

**TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM  
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG  
KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN CUỐI KỲ  
MÔN PHÁT TRIỂN ỨNG DỤNG WEB VỚI NODEJS  
WEBSITE BÁN ĐIỆN THOẠI DI ĐỘNG**

*Người hướng dẫn:* **ThS VŨ ĐÌNH HỒNG**

*Người thực hiện:* **NGUYỄN VĂN HẬU – 52200292**

**NGUYỄN HÀ MY – 52200066**

**ĐOÀN THỐNG LĨNH – 52200013**

*Mã nhóm :* **ĐA14**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2024**

**TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM  
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG  
KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN CUỐI KỲ**  
**MÔN PHÁT TRIỂN ỨNG DỤNG WEB VỚI NODEJS**

**WEBSITE BÁN ĐIỆN THOẠI DI ĐỘNG**

*Người hướng dẫn:* **ThS VŨ ĐÌNH HỒNG**

*Người thực hiện:* **NGUYỄN VĂN HẬU – 52200292**

**NGUYỄN HÀ MY – 52200066**

**ĐOÀN THỐNG LĨNH – 52200013**

*Mã nhóm :* **ĐA14**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2014**

## LỜI CẢM ƠN

Lời đầu tiên, chúng em xin gửi lời cảm ơn chân thành đến các thầy cô của trường Đại học Tôn Đức Thắng.

Trong những năm học tập và rèn luyện tại trường chúng em đã nhận được rất nhiều sự giúp đỡ từ thầy Vũ Đình Hồng. Thầy là người đã truyền lửa, truyền cảm hứng là tấm gương sáng cho chúng em noi theo. Thầy đã truyền cho chúng em những kiến thức bổ ích, những bài học cuộc sống, những câu chuyện ý nghĩa, không chỉ là kiến thức trong sách vở mà còn cả những bài học trong cuộc sống này. Với lòng biết ơn sâu sắc từ trong đáy lòng của chúng em, chúng em xin gửi lời cảm ơn sâu sắc đến các thầy cô đã dẫn dắt em trong môn Phát triển ứng dụng web với Nodejs.

Trong quá trình thực hiện đồ án, em khó có thể tránh khỏi những sai sót, rất mong quý thầy cô có thể đóng góp ý kiến để em có được thêm nhiều bài học và rút kinh nghiệm cho lần sau.

## **ĐỒ ÁN ĐƯỢC HOÀN THÀNH TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Tôi xin cam đoan đây là sản phẩm đồ án của riêng chúng tôi và được sự hướng dẫn của thầy Vũ Đình Hồng;. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình.** Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày tháng năm 2024*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

*Nguyễn Văn Hậu*

*Đoàn Thống Lĩnh*

*Nguyễn Hà My*

## **PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN**

### **Phần xác nhận của GV hướng dẫn**

---

---

---

---

---

---

---

---

Tp. Hồ Chí Minh, ngày      tháng      năm  
(kí và ghi họ tên)

### **Phần đánh giá của GV chấm bài**

---

---

---

---

---

---

---

---

Tp. Hồ Chí Minh, ngày      tháng      năm  
(kí và ghi họ tên)

## TÓM TẮT

Đồ án này phát triển website cung cấp chức năng Point of Sale với mục đích bán hàng tại cửa hàng bán lẻ điện thoại, phụ kiện. Người dùng ứng dụng web này là nhân viên bán hàng và quản trị viên tại cửa hàng điện thoại. Người mua điện thoại không phải là người dùng ứng dụng web này. Về cơ bản, ứng dụng web này cần cung cấp các chức năng như:

- Quản lý nhân viên ( dưới quyền admin)
- Quản lý khách hàng
- Quản lý sản phẩm
- Quản lý giao dịch (đơn hàng)
- Thống kê và báo cáo theo thời gian

## MỤC LỤC

LỜI CẢM ƠN .....	i
ĐỒ ÁN ĐƯỢC HOÀN THÀNH .....	ii
TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG.....	ii
PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN .....	iii
TÓM TẮT .....	iv
MỤC LỤC.....	1
DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT .....	3
DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ .....	4
CHƯƠNG 1 – GIỚI THIỆU ĐỀ TÀI.....	5
1.1    Tổng quan về đề tài.....	5
1.2    Vì sao cần thực hiện đề tài? .....	5
1.3 Những yêu cầu của đề tài cần thực hiện .....	6
CHƯƠNG 2 – CƠ SỞ LÝ THUYẾT .....	9
CHƯƠNG 3 – PHÂN TÍCH THIẾT KẾ .....	10
3.1    Sơ đồ usecase tổng quát .....	10
3.2    Mô hình quan hệ .....	10
3.3    Mô hình ERD.....	11
3.4    Mô hình phát triển ứng dụng .....	11
3.5    Thiết kế cơ sở dữ liệu.....	12
CHƯƠNG 4 – HIỆN THỰC HỆ THỐNG .....	15
4.1    Công nghệ sử dụng .....	15
4.2    Kiến trúc và các thành phần chính.....	15
4.3    Cấu hình chính của hệ thống.....	15
4.4    Mô tả luồng hoạt động .....	16
4.5    Những phần code chính trong hệ thống.....	16
4.5.1    Khởi tạo ứng dụng với Express và các thiết lập cơ bản.....	16

4.5.2	Thiết lập Handlebars làm View Engine .....	18
4.5.3	Định nghĩa các Route chính .....	19
4.5.4	Xử lý lỗi 404 và các lỗi khác .....	20
4.5.5	Sử dụng Tailwind CSS .....	20
CHƯƠNG 5 – KẾT QUẢ ĐẠT ĐƯỢC .....		21
CHƯƠNG 6 – KẾT LUẬN.....		29
6.1	Ưu điểm và khuyết điểm của đề tài .....	29
6.1.1	Ưu điểm của đề tài .....	29
6.1.2	Khuyết điểm của đề tài .....	30
6.2	Những gì đã làm được và chưa làm được trong đề tài.....	31
6.2.1	Những gì đã làm được trong đề tài .....	31
6.2.2	Những gì chưa làm được trong đề tài .....	33
6.3	Hướng phát triển trong tương lai .....	33
6.3.1.	Tích hợp đặt hàng trực tuyến .....	33
6.3.2.	Tích hợp các phương thức thanh toán trực tuyến .....	33
6.3.3	Phát triển hệ thống chăm sóc khách hàng (CRM) .....	33



## **DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT**

### **CÁC KÝ HIỆU**

### **CÁC CHỮ VIẾT TẮT**

CRM: Customer Relationship Management

## **DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ**

### **DANH MỤC HÌNH**

Hình 1: Usecase tổng quát.....	10
Hình 2: Mô hình quan hệ .....	10
Hình 3: Mô hình ERD .....	11
Hình 4: Khởi tạo ứng dụng với Express và các thiết lập cơ bản.....	17
Hình 5: Thiết lập Handlebars làm View Engine .....	18
Hình 6: Định nghĩa các Route chính .....	19
Hình 7: Xử lý lỗi 404 và các lỗi khác .....	20
Hình 8: Sử dụng Tailwind CSS.....	20
Hình 9: Giao diện trang chủ khi admin đăng nhập .....	21
Hình 10: Giao diện trang chủ khi mở sidebar .....	21
Hình 11: Giao diện trang chủ khi nhân viên đăng nhập.....	22
Hình 12: Giao diện quản lý nhân viên .....	22
Hình 13: Giao diện quản lý khách hàng với vai trò admin .....	23
Hình 14: Giao diện quản lý khách hàng với vai trò nhân viên .....	23
Hình 15: Giao diện quản lý sản phẩm với vai trò admin .....	24
Hình 16: Giao diện quản lý sản phẩm với vai trò nhân viên .....	24
Hình 17: Giao diện quản lý giao dịch với vai trò admin.....	25
Hình 18: Giao diện quản lý giao dịch với vai trò nhân viên .....	25
Hình 19: Giao diện báo cáo thống kê với vai trò admin .....	26
Hình 20: Giao diện báo cáo thống kê với vai trò nhân viên .....	26
Hình 21: Giao diện tạo giao dịch .....	27
Hình 22: Giao diện tạo giao dịch .....	27

### **DANH MỤC BẢNG**

## **CHƯƠNG 1 – GIỚI THIỆU ĐỀ TÀI**

### **1.1 Tổng quan về đề tài**

Đề án này phát triển website cung cấp chức năng Point of Sale (điểm bán hàng) với mục đích bán hàng tại cửa hàng bán lẻ điện thoại, phụ kiện. Người dùng ứng dụng web này là nhân viên bán hàng và quản trị viên tại cửa hàng điện thoại. Người mua điện thoại không phải là người dùng ứng dụng web này. Về cơ bản, ứng dụng web này cần cung cấp các chức năng như:

- Quản lý nhân viên (dưới quyền admin)
- Quản lý tài khoản
- Quản lý khách hàng
- Quản lý sản phẩm
- Quản lý giao dịch (đơn hàng)
- Thống kê và báo cáo theo thời gian

### **1.2 Vì sao cần thực hiện đề tài?**

Trong thời đại số hóa ngày nay, việc quản lý bán hàng truyền thống đã gặp nhiều hạn chế và khó khăn như mất thời gian trong việc ghi nhận thông tin, thiếu chính xác trong quản lý tồn kho, và thiếu khả năng tương tác trực tuyến. Trên thực tế, nhiều doanh nghiệp nhỏ và vừa đang có xu hướng chuyển sang sử dụng các nền tảng trực tuyến để quản lý hoạt động kinh doanh của mình. Việc sử dụng các hệ thống bán hàng trực tuyến không chỉ tăng hiệu quả vận hành mà còn cải thiện trải nghiệm khách hàng thông qua quy trình mua hàng nhanh chóng và thuận tiện.

Xu hướng thị trường hiện nay cho thấy các cửa hàng điện thoại đang tìm kiếm các giải pháp để tối ưu quy trình bán hàng, quản lý hàng tồn kho một cách tự động, và tăng khả năng tương tác với khách hàng. Do đó, việc phát triển website quản lý bán điện thoại không chỉ đáp ứng được nhu cầu hiện tại của thị trường mà còn giúp cửa hàng điện thoại tối ưu hóa quy trình kinh doanh, tăng năng suất và khả năng cạnh tranh.

Mục tiêu của đề tài này là xây dựng một website bán điện thoại có khả năng quản lý sản phẩm, giao dịch bán hàng, khách hàng và đưa ra các báo cáo thống kê nhằm giúp quản trị viên và nhân viên tối ưu hóa công việc bán hàng và quản lý kinh doanh hiệu quả hơn.

