọn xem chi tiết ở một đơn hàng.
Điều kiện trước	Người dùng đã nhấn nút "Thanh toán" ở trang thanh toán khi đặt hàng

Điều kiện sau	Người dùng xem được thông tin chi tiết của đơn hàng.
Tác dụng phụ	Không có tác động phụ được mô tả.

#### 2.1.4. Người dùng mua hàng

##### 2.1.4.1. Thêm vào giỏ hàng

Chức năng	Mua sản phẩm
Mô tả	Người dùng chọn sản phẩm mình muốn để thêm vào giỏ hàng
Mô tả đầu vào nguồn	Trong mỗi trang sản phẩm sẽ có nút thêm vào giỏ hàng
Mô tả đầu ra và đích đến	Giỏ hàng sẽ cập nhật sản phẩm đã được chọn của người dùng
Mô tả hành động	Người dùng xem các sản phẩm và thực hiện thêm vào giỏ hàng với số lượng mong muốn
Điều kiện trước	Người dùng thực hiện nút thêm vào giỏ hàng
Điều kiện sau	Giỏ hàng cập nhật các sản phẩm đã được chọn
Tác dụng phụ	Không có tác động phụ được mô tả.

##### 2.1.4.2. Sửa/Xoá số lượng

Chức năng	Sửa/Xoá số lượng
Mô tả	Người dùng sửa/xoá số lượng sản phẩm đã chọn
Mô tả đầu vào nguồn	Thực hiện xoá/sửa bằng các nút có trong giao diện giỏ hàng
Mô tả đầu ra và đích đến	Số lượng sản phẩm được cập nhật, sau khi thao tác của người dùng
Mô tả hành động	Người dùng chọn các nút xoá, thêm sản phẩm ở kế bên hình mini của mỗi sản phẩm trong giỏ hàng của mình
Điều kiện trước	Giỏ hàng đã được chọn một số sản phẩm
Điều kiện sau	Số lượng được cập nhật chính xác theo các thao tác
Tác dụng phụ	

## **Chương 3. PHÂN TÍCH HỆ THỐNG**

### **Phân tích hệ thống**

Nhóm quyết định lựa chọn kiến trúc phân lớp (layered architecture) cho dự án vì những lý do sau:

Dễ thiết kế: Kiến trúc này không quá phức tạp trong việc thiết kế, giúp các thành viên trong nhóm dễ dàng tiếp cận và hiểu được mô hình thiết kế này.

Tổ chức rõ ràng: Các chức năng và thành phần của ứng dụng được tổ chức theo cách thức rõ ràng, giúp nhận biết các tầng của hệ thống và hiểu rõ hơn về chức năng của từng tầng.

Dễ mở rộng: Kiến trúc phân lớp cung cấp sự linh hoạt, dễ dàng để xây dựng thêm các hệ thống hoặc chức năng mới cho ứng dụng.

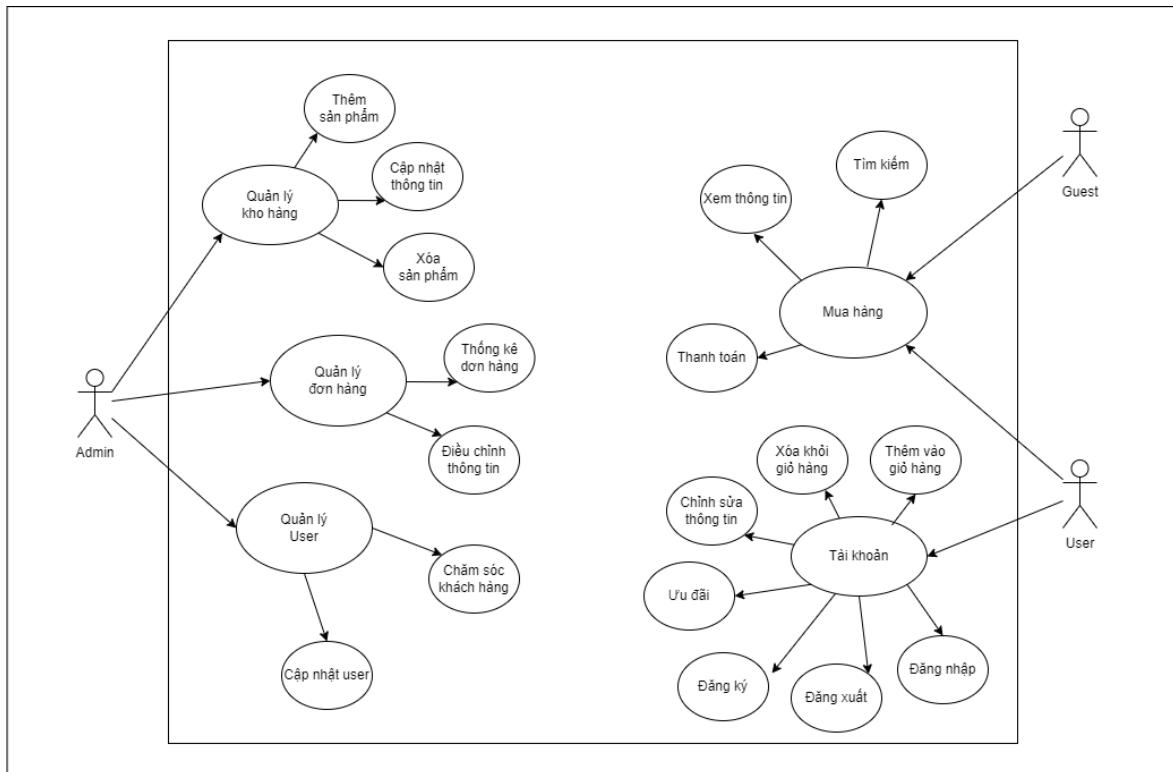
Chương này hướng đến việc xác định các tác nhân của hệ thống, các kịch bản sử dụng điển hình của hệ thống, và mục tiêu của các tác nhân đó. Từ thông tin này, chúng ta sẽ tiến hành phân tích hệ thống dưới dạng mô hình hóa, bao gồm các sơ đồ use-case.

### **SƠ ĐỒ USE CASE**

Sơ đồ use-case giúp mô hình hóa các tương tác giữa từng thành phần, hỗ trợ quy trình khơi gợi yêu cầu. Mỗi use-case đại diện cho một nhiệm vụ rời rạc (discrete task) liên quan đến sự tương tác từ bên ngoài với hệ thống. Các tác nhân (actor) trong use-case có thể là người cũng có thể là hệ thống khác.

Phần này sẽ mô tả các yêu cầu chức năng, bao gồm tên của hầu hết các use cases và actors được mong đợi. Đồng thời, mỗi use-case sẽ có những mô tả chi tiết cho chúng.

## 1. Sơ đồ use-case tổng quát



## 2. Danh sách các Actors

No	ID Actor	Actor	Mô tả
1	A01	Admin	Tác nhân admin được dùng tất cả các tính năng của hệ thống và được chỉnh sửa các tính năng.
2	A02	System	Tác nhân hệ thống là cơ sở dữ liệu, giúp truy xuất dữ liệu khi cần hiển thị trên giao diện cho tác nhân khác
3	A03	Customer	Tác nhân khách hàng khi đã có tài khoản, đăng nhập vào hệ thống sẽ được dùng tất cả những tính năng dành cho khách hàng
4	A04	Guest	Tác nhân người dùng khách được xem giới thiệu về hệ thống, xem các tính năng hệ thống cung cấp như xem các loại sản phẩm nhưng chỉ được sử dụng tính năng khi đã đăng ký tài khoản

Bảng 3.1 Danh sách các actors và mô tả của chúng

### 3. Danh sách các use-cases

Danh sách các use-cases chính đã được mô hình hóa từ use-case tổng quát gồm ý nghĩa và mô tả chức năng của use-case đó.

No	ID UC	UC	Ý nghĩa	Mô tả chức năng
1	UC01	Quản lý tài khoản	Actors có thể thực hiện được các tính năng khi làm việc với hệ thống bao gồm việc phải có tài khoản mới được truy cập vào hệ thống.	- Đăng ký. - Đăng nhập. - Đăng xuất.
2	UC02	Quản lý sản phẩm	Admin có thể quản lý thông tin sản phẩm (như xem, thêm, xóa, cập nhật)	- Thêm sản phẩm - Xóa sản phẩm - Cập nhật sản phẩm - Xem thông tin sản phẩm
3	UC03	Quản lý khách hàng	Admin có quyền quản lý khách hàng (thêm, xóa, cập nhật).	- Xem - Xóa - Cập nhật - Tìm kiếm
4	UC04	Quản lý đơn hàng	Người dùng có thể xem và theo dõi đơn hàng của mình. Admin có thể xóa, sửa thông tin đơn hàng	- Xem - Thêm - Xóa - Sửa

5	UC05	Quản lý hóa đơn	Giúp hệ thống quản lý tốt những hóa đơn đã được đặt/ thanh toán một cách hiệu quả	<ul style="list-style-type: none"> <li>- Xóa hóa đơn chưa thanh toán</li> <li>- Xem tình trạng hóa đơn</li> <li>- Thêm hóa đơn</li> <li>- Tìm kiếm</li> </ul>
---	------	-----------------	---	---

