

**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**

# **Hệ Thống POS cho Cửa Hàng Điện Thoại**

*Người hướng dẫn:* **GVHD. VŨ ĐÌNH HỒNG**

*Người thực hiện:* **PHẠM QUỐC ANH – 52000626**

**LƯƠNG GIA BẢO – 52000630**

**NGUYỄN TRẦN QUANG HUY – 52000668**

**Lớp : 20050201**

**Khoá : 24**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023**

**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**

# **Hệ Thống POS cho Cửa Hàng Điện Thoại**

Người hướng dẫn: **GVHD. VŨ ĐÌNH HỒNG**  
Người thực hiện: **PHẠM QUỐC ANH – 52000626**  
**LƯƠNG GIA BẢO – 52000630**  
**NGUYỄN TRẦN QUANG HUY – 52000668**  
Lớp : **20050201**  
Khoá : **24**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023**

## LỜI CẢM ƠN

Để hoàn thành bài báo cáo cuối giữa này, chúng em xin gửi lời cảm ơn đến thầy Vũ Đình Hồng đã hướng dẫn, góp ý và có những nhận xét tích cực trong suốt quá trình làm bài.

Ngoài ra chúng em cũng xin chân thành cảm ơn khoa Công nghệ thông tin đã tạo cơ hội để chúng em có cơ hội thực hiện và hoàn thành bài báo cáo này một cách suôn sẻ.

Bài báo cáo của chúng em được thực hiện thông qua các kiến thức từ bài giảng, thông tin từ sách tham khảo và các trang tài liệu. Trong quá trình làm bài, bởi vì thời gian có hạn nên không thể tránh khỏi có sai sót. Vì vậy chúng em mong nhận được sự quan tâm, đánh giá từ thầy để bài báo cáo của chúng em có thể được hoàn thiện hơn.

Cuối cùng, chúng em xin gửi lời chúc sức khỏe đến thầy và các thầy cô, mong thầy cô sẽ hoàn thành tốt công tác giảng dạy của mình. Em xin gửi lời chúc đến thầy Vũ Đình Hồng sẽ luôn khỏe mạnh, ngày càng thành công trong công cuộc truyền đạt kiến thức và những công việc khác trong cuộc sống. Em xin chân thành cảm ơn.

## **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 GVHD 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 10 tháng 12 năm 2023*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

*Phạm Quốc Anh*

*Lương Gia Bảo*

*Nguyễn Trần Quang Huy*

## 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

Bài báo cáo của chúng em tập trung vào việc giới thiệu và trình bày về quá trình phát triển một trang web cung cấp chức năng Điểm bán hàng (Point of Sale), hay được viết tắt là POS, hướng tới việc bán lẻ điện thoại và phụ kiện. Chúng em đã xây dựng một giao diện người dùng cho phép admin và nhân viên thực hiện việc quản lý dữ liệu, đơn hàng mua sắm điện thoại và các sản phẩm phụ kiện liên quan một cách thuận tiện.

Trong quá trình phát triển, chúng em đã áp dụng những kiến thức được học từ các buổi giảng và thực hành trên lớp, cũng như nghiên cứu các phương pháp và công nghệ mới từ nguồn tài nguyên bên ngoài. Chúng em đã đáp ứng đúng yêu cầu đã đề ra và đã xây dựng một trang web có tính ổn định và khả dụng để hỗ trợ quá trình quản lý dữ liệu và đơn hàng.

Mặc dù trang web vẫn còn một số điểm chưa hoàn thiện và thiếu sót, nhưng chúng em kỳ vọng rằng nó có thể đáp ứng được mong đợi của người sử dụng và giáo viên. Chúng em rất mong có sự đánh giá và phản hồi từ thầy để có thể cải thiện và hoàn thiện trang web trong các bài tập sau.

Để có cái nhìn chi tiết hơn về trang web và quá trình phát triển, chúng em kính mời thầy đọc chi tiết trong phần trình bày dưới đây. Chúng em rất mong nhận được những ý kiến đóng góp và hướng dẫn từ thầy để nâng cao chất lượng của sản phẩm trong tương lai.

## MỤC LỤC

LỜI CẢM ƠN .....	i
ĐỒ ÁN ĐƯỢC HOÀN THÀNH TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG .....	ii
TÓM TẮT .....	iv
MỤC LỤC .....	1
DANH MỤC HÌNH VẼ, ĐỒ THỊ .....	3
DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT .....	4
CHƯƠNG 1 – TỔNG QUAN .....	5
1.1 Giới thiệu .....	5
1.2 Lý do chọn đề tài .....	5
1.3 Đối tượng sử dụng .....	5
1.4 Các tính năng chính .....	6
CHƯƠNG 2 – TÌM HIỂU CÔNG NGHỆ XÂY DỰNG CHO WEBSITE .....	8
2.1 Express .....	8
2.2 MongoDB .....	9
2.3 Cloudinary và Sharp .....	9
2.4 Passport, Passport-local .....	10
CHƯƠNG 3 – THIẾT KẾ HỆ THỐNG WEBSITE TECH HUT .....	11
3.1 Kiến trúc hệ thống .....	11
3.1.1 Template Engine và Layout: .....	11
3.1.2 Biên Dịch và Xử Lý Ảnh: .....	11
3.1.3 Cơ Sở Dữ Liệu và Bảo Mật: .....	11
3.1.4 Framework và Middleware Web: .....	12
3.1.5 Công Cụ và Tiện Ích Hỗ Trợ: .....	12
3.2 Các sơ đồ diagram .....	13
3.2.1 Sơ đồ Use Case .....	13
3.2.2 Sơ đồ tuần tự (Sequence) .....	15

3.2.3 Sơ đồ mối quan hệ (ERD).....	19
3.2.4 Sơ đồ kiến trúc .....	20
CHƯƠNG 4 – THIẾT KẾ GIAO DIỆN WEBSITE.....	21
4.1 Nguyên tắc thiết kế giao diện:.....	21
4.2 Wireframes .....	22
4.3 Giao diện Người dùng (UI) và mô tả chức năng của từng trang .....	24
4.4 Phần quan trọng (code) .....	28
4.4.1 Phân quyền hệ thống.....	28
4.4.2 Biểu đồ doanh thu .....	29
4.4.3 In hóa đơn .....	30
CHƯƠNG 5 – HIỆN THỰC HỆ THỐNG .....	31
5.1 Triển khai và chạy hệ thống.....	31
5.1.1 Yêu cầu hệ thống .....	31
5.1.2 Triển khai mã nguồn .....	31
5.1.3 Kết nối cơ sở dữ liệu.....	32
5.2 Hướng dẫn sử dụng và lưu ý.....	32
5.3 Video demo .....	32
CHƯƠNG 6 – KẾT LUẬN.....	33
6.1 Tóm tắt .....	33
6.2 Học được .....	33
6.3 Ưu điểm và nhược điểm.....	34
6.4 Hướng phát triển thêm trong tương lai .....	35
TÀI LIỆU THAM KHẢO.....	37
PHỤ LỤC .....	38
Một số hình ảnh khác của trang web.....	41