### 1.3 Những yêu cầu của đề tài cần thực hiện

Những yêu cầu của đề tài cần thực hiện:

- **Quản lý nhân viên:** chức năng này chỉ được thực hiện bởi admin (quản trị viên), với chức năng này admin có thể xem danh sách nhân viên, xem thông tin chi tiết của nhân viên, khóa hoặc mở khóa tài khoản nhân viên, tạo tài khoản cho nhân viên: nhân viên bán hàng không thể tự tạo tài khoản, tài khoản của họ sẽ do quản trị viên tạo. Khi tạo tài khoản cho nhân viên bán hàng, quản trị viên cần cung cấp các thông tin sau: tên đầy đủ và địa chỉ gmail. Sau đó, một email sẽ được gửi đến hộp thư của nhân viên bán hàng, thông báo cho họ rằng một tài khoản đã được tạo với một liên kết để đăng nhập vào hệ thống. Liên kết chỉ có giá trị trong 1 phút. Sau thời gian này, nhân viên bán hàng sẽ cần yêu cầu bộ phận hỗ trợ của quản trị viên gửi lại một email khác với một liên kết 1 phút khác. Nhân viên kinh doanh mới được yêu cầu sử dụng liên kết để đăng nhập vào hệ thống lần đầu tiên. Họ không thể truy cập trực tiếp vào biểu mẫu đăng nhập và đăng nhập như các nhân viên cũ khác, nếu họ cố gắng làm như vậy, họ sẽ nhận được thông báo lỗi đại loại như "Vui lòng đăng nhập bằng cách nhấp vào liên kết trong email của bạn". Sau khi đăng nhập vào hệ thống lần đầu tiên, nhân viên bán hàng buộc phải tạo mật khẩu mới (không cần nhập lại mật khẩu cũ). Nếu không, họ sẽ không thể truy cập bất kỳ chức năng hệ thống nào khác ngoại trừ đăng xuất.
- **Quản lý tài khoản:** chức năng này dành cho cả admin và nhân viên (những người sử dụng hệ thống) họ có thể xem thông tin tài khoản của mình và thay đổi ảnh, mật khẩu tài khoản của mình.

- **Quản lý sản phẩm:** chức năng này dành cho cả admin và nhân viên nhưng nhân viên chỉ được phép xem danh sách sản phẩm và thông tin chi tiết của sản phẩm đó nhưng không thể thấy được giá nhập vào của sản phẩm, đối với admin thì có quyền thêm, xem mọi thông tin, chỉnh sửa sản phẩm và đặc biệt chỉ được xóa những sản phẩm chưa được mua lần nào.
- **Quản lý khách hàng:** chức năng này dành cho cả admin và nhân viên, với chức năng này thì tài khoản khách hàng sẽ được tạo trong lần đầu tiên mua hàng, khi thanh toán tại quầy thanh toán, nhân viên bán hàng sẽ hỏi số điện thoại của khách hàng. Tính năng thanh toán sẽ tự động hiển thị tên và địa chỉ của khách nếu khách đã mua trước đó. Nếu là lần đầu khách hàng mua hàng, nhân viên sẽ cần nhập họ tên và địa chỉ của khách hàng, hệ thống sẽ tự động tạo tài khoản mới cho khách hàng tại tính năng thanh toán. Không thể tạo tài khoản khách hàng theo cách thủ công, trên giao diện bán hàng, nhân viên có thể dễ dàng xem thông tin cá nhân của khách hàng (tên, số điện thoại, địa chỉ) và lịch sử mua hàng của họ bao gồm các thông tin như: tổng số tiền, số tiền khách hàng cho, số tiền thừa đã trả lại, ngày mua, số lượng sản phẩm. Nếu cần, nhân viên có thể xem chi tiết đơn hàng để xem thông tin cụ thể hơn như: danh sách sản phẩm và giá bán của chúng.
- **Quản lý giao dịch:** Trong chức năng này, nhân viên bán hàng sẽ nhập các sản phẩm cần mua thông qua nhiều cách khác nhau như tìm kiếm hoặc nhập mã vạch. Các sản phẩm được thêm vào sẽ được hiển thị trong chế độ xem danh sách, nơi chúng ta có thể xem danh sách các sản phẩm đang được mua cùng với các thông tin như: số lượng mặt hàng, đơn giá và tổng số tiền của từng sản phẩm. Ở giao diện tiếp theo thường sẽ có các thông tin tóm tắt như: tổng số tiền của toàn bộ đơn hàng, nơi nhập thông tin khách hàng (số điện thoại), cùng với các thông tin liên quan khác trong quá trình thanh toán. Chẳng hạn như số tiền khách hàng đưa ra, số tiền khách hàng nhận lại. Khi một sản phẩm được thêm vào (dù bằng tìm kiếm hay mã vạch) sẽ được hiển thị trong danh sách ngay lập tức mà không cần chờ xác nhận (để nhân viên bán

hàng dễ thao tác hơn), các thông số như tổng số tiền cũng được cập nhật tự động. Khi một sản phẩm được cập nhật (ví dụ: cập nhật số lượng) hoặc bị xóa khỏi danh sách, các thông số cũng sẽ được cập nhật tự động. Tại giao diện thanh toán này, như đã đề cập trước đó, nhân viên bán hàng có một cách dễ dàng để nhập số điện thoại của khách hàng hoặc nhập tên đầy đủ và địa chỉ nếu đó là khách hàng mới. Sau khi nhập đủ thông tin, quá trình thanh toán sẽ được hoàn tất và hóa đơn sẽ được in ra (mô phỏng bằng cách in file hóa đơn pdf).

## CHƯƠNG 2 – CƠ SỞ LÝ THUYẾT

Trong phần cơ sở lý thuyết, các kiến thức được sử dụng cho đề tài này bao gồm:

**Node.js:** Đây là một môi trường chạy JavaScript trên máy chủ, được sử dụng phổ biến để phát triển các ứng dụng web nhanh chóng và có khả năng xử lý nhiều yêu cầu đồng thời. Đối với đề tài này, Node.js giúp đảm bảo khả năng phản hồi nhanh và xử lý dữ liệu một cách liên tục.

**Express.js:** Là một framework cho Node.js, Express.js giúp việc xây dựng các API và ứng dụng web dễ dàng hơn thông qua việc cung cấp các phương thức HTTP mạnh mẽ. Express.js được sử dụng để xây dựng cấu trúc cho website, từ việc định nghĩa các route cho tới việc xử lý yêu cầu của người dùng.

**MongoDB:** Là hệ quản trị cơ sở dữ liệu NoSQL, MongoDB phù hợp với các ứng dụng web có sự thay đổi dữ liệu liên tục và đòi hỏi sự linh hoạt trong việc lưu trữ dữ liệu. Đối với hệ thống này, MongoDB được sử dụng để lưu trữ thông tin sản phẩm, khách hàng, và giao dịch.

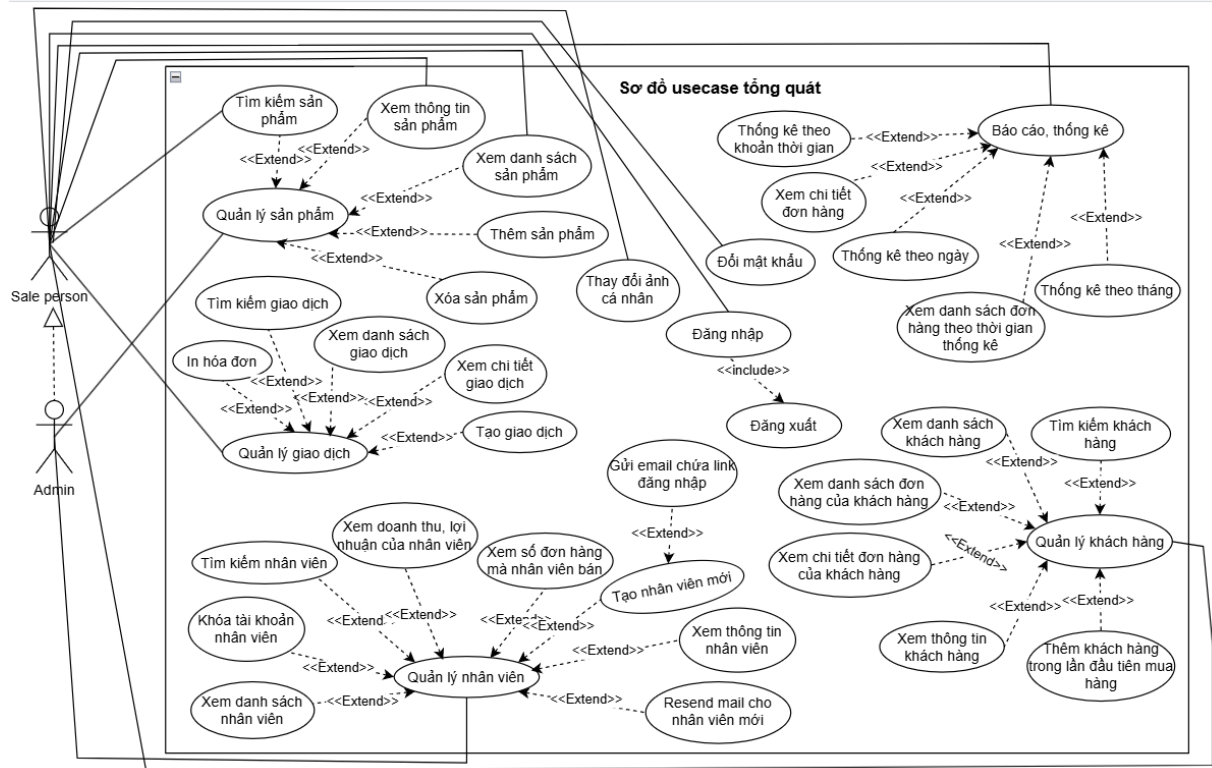
**Handlebars (HBS):** Đây là một công cụ render template cho Node.js, giúp tạo các trang HTML động một cách dễ dàng. Handlebars được sử dụng để xây dựng giao diện của website, cho phép hiển thị dữ liệu từ cơ sở dữ liệu lên trang web một cách trực quan.

**RESTful API:** Sử dụng mô hình REST để tạo API giúp các thành phần trong hệ thống có thể tương tác với nhau một cách hiệu quả. Các API này sẽ xử lý các yêu cầu từ người dùng như thêm sản phẩm mới, xem thông tin khách hàng, hoặc tạo báo cáo doanh số.

**Tailwind CSS:** Đây là một framework CSS tiện dụng cho việc xây dựng giao diện người dùng. Tailwind CSS cho phép thiết kế giao diện bằng cách sử dụng các lớp tiện ích có sẵn, giúp tiết kiệm thời gian và dễ dàng tùy chỉnh. Đặc biệt, Tailwind CSS mang lại sự linh hoạt và dễ bảo trì so với việc sử dụng các CSS truyền thống, do các thành phần giao diện được xây dựng từ những lớp tiện ích nhỏ và độc lập.