Bảng 3.2 Danh sách các actors và mô tả của chúng

#### 4. Đặc tả use-case

Các thành phần	Mô tả
Use Case Name	Tên của use-case, bắt đầu bằng một động từ. Ví dụ: Quản lý tài khoản.
Use Case ID	Định danh của use-case (ID: Identification). Ví dụ: UC01
Description	Mô tả ngắn gọn về use-case. Mô tả này nằm trong khoảng từ một dòng đến một đoạn văn mô tả mục đích và cách sử dụng nó
Stereotype & Package	Mô tả khuôn mẫu và gói mà trường hợp sử dụng này thuộc về. Đây là thông tin tùy chọn và có thể không phải lúc nào cũng được ghi lại, mặc dù nó sẽ được nhập dễ dàng trong một công cụ mô hình hóa
Actors	Danh sách các actor có liên quan đến use-case
Preconditions	Điều kiện tiên quyết là những điều kiện cần được thỏa mãn trước khi thực hiện được mô tả khi use-case có thể bắt đầu
Postconditions	Điều kiện sau là điều kiện phải được đáp ứng khi kết thúc use-case
Use Case Relationships	Relationships của use-case (gồm 3 loại: Include, Extend và Inherited), các actor có quan hệ giữa use-case với use-case

	<p>Luồng thành công chính, trường hợp mọi bước trong use-case đều được thực hiện thành công. Ví dụ như:</p> <ol style="list-style-type: none"> <li>1. &lt;mô tả&gt;</li> <li>2. &lt;mô tả&gt; (A1, E1, E2)</li> <li>3. &lt;mô tả&gt; (A2, E3)</li> </ol> <p><b>&lt;&lt;include&gt;&gt; &lt;mô tả của use-case included&gt;</b></p> <p><b>&lt;&lt;extends&gt;&gt; &lt;mô tả của use-case extended&gt;</b></p>
Basic Flow	A1 – Các mô tả tùy chọn ở đây là các luồng thay thế trong các điều kiện được chỉ định trong các bước trong basic flow
Exceptions	E1 – Các mô tả tùy chọn ở đây chỉ định các hành động được thực hiện trong các điều kiện ngoại lệ gấp phải trong basic flow của ca sử dụng. Về mặt kỹ thuật, điều này có thể đại diện cho các hành động để được thực hiện trong trường hợp có lỗi
Constraints	Các ràng buộc hoặc giới hạn đặc biệt đã được lập thành văn bản có liên quan đến việc sử dụng trường hợp
User Interface Specifications	Số và tên của thông số kỹ thuật giao diện người dùng liên quan đến use-case, bao gồm cả thông số kỹ thuật màn hình web, nếu như có sẵn. Lưu ý rằng đây không phải là thiết kế giao diện người dùng mà chỉ đơn giản là tham chiếu đến các màn hình/biểu mẫu có khả năng sẽ được tác nhân sử dụng khi giao tiếp với hệ thống
Metrics (Complexity)	Bất cứ thứ gì cần đo lường có liên quan đến các use-case sẽ được đưa vào đây – ví dụ: độ phức tạp của use-case: đơn giản/trung bình/phức tạp
Priority	Tầm quan trọng của chức năng được mô tả bởi use-case: cao/trung bình/thấp
Status	Trạng thái đầy đủ của tài liệu của use-case: ban đầu/chính/cuối cùng. Cái này sẽ cho biết mức độ phát triển của use-case

#### 4.1. Quản lý tài khoản

##### Đăng ký

Các thành phần	Mô tả
Use Case Name	Đăng ký.
Description	Cho phép khách hàng đăng ký để trở thành thành viên của ứng dụng
Actors	Guest
Preconditions	Thiết bị của actor đã được kết nối Internet khi đăng ký. Username và Email chưa tồn tại chúa cơ sở dữ liệu
Postconditions	Tạo tài khoản mới cho người dùng nếu đăng ký thành công, thông báo không đăng ký được nếu xảy ra lỗi.
Basic Flow	<ol style="list-style-type: none"> <li>Truy cập vào trang web, sau đó chọn đăng nhập.</li> <li>Chọn đăng ký tài khoản và hiển thị màn hình đăng ký.</li> <li>Người dùng nhập các thông tin (Username, email, password) để đăng ký tài khoản.</li> <li>Hệ thống sẽ tạo tài khoản cho người dùng, lưu vào cơ sở dữ liệu, thông báo đã đăng ký thành công. Đưa người dùng về lại trang chủ.</li> </ol>
Alternative Flow	Nếu tên đăng nhập đã tồn tại thì hệ thống đưa ra thông báo và yêu cầu người dùng lựa chọn lại tên đăng nhập khác
Constraints	Tác nhân đăng ký không được bỏ trống các trường dữ liệu Username/Email / Password và ConfirmPassword

Bảng XX Bảng đặc tả use-case đăng ký

##### Đăng nhập

Các thành phần	Mô tả
Use Case Name	Đăng nhập.

Description	Các tác nhân muốn đăng nhập vào hệ thống TOPSHIU
Actors	Admin /Customer/
Preconditions	Thiết bị của actor đã được kết nối Internet khi đăng nhập. Thông tin tài khoản phải được đăng ký thành công trước đó.Nếu chưa thì phải đăng kí tài khoản
Postconditions	Actor đăng nhập thành công. Hệ thống ghi nhận actor đã đăng nhập. Actor có thể sử dụng các tính năng trên hệ thống
Basic Flow	1. Người dùng chọn tính năng đăng nhập trên giao diện. 2. Hệ thống hiển thị màn hình đăng nhập 3. Người dùng nhập thông tin Email và password để đăng nhập tài khoản. 4. Hệ thống xác thực người dùng qua tên đăng nhập và mật khẩu, hiển thị màn hình chính với các chức năng tương ứng.
Constraints	Tác nhân đăng nhập không được bỏ trống các trường dữ liệu Email và Password

Bảng 3.3 Bảng đặc tả use-case đăng nhập

### Đăng xuất

Các thành phần	Mô tả
Use Case Name	Đăng xuất.
Description	Các tác nhân muốn đăng xuất khỏi hệ thống
Actors	Guest
Preconditions	Thiết bị của actor đã được kết nối Internet

	Trước đó, người dùng phải đăng nhập vào hệ thống
Postconditions	Trở về Trang chủ
Basic Flow	<ol style="list-style-type: none"> <li>1. Người dùng chọn tính năng Đăng xuất.</li> <li>2. Người dùng nhấn đăng xuất.</li> <li>3. Hệ thống đăng xuất khỏi ứng dụng với tên người dùng đó. Trở về màn hình Trang chủ.</li> </ol>

### 4.3. Quản lý khách hàng

Các thành phần	Mô tả
Use Case Name	Quản lý khách hàng.
Description	Admin có thể xem danh sách khách hàng, xem thông tin chi tiết của từng khách hàng và có thể thêm, xóa, cập nhật (đối với admin) và tìm kiếm thông tin khách hàng theo nhu cầu
Actors	Admin
Preconditions	Đã đăng nhập thành công vào hệ thống và đi đến trang dashboard
Postconditions	Thực hiện thành công sẽ hiện thị thông tin hay lưu thông tin mới. Nếu không thì trạng thái không thay đổi
Basic Flow	<ol style="list-style-type: none"> <li>1. Đến trang quản lý Dashboard, chọn quản lý khách hàng.</li> <li>2. Hiển thị màn hình danh sách thông tin khách hàng.</li> <li>3. Có thể xem thông tin chi tiết, tìm kiếm thông tin hoặc chọn khách hàng cần cập nhật hoặc xóa. Có thể chọn thêm mới khách hàng</li> <li>4. Lưu thông tin của khách hàng khi thao tác hợp lệ</li> </ol>

Bảng 3.4 Bảng đặc tả use-case quản lý khách hàng.

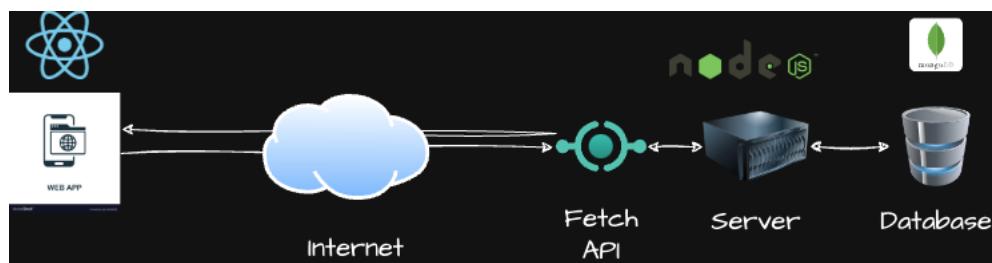
#### 4.4. Quản lý hóa đơn

Các thành phần	Mô tả
Use Case Name	Quản lý hóa đơn
Description	Admin mới có thể xem danh sách hóa đơn, xem thông tin chi tiết của từng hóa đơn và có thể thêm, xóa.
Actors	Admin
Preconditions	Đã đăng nhập thành công vào hệ thống và đi đến trang dashboard
Postconditions	Thực hiện thành công sẽ hiển thị thông tin hay lưu thông tin mới. Nếu không thì trạng thái không thay đổi
Basic Flow	<ol style="list-style-type: none"> <li>1. Đi đến trang quản lý Dashboard, chọn quản lý hóa đơn.</li> <li>2. Hiển thị màn hình danh sách thông tin hóa đơn</li> <li>3. Có thể xem thông tin chi tiết, tìm kiếm thông tin hoặc chọn hóa đơn cần cập nhật hoặc xóa. Có thể chọn thêm mới hóa đơn</li> <li>4. Lưu thông tin của hóa đơn khi thao tác hợp lệ</li> </ol>

Bảng xx Bảng đặc tả use-case quản lý hóa đơn.

### Chương 4. THIẾT KẾ HỆ THỐNG

#### 4.1.1. Kiến trúc hệ thống



Hình 9: Kiến trúc hệ thống

Về kiến trúc hệ thống, nhóm chúng em xây dựng theo dạng Web API và chia thành 2 phần là Front End và Back End. Chúng sẽ giao tiếp với nhau thông qua API và sử dụng Fetch API.

#### 4.1.2. Mô tả các thành phần

Đối với phần Front End, chúng tôi đã lựa chọn ReactJS để xây dựng giao diện người dùng. ReactJS giúp chúng tôi tiết kiệm thời gian trong quá trình phát triển giao diện và xử lý các tính năng so với việc sử dụng code thuần. Dữ liệu chủ yếu được lấy từ API được cung cấp bởi Back End.