## **DANH MỤC HÌNH VẼ, ĐỒ THỊ**

Hình 2.1 Giới thiệu về Express trong NodeJS .....	8
Hình 3.1 Use Case Nhân Viên .....	13
Hình 3.2 Use Case Admin.....	14
Hình 3.3 Sơ Đồ Tuần Tự - Quản Lý Tài Khoản và Nhân Viên .....	15
Hình 3.4 Sơ Đồ Tuần Tự - Quản Lý Danh Mục Và Sản Phẩm .....	16
Hình 3.5 Sơ Đồ Tuần Tự - Quản Lý Khách Hàng .....	17
Hình 3.6 Sơ Đồ Tuần Tự - Xử Lý Giao Dịch .....	18
Hình 3.7 Thiết kế hệ thống quản lý bán lẻ ERD .....	19
Hình 3.8 Tổng quan kiến trúc của trang web .....	20
Hình 4.1 Khung sườn cho trang đăng nhập .....	22
Hình 4.2 Khung sườn cho trang chủ .....	22
Hình 4.3 Khung sườn cho trang Check Out.....	23
Hình 4.4 Khung sườn cho trang Report .....	23
Hình 4.5 Giao diện trang đăng nhập .....	24
Hình 4.6 Giao diện trang quên mật khẩu .....	24
Hình 4.7 Giao diện trang chủ Tech Hut .....	25
Hình 4.8 Giao diện trang danh sách sản phẩm.....	25
Hình 4.9 Giao diện trang thanh toán .....	26
Hình 4.10 Giao diện quản lý khách hàng .....	26
Hình 4.11 Giao diện trang quản lý nhân viên .....	27
Hình 4.12 Giao diện trang thông tin cá nhân .....	27
Hình 4.13 Giao diện báo cáo doanh thu .....	28

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

### **CÁC CHỮ VIẾT TẮT**

POS	Point of Sale
JS	Javascript
API	Application Programming Interface
DB	Database
CDN	Content Delivery Network (Mạng Phân Phối Nội Dung)
DDOS	Distributed Denial of Service (Tấn công từ chối dịch vụ)

# CHƯƠNG 1 – TỔNG QUAN

## 1.1 Giới thiệu

Chào mừng thầy đến với bài báo cáo của chúng em! Trong bài báo cáo này, chúng em xin trình bày về dự án trang web Point of Sale (POS) mang tên Tech Hut được phát triển nhằm đáp ứng nhu cầu quản lý và thực hiện giao dịch một cách hiệu quả tại cửa hàng bán lẻ điện thoại và phụ kiện. Mục tiêu hàng đầu của dự án là tạo ra một nền tảng linh hoạt, giúp tối ưu hóa trải nghiệm mua sắm và quản lý cho cả nhân viên bán hàng và quản trị viên.

## 1.2 Lý do chọn đề tài

Chúng em quyết định lựa chọn đề tài này và xây dựng một trang web POS vì trên thực tế, với sự gia tăng nhanh chóng của thị trường và nhu cầu ngày càng đa dạng, việc có một hệ thống POS linh hoạt và dễ sử dụng không chỉ giúp tối ưu hóa hiệu suất mà còn giúp cửa hàng duy trì và phát triển một cách hiệu quả.

Dự án Tech Hut này không chỉ là một cơ hội để thực hành kỹ năng lập trình và phát triển web mà còn là một hướng tiếp cận thực tế để giải quyết vấn đề thực tế trong môi trường kinh doanh đầy thách thức với tụi em. Thông qua trang web, chúng em hy vọng mang lại giá trị cho doanh nghiệp và đồng thời tạo ra một sản phẩm sáng tạo và hữu ích trong lĩnh vực quản lý cửa hàng bán lẻ điện thoại và phụ kiện.

## 1.3 Đối tượng sử dụng

Trang web Tech Hut này được tạo ra với sự tập trung chủ yếu đối với hai đối tượng chính là:

- Nhân Viên Bán Hàng:
  - Thực hiện quy trình bán hàng một cách nhanh chóng và thuận tiện.
  - Dễ dàng quản lý thông tin sản phẩm và giao dịch.
  - Tiếp cận thông tin về khách hàng và lịch sử mua sắm một cách hiệu quả.

- **Quản Trị Viên:**
  - Quản lý tài khoản và quyền truy cập nhân viên một cách linh hoạt.
  - Thêm, sửa, xóa thông tin sản phẩm và nhân viên.
  - Xem và phân tích báo cáo doanh số bán hàng.

#### 1.4 Các tính năng chính

Trang web POS không chỉ giúp quản lý cửa hàng tiến hành bán hàng một cách dễ dàng mà còn mang lại nhiều tính năng quản lý quan trọng, bao gồm:

- **Quản Lý Tài Khoản:** Trong trang web POS, chúng em tập trung vào việc quản lý tài khoản một cách linh hoạt và an toàn. Ngoài ra còn cung cấp khả năng tạo và quản lý tài khoản quản trị viên và nhân viên bán hàng một cách thuận tiện. Đặc biệt, khi tạo tài khoản mới, thì hệ thống sẽ tự động gửi thông báo qua email kèm theo liên kết đăng nhập, giúp nhân viên dễ dàng truy cập vào hệ thống..
- **Quản Lý Sản Phẩm:** Với khả năng quản lý sản phẩm đa dạng, admin có thể xem, thêm mới, cập nhật và xóa thông tin về sản phẩm một cách linh hoạt. Chúng tôi tập trung vào việc hạn chế quyền chỉnh sửa đối với nhân viên bán hàng, giúp đảm bảo tính nhất quán và chính xác của dữ liệu sản phẩm..
- **Quản Lý Khách Hàng:** Quản lý thông tin khách hàng là một phần quan trọng trong hệ thống POS nên chúng em cho phép nhập thông tin khách hàng mới và tự động lưu lại, đồng thời cung cấp khả năng xem thông tin chi tiết và lịch sử mua sắm của khách hàng, giúp cải thiện trải nghiệm mua hàng và tương tác với khách hàng.
- **Xử Lý Giao Dịch:** Để giúp thực hiện cuộc giao dịch một cách nhanh chóng và hiệu quả. Nhân viên có thể thêm sản phẩm vào đơn hàng thông qua tìm kiếm hoặc quét mã vạch. Chúng em cũng cung cấp tính năng tự động tính

tổng giá trị của đơn hàng và hoàn tất quá trình thanh toán một cách thuận tiện nhất.