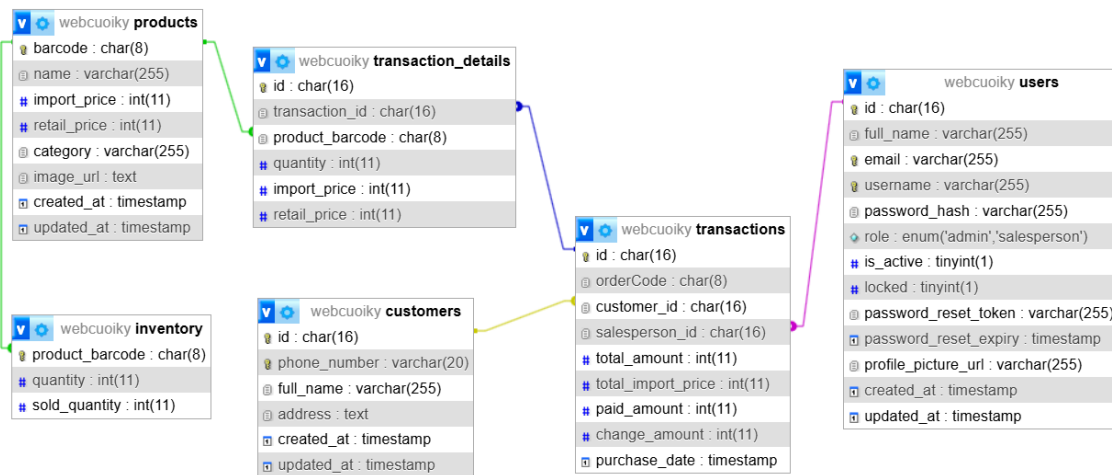
## CHƯƠNG 3 – PHÂN TÍCH THIẾT KẾ

### 3.1 Sơ đồ usecase tổng quát



Hình 1: Usecase tổng quát

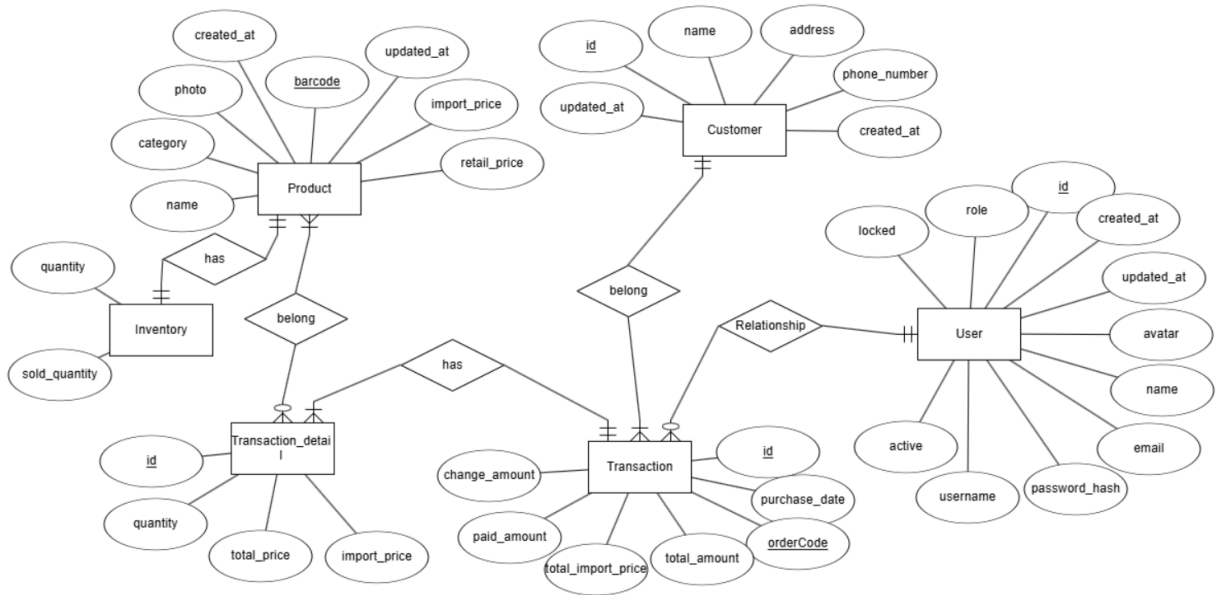
### 3.2 Mô hình quan hệ



Hình 2: Mô hình quan hệ



### 3.3 Mô hình ERD



Hình 3: Mô hình ERD

### 3.4 Mô hình phát triển ứng dụng

Ứng dụng web ở đề án này được phát triển theo mô hình thác nước. Mô hình thác nước là một mô hình quản lý dự án dễ hiểu, dễ tiếp cận và dễ quản lý, dựa vào quy trình thiết kế tuần tự và liên tiếp, giai đoạn mới chỉ được bắt đầu khi giai đoạn trước đã kết thúc. Quy trình phát triển:

- Giai đoạn Yêu cầu: Thực hiện khảo sát chi tiết yêu cầu về chức năng, giao diện, phương pháp quản lý tiệm net như quản lý doanh thu, quản lý nhân viên, quản lý khách hàng từ phía khách hàng và các yêu cầu về bảo mật.
- Giai đoạn Thiết kế:
  - Tiến hành thiết kế tổng thể các đặc tả, các sơ đồ mô tả chi tiết các chức năng, sơ đồ hoạt động phần mềm, bao gồm sơ đồ lớp, sơ đồ ERD, sơ đồ quan hệ.
  - Thiết kế cơ sở dữ liệu lưu trữ các dữ liệu về hệ thống, trang thiết bị, nhân viên.
- Giai đoạn Lập trình (coding): Lập trình viên thực hiện việc lập kế hoạch cho việc coding, xây dựng module, chức năng dựa trên các tài liệu đặc tả, thiết kế.

- Giai đoạn Kiểm thử: Tích hợp các bộ phận lại để thực hiện việc kiểm tra và sửa lỗi để hệ thống hoạt động chính xác và đúng theo tài liệu đặc tả yêu cầu.
- Giai đoạn triển khai: Triển khai hệ thống trong môi trường khách hàng, cài đặt hệ thống trên máy chủ, các máy quản lý, máy khách và thiết lập các cấu hình hệ thống.
- Giai đoạn Bảo trì: Bảo trì hệ thống khi phát hiện lỗi hay có thay đổi từ phía khách hàng, cập nhật hệ thống để giải quyết lỗi hoặc điều chỉnh theo yêu cầu khách hàng, tiếp tục cải thiện, tối ưu hệ thống.

### 3.5 Thiết kế cơ sở dữ liệu

```
User {
    id: ObjectId(...)
    name: string
    username: string
    avatar: string
    password: string -> hashing password
    active: int
    lock: int
    role: string
    email: string
    created_at: Date
    updated_at: Date
}
```

```
Product {
    id: ObjectId(...)
    barcode: string
    name: string
    import_price: int
    retail_price: int
}
```

```

        category: string
        created_at: date
        updated_at: date
        photo: string
        quantity: int
        sold_quantity: int
    }
    Customer {
        id: ObjectId(...)
        name: string
        address: string
        phone_number: string
        created_at: date
        update_at: date
    }
    Transaction{
        id: ObjectId(...)
        orderCode: string
        customer_Id: ObjectId(...)
        sale_Id: ObjectId(...)
        total_amount: int
        total_import_price: int
        change_amount: int
        paid_amount: int
        purchase_date: date
        details: ObjectId(...)
        [{

```

```
    _id: ObjectId()
    product_barcode: String,
    quantity: Number,
    import_price: Number,
    retail_price: Number,
  }, ...]
}
```

## CHƯƠNG 4 – HIỆN THỰC HỆ THỐNG

### 4.1 Công nghệ sử dụng

Hệ thống được xây dựng sử dụng Node.js cùng với Express.js làm framework chính để quản lý các route và xử lý các yêu cầu HTTP. Đây là các công nghệ được chọn nhờ tính dễ dàng tích hợp và hiệu suất cao khi xây dựng các ứng dụng web. Ngoài ra, các công nghệ hỗ trợ khác bao gồm:

- MongoDB với Mongoose: Để quản lý dữ liệu và kết nối tới cơ sở dữ liệu MongoDB.
- Express-Handlebars: Hệ thống sử dụng Express-Handlebars để tạo các giao diện người dùng động với các phần mở rộng .hbs. Hệ thống view được cấu hình để bao gồm các helper như currencyFormat, eq, và json để hỗ trợ việc hiển thị dữ liệu trên giao diện một cách trực quan.
- TailwindCSS: Hệ thống sử dụng TailwindCSS để quản lý giao diện người dùng, giúp tăng tính thẩm mỹ và tính phù hợp trên nhiều thiết bị. Tập cấu hình tailwind.config.js định nghĩa các breakpoint và theme cho hệ thống, giúp giao diện có thể thích nghi với nhiều kích thước màn hình khác nhau, từ sm (640px) cho tới 2xl (1920px), đảm bảo trải nghiệm tốt trên các thiết bị từ điện thoại tới máy tính để bàn

### 4.2 Kiến trúc và các thành phần chính

Ứng dụng được thiết kế với mô hình kiến trúc MVC (Model-View-Controller), gồm các thành phần chính:

- Controller: Chịu trách nhiệm xử lý các yêu cầu từ người dùng và tương tác với Model để lấy dữ liệu.
- Model: Được xây dựng bằng Mongoose để kết nối và thao tác với cơ sở dữ liệu MongoDB.
- View: Sử dụng Express-Handlebars để tạo các giao diện người dùng động.

### 4.3 Cấu hình chính của hệ thống

Tập index.js là điểm khởi đầu chính của ứng dụng, với một số thiết lập như sau:

- Sử dụng các thư viện cần thiết như `express`, `express-session`, `method-override`, `connect-flash`, và `express-handlebars` để quản lý phiên, xử lý dữ liệu, và cung cấp thông báo phản hồi tới người dùng.
- Thiết lập middleware cho Express bao gồm `bodyParser` để phân tích các yêu cầu HTTP, `session` để quản lý phiên đăng nhập của người dùng, và `flash` để hiển thị các thông báo flash khi có hành động thành công hay thất bại.
- Định nghĩa hệ thống route để quản lý người dùng, khách hàng, sản phẩm, giao dịch, và báo cáo thông qua các tệp như `usersRoutes`, `customersRoutes`, `productsRoutes`, `transactionsRoutes` và `reportsRoutes`.

#### **4.4 Mô tả luồng hoạt động**

Khi người dùng truy cập vào hệ thống, họ sẽ gửi yêu cầu HTTP đến server. Yêu cầu này được Router tiếp nhận và chuyển tiếp tới Controller phù hợp.

Controller xử lý yêu cầu bằng cách tương tác với Model để lấy hoặc cập nhật dữ liệu từ MongoDB.

Sau khi dữ liệu được lấy hoặc cập nhật, Controller sẽ truyền dữ liệu tới View để hiển thị kết quả cho người dùng. Express-Handlebars được sử dụng để kết hợp dữ liệu và mẫu giao diện, tạo ra HTML động để trả về cho người dùng.