Chúng em thực hiện các yêu cầu mạng thông qua Fetch API để gửi các yêu cầu HTTP bao gồm GET, POST, PUT, DELETE. Điều này giúp chúng tôi tương tác hiệu quả với Back End và nhận dữ liệu cần thiết để hiển thị trên giao diện người dùng.

Chúng em quyết định sử dụng Fetch API thay vì Axios để thực hiện các yêu cầu mạng. Dưới đây là một ví dụ về cách chúng tôi sử dụng **Fetch API** để thực hiện một yêu cầu GET:

```
// Thực hiện một yêu cầu GET sử dụng Fetch API
const fetchData = async () => {
  try {
    const response = await fetch('https://api.example.com/data');
    if (!response.ok) {
      throw new Error('Network response was not ok');
    }

    const data = await response.json();
    console.log(data);
    // Xử lý dữ liệu ở đây (cập nhật state, hiển thị trên giao diện, ...)
  } catch (error) {
    console.error('Error:', error.message);
    // Xử lý lỗi ở đây (hiển thị thông báo lỗi, ...)
  }
};

// Gọi fetchData khi component mount
useEffect(() => {
  fetchData();
}, []);
```

Chúng em tin rằng việc sử dụng Fetch API sẽ giúp chúng em duy trì mã nguồn dễ đọc và hiệu quả trong quá trình phát triển dự án.

Backend của đồ án của chúng em được xây dựng trên nền tảng NodeJS sử dụng framework ExpressJS để tạo ra một máy chủ (Server) mạnh mẽ. Chức năng chính của Server là nhận các yêu cầu từ phía Frontend và cung cấp phản hồi thông tin tương ứng. Để thực hiện điều này, chúng em đã xây dựng các API theo kiến trúc Fetch API, chủ yếu sử dụng 4 phương thức là GET, POST, PUT và DELETE.

- **Model:** Model đóng vai trò như một mẫu dữ liệu, chịu trách nhiệm thực hiện các thao tác liên quan đến cơ sở dữ liệu. Cụ thể, nó chứa các hàm và phương thức thực hiện các truy vấn trực tiếp vào cơ sở dữ liệu. Controller sẽ sử dụng các hàm và phương thức này để lấy dữ liệu và truyền nó đến các yêu cầu từ phía client.
- **Controller:** Controller nhận yêu cầu từ client và có trách nhiệm xử lý chúng. Nó tương tác với các model để lấy dữ liệu cần thiết và sau đó gửi dữ liệu này đến client. Controller đóng vai trò như một cầu nối giữa model và router, quản lý luồng thông tin trong hệ thống.
- **Router:** Router định tuyến các yêu cầu từ client đến các hàm xử lý tương ứng trong controller. Nó giúp định danh các URL và hành động tương ứng, định rõ cách các yêu cầu sẽ được xử lý.
- **Middleware:** Middleware là một thành phần quan trọng trong ExpressJS, chúng giúp lọc và kiểm soát các yêu cầu khi chúng đi qua server. Middleware có thể thực hiện các tác vụ như xác thực người dùng, kiểm soát truy cập, và các xử lý trung gian khác để bảo vệ hệ thống khỏi các yêu cầu không hợp lệ.
- **Database:** Cơ sở dữ liệu đóng vai trò lưu trữ, truy cập và quản lý dữ liệu. Chúng em sử dụng nó để lưu trữ thông tin từ client, cũng như để cung cấp dữ liệu khi có yêu cầu từ phía Frontend. Các thao tác truy vấn đến cơ sở dữ liệu được thực hiện thông qua model.

Qua cấu trúc này, chúng em đảm bảo sự tổ chức và quản lý chặt chẽ trong quá trình xử lý yêu cầu và dữ liệu, giúp tối ưu hóa hiệu suất và duy trì mã nguồn dễ bảo trì.

Ngoài những đặc điểm cơ bản đã mô tả, sự linh hoạt và mạnh mẽ của ExpressJS còn nhiều ưu điểm và tính năng khác giúp nó trở thành một lựa chọn phổ biến trong việc xây dựng Backend cho ứng dụng web và API. Dưới đây là một số điểm mở rộng về ExpressJS:

- **Middleware Rich:** ExpressJS cung cấp một loạt các middleware tích hợp sẵn và rất dễ mở rộng. Các middleware này giúp thực hiện nhiều nhiệm vụ khác nhau như xác thực, ghi log, nén dữ liệu, và nhiều tác vụ khác mà không cần phải triển khai từ đầu.
- **Template Engine:** Mặc dù ExpressJS không có một template engine mặc định như các framework khác, nhưng nó hỗ trợ tích hợp dễ dàng với nhiều template engines phổ biến như EJS, Pug (Jade), Handlebars, giúp tạo ra các

trang HTML động.

- **Phát triển API:** ExpressJS thường được sử dụng để xây dựng các API RESTful, nhưng nó cũng có thể được sử dụng để phát triển API GraphQL. Sự đa dạng trong việc xử lý yêu cầu và phản hồi giúp đơn giản hóa quá trình phát triển API.
- **Sự Mở Rộng:** ExpressJS là một framework nhẹ và không có quy định cứng nhắc về cấu trúc dự án. Điều này giúp các nhóm phát triển linh hoạt khi xây dựng ứng dụng, từ việc chọn cơ sở dữ liệu (MongoDB, MySQL, PostgreSQL) cho đến cách tổ chức cấu trúc dự án.

Với những đặc điểm này, ExpressJS không chỉ là một framework chọn lựa phổ biến mà còn là một công cụ mạnh mẽ giúp đội phát triển xây dựng các ứng dụng web và API hiệu quả và dễ bảo trì.

Đối với mô hình của chúng em, chúng em đã chọn **MongoDB** làm hệ quản lý cơ sở dữ liệu cho Backend. MongoDB là một hệ quản lý cơ sở dữ liệu phi quan hệ (NoSQL), sử dụng kiểu dữ liệu JSON-like được gọi là BSON (Binary JSON) để lưu trữ dữ liệu. Dưới đây là một số điểm mở rộng về việc sử dụng MongoDB trong mô hình của chúng em:

- **Linh hồn Hệ Cơ Sở Dữ Liệu NoSQL:** MongoDB là một hệ quản lý cơ sở dữ liệu NoSQL, điều này nghĩa là nó không yêu cầu sự cố định cứng về cấu trúc dữ liệu và cho phép lưu trữ dữ liệu một cách linh động. Điều này rất phù hợp với mô hình dự án có thể phải đổi mặt với sự thay đổi và mở rộng liên tục.
- **Dữ Liệu JSON-Like và BSON:** MongoDB lưu trữ dữ liệu dưới dạng BSON, một định dạng tương tự JSON. Điều này giúp dễ dàng tích hợp với các ngôn ngữ lập trình và đảm bảo tính linh hoạt trong việc thêm, sửa, và xóa dữ liệu.
- **Khả Năng Mở Rộng và Replica Set:** MongoDB hỗ trợ khả năng mở rộng ngang, cho phép bạn mở rộng cơ sở dữ liệu bằng cách thêm các node vào cluster. Đồng thời, MongoDB cung cấp khả năng sao chép (replica set) để tăng tính sẵn sàng và đảm bảo an toàn cho dữ liệu.
- **Tích Hợp Tốt với JavaScript và NodeJS:** Vì MongoDB sử dụng BSON (tương tự JSON) làm định dạng lưu trữ dữ liệu, nó tích hợp rất tốt với JavaScript và NodeJS. Điều này giúp giảm thiểu sự phức tạp trong quá trình làm việc với dữ liệu trong cả Frontend và Backend.
- **Query Linh Hoạt:** MongoDB cung cấp một ngôn ngữ truy vấn mạnh mẽ cho phép bạn truy xuất dữ liệu một cách linh hoạt. Các truy vấn có thể được viết bằng cách sử dụng JavaScript, và nó hỗ trợ nhiều toán tử và phương thức để xử lý dữ liệu.