- **Báo Cáo và Phân Tích:** Để hỗ trợ cho kinh doanh, hệ thống của chúng em còn cung cấp các báo cáo doanh số bán hàng theo thời gian. Từ thông tin tổng quan đến chi tiết về doanh số, admin có thể dễ dàng xem và phân tích, giúp họ điều chỉnh chiến lược kinh doanh và tối ưu hóa hiệu suất bán hàng.

## CHƯƠNG 2 – TÌM HIỂU CÔNG NGHỆ XÂY DỰNG CHO WEBSITE

### 2.1 Express



Hình 2.1 Giới thiệu về Express trong NodeJS

Express<sup>1</sup> là một framework phát triển ứng dụng web bằng ngôn ngữ lập trình JavaScript và được sử dụng chủ yếu trong môi trường Node.js<sup>2</sup>. Nó giúp đơn giản hóa quá trình xây dựng ứng dụng web bằng cách cung cấp một cấu trúc tổ chức linh hoạt và hiệu quả. Với các tính năng như routing mạnh mẽ, middleware linh hoạt, và khả năng xử lý request và response một cách dễ dàng, Express là lựa chọn phổ biến cho việc phát triển cả ứng dụng web và API. Nó tối ưu cho các dự án từ nhỏ đến lớn và cung cấp sự linh hoạt để tích hợp với nhiều loại cơ sở dữ liệu và frontend frameworks. Nhờ vào sự đơn giản và tích hợp mạnh mẽ, Express đóng vai trò quan trọng trong việc nhanh chóng xây dựng các ứng dụng web hiệu quả.

---

<sup>1</sup> "Express - Node.js Web Application Framework."

<sup>2</sup> "Node.js."

## 2.2 MongoDB

MongoDB<sup>3</sup> là một hệ quản trị cơ sở dữ liệu NoSQL phổ biến, thường được sử dụng kết hợp với Express khi phát triển ứng dụng web. Được biết đến với tính nhất quán và khả năng mở rộng, MongoDB cung cấp một cơ sở dữ liệu linh hoạt và không yêu cầu schema nào cố định. Với khả năng mở rộng và hỗ trợ tốt cho dữ liệu không gian, MongoDB là lựa chọn phổ biến cho việc lưu trữ và quản lý dữ liệu trong các ứng dụng web hiện đại. Cộng đồng phát triển đông đảo và các công cụ như Mongoose giúp tương tác dễ dàng hơn với cơ sở dữ liệu MongoDB trong môi trường Node.js.

## 2.3 Cloudinary và Sharp

Cloudinary<sup>4</sup> và Sharp<sup>5</sup> là hai công nghệ quan trọng thường được sử dụng đồng thời trong phát triển ứng dụng web để quản lý và xử lý hình ảnh. Cloudinary là một dịch vụ đám mây cung cấp giải pháp lưu trữ và quản lý đa phương tiện trực tuyến, tập trung chủ yếu vào hình ảnh và video. Nó tối ưu hóa hình ảnh, hỗ trợ xử lý ảnh động và tĩnh, và quản lý quyền truy cập một cách an toàn.

Còn Sharp, một thư viện xử lý hình ảnh trong Node.js, là một công cụ hiệu quả và mạnh mẽ, thích hợp cho các yêu cầu về hiệu suất cao. Nó hỗ trợ nhiều chức năng như thay đổi kích thước, chuyển đổi định dạng và thêm văn bản, đồng thời có đa luồng để xử lý đồng thời nhiều hình ảnh.

Khi kết hợp cả hai công nghệ này, thì thường sử dụng Sharp để thực hiện các thao tác xử lý hình ảnh mạnh mẽ trước khi tải lên Cloudinary. Điều này giúp tối ưu hóa hình ảnh, đảm bảo chúng đáp ứng tốt với yêu cầu về kích thước, chất lượng và tốc độ tải trang. Cloudinary sau đó quản lý lưu trữ, tối ưu hóa và phục vụ hình ảnh qua CDN một cách

---

<sup>3</sup> "MongoDB: The Developer Data Platform | MongoDB."

<sup>4</sup> "Image and Video Upload, Storage, Optimization and CDN."

<sup>5</sup> "Sharp."

hiệu quả. Sự kết hợp giữa Cloudinary và Sharp mang lại sự linh hoạt và hiệu suất cho việc quản lý đa phương tiện trong ứng dụng web.

## 2.4 Passport, Passport-local

Passport<sup>6</sup> và Passport-local đều là các thư viện quan trọng trong phát triển ứng dụng web để quản lý xác thực và thông tin đăng nhập của người dùng. Passport là một middleware xác thực linh hoạt, hỗ trợ nhiều chiến lược xác thực và tích hợp dễ dàng vào ứng dụng Express. Trong khi đó, Passport-local là một chiến lược xác thực cụ thể, tập trung vào việc xác thực thông tin đăng nhập local từ username và password.

Khi kết hợp cả hai, người phát triển thường sử dụng Passport để quản lý quá trình xác thực chung và tích hợp Passport-local như một chiến lược con để xác thực thông tin đăng nhập cụ thể từ người dùng. Sự kết hợp này mang lại linh hoạt và thuận tiện trong quá trình xác thực, giữ cho quản lý thông tin đăng nhập trở nên an toàn và hiệu quả.

---

<sup>6</sup> "Passport."



## CHƯƠNG 3 – THIẾT KẾ HỆ THỐNG WEBSITE TECH HUT

### 3.1 Kiến trúc hệ thống

Ở trong trang web của chúng em, như những bước xây nhà, đều cần có các công cụ và vật liệu để tạo ra một căn nhà. Ở đây, các công cụ và vật liệu là những công nghệ, phần mềm, modules và cách chúng tương tác với nhau giống như cách chúng ta xây dựng căn nhà. Dưới đây là chi tiết về các thành phần chính mà trang web sử dụng:

#### 3.1.1 *Template Engine và Layout:*

- **EJS<sup>7</sup> (Embedded JavaScript):** Là một template engine giúp xây dựng giao diện người dùng một cách dễ dàng và linh hoạt.
- **express-ejs-layouts:** Đây là một middleware hỗ trợ việc quản lý layout cho EJS, giúp tái sử dụng code và quản lý cấu trúc trang web hiệu quả hơn.