Session và Flash message cũng được sử dụng trong quá trình này để lưu trữ thông tin người dùng và hiển thị thông báo (ví dụ: đăng nhập thành công, thêm sản phẩm thành công).

#### **4.5 Những phần code chính trong hệ thống**

##### **4.5.1 Khởi tạo ứng dụng với Express và các thiết lập cơ bản**

```

JS index.js > ...
1  const express = require('express');
2  const exphbs = require('express-handlebars');
3  const db = require('./src/config/db');
4  const path = require('path');
5  const app = express();
6  const port = 3000;
7  const methodOverride = require('method-override');
8  app.use(methodOverride('_method'));
9  app.use(express.urlencoded({ extended: true }));
10 app.use(express.json());
11 const bodyParser = require('body-parser');
12 app.use(bodyParser.urlencoded({ extended: true }));
13 const session = require('express-session');
14 const flash = require('connect-flash');
15
16 app.use(session({
17   secret: process.env.SESSION_SECRET || 'nodejs',
18   resave: true,
19   saveUninitialized: true
20 }));
21 app.use(flash());
22 app.use((req, res, next) => {
23   if (req.session.user) {
24     res.locals.user = req.session.user;
25   } else {
26     res.locals.user = null;
27   }
28   res.locals.success = req.flash('success');
29   res.locals.error = req.flash('error');
30   next();
31 });

```

Hình 4: Khởi tạo ứng dụng với Express và các thiết lập cơ bản

- Khởi tạo ứng dụng với Express: `express()` khởi tạo ứng dụng Node.js.
- Method Override: `methodOverride('_method')` cho phép gửi các yêu cầu HTTP khác ngoài GET và POST (như PUT, DELETE) thông qua query parameter `_method`.
- Session và Flash: Sử dụng `express-session` để quản lý phiên đăng nhập của người dùng và `connect-flash` để hiển thị các thông báo thành công hoặc thất bại.

### 4.5.2 Thiết lập Handlebars làm View Engine

```

38  const hbs = exphbs.create({
39    extname: '.hbs',
40    defaultLayout: 'layout',
41    layoutsDir: path.join(__dirname, 'src/views/layouts'),
42    partialsDir: path.join(__dirname, 'src/views/partials'),
43    helpers: {
44      currencyFormat: currencyFormat,
45      eq: function (a, b) {
46        return a === b;
47      },
48      count: function (a) {
49        return a + 1;
50      },
51      dif: function (a, b) {
52        return a !== b;
53      },
54      json: function(context) {
55        return JSON.stringify(context); // Chuyển đối tượng thành JSON string
56      },
57      eqo: function (a, b, options) {
58        if (a === b) {
59          return options.fn(this);
60        }
61        return options.inverse(this);
62      }
63    },
64  });
65
66
67  app.engine('hbs', hbs.engine);
68  app.set('view engine', 'hbs');
69  app.set('views', path.join(__dirname, 'src', 'views'));

```

Hình 5: Thiết lập Handlebars làm View Engine

Express-Handlebars: Thiết lập Handlebars làm view engine với extension là .hbs, giúp phát triển giao diện động dễ dàng hơn.

Helper Functions: Các helper như currencyFormat, count, dif, eqo, json và eq giúp định dạng dữ liệu hiển thị trên giao diện.



### 4.5.3 Định nghĩa các Route chính

```

72 const usersRoutes = require('./src/routes/usersRoutes');
73 const userRoutes = require('./src/routes/userRoutes');
74 const customersRoutes = require('./src/routes/customersRoutes');
75 const customerRoutes = require('./src/routes/customerRoutes');
76 const productsRoutes = require('./src/routes/productsRoutes');
77 const productRoutes = require('./src/routes/productRoutes');
78 const reportsRoutes = require('./src/routes/reportsRoutes');
79 const transactionsRoutes = require('./src/routes/transactionsRoutes');
80 const transactionRoutes = require('./src/routes/transactionRoutes');
81 const authRoutes = require('./src/routes/authRoutes');
82 const { loggedIn, isAuth, isAdmin } = require('./src/middleware/auth.js');
83 const { handlebars } = require('hbs');
84
85 // use routes
86 app.get('/', isAuth, (req, res) => {
87   res.redirect('/login');
88 });
89 app.use('/', authRoutes);
90 // users
91 app.get('/dashboard', loggedIn, (req, res) => {
92   const user = req.session.user;
93   res.render('index', {user});
94 });
95 app.use('/users', loggedIn, usersRoutes);
96 app.use('/user', loggedIn, userRoutes);
97 // customers
98 app.use('/customer', loggedIn, customerRoutes);
99 app.use('/customers', loggedIn, customersRoutes);
100 // products
101 app.use('/products', loggedIn, productsRoutes);
102 app.use('/product', loggedIn, productRoutes);
103 // transactions
104 app.use('/transactions', loggedIn, transactionsRoutes);
105 app.use('/transaction', loggedIn, transactionRoutes);
106 // reports
107 app.use('/reports', loggedIn, reportsRoutes);

```

Hình 6: Định nghĩa các Route chính

Các Route quản lý: Các route /users, /products, /transactions được yêu cầu thông qua middleware loggedIn để đảm bảo người dùng đã đăng nhập trước khi truy cập.

Route /dashboard: Đây là trang chính (dashboard) hiển thị thông tin người dùng đã đăng nhập.

#### 4.5.4 Xử lý lỗi 404 và các lỗi khác

```

118 app.use((req, res, next) => {
119   console.log(`404 Error - Route Not Found: ${req.originalUrl}`);
120   const error = new Error('Not Found');
121   error.status = 404;
122   next(error);
123 });
124
125 app.use((error, req, res, next) => {
126   res.status(error.status || 500);
127   console.log(error.message);
128   res.render('error', {
129     layout: 'loginLayout',
130     message: error.message,
131     status: error.status
132   });
133 });

```

Hình 7: Xử lý lỗi 404 và các lỗi khác

Xử lý lỗi 404: Nếu không tìm thấy route yêu cầu, hệ thống sẽ trả về lỗi 404 và ghi lại URL không tìm thấy.

Xử lý lỗi khác: Bất kỳ lỗi nào khác sẽ được bắt và render qua view error để thông báo cho người dùng.

#### 4.5.5 Sử dụng Tailwind CSS

```

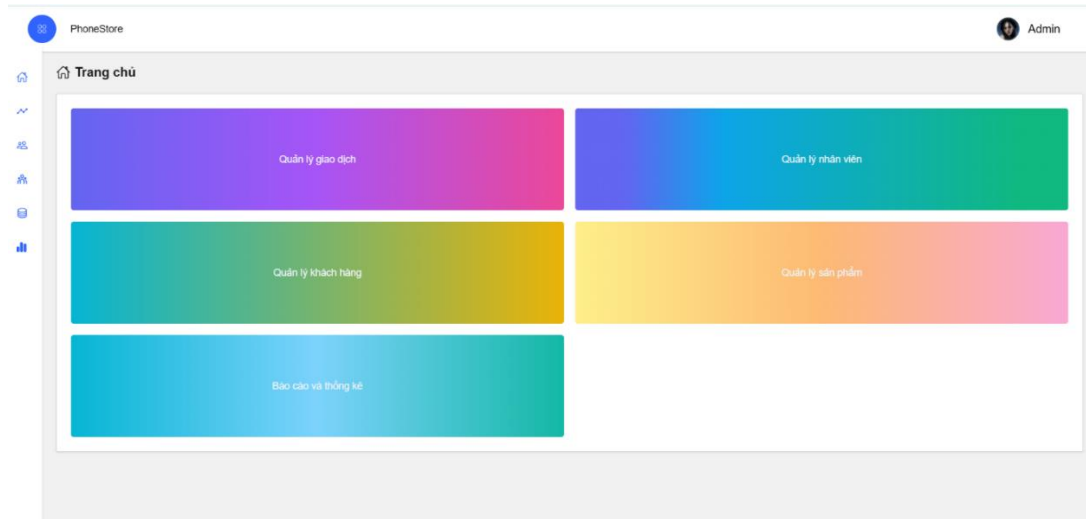
1  /** @type {import('tailwindcss').Config} */
2  module.exports = {
3    content: ['./src/views/**/*.hbs', './src/public/**/*.js', './src/public/**/*.html'],
4    theme: {
5      screens: {
6        sm: "640px",
7        md: "960px",
8        lg: "1280px",
9        xl: "1440px",
10       "2xl": "1920px",
11     },
12     container: {
13       padding: {
14         DEFAULT: "1.5625rem",
15         // md: '4.96875rem',
16         // sm: '1.25rem',
17         lg: "7.5rem",
18         // xl: '1.5625rem',
19       },
20     },
21     screens: {
22       md: "960px",
23       lg: "1280px",
24       xl: "1440px",
25       "2xl": "1920px",
26     },
27     center: true,
28   },
29   aspectRatio: {
30     auto: "auto",
31     square: "1 / 1",

```

Hình 8: Sử dụng Tailwind CSS

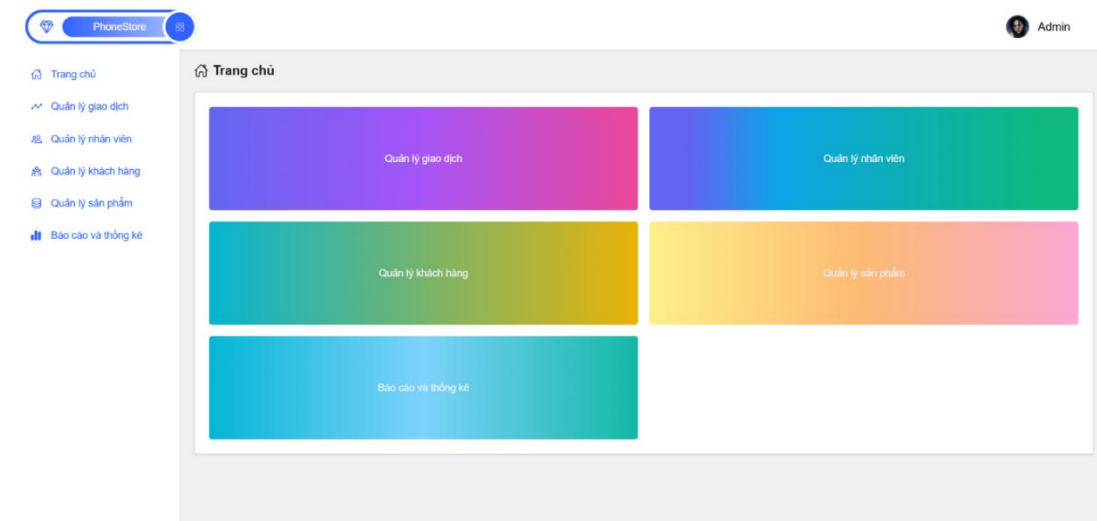
## CHƯƠNG 5 – KẾT QUẢ ĐẠT ĐƯỢC

Dưới đây là một số chức năng chính của hệ thống:



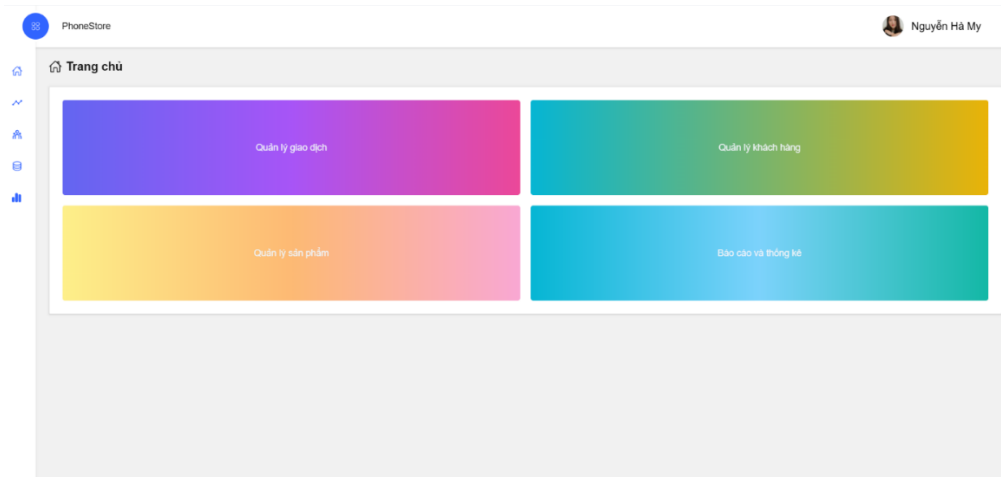
Hình 9: Giao diện trang chủ khi admin đăng nhập

- Đây là giao diện trang chủ với vai trò là admin sẽ có đầy đủ 5 chức năng: quản lý nhân viên, quản lý giao dịch, quản lý sản phẩm, quản lý khách hàng, báo cáo và thống kê. Chúng ta cũng có thể mở sidebar hoặc có thể bấm vào các icon bên trái để chọn chức năng mà không cần quay về home. Hoặc chúng ta có thể mở sidebar như hình dưới khi bấm vào icon bên cạnh chữ PhoneStore.



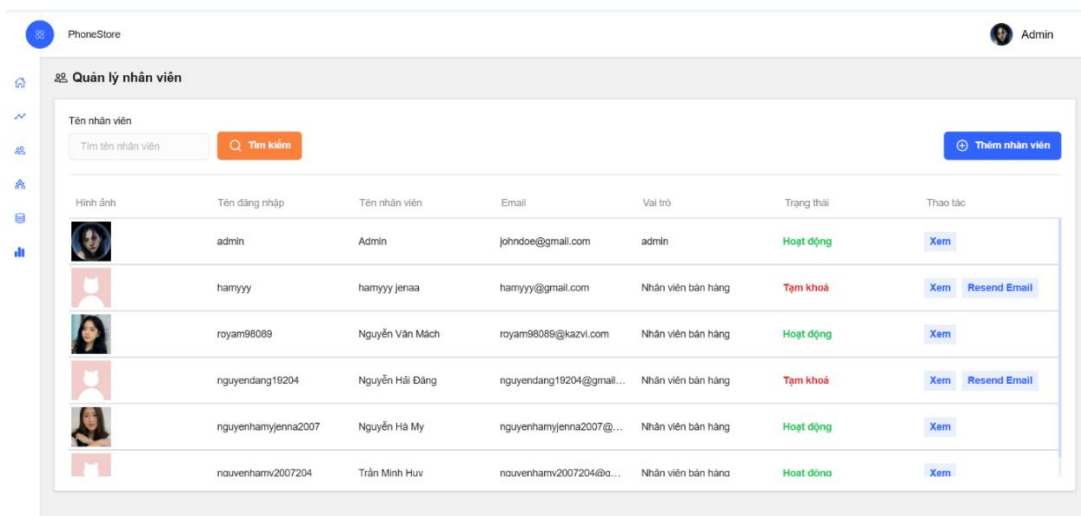
Hình 10: Giao diện trang chủ khi mở sidebar

- Đây là giao diện trang chủ với vai trò là nhân viên sẽ có 4 chức năng: quản lý giao dịch, quản lý sản phẩm, quản lý khách hàng, báo cáo và thống kê và không có quản lý nhân viên. Tương tự như trên chúng ta cũng có thể mở sidebar hoặc có thể bấm vào các icon bên trái để chọn chức năng mà không cần quay về home.

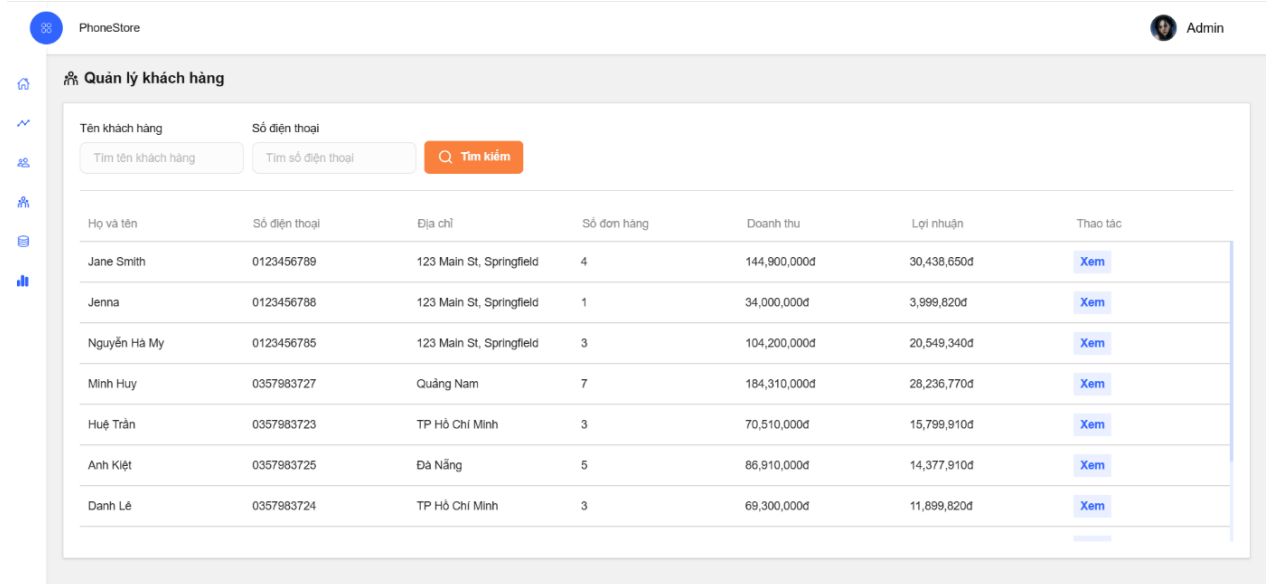


Hình 11: Giao diện trang chủ khi nhân viên đăng nhập

- Đây là giao diện quản lý nhân viên sẽ hiển thị một số thông tin như hình ảnh nhân viên, tên đăng nhập, tên nhân viên, email, vai trò, trạng thái của nhân viên. Chúng ta có thể xem chi tiết khi bấm vào nút xem và ở đó có thể chỉnh sửa khi bấm nút sửa.



Hình 12: Giao diện quản lý nhân viên



PhoneStore Admin

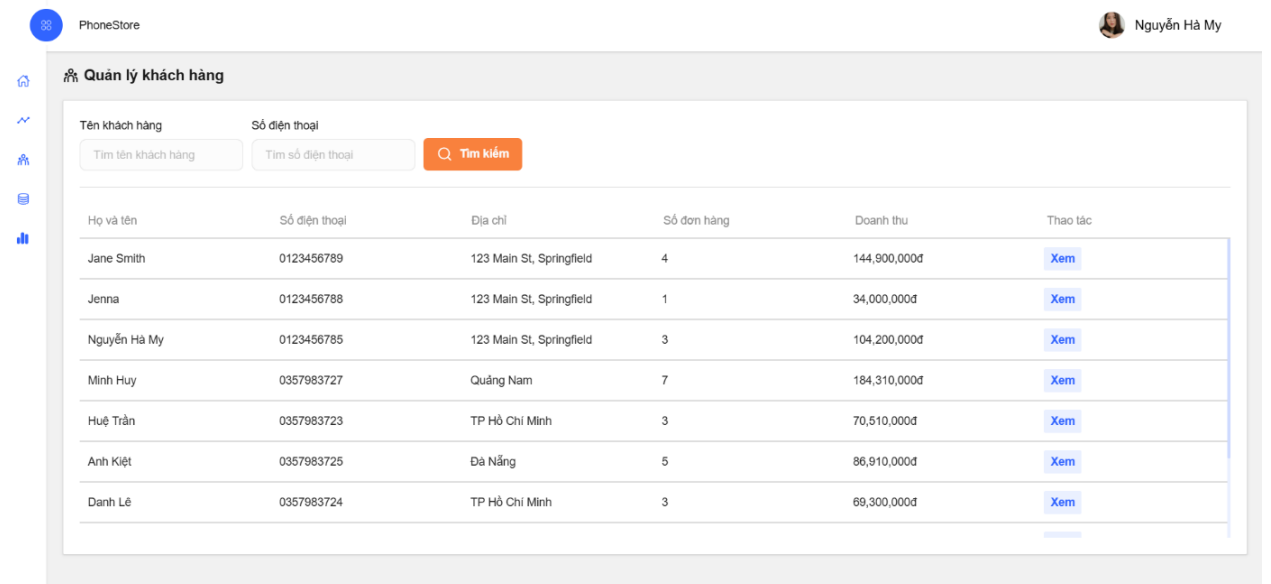
**Quản lý khách hàng**

Tên khách hàng:  Số điện thoại:  [Tìm kiếm](#)

Họ và tên	Số điện thoại	Địa chỉ	Số đơn hàng	Doanh thu	Lợi nhuận	Thao tác
Jane Smith	0123456789	123 Main St, Springfield	4	144,900,000đ	30,438,650đ	<a href="#">Xem</a>
Jenna	0123456788	123 Main St, Springfield	1	34,000,000đ	3,999,820đ	<a href="#">Xem</a>
Nguyễn Hà My	0123456785	123 Main St, Springfield	3	104,200,000đ	20,549,340đ	<a href="#">Xem</a>
Minh Huy	0357983727	Quảng Nam	7	184,310,000đ	28,236,770đ	<a href="#">Xem</a>
Huệ Trần	0357983723	TP Hồ Chí Minh	3	70,510,000đ	15,799,910đ	<a href="#">Xem</a>
Anh Kiệt	0357983725	Đà Nẵng	5	86,910,000đ	14,377,910đ	<a href="#">Xem</a>
Danh Lê	0357983724	TP Hồ Chí Minh	3	69,300,000đ	11,899,820đ	<a href="#">Xem</a>

Hình 13: Giao diện quản lý khách hàng với vai trò admin

- Đây là giao diện quản lý khách hàng, trên đó là những chi tiết về họ tên, số điện thoại, địa chỉ, số đơn hàng, doanh thu từ khách hàng đó. Và có thể bấm nút xem để xem thông tin về khách hàng và các đơn hàng của khách hàng đó. Đối với nhân viên thì họ sẽ không được xem lợi nhuận từ những khách hàng như hình dưới đây.