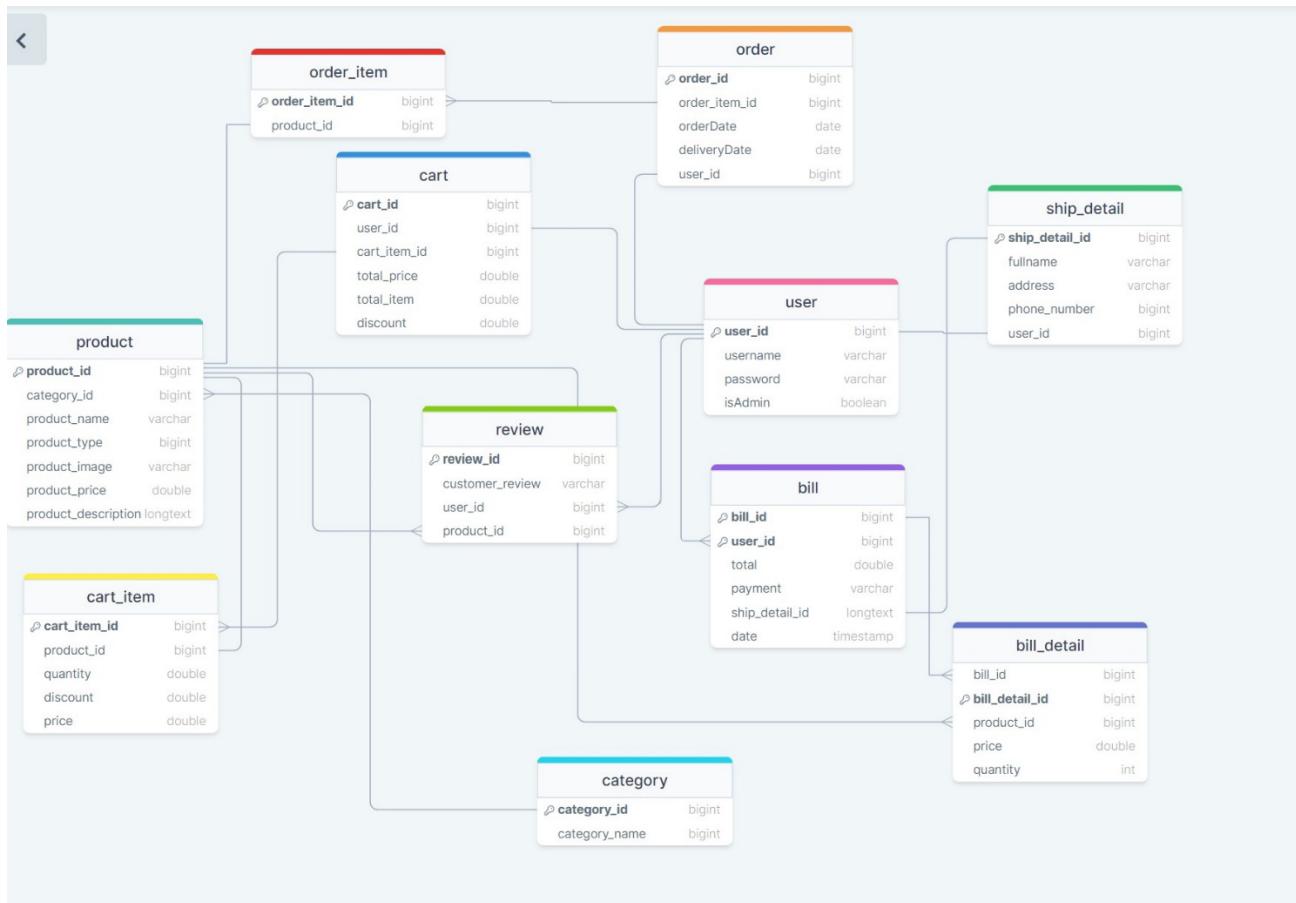
- **Document-Oriented Database:** MongoDB là một cơ sở dữ liệu dạng document-oriented, nghĩa là dữ liệu được tổ chức trong các tài liệu (documents) JSON-like. Điều này làm cho việc lưu trữ và truy xuất dữ liệu trở nên thuận tiện, đặc biệt khi có mối quan hệ phức tạp giữa các dữ liệu.

Tổng cộng, sự kết hợp giữa ExpressJS và MongoDB trong mô hình của chúng em tạo ra một hệ thống có khả năng mở rộng, linh hoạt, và dễ quản lý. Nó giúp đội phát triển xây dựng ứng dụng web và API với khả năng mở rộng tốt và xử lý hiệu quả cả trong môi trường phát triển và triển khai.

## Chương 5. THIẾT KẾ CƠ SỞ DỮ LIỆU

Để phát triển phần Back End của dự án, nhóm chúng tôi đã chọn sử dụng Nodejs và cơ sở dữ liệu sử dụng MongoDB. MongoDB là một hệ quản trị cơ sở dữ liệu nguồn mở, thuộc loại NoSQL, được sử dụng rộng rãi bởi hàng triệu người dùng. Đặc điểm của MongoDB là hướng tới việc lưu trữ dữ liệu theo cấu trúc tài liệu, sử dụng định dạng JSON thay vì lưu trữ dữ liệu dưới dạng bảng như cơ sở dữ liệu quan hệ, giúp tối ưu hóa truy vấn và tăng tốc độ xử lý dữ liệu.

**Flowchart:** Theo đề xuất ban đầu



Hình xx Biểu đồ cơ sở dữ liệu để xuất.

**Product:** chứa các thông tin sản phẩm

Tên trường dữ liệu	Loại dữ liệu	Chú thích	Ràng buộc
Product_id	Number	Mã sản phẩm	Bắt buộc có dữ liệu
Product_name	String	Tên sản phẩm	Bắt buộc có dữ liệu
Product_image	String	Hình ảnh dưới dạng link	Bắt buộc có dữ liệu
Product_type	String	Loại sản phẩm	Bắt buộc có dữ liệu
Old_price	Number	Giá cũ của sản phẩm	Bắt buộc có dữ liệu

New_price	Number	Giá mới của sản phẩm	Bắt buộc có dữ liệu
Product_Description	Boolean	Mô tả sản phẩm	Bắt buộc có dữ liệu

Bảng xx Model sản phẩm

User:

Tên trường dữ liệu	Loại dữ liệu	Chú thích	Ràng buộc
User_Id	Number	ID tài khoản	Bắt buộc có dữ liệu
email	String	Tên email	Bắt buộc có dữ liệu
password	String	Mật khẩu của tài khoản	Bắt buộc có dữ liệu

Bảng xx Model người dùng

Cart: Chứa các thông tin của đối tượng sản phẩm

Tên trường dữ liệu	Loại dữ liệu	Chú thích	Ràng buộc
User_ID	Number	ID tài khoản	Bắt buộc có dữ liệu
Cart_item_ID	Number	ID của mặt hàng trong giỏ hàng	Bắt buộc có dữ liệu
Total_Item	Number	Tổng số lượng Item	Bắt buộc có dữ liệu
Total_price	Number	Tổng giá tiền	Bắt buộc có dữ liệu

Bảng xx Model Giỏ hàng

Cart Item: lưu trữ các sản phẩm trong giỏ hàng

Tên trường dữ liệu	Loại dữ liệu	Chú thích	Ràng buộc
Cart_item_ID	Number	ID của mặt hàng trong giỏ hàng	Bắt buộc có dữ liệu
Product_price	Number	Giá của sản phẩm	Bắt buộc có dữ liệu

Bảng xx Model sản phẩm giỏ hàng

**Order:** lưu thông tin đơn hàng của người dùng

Tên trường dữ liệu	Loại dữ liệu	Chú thích	Ràng buộc
Order_ID	Number	Id đơn hàng	Bắt buộc có dữ liệu
Order_Date	String	Ngày đặt hàng	Bắt buộc có dữ liệu
User_ID	Number	ID người dùng	Bắt buộc có dữ liệu

Bảng xx Model Order

**Bill:** lưu trữ các thông tin của hóa đơn

Tên trường dữ liệu	Loại dữ liệu	Chú thích	Ràng buộc
Bill_ID	Number	ID hóa đơn	Bắt buộc có dữ liệu
User_ID	Number	Id người dùng	Bắt buộc có dữ liệu
Total_price	Number	Tổng giá tiền	Bắt buộc có dữ liệu
Payment	String	Phương thức thanh toán	Bắt buộc có dữ liệu
Address	Date	Địa chỉ của người nhận	Bắt buộc có dữ liệu

Bảng xx Model đơn hàng.

## Chương 6. THIẾT KẾ GIAO DIỆN VÀ XỬ LÝ

### 6.1. Danh sách các màn hình và sơ đồ liên kết các màn

#### 6.1.1. Sơ đồ liên kết các màn hình

#### 6.1.2. Danh sách các màn hình

Màn hình	Mô tả
Login	Màn hình đăng nhập trang web

<b>Register</b>	Màn hình đăng ký tài khoản
<b>Home</b>	Màn hình trang chủ của trang web
<b>Products</b>	Màn hình chứa danh sách các sản phẩm của trang web
<b>News</b>	Màn hình chứa các tin tức về điện thoại Apple và phụ kiện điện thoại.
<b>DetailProduct</b>	Màn hình chi tiết sản phẩm
<b>Promotion</b>	Màn hình danh sách khuyến mãi
<b>Cart</b>	Màn hình giỏ hàng
<b>Payment</b>	Màn hình thanh toán
<b>Orders Detail</b>	Màn hình chi tiết đơn hàng
<b>Address</b>	Màn hình địa chỉ nhận hàng của khách hàng

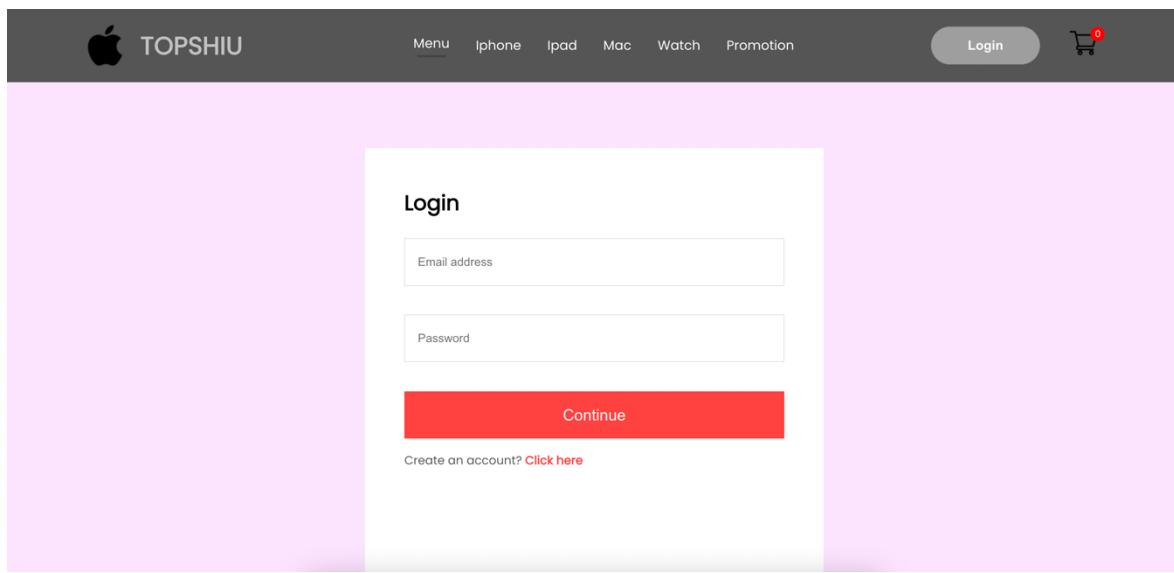
Bảng 78: Danh sách các màn hình

Màn hình	Mô tả
<b>Quản lý sản phẩm</b>	Màn hình quản lý của admin
<b>Thêm sản phẩm</b>	Màn hình chứa các trường để thêm sản phẩm mới