#### 3.1.2 *Biên Dịch và Xử Lý Ảnh:*

- **node-sass-middleware:** Dùng để biên dịch mã nguồn SASS<sup>8</sup> thành CSS, giúp quản lý mã nguồn và tăng khả năng tái sử dụng.
- **multer:** Là một middleware giúp xử lý việc tải lên (upload) file từ client lên server.
- **sharp:** Thư viện mạnh mẽ giúp điều chỉnh kích thước, nén ảnh và giảm dung lượng trước khi tải lên Cloudinary, tối ưu hóa hiệu suất.

#### 3.1.3 *Cơ Sở Dữ Liệu và Bảo Mật:*

- **mongoose:** Được sử dụng để kết nối và tương tác với MongoDB, cung cấp giao diện linh hoạt để thao tác với dữ liệu.
- **connect-mongo:** Lưu trữ session vào MongoDB, giúp duy trì trạng thái đăng nhập của người dùng.
- **passport:** Là một framework cho xác thực, cung cấp cơ sở để xây dựng các chiến lược xác thực khác nhau.

---

<sup>7</sup> "EJS -- Embedded JavaScript Templates."

<sup>8</sup> "Sass: Syntactically Awesome Style Sheets."

- **passport-local:** Sử dụng để xác thực tại local, không sử dụng xác thực qua bên thứ ba.
- **passport-local-mongoose:** Tích hợp Passport và chiến lược local với MongoDB, giúp quản lý tài khoản người dùng hiệu quả.
- **bcryptjs:** Sử dụng để hash mật khẩu, cung cấp bảo mật cho thông tin đăng nhập.
- **cookie-session:** Quản lý session thông qua cookie, giúp theo dõi trạng thái đăng nhập của người dùng.

#### 3.1.4 *Framework và Middleware Web:*

- **express:** Là một framework web Node.js giúp xây dựng ứng dụng web nhanh chóng và hiệu quả.
- **cors:** Middleware cho phép truy cập cross-origin, quan trọng cho việc tương tác giữa các domain khác nhau.
- **express-validator:** Thư viện giúp kiểm tra và xác nhận dữ liệu đầu vào từ client, ngăn chặn các lỗ hổng bảo mật.
- **toastr<sup>9</sup>:** Được sử dụng để hiển thị flash message, thông báo cho người dùng về các tình trạng hoạt động.

#### 3.1.5 *Công Cụ và Tiện Ích Hỗ Trợ:*

- **dotenv:** Quản lý biến môi trường, giúp cấu hình ứng dụng dễ dàng và bảo mật.
- **currency.js:** Định dạng tiền tệ, hỗ trợ xử lý và hiển thị số tiền một cách chính xác.
- **moment:** Thư viện quản lý thời gian, giúp định dạng và xử lý thông tin thời gian một cách linh hoạt.
- **winston<sup>10</sup>:** Thư viện logging, cung cấp khả năng ghi log mạnh mẽ giúp theo dõi và debug ứng dụng.

---

<sup>9</sup> "Toastr."

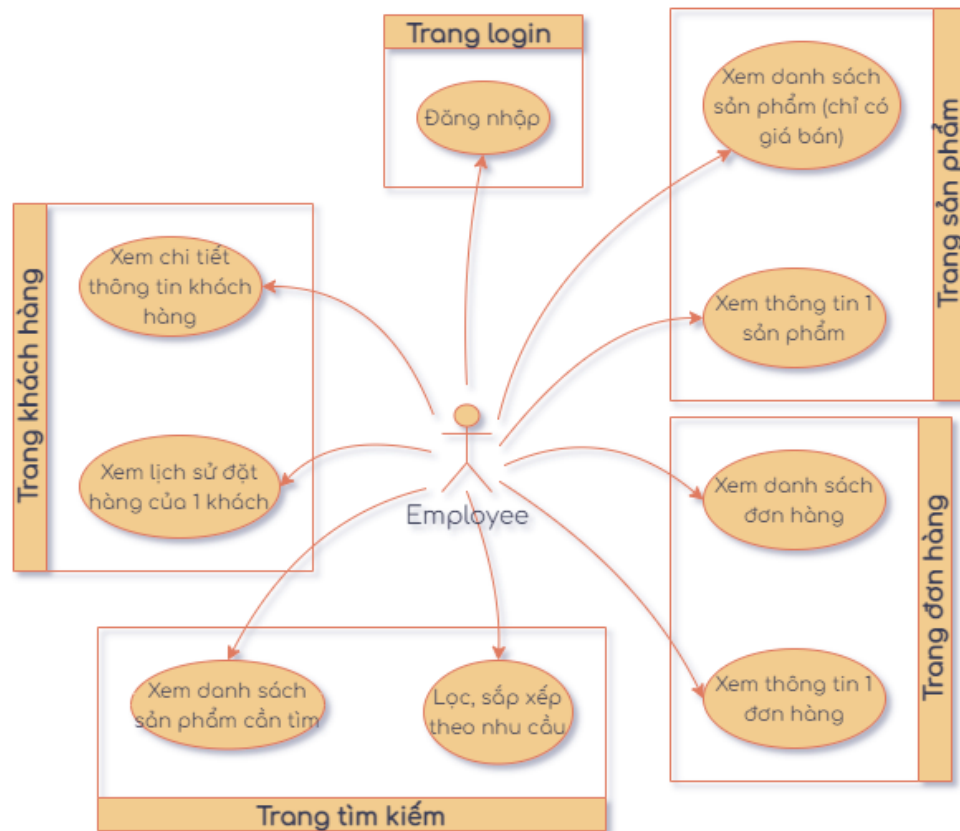
<sup>10</sup> "Winston."

- **nodemailer<sup>11</sup>**: Thư viện gửi email, hỗ trợ chức năng gửi thông báo qua email trong ứng dụng.
- **express-rate-limit**: Middleware giúp hạn chế số lượng request từ một nguồn cụ thể, bảo vệ ứng dụng khỏi tấn công DDOS.
- **cloudinary**: Dịch vụ lưu trữ ảnh trên cloud, giúp quản lý và phục vụ hình ảnh một cách hiệu quả.