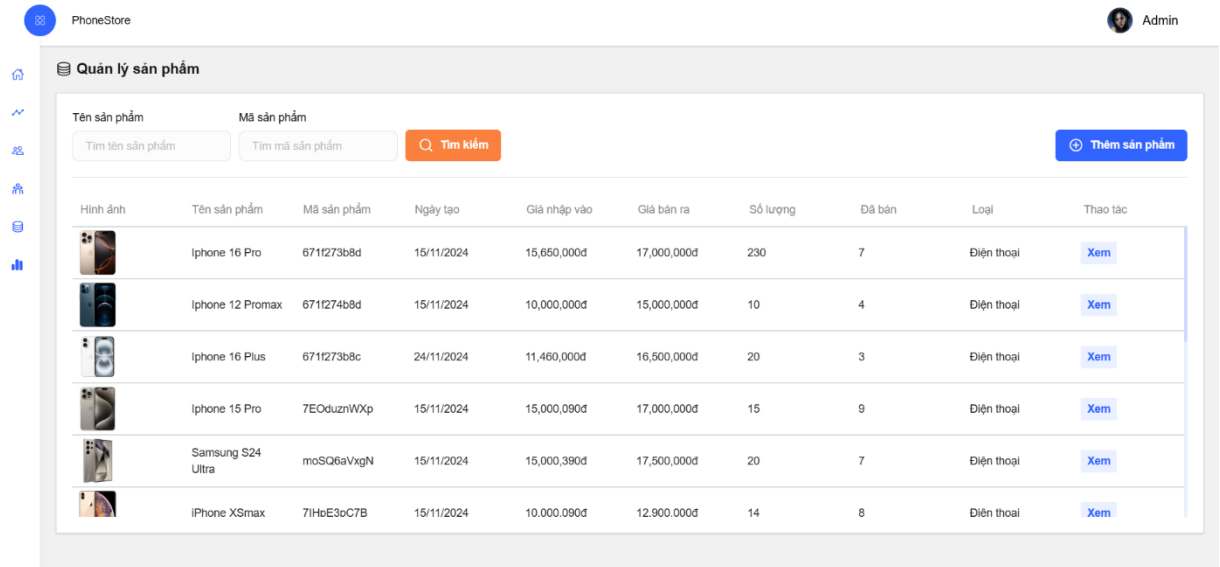
PhoneStore Nguyễn Hà My

**Quản lý khách hàng**

Tên khách hàng:  Số điện thoại:  [Tìm kiếm](#)

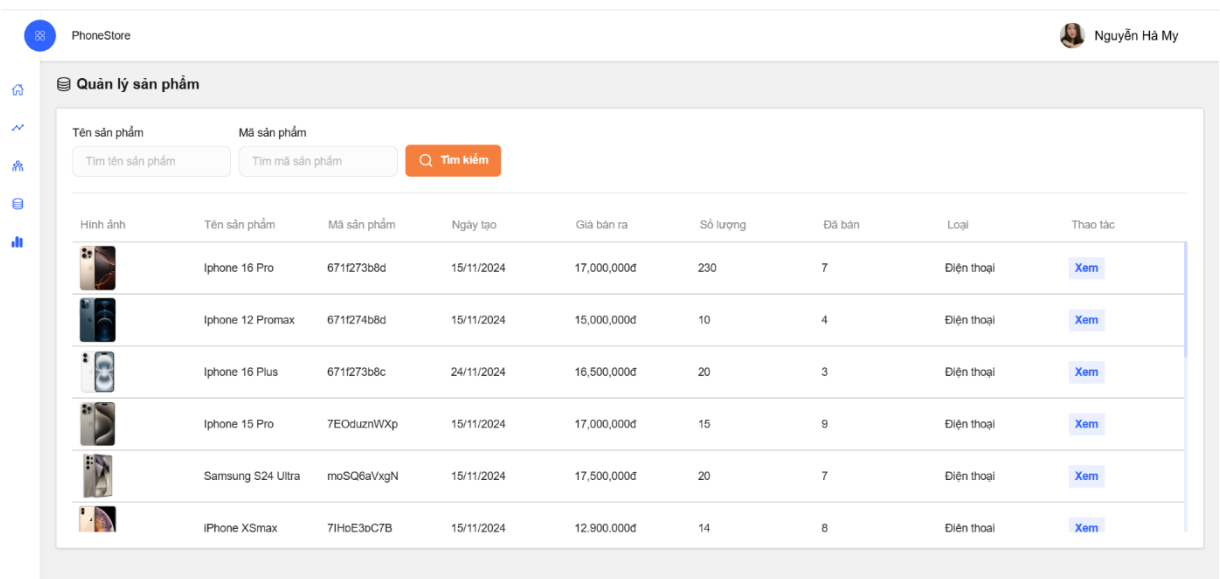
Họ và tên	Số điện thoại	Địa chỉ	Số đơn hàng	Doanh thu	Thao tác
Jane Smith	0123456789	123 Main St, Springfield	4	144,900,000đ	<a href="#">Xem</a>
Jenna	0123456788	123 Main St, Springfield	1	34,000,000đ	<a href="#">Xem</a>
Nguyễn Hà My	0123456785	123 Main St, Springfield	3	104,200,000đ	<a href="#">Xem</a>
Minh Huy	0357983727	Quảng Nam	7	184,310,000đ	<a href="#">Xem</a>
Huệ Trần	0357983723	TP Hồ Chí Minh	3	70,510,000đ	<a href="#">Xem</a>
Anh Kiệt	0357983725	Đà Nẵng	5	86,910,000đ	<a href="#">Xem</a>
Danh Lê	0357983724	TP Hồ Chí Minh	3	69,300,000đ	<a href="#">Xem</a>

Hình 14: Giao diện quản lý khách hàng với vai trò nhân viên



Hình 15: Giao diện quản lý sản phẩm với vai trò admin

- Đây là giao diện quản lý sản phẩm, trên đó là những chi tiết về tên, mã sản phẩm, hình ảnh, ngày tạo, giá nhập vào, giá bán ra, số lượng và số lượng đã bán. Và có thể bấm nút xem để xem thông tin về sản phẩm. Đối với nhân viên thì họ sẽ không được xem giá nhập vào của sản phẩm và không có quyền xóa sản phẩm. Và admin chỉ được xóa với những sản phẩm có số lượng đã bán là 0.



Hình 16: Giao diện quản lý sản phẩm với vai trò nhân viên

PhoneStore Admin

**Quản lý giao dịch**

Số lượng đơn hàng: **31**

Doanh thu: **779,280,000đ**

Lợi nhuận: **135,951,740đ**

Tổng tiền khách trả: **852,343,917đ**

Tổng tiền hoàn trả: **73,063,917đ**

Tên người bán hàng:  Tìm tên người bán hàng

Tên khách hàng:  Tìm tên khách hàng

Mã đơn hàng:  Tìm mã đơn hàng

**Tìm kiếm**

**Tạo giao dịch**

Mã đơn hàng	Ngày mua hàng	Nhân viên bán	Tên khách hàng	Tổng giá nhập(1)	Tổng đơn hàng(2)	Tiền khách trả(3)	Tiền hoàn trả(3-2)	Lợi nhuận(2-1)	Thao tác
53456354765	20/10/2024	Admin	Jane Smith	11,460,000đ	16,500,000đ	22,435,500đ	5,935,500đ	5,040,000đ	<a href="#">Xem</a>
53456354767	20/11/2024	hamyyy jena	Jenna	30,000,180đ	34,000,000đ	34,543,452đ	543,452đ	3,999,820đ	<a href="#">Xem</a>
53456354768	15/10/2024	Admin	Jane Smith	45,001,170đ	52,500,000đ	52,534,532đ	34,532đ	7,498,830đ	<a href="#">Xem</a>
53456354769	14/11/2024	hamyyy jena	Nguyễn Hà My	15,650,000đ	17,000,000đ	17,000,000đ	0đ	1,350,000đ	<a href="#">Xem</a>
53456354164	22/11/2024	Admin	Jane Smith	33,000,000đ	46,000,000đ	46,540,000đ	540,000đ	13,000,000đ	<a href="#">Xem</a>

Hình 17: Giao diện quản lý giao dịch với vai trò admin

- Đây là giao diện quản lý giao dịch với vai trò là admin. Và có thể bấm nút xem để xem thông tin chi tiết hơn về giao dịch đó. Đối với nhân viên thì họ sẽ không được xem lợi nhuận từ những giao dịch như hình dưới đây.