Bảng 78: Danh sách các quản lý

## 6.2. Chi tiết màn hình.

### 6.2.1. Màn hình Login



Hình 6.2.1: Màn hình Login

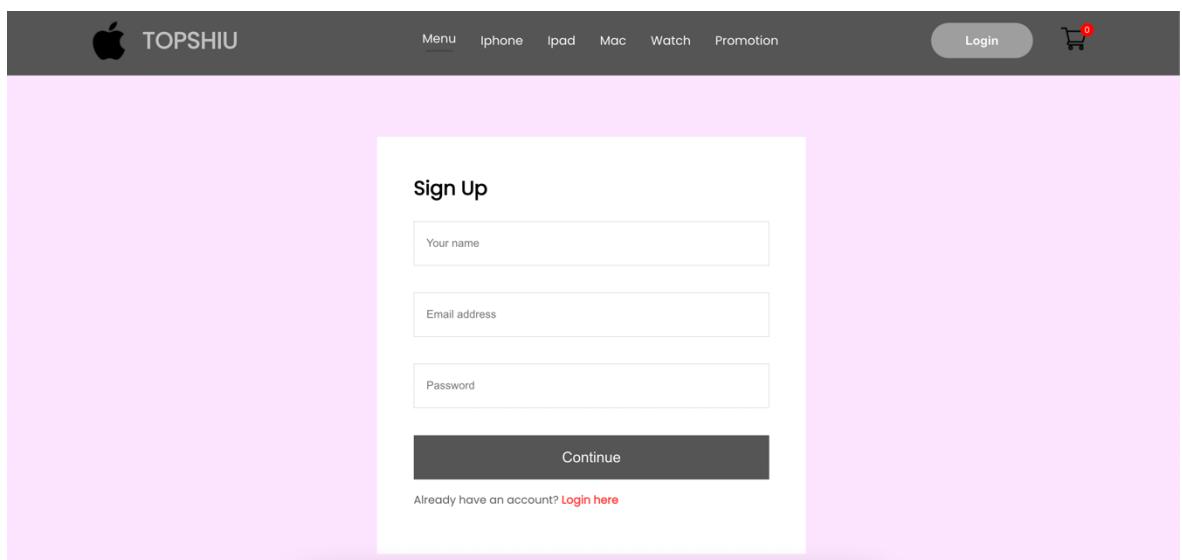
S T T	Tên đối tượng	Kiểu	Ý tưởng	Điều kiện
1	Email	Text	Trường nhập email đăng nhập đã đăng kí	Bắt buộc phải nhập nội dung và đúng tiêu chuẩn email
2	Password	Password	Trường nhập mật khẩu theo email đã đăng kí	Bắt buộc phải nhập nội dung và đúng mật khẩu
3	Login	Button	Đăng nhập với tài khoản được nhập	Click vào button hoặc nhấn enter
4	Signup	Link	Dẫn đến trang đăng kí nếu chưa có tài khoản	Click vào chữ “Click here”

Bảng 6.2.1: Bảng mô tả các đối tượng trên màn hình Login

S T T	Biến cố	Tên hàm	Mô tả xử lý	Ý nghĩa
1	Click “Đăng nhập”	handle Submit (e)	Khi click vào nút này thì sẽ điều hướng đến trang chủ nếu đăng nhập thành công	Đăng nhập
2	Click “đăng kí”	handle Res(e)	Khi click vào nút này thì sẽ điều hướng đến màn hình đăng kí	Đăng kí tài khoản

Bảng 6.2.1: Danh sách các biến cố và xử lý trên màn hình Login

### 6.2.2. Màn hình Register



Hình 6.2.2: Màn hình Register

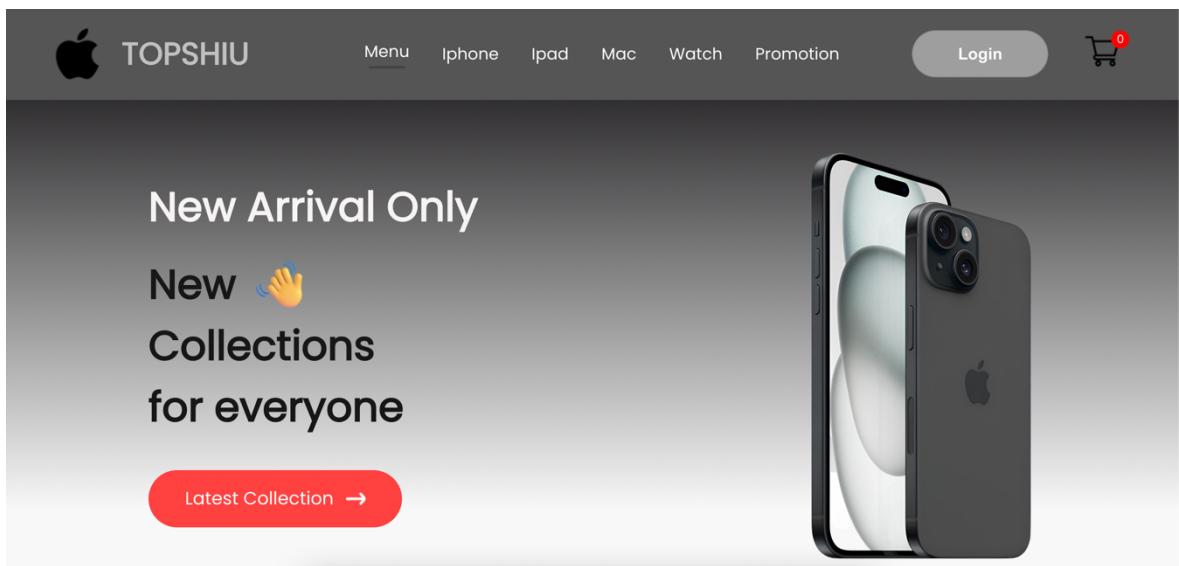
S T T	Tên đối tượng	Kiểu	Ý tưởng	Điều kiện
1	name	Text	Trường nhập tên của người dùng	Bắt buộc phải nhập nội dung
2	email	Text	Trường nhập email của người dùng	Bắt buộc phải nhập email hợp lệ
3	password	Password	Trường nhập password người dùng	Bắt buộc phải nhập mật khẩu từ 8 ký tự trở lên, có kèm kí tự in hoa và kí tự đặc biệt
4	Đăng kí	Button	Mục điều hướng đến màn hình home	

Bảng 6.2.2: Bảng mô tả các đối tượng trên màn hình Register

S T T	Biến cód	Tên hàm	Mô tả xử lý	Ý nghĩa
1	Click “Đăng kí”	handle Submit (e)	Khi click vào nút này thì sẽ điều hướng đến trang chủ nếu đăng kí thành công	Đăng kí tài khoản, hiển thị màn hình Home
2	Click “Quay vềđăng nhập”		Khi click sẽ quay về trang đăng nhập	Hiển thị màn hình Login

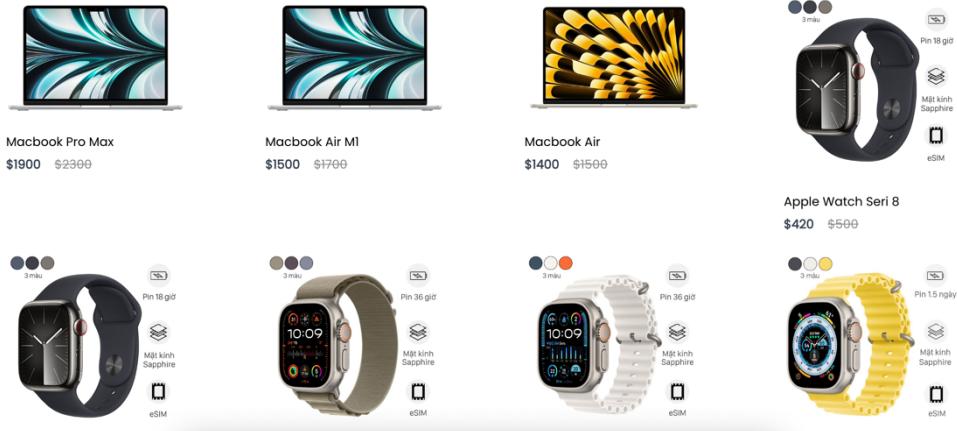
Bảng 6.2.2: Danh sách các biến cód và xử lý trên màn hình Register