### 3.2 Các sơ đồ diagram

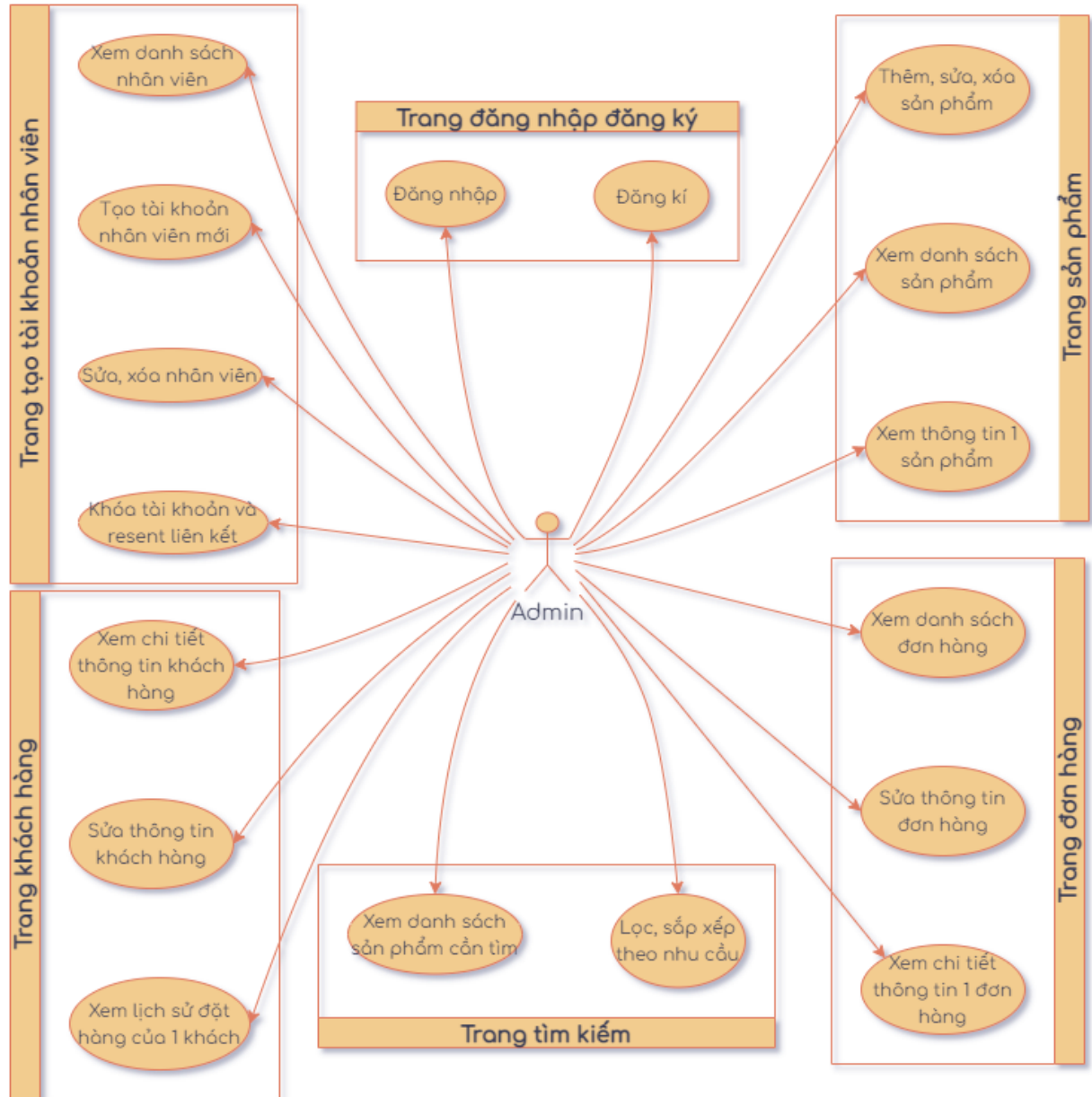
Để cho việc hình dung cách hoạt động của trang web được rõ hơn sau đây là các sơ đồ diagram của toàn dự án.

#### 3.2.1 Sơ đồ Use Case



Hình 3.1 Use Case Nhân Viên

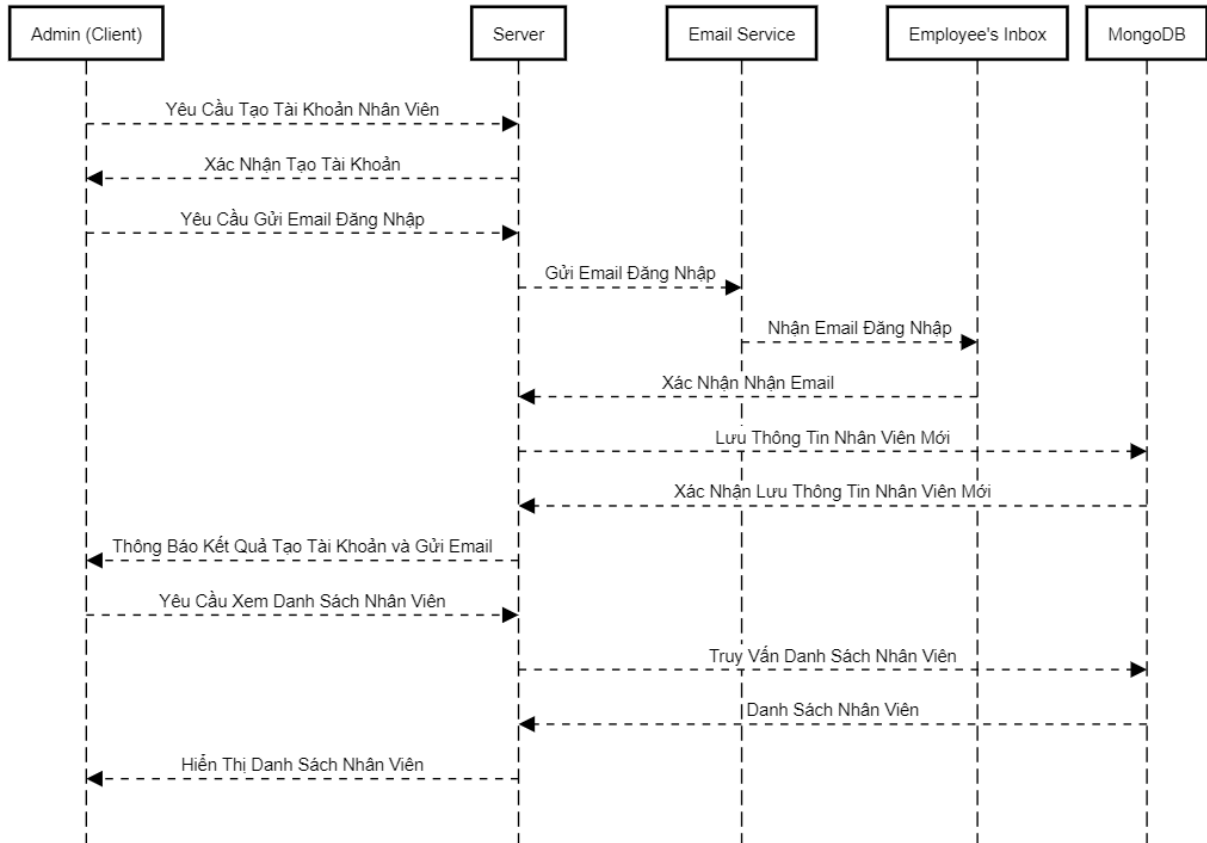
<sup>11</sup> "Nodemailer :: Nodemailer."