PhoneStore Nguyễn Hà My

**Quản lý giao dịch**

Số lượng đơn hàng: **31**

Doanh thu: **779,280,000đ**

Tổng tiền khách trả: **852,343,917đ**

Tổng tiền hoàn trả: **73,063,917đ**

Tên người bán hàng:  Tìm tên người bán hàng

Tên khách hàng:  Tìm tên khách hàng

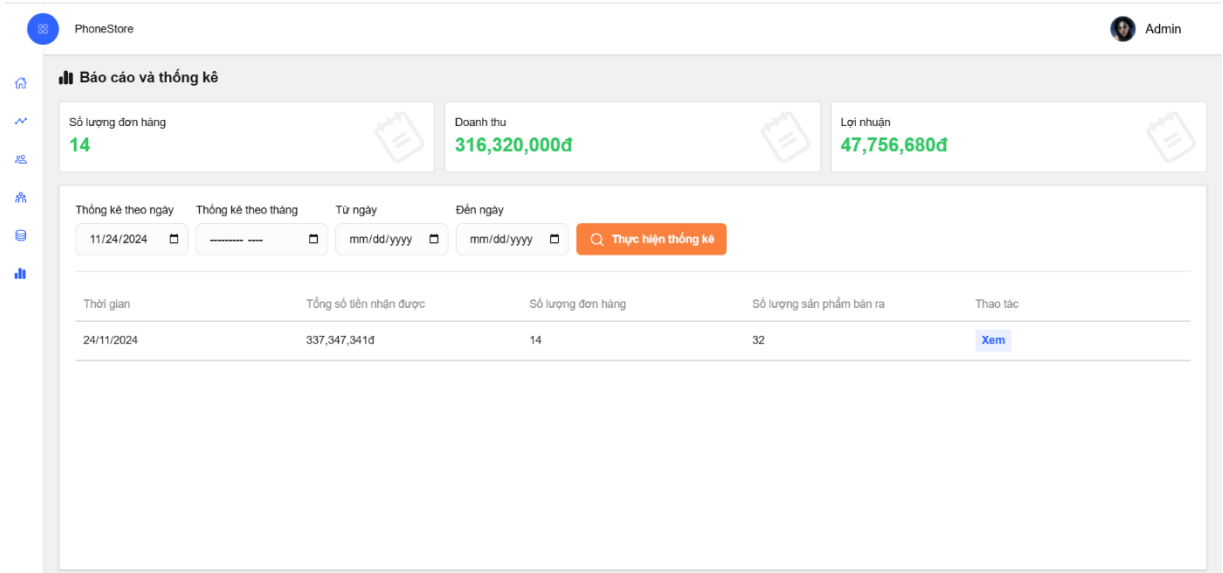
Mã đơn hàng:  Tìm mã đơn hàng

**Tìm kiếm**

**Tạo giao dịch**

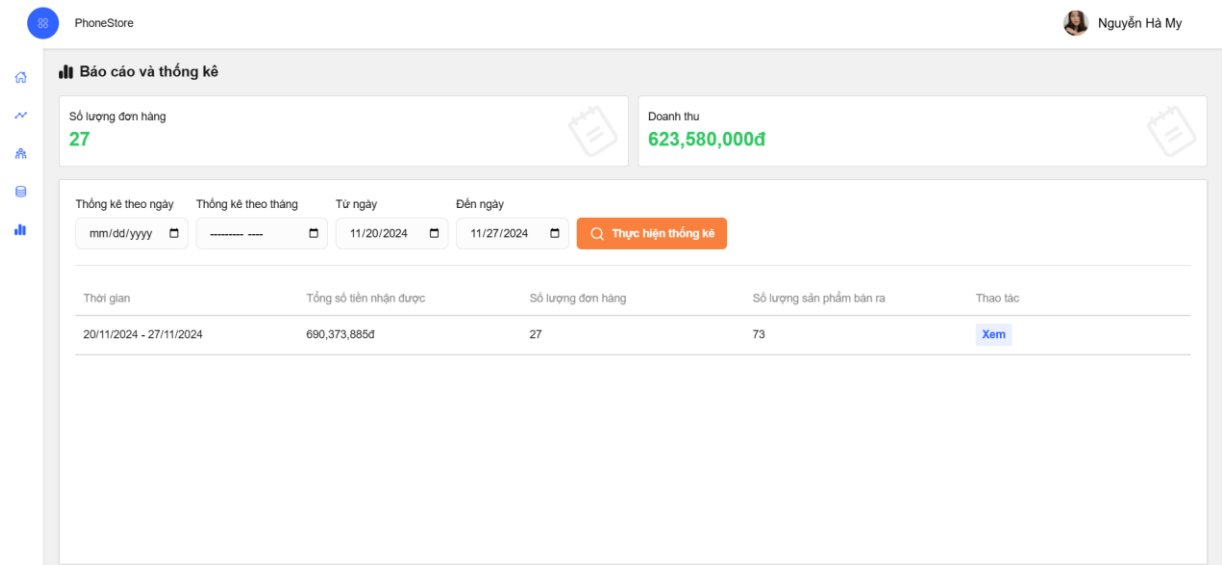
Mã đơn hàng	Ngày mua hàng	Nhân viên bán	Tên khách hàng	Tổng đơn hàng(1)	Tiền khách trả(2)	Tiền hoàn trả(2-1)	Thao tác
53456354765	20/10/2024	Admin	Jane Smith	16,500,000đ	22,435,500đ	5,935,500đ	<a href="#">Xem</a>
53456354767	20/11/2024	hamyyy jena	Jenna	34,000,000đ	34,543,452đ	543,452đ	<a href="#">Xem</a>
53456354768	15/10/2024	Admin	Jane Smith	52,500,000đ	52,534,532đ	34,532đ	<a href="#">Xem</a>
53456354769	14/11/2024	hamyyy jena	Nguyễn Hà My	17,000,000đ	17,000,000đ	0đ	<a href="#">Xem</a>
53456354164	22/11/2024	Admin	Jane Smith	46,000,000đ	46,540,000đ	540,000đ	<a href="#">Xem</a>

Hình 18: Giao diện quản lý giao dịch với vai trò nhân viên



Hình 19: Giao diện báo cáo thống kê với vai trò admin

- Dưới đây là giao diện báo cáo thống kê, trên đó là những chi tiết về thời gian, số lượng đơn hàng, doanh thu, số tiền nhận được, số lượng sản phẩm bán ra. Và có thể bấm nút xem để xem thông tin từng đơn hàng của thời gian mà thực hiện thống kê. Đối với nhân viên thì họ sẽ không được xem lợi nhuận.



Hình 20: Giao diện báo cáo thống kê với vai trò nhân viên



PhoneStore Admin

**Tạo giao dịch**

**Thông tin khách hàng**

Số điện thoại: 0357983727

Tiền đã nhận:

**Tìm kiếm sản phẩm**

**Danh sách sản phẩm đã chọn**

Sản phẩm	Số tiền	Số lượng	Thao tác
Hiện tại chưa có sản phẩm được chọn!			

**Thông tin hoá đơn**

**Thông tin khách hàng**

Tên khách hàng:

Số điện thoại: 0357983727

Địa chỉ:

**Danh sách đơn hàng**

Tên SP	SL	Thành tiền
Hiện tại chưa có sản phẩm được chọn!		

Tổng tiền nhận: 0 đ

Tổng đơn hàng: 0 đ

Số tiền phải trả lại: 0 đ

Hình 21: Giao diện tạo giao dịch

PhoneStore Nguyễn Hà My

**Tạo giao dịch**

**Thông tin khách hàng**

Số điện thoại: 0357983720

Họ và tên:

Địa chỉ:

Tiền đã nhận:

**Tìm kiếm sản phẩm**

**Danh sách sản phẩm đã chọn**

Sản phẩm	Số tiền	Số lượng	Thao tác
Hiện tại chưa có sản phẩm được chọn!			

**Thông tin hoá đơn**

**Thông tin khách hàng**

Tên khách hàng:

Số điện thoại: 0357983720

Địa chỉ:

**Danh sách đơn hàng**

Tên SP	SL	Thành tiền
Hiện tại chưa có sản phẩm được chọn!		

Tổng tiền nhận: 0 đ

Tổng đơn hàng: 0 đ

Số tiền phải trả lại: 0 đ

Khách hàng chưa từng mua hàng. Vui lòng điền thông tin khách hàng

Hình 22: Giao diện tạo giao dịch

- Dưới đây là giao diện tạo giao dịch, khi nhập đủ 10 số vào ô nhập số điện thoại thì nút kiểm tra thông tin sẽ hiện để kiểm tra khách hàng đã tồn tại chưa, nếu chưa thì thực hiện nhập tên và địa chỉ, còn nếu rồi thì sẽ tự động hiện tên và địa chỉ của khách hàng đó, nhân viên bán hàng sẽ nhập các sản phẩm cần mua thông qua nhiều cách khác nhau như tìm kiếm hoặc nhập mã vạch. Các sản phẩm được thêm vào sẽ

được hiển thị trong chế độ xem danh sách, nơi chúng ta có thể xem danh sách các sản phẩm đang được mua cùng với các thông tin như: số lượng mặt hàng, đơn giá và tổng số tiền của từng sản phẩm. Ở giao diện thông tin hóa đơn có các thông tin tóm tắt như: tổng số tiền của toàn bộ đơn hàng, nơi nhập thông tin khách hàng (số điện thoại), cùng với các thông tin liên quan khác trong quá trình thanh toán. Chẳng hạn như số tiền khách hàng đưa ra, số tiền khách hàng nhận lại. Khi một sản phẩm được thêm vào (dù bằng tìm kiếm hay mã vạch) sẽ được hiển thị trong danh sách ngay lập tức mà không cần chờ xác nhận (để nhân viên bán hàng dễ thao tác hơn), các thông số như tổng số tiền cũng được cập nhật tự động. Khi một sản phẩm được cập nhật (ví dụ: cập nhật số lượng) hoặc bị xóa khỏi danh sách, các thông số cũng sẽ được cập nhật tự động.

## CHƯƠNG 6 – KẾT LUẬN

### 6.1 Ưu điểm và khuyết điểm của đề tài

#### 6.1.1 Ưu điểm của đề tài

- Tự động hóa quản lý bán hàng: Hệ thống giúp tự động hóa việc quản lý sản phẩm, khách hàng và giao dịch, từ đó giảm thiểu tối đa các sai sót do nhập liệu thủ công. Nhân viên có thể cập nhật và tra cứu thông tin một cách nhanh chóng, giúp tối ưu quy trình bán hàng và đảm bảo tính chính xác trong quản lý tồn kho.
- Giao diện thân thiện, dễ sử dụng: Website được thiết kế với Tailwind CSS, giúp giao diện trở nên trực quan, hiện đại và thân thiện với người dùng. Các nút, form nhập liệu và bảng thông tin đều dễ sử dụng, ngay cả với người không có nhiều kinh nghiệm về công nghệ. Điều này giúp nâng cao trải nghiệm người dùng, cho phép nhân viên thao tác nhanh và hiệu quả.
- Quản lý hiệu quả khách hàng và tạo sự cá nhân hóa: Hệ thống lưu trữ thông tin khách hàng, bao gồm lịch sử mua hàng, giúp quản trị viên và nhân viên có thể quản lý khách hàng dễ dàng, cung cấp dịch vụ chăm sóc khách hàng tốt hơn. Việc nắm rõ lịch sử mua hàng của khách giúp cửa hàng cung cấp dịch vụ cá nhân hóa, từ đó tăng cường sự hài lòng của khách hàng.
- Tính năng báo cáo và thống kê: Tính năng báo cáo doanh số giúp quản trị viên có thể theo dõi chi tiết tình hình kinh doanh của cửa hàng. Báo cáo này giúp đưa ra các quyết định chiến lược, cải thiện hoạt động kinh doanh và tối ưu hóa hàng tồn kho.
- Hệ thống phân quyền người dùng: Hệ thống phân quyền rõ ràng cho quản trị viên và nhân viên bán hàng, đảm bảo người dùng chỉ có quyền truy cập vào những chức năng cần thiết. Việc này giúp tăng cường bảo mật, giảm thiểu các lỗi thao tác do nhân viên không có đủ hiểu biết về quản trị.
- Dễ dàng mở rộng và tích hợp: Việc sử dụng Node.js và MongoDB giúp hệ thống dễ dàng mở rộng và tích hợp thêm các tính năng mới trong tương lai. Những công nghệ

này cũng mang lại khả năng xử lý lượng lớn dữ liệu một cách hiệu quả, đảm bảo hệ thống hoạt động ổn định.