### 6.2.3. Màn hình Home

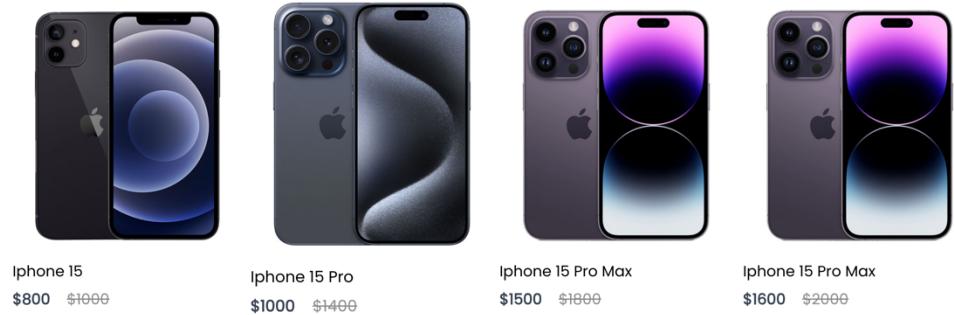


Hình 6.2.3: Màn hình Home

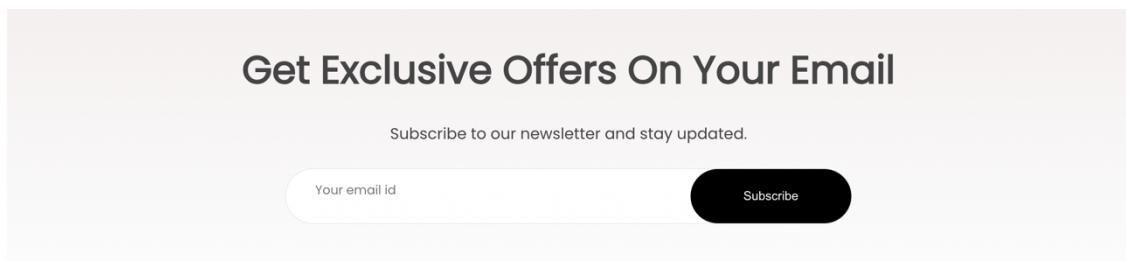
## NEW COLLECTIONS



## POPULAR



Hình 6.2.3: Màn hình Home



Company Products Offices About Contact



*Hình 6.2.3: Màn hình Home*

**Bảng mô tả các đối tượng trên màn hình**

S T T	Tên đối tượng	Kiểu	Ý tưởng	Điều kiện
1	Logo	Image	Quay lại trang home	
2	Iphone, Ipad, Mac, Watch	Button	Xem danh sách các sản phẩm theo mục	
3	Promotion	Button	Tin tức, khuyến mãi	
4	Giỏ hàng	Icon button	Xem giỏ hàng	
5	User	Icon button	Đăng nhập, đăng xuất, xem thông tin user	

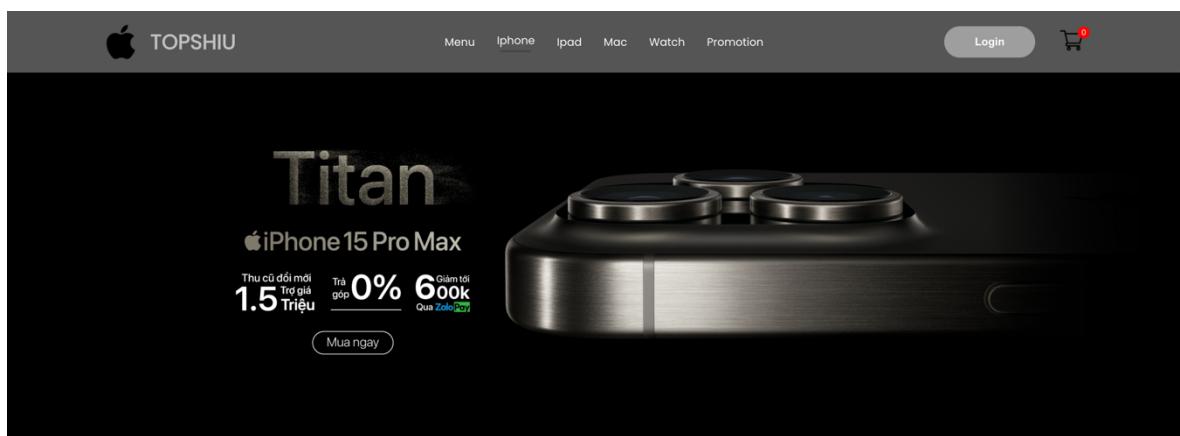
*Bảng 6.2.3:: Bảng mô tả các đối tượng trên màn hình Home*

S T T	Biến cố	Tên hàm	Mô tả xử lý	Ý nghĩa

1	Click vào Logo		Hiển thị màn hình introduction	
2	Click vào các chữ trên menu		Hiển thị danh sách tương ứng	
3	Click vào icon tìm kiếm		Hiển thị ô tìm kiếm	
4	Click vào button giỏ hàng		Vào trang giỏ hàng	
5	Click vào icon user		Vào trang đăng nhập nếu chưa đăng nhập Hiện menu nếu đăng nhập rồi	

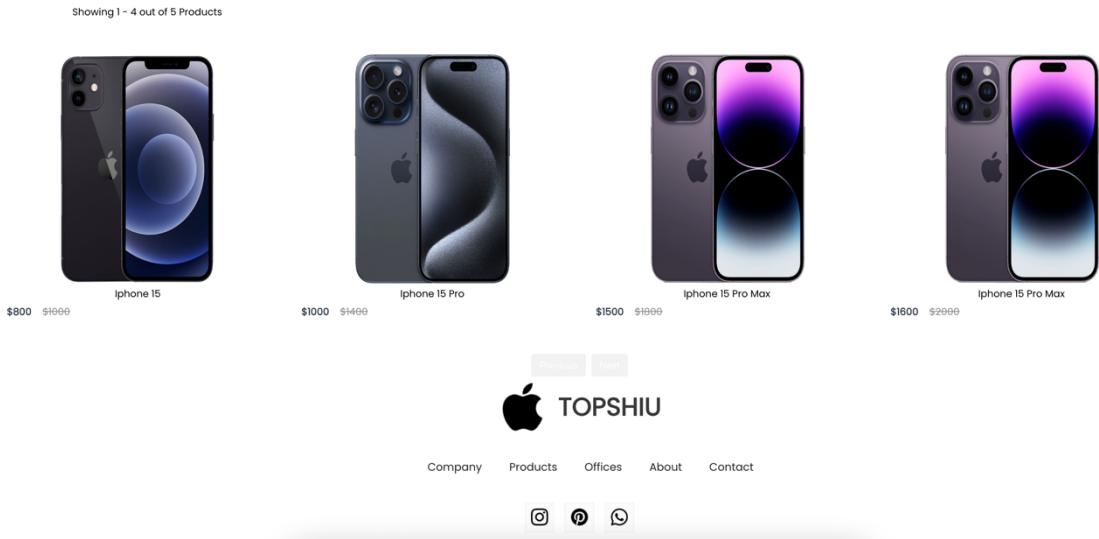
Bảng 6.2.3: Danh sách các biến cố và xử lý trên màn hình Home

#### 6.2.4. Màn hình Products



Showing 1 - 4 out of 5 Products

Hình 6.2.4: Màn hình Products



Hình 6.2.4: Màn hình Products

S T T	Tên đối tượng	Kiểu	Ý tưởng	Điều kiện
1	Breadcrumb	div	Hiển thị breadcrumb danh sách sản phẩm	
2	Title	H1	Hiển thi loại sản phẩm	
3	Category	Div	Trường hiển thị phân loại con của sản phẩm	
4	Banner	Image	Hiển thị quảng cáo	
5	Product card	Icon button	Hiển thị hình ảnh, tên và giá của sản phẩm	

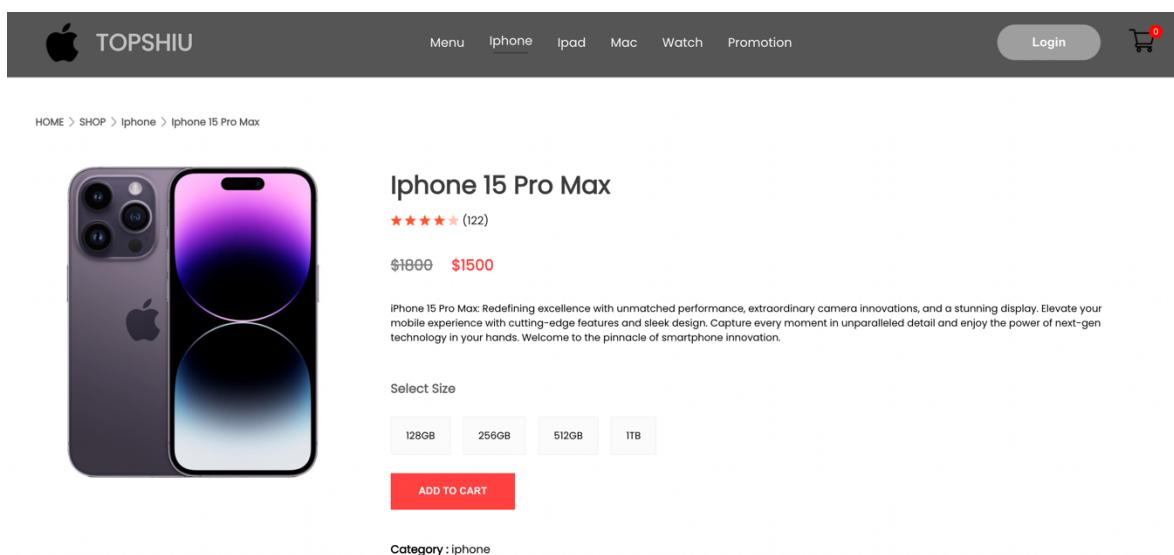
Bảng 6.2.4: Bảng mô tả các đối tượng trên màn hình Products

S T T	Biến cố	Tên hàm	Mô tả xử lý	Ý nghĩa

1	Click vào Breadcrumb	navigate	Tuỳ vào phần được click mà điều hướng theo	Di chuyển nhanh chóng giữa các màn hình liên quan
2	Click và categort	navigate	Điều hướng đến danh sách sản phẩm tiếp theo	Lọc sản phẩm
3	Click vào ProductCard	navigate	Điều hướng đến trang chi tiết sản phẩm	Di chuyển đến trang chi tiết sản phẩm mà người dùng quan tâm
4	Chọn mục select	navigate	Vào trang giỏ hàng	

Bảng 6.2.4: Danh sách các biến cố và xử lý trên màn hình Products

## 6.2.5. Màn hình DetailProduct



Hình 6.2.5: Màn hình DetailProduct

Description      Reviews (122)

An e-commerce website is an online platform that facilitates the buying and selling of products or services over the internet.  
E-commerce websites typically display products or services along with detailed descriptions, images, prices, and any available variations (e.g., sizes, colors).