Hình 3.2 Use Case Admin

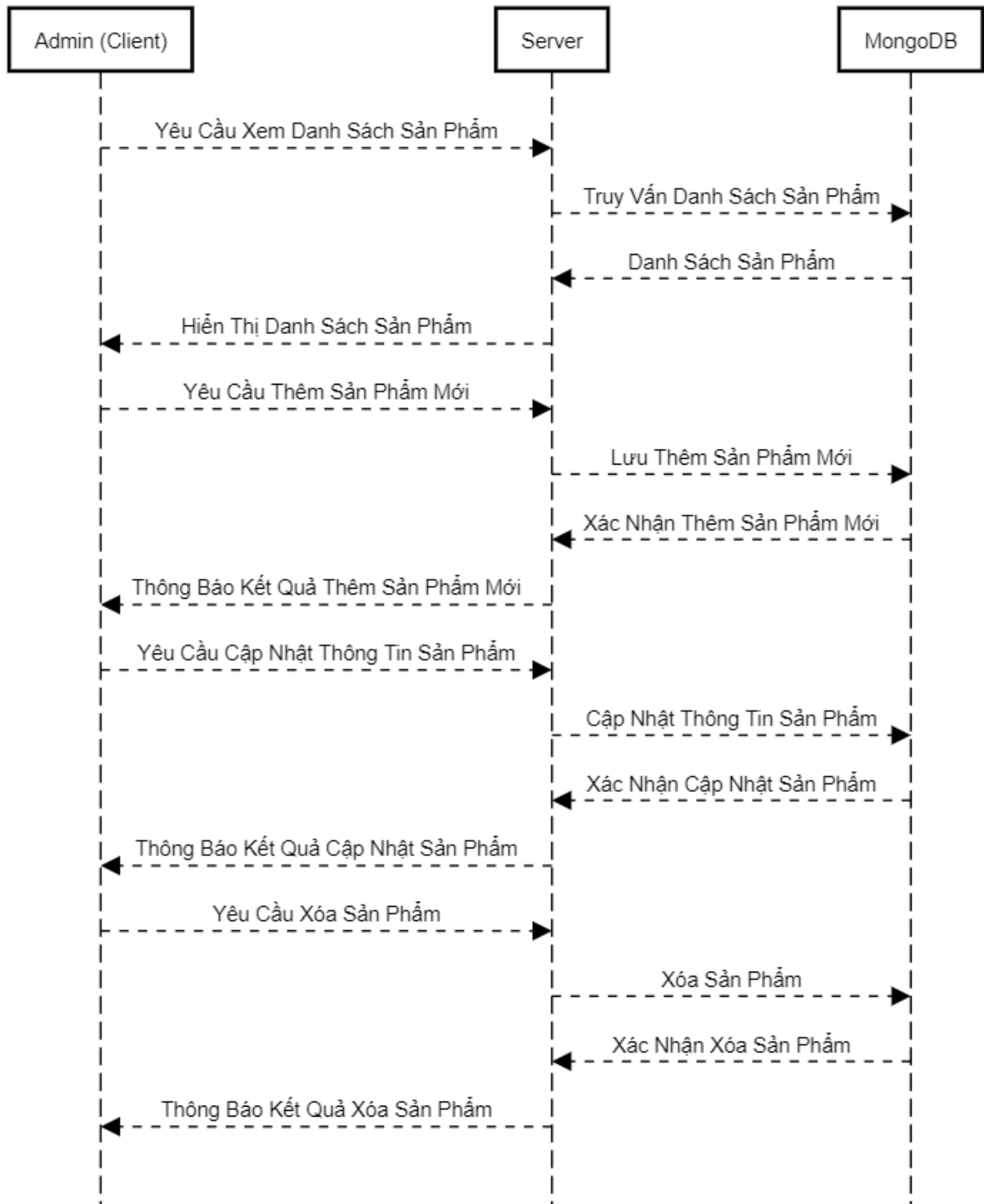
### 3.2.2 Sơ đồ tuần tự (Sequence)

Sơ Đồ Tuần Tự - Quản Lý Tài Khoản và Nhân Viên(Admin)