### **6.1.2 Khuyết điểm của đề tài**

- Chưa có tính năng đặt hàng trực tuyến cho khách hàng: Hệ thống hiện tại chỉ hỗ trợ việc bán hàng tại cửa hàng và không có chức năng đặt hàng trực tuyến cho khách hàng. Điều này làm giảm khả năng tiếp cận đến nhiều đối tượng khách hàng hơn, đặc biệt trong thời đại thương mại điện tử đang phát triển mạnh mẽ.
- Thiếu hỗ trợ các phương thức thanh toán đa dạng: Hệ thống chưa tích hợp các phương thức thanh toán trực tuyến như thẻ tín dụng, ví điện tử (MoMo, ZaloPay). Điều này có thể gây khó khăn cho khách hàng khi muốn mua hàng nhưng không có sẵn tiền mặt hoặc không muốn thanh toán trực tiếp tại cửa hàng.
- Quản lý khuyến mãi còn hạn chế: Hiện tại, hệ thống chưa có chức năng quản lý các chương trình khuyến mãi và giảm giá phức tạp. Điều này làm hạn chế khả năng thúc đẩy doanh số bán hàng, đặc biệt trong các dịp lễ tết khi cửa hàng thường có nhu cầu triển khai các chương trình khuyến mãi để thu hút khách hàng.
- Chưa tích hợp ứng dụng di động: Hệ thống chưa có ứng dụng di động, điều này gây bất tiện cho nhân viên khi muốn cập nhật thông tin hoặc kiểm tra tồn kho khi không có sẵn máy tính. Một ứng dụng di động sẽ giúp tăng cường khả năng di động cho hệ thống, đặc biệt cho các nhân viên thường xuyên phải làm việc ngoài cửa hàng.
- Chưa có tính năng tự động hóa quy trình quản lý: Hệ thống chưa được tích hợp đầy đủ tính năng tự động hóa quy trình, ví dụ như tự động gửi email chăm sóc khách hàng hoặc tự động cập nhật trạng thái đơn hàng cho khách. Điều này làm tăng khối lượng công việc cho nhân viên và giảm khả năng đáp ứng kịp thời cho khách hàng.
- Khả năng bảo mật còn cần được cải thiện: Hệ thống hiện tại có thể cần thêm các biện pháp bảo mật như xác thực hai lớp (2FA) hoặc mã hóa dữ liệu để bảo vệ thông tin của người dùng tốt hơn. Việc bảo vệ dữ liệu khách hàng và giao dịch là rất quan trọng, đặc biệt với các thông tin nhạy cảm như mật khẩu và thông tin thanh toán.

## 6.2 Những gì đã làm được và chưa làm được trong đề tài

### 6.2.1 Những gì đã làm được trong đề tài

- **Quản lý nhân viên:** Nhóm em đã hoàn thành được tất cả các chức năng này, chức năng này chỉ được thực hiện bởi admin (quản trị viên), admin có thể xem danh sách nhân viên, xem thông tin chi tiết của nhân viên, khóa hoặc mở khóa tài khoản nhân viên, tạo tài khoản cho nhân viên: nhân viên bán hàng không thể tự tạo tài khoản, tài khoản của họ sẽ do quản trị viên tạo. Khi tạo tài khoản cho nhân viên bán hàng, quản trị viên cần cung cấp các thông tin sau: tên đầy đủ và địa chỉ gmail. Sau đó, một email sẽ được gửi đến hộp thư của nhân viên bán hàng, thông báo cho họ rằng một tài khoản đã được tạo với một liên kết để đăng nhập vào hệ thống. Liên kết chỉ có giá trị trong 1 phút. Sau thời gian này, nhân viên bán hàng sẽ cần yêu cầu bộ phận hỗ trợ của quản trị viên gửi lại một email khác với một liên kết 1 phút khác. Nhân viên kinh doanh mới được yêu cầu sử dụng liên kết để đăng nhập vào hệ thống lần đầu tiên. Họ không thể truy cập trực tiếp vào biểu mẫu đăng nhập và đăng nhập như các nhân viên cũ khác, nếu họ cố gắng làm như vậy, họ sẽ nhận được thông báo lỗi đại loại như "Vui lòng đăng nhập bằng cách nhấp vào liên kết trong email của bạn". Sau khi đăng nhập vào hệ thống lần đầu tiên, nhân viên bán hàng buộc phải tạo mật khẩu mới (không cần nhập lại mật khẩu cũ). Nếu không, họ sẽ không thể truy cập bất kỳ chức năng hệ thống nào khác ngoại trừ đăng xuất.
- **Quản lý tài khoản:** Chức năng này dành cho cả admin và nhân viên (những người sử dụng hệ thống) họ có thể xem thông tin tài khoản của mình và thay đổi ảnh, mật khẩu tài khoản của mình. Với chức năng này nhóm em đã thực hiện được tất cả các chức năng.
- **Quản lý sản phẩm:** Nhóm em đã hoàn thành được tất cả các chức năng này, chức năng này dành cho cả admin và nhân viên nhưng nhân viên chỉ được phép xem danh sách sản phẩm và thông tin chi tiết của sản phẩm đó nhưng không thể thấy được giá

nhập vào của sản phẩm, đối với admin thì có quyền thêm, xem mọi thông tin, chỉnh sửa sản phẩm và đặc biệt chỉ được xóa những sản phẩm chưa được mua lần nào.

- **Quản lý khách hàng:** Nhóm em đã hoàn thành được tất cả các chức năng này, chức năng này dành cho cả admin và nhân viên, với chức năng này thì tài khoản khách hàng sẽ được tạo trong lần đầu tiên mua hàng, khi thanh toán tại quầy thanh toán, nhân viên bán hàng sẽ hỏi số điện thoại của khách hàng. Tính năng thanh toán sẽ tự động hiển thị tên và địa chỉ của khách nếu khách đã mua trước đó. Nếu là lần đầu khách hàng mua hàng, nhân viên sẽ cần nhập họ tên và địa chỉ của khách hàng, hệ thống sẽ tự động tạo tài khoản mới cho khách hàng tại tính năng thanh toán. Không thể tạo tài khoản khách hàng theo cách thủ công, trên giao diện bán hàng, nhân viên có thể dễ dàng xem thông tin cá nhân của khách hàng (tên, số điện thoại, địa chỉ) và lịch sử mua hàng của họ bao gồm các thông tin như: tổng số tiền, số tiền khách hàng cho, số tiền thừa đã trả lại, ngày mua, số lượng sản phẩm. Nếu cần, nhân viên có thể xem chi tiết đơn hàng để xem thông tin cụ thể hơn như: danh sách sản phẩm và giá bán của chúng.
- **Quản lý giao dịch:** Nhóm em đã hoàn thành được tất cả các chức năng này, trong chức năng này, nhân viên bán hàng sẽ nhập các sản phẩm cần mua thông qua nhiều cách khác nhau như tìm kiếm hoặc nhập mã vạch. Các sản phẩm được thêm vào sẽ được hiển thị trong chế độ xem danh sách, nơi chúng ta có thể xem danh sách các sản phẩm đang được mua cùng với các thông tin như: số lượng mặt hàng, đơn giá và tổng số tiền của từng sản phẩm. Ở giao diện tiếp theo thường sẽ có các thông tin tóm tắt như: tổng số tiền của toàn bộ đơn hàng, nơi nhập thông tin khách hàng (số điện thoại), cùng với các thông tin liên quan khác trong quá trình thanh toán. Chẳng hạn như số tiền khách hàng đưa ra, số tiền khách hàng nhận lại. Khi một sản phẩm được thêm vào (dù bằng tìm kiếm hay mã vạch) sẽ được hiển thị trong danh sách ngay lập tức mà không cần chờ xác nhận (để nhân viên bán hàng dễ thao tác hơn), các thông số như tổng số tiền cũng được cập nhật tự động. Khi một sản phẩm được

cập nhật (ví dụ: cập nhật số lượng) hoặc bị xóa khỏi danh sách, các thông số cũng sẽ được cập nhật tự động. Tại giao diện thanh toán này, như đã đề cập trước đó, nhân viên bán hàng có một cách dễ dàng để nhập số điện thoại của khách hàng hoặc nhập tên đầy đủ và địa chỉ nếu đó là khách hàng mới.

### ***6.2.2 Những gì chưa làm được trong đề tài***

- Sau khi nhập đủ thông tin, quá trình thanh toán sẽ được hoàn tất và hóa đơn sẽ được in ra (mô phỏng bằng cách in file hóa đơn pdf). Với chức năng này thì nhóm em đã thực hiện in được hóa đơn nhưng còn bị lỗi font chữ.

## **6.3 Hướng phát triển trong tương lai**

### ***6.3.1. Tích hợp đặt hàng trực tuyến***

Hiện tại, hệ thống chỉ hỗ trợ việc bán hàng qua cửa hàng và không có chức năng đặt hàng trực tuyến cho khách hàng. Hướng phát triển trong tương lai là tích hợp chức năng đặt hàng trực tuyến, cho phép khách hàng có thể tự lựa chọn và đặt mua sản phẩm từ website.

Các tính năng bổ sung có thể bao gồm: giỏ hàng, quản lý đơn hàng, và thông báo trạng thái đơn hàng đến khách hàng.

### ***6.3.2. Tích hợp các phương thức thanh toán trực tuyến***

Để tạo thuận tiện cho khách hàng trong việc thanh toán, hệ thống có thể được mở rộng để tích hợp các phương thức thanh toán trực tuyến như thẻ tín dụng, thẻ ghi nợ, ví điện tử (Momo, ZaloPay, v.v.), và chuyển khoản ngân hàng.

Điều này sẽ giúp nâng cao trải nghiệm khách hàng và đáp ứng được xu hướng thanh toán không dùng tiền mặt.

### ***6.3.3 Phát triển hệ thống chăm sóc khách hàng (CRM)***

Tích hợp hệ thống chăm sóc khách hàng (CRM) để quản lý lịch sử mua hàng, hỗ trợ khách hàng và gửi thông báo chăm sóc định kỳ. Hệ thống có thể tự động gửi email hoặc thông báo qua SMS để giữ liên lạc với khách hàng về các chương trình khuyến mãi hoặc sản phẩm mới.

Có thể thêm tính năng đánh giá sản phẩm và gửi phản hồi của khách hàng, giúp cải thiện chất lượng dịch vụ và sản phẩm.



## TÀI LIỆU THAM KHẢO

### Tiếng Việt

### Tiếng Anh

- [1] Tailwind Labs, Tailwind CSS: Rapidly Build Modern Websites Without Ever Leaving Your HTML. [Online]. Available: <https://tailwindcss.com/>. [Accessed: Nov. 29, 2024].
- [2] StrongLoop, IBM, and others, *Express - Node.js Web Application Framework*. [Online]. Available: <https://expressjs.com/>. [Accessed: Nov. 29, 2024].
- [3] Node.js Foundation, Node.js: JavaScript Runtime Built on Chrome's V8 JavaScript Engine. [Online]. Available: <https://nodejs.org/>. [Accessed: Nov. 29, 2024].
- [4] Handlebars.js, Handlebars: Minimal Templating on Steroids. [Online]. Available: <https://handlebarsjs.com/>. [Accessed: Nov. 29, 2024].