### Related Products



*Hình 6.2.5: Màn hình DetailProduct*

Số thứ tự	Tên đối tượng	Kiểu	Ý tưởng	Điều kiện
1	Name	H1	Hiển thị tên sản phẩm	
2	Dungluong	Li	Dung lượng sản phẩm	
3	Ram	Li	Dung lượng ram	
4	Detail	Div	Hiển thị mô tả chi tiết	
5	Add	Icon button	Thêm sản phẩm vào giỏ hàng	
6	Rating	Tab	Hiển thị đánh giá theo 1-5 sao	

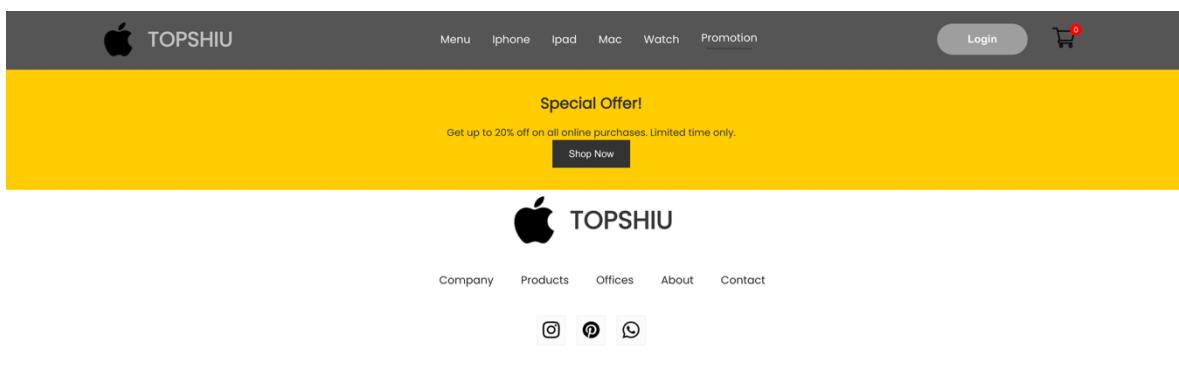
*Bảng 6.2.5: Bảng mô tả các đối tượng trên màn hình*

*Products*

S T T	Biến cố	Tên hàm	Mô tả xử lý	Ý nghĩa
1	Click vào dungluong	setLstDungluong	Chọn dung lượng	
2	Click vào ram	setLstRam	Chọn RAM	Người dùng tự ý chọn
3	Click vào Detail	setLstDetail	Hiển thị chi tiết thông số	Người dùng tự ý chọn
4	Click vào Add	setAdd	Vào trang giỏ hàng	Thêm sản phẩm muốn vào giỏ để thanh toán

Bảng 6.2.5: Danh sách các biến cố và xử lý trên màn hình Products

### 6.2.6. Màn hình Promotion



Hình 6.2.6: Màn hình Promotion

S T T	Tên đối tượng	Kiểu	Ý tưởng	Điều kiện
1	Promotion	Div	Hiển thị tiêu đề khuyến mãi	
2	Promotioncode	H1	Code khuyến mãi	

Bảng 6.2.6: Bảng mô tả các đối tượng trên màn hình

Products

### 6.2.7. Màn hình Cart

The screenshot shows the TOPSHIU website's cart page. At the top, there is a navigation bar with the logo, menu items (Menu, Iphone, Ipad, Mac, Watch, Promotion), a login button, and a shopping cart icon with a red notification badge. Below the navigation, the cart table lists one item: "Ipad Pro max" with a price of "\$1400", quantity "1", and a "Remove" link. Underneath the cart table, the "Cart Totals" section displays the subtotal (\$1400), shipping fee (Free), and total (\$1400). There is a field for entering a promo code and a "Submit" button. At the bottom, a red "PROCEED TO CHECKOUT" button is visible.

Hình 6.2.7: Màn hình Cart

S T T	Tên đối tượng	Kiểu	Ý tưởng	Điều kiện
1	Listproduct_cart	List	Danh sách các sản phẩm đã chọn	
2	Promotion	Textbox	Nhập code khuyến mãi	
3	Proceed	Button	Dẫn đến trang thanh toán	

*Bảng 6.2.7:: Bảng mô tả các đối tượng trên màn hình Giỏ hàng*

Số thứ tự	Biến cờ	Tên hàm	Mô tả xử lý	Ý nghĩa
1	Nhập mã code	Putcode	Nhập mã khuyến mãi	Giảm giá cho sản phẩm
2	Click vào Proceed	toProceed	Đến trang thanh toán	Đến trang thanh toán tiếp theo (địa chỉ, phương thức)

*Bảng 6.2.7: Danh sách các biến cờ và xử lý trên màn hình Giỏ hàng*

### 6.2.8. Màn hình Orders Detail

The screenshot shows the 'TOPSHIU' website's Order Details page. At the top, there is a navigation bar with links for Menu, Iphone, Ipad, Mac, Watch, Promotion, Login, and a shopping cart icon with a red notification. The main content area is divided into several sections:

- Payment Method:** A section where 'Cash on Delivery' is selected. Other options like 'Bank Transfer' and 'Momo' are also listed.
- Order Summary:** A table showing an ordered product: 'Ipad Pro max' quantity 'x1' at '\$1400'.
- Shipping Information:** Details for delivery to 'Phan Hieu Luon' at '214 đường số 11' in 'Binh Duong' with phone number '0931328505'.
- Total Amount:** '\$1400'.
- Payment Method:** Shows 'cashOnDelivery'.
- Confirm Order:** A green button at the bottom of the payment method section.

*Hình 6.2.8: Màn hình Order Details*

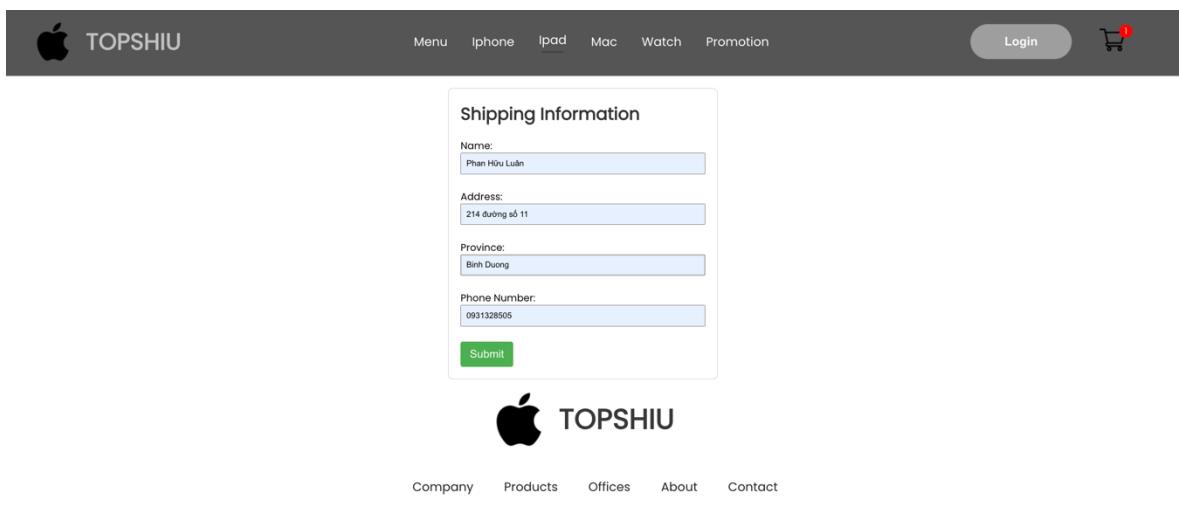
S T T	Tên đối tượng	Kiểu	Ý tưởng	Điều kiện
1	Ordersummary	Div	Tóm tắt lại đơn hàng	Đã nhập địa chỉ và chọn phương thức thanh toán
2	Payment method	Button	Chọn phương thức thanh toán	

Bảng 6.2.8: Bảng mô tả các đối tượng trên màn hình chi tiết đơn hàng

S T T	Biến cód	Tên hàm	Mô tả xử lý	Ý nghĩa
1	Select Payment	SelectPayment	Chọn các phương thức thanh toán	

Bảng 6.2.8: Danh sách các biến cód và xử lý trên màn hình chi tiết Giỏ hàng

### 6.2.9. Màn hình Address



Hình 6.2.8: Màn hình nhập Địa chỉ

S T T	Tên đối tượng	Kiểu	Ý tưởng	Điều kiện
1	Name	Textbox	Tên người nhận	
2	Address	Textbox	Địa chỉ nhà	
3	Province	Textbox	Tỉnh	
4	Phone Number	Textbox	Số điện thoại	Đúng theo format số điện thoại mới nhận input

Bảng 6.2.9: Bảng mô tả các đối tượng trên màn hình Address

S T T	Biến cõ	Tên hàm	Mô tả xử lý	Ý nghĩa
1	Click vào Lưu lại	saveaddress	Gửi yêu cầu cập nhật thông tin user	

Bảng 6.2.9: Danh sách các biến cõ và xử lý trên màn  
hình Address

## 6.2.10. Các màn hình Dashboard

### 6.2.10.1. Màn hình quản lý sản phẩm

The screenshot shows the 'SHOPPER Admin Panel' interface. On the left, there is a sidebar with two buttons: 'Add Product' (with a shopping cart icon) and 'Product List' (with a document icon). The main area is titled 'All products List' and displays a table of product information. The columns are: products, Title, Old price, New price, Category, Update, and Remove. The data in the table is as follows:

products	Title	Old price	New price	Category	Update	Remove
	Iphone 15	1000\$	800\$	iphone	<input checked="" type="checkbox"/>	X
	Iphone 15 Pro	1400\$	1000\$	iphone	<input checked="" type="checkbox"/>	X
	Iphone 15 Pro Max	1800\$	1500\$	iphone	<input checked="" type="checkbox"/>	X
	Iphone 15 Pro Max	2000\$	1600\$	iphone	<input checked="" type="checkbox"/>	X
	Iphone 15 Pro Max	2000\$	1600\$	iphone	<input checked="" type="checkbox"/>	X

Hình 6.2.1 Màn hình quản lí sản phẩm

Hình 6.2.1 Màn hình cập nhật sản phẩm

STT	Tên đối tượng	Kiểu	Ý tưởng	Điều kiện
1	Update	Link	Cập nhật các sản phẩm trong danh sách	

Bảng 6.2.1 Bảng mô tả các đối tượng trong quản lý sản phẩm.