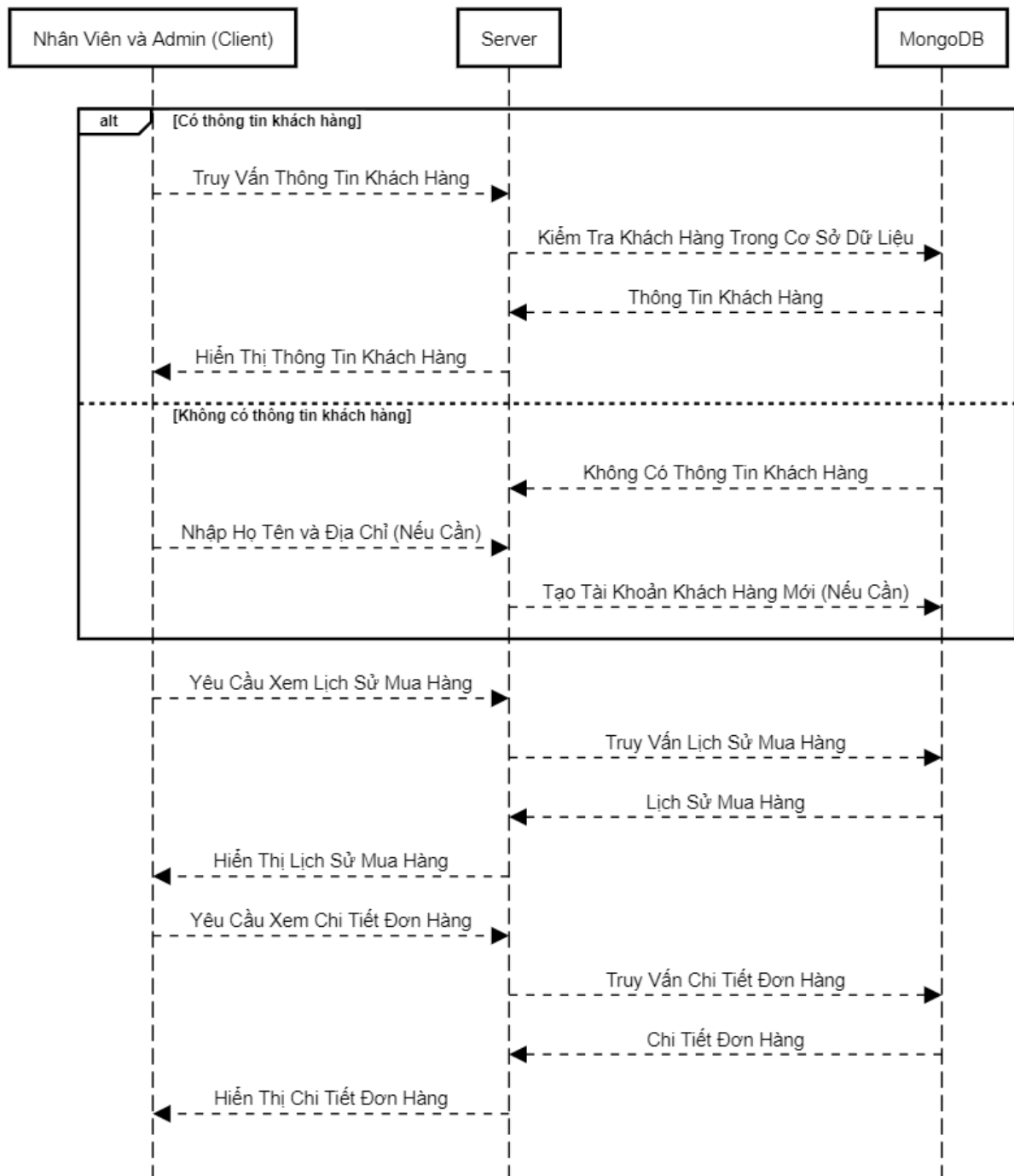
Hình 3.3 Sơ Đồ Tuần Tự - Quản Lý Tài Khoản và Nhân Viên