STT	Biến cỏ	Tên hàm	Mô tả xử lý	Ý nghĩa
1	Click vào biểu tượng trong cột Update	UpdateProduct, editProduct.	Khi click vào biểu tượng này sẽ hiển thị một cửa sổ cho phép cập nhật thông tin sản phẩm	Cập nhật các sản phẩm trong danh sách

Bảng 6.2.1 Bảng mô danh sách và biến cỏ trên màn hình

### 6.2.10.2. Màn hình thêm sản phẩm

The screenshot shows the 'SHOPPER Admin Panel' interface. On the left, a sidebar has two buttons: 'Add Product' and 'Product List'. The main area is titled 'Add Product' and contains the following fields:

- Product title:** An input field with placeholder text 'Type here'.
- Product description:** A large text area with placeholder text 'Type here'.
- Price:** An input field with placeholder text 'Type here'.
- Offer Price:** An input field with placeholder text 'Type here'.
- Product category:** A dropdown menu currently set to 'None'.
- Product image:** A button labeled 'Upload' with a cloud icon.

At the bottom center is a blue 'ADD' button.

Hình XX Màn hình thêm sản phẩm

STT	Tên đối tượng	Kiểu	Ý tưởng	Điều kiện
1	Product title	Input	Nhập tên sản phẩm	
2	Product description	Input	Nhập mô tả sản phẩm	
3	Price	Input	Nhập giá cũ sản phẩm	Yêu cầu nhập đúng số liệu
4	Offer price	Input	Nhập giá mới sản phẩm	Yêu cầu nhập đúng số liệu
5	Product category	Select	Chọn loại sản phẩm	
6	Product image	Select	Tải lên ảnh sản phẩm	

7	Add	Button	Đăng tải sản phẩm lên front end	
---	-----	--------	---------------------------------	--

Bảng 6.2.1 Bảng mô tả các đối tượng của thêm sản phẩm

STT	Biến cõ	Tên hàm	Mô tả xử lý	Ý nghĩa
1	Click “ADD”	AddProduct	Khi click vào nút này, một hộp thoại được hiển thị trạng thái thêm sản phẩm	Đăng tải sản phẩm lên website theo các thông tin đã nhập.

Bảng 6.2.1 Bảng mô tả danh sách và biến cõ trên màn hình thêm sản phẩm.

## Chương 7. CÀI ĐẶT VÀ KIỂM THỦ - ĐỊNH HƯỚNG PHÁT TRIỂN

### 7.1. Các bước cài đặt, deploy đồ án.

Link source code: [https://github.com/huyf-51/full\\_project.git](https://github.com/huyf-51/full_project.git)

Link source code Front End: [https://github.com/huyf-51/full\\_project/tree/main/e-commerce-frontend-main](https://github.com/huyf-51/full_project/tree/main/e-commerce-frontend-main)

Link source code Admin portal: [https://github.com/huyf-51/full\\_project/tree/main/e-commerce-admin](https://github.com/huyf-51/full_project/tree/main/e-commerce-admin)

Link source code Back End: [https://github.com/huyf-51/full\\_project/tree/main/e-commerce-backend](https://github.com/huyf-51/full_project/tree/main/e-commerce-backend)

Các bước chạy dự án ở local:

+ Bước 1: Mở terminal, gõ lệnh

```
git clone https://github.com/huyf-51/full_project.git
cd <Thư mục>
```

+ Bước 2: Gõ lệnh **npm install** để cài đặt các node-modules cho dự án.

+ Bước 3: Gõ lệnh **npm start** để chạy dự án.

## **7.2. Kiểm thử:**

### **7.2.1. Performance Testing**

Kiểm tra và đánh giá hiệu suất của một ứng dụng hoặc hệ thống để đảm bảo rằng nó đáp ứng được yêu cầu về tải trọng, thời gian phản hồi, và khả năng chịu tải.

Công cụ: **JMeter**.

Các bước kiểm thử:

Bước 1: Cài đặt Jmeter

Bước 2: Tạo yêu cầu HTTP để mô phỏng các hành động trên website như đăng nhập, xem trang, mua hàng.

Bước 3: Chạy kịch bản kiểm thử và theo dõi các chỉ số hiệu suất, Thu thập dữ liệu về thời gian phản hồi, lỗi, throughput, và các chỉ số khác.

Bước 4: Phân tích kết quả: Tìm các vấn đề hiệu suất như tải trọng không chịu được, thời gian phản hồi lớn, lỗi, hoặc nghẽn.

Bước 5: Tối ưu hóa.

## **7.3. Ưu nhược điểm đồ án**

### **7.3.1. Ưu điểm**

- Các tính năng đơn giản, dễ sử dụng song vẫn đáp ứng được nhu cầu sử dụng thực tế.
- Số lượng sản phẩm đa dạng (mac, iphone, ipad, âm thanh) có tính tham khảo.
- Có áp dụng các phương thức bảo mật.
- Thực hiện xây dựng trang quản lý cửa hàng
- Có thực hiện Responsive trang web cho các thiết bị di động.
- Có giao diện Administrator để quản lý sản phẩm, người dùng.

### **7.3.2. Nhược điểm**

- Chưa có tính năng tìm kiếm sản phẩm chi tiết

- Bên phần Administrator tính năng lọc sản phẩm, quản lý số lượng sản phẩm chưa hoàn thiện.
- Một vài thiếu sót trong các chức năng như chưa định dạng sẵn ngày tháng ở một số nơi nhập liệu.
- Hiệu suất web chưa cao, một vài xử lý chưa được tối ưu.
- Data chưa đủ để thực hiện vẽ biểu đồ chart để đánh giá và định hướng trong kinh doanh

#### **7.4. Định hướng phát triển đồ án**

- Thực hiện hoàn thiện chức năng Tìm kiếm sản phẩm ở phần Front-End
- Thêm module quản lý sản phẩm chi tiết bao gồm các tính năng như lọc, thống kê số lượng sản phẩm.
- Phát triển thêm một số tính năng cho trang web như chức năng tìm kiếm sản phẩm nhờ model AI gợi ý sản phẩm.
- Tối ưu một số phần code nhằm cải thiện hiệu suất trang web.
- Xây dựng website đáp ứng trên mỗi thiết bị và kích thước khung hình khác nhau

### **Chương 8. KẾT LUẬN**

Website bán điện thoại và phụ kiện đã được nhóm làm việc xuyên suốt từ bước thiết kế ban đầu đến triển khai xây dựng và cuối cùng kiểm thử như một trang web chuyên nghiệp.

Trang web dễ sử dụng, thân thiện với người dùng. Điều này giúp người dùng có được một trải nghiệm người dùng tốt nhất và giúp khách hàng dễ dàng tìm kiếm và mua sắm các sản phẩm điện thoại.

Chức năng của trang web chưa được đầy đủ nhưng đáp ứng đủ nhu cầu của người dùng và người kinh doanh. Nhóm đã thực hiện tạo nên nhiều tính năng cơ bản song vẫn quan trọng tìm kiếm sản phẩm, sắp xếp theo giá, danh mục sản phẩm và xem thông tin chi tiết của mỗi sản phẩm. Điều này cho phép người dùng có thể tìm kiếm và so sánh các điện thoại dễ dàng để lựa chọn sản phẩm phù hợp với nhu cầu của họ.

Tiếp đó, nhóm cũng đã thực hiện tạo nên chức giỏ hàng và thanh toán tiện lợi.

Người dùng có thể thêm sản phẩm vào giỏ hàng, điều chỉnh số lượng và tiến hành thanh toán thông qua các phương thức thanh toán phổ biến Banking hay momo. Điều này giúp tăng tính tiện lợi và tránh các rào cản trong quá trình mua sắm trực tuyến.

Đi sâu vào backend, nhóm cũng đã thực hiện một công việc tốt trong việc quản

lý cơ sở dữ liệu sản phẩm. Các thông tin về sản phẩm, bao gồm hình ảnh, mô tả và giá cả, được tổ chức một cách cẩn thận và chính xác. Điều này đảm bảo rằng khách hàng có thể có được thông tin chi tiết và chính xác về sản phẩm trước khi quyết định mua hàng.

Tuy nhiên, vẫn còn một số khuyết điểm và thiếu sót mà nhóm cần thực hiện để nâng cao trải nghiệm người dùng.

Website bán điện thoại của nhóm đã thể hiện kết quả làm việc và sự cố gắng của từng thành viên khi thực hiện đồ án. Nhóm đã thành công trong việc thiết kế và xây dựng một giao diện đẹp đẽ và bắt mắt, đồng thời tích hợp các chức năng quan trọng và thiết kế quản lý cơ sở dữ liệu một cách tốt nhất. Mặc dù còn nhiều điểm cần khắc phục và sửa đổi cải tiến một số chức năng nhưng các thành viên trong nhóm nhìn chung đã mang lại một kết quả tốt và đáng mong đợi.