### Sơ Đồ Tuần Tự - Quản Lý Danh Mục Sản Phẩm (Admin)

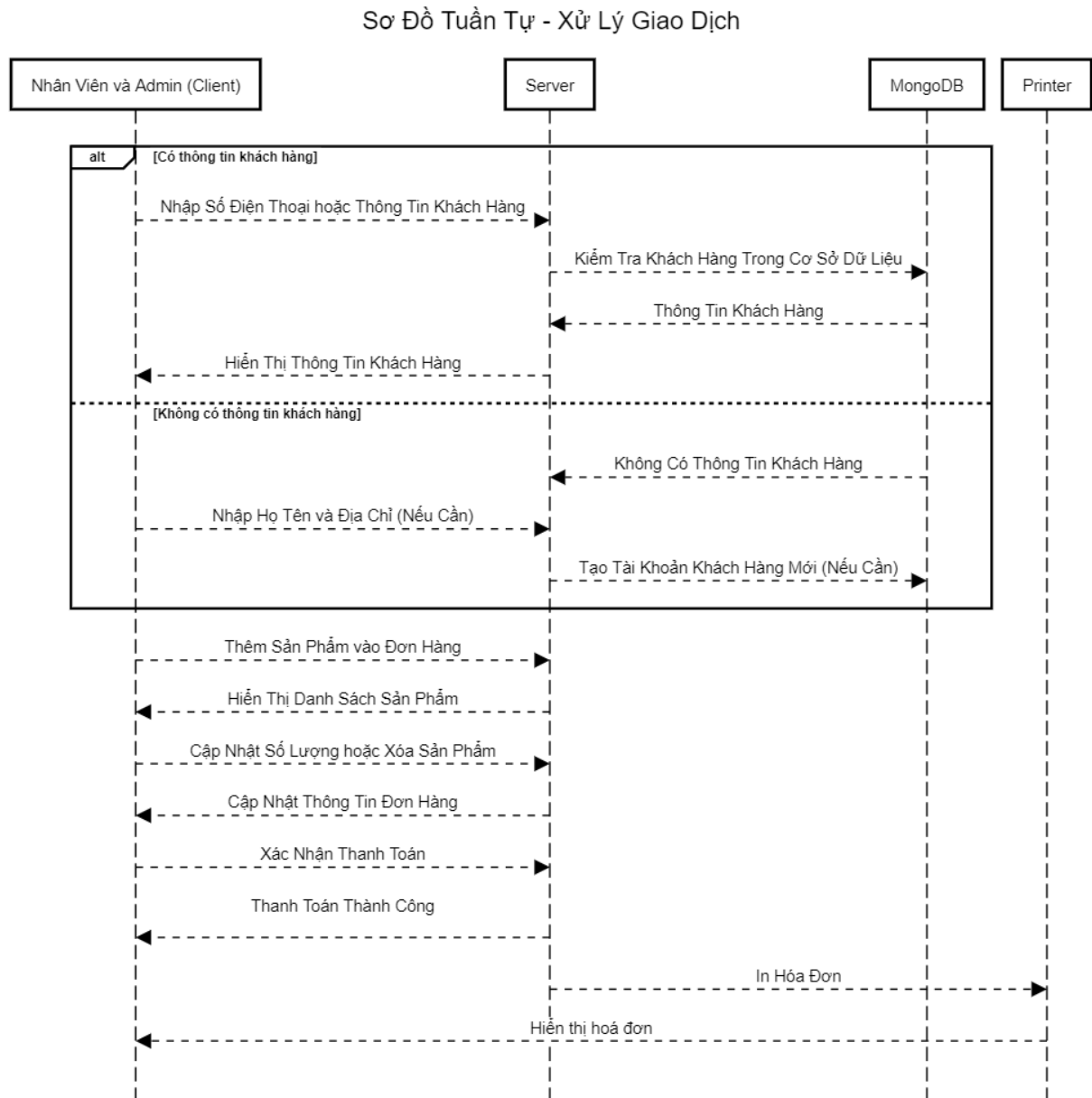


Hình 3.4 Sơ Đồ Tuần Tự - Quản Lý Danh Mục Và Sản Phẩm

### Sơ Đồ Tuần Tự - Quản Lý Khách Hàng



Hình 3.5 Sơ Đồ Tuần Tự - Quản Lý Khách Hàng



Hình 3.6 Sơ Đồ Tuần Tự - Xử Lý Giao Dịch

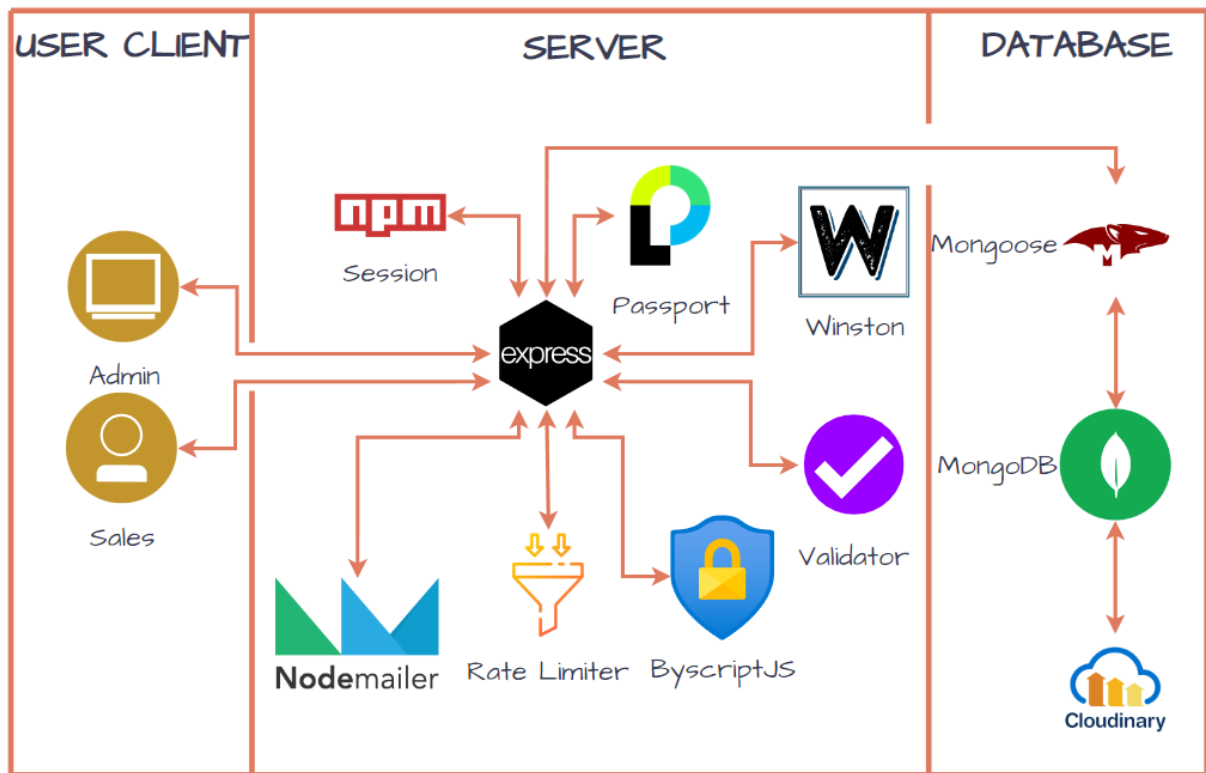


### 3.2.3 Sơ đồ mối quan hệ (ERD)



Hình 3.7 Thiết kế hệ thống quản lý bán lẻ ERD

### 3.2.4 Sơ đồ kiến trúc



Hình 3.8 Tổng quan kiến trúc của trang web

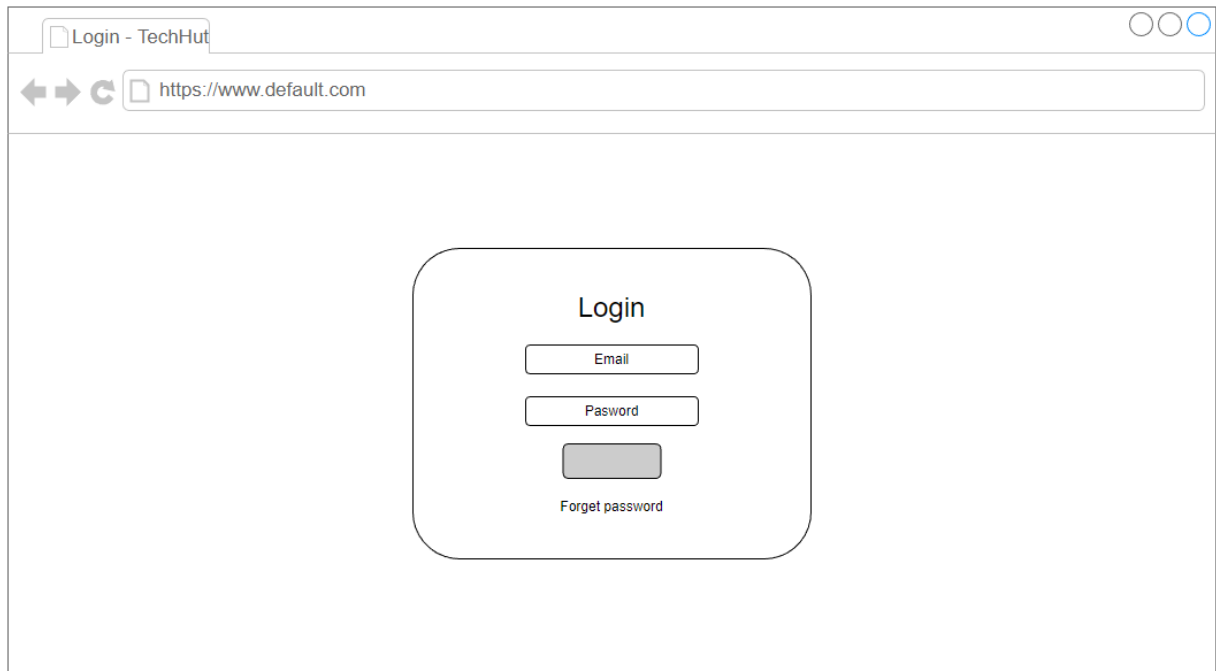
## CHƯƠNG 4 – THIẾT KẾ GIAO DIỆN WEBSITE

### 4.1 Nguyên tắc thiết kế giao diện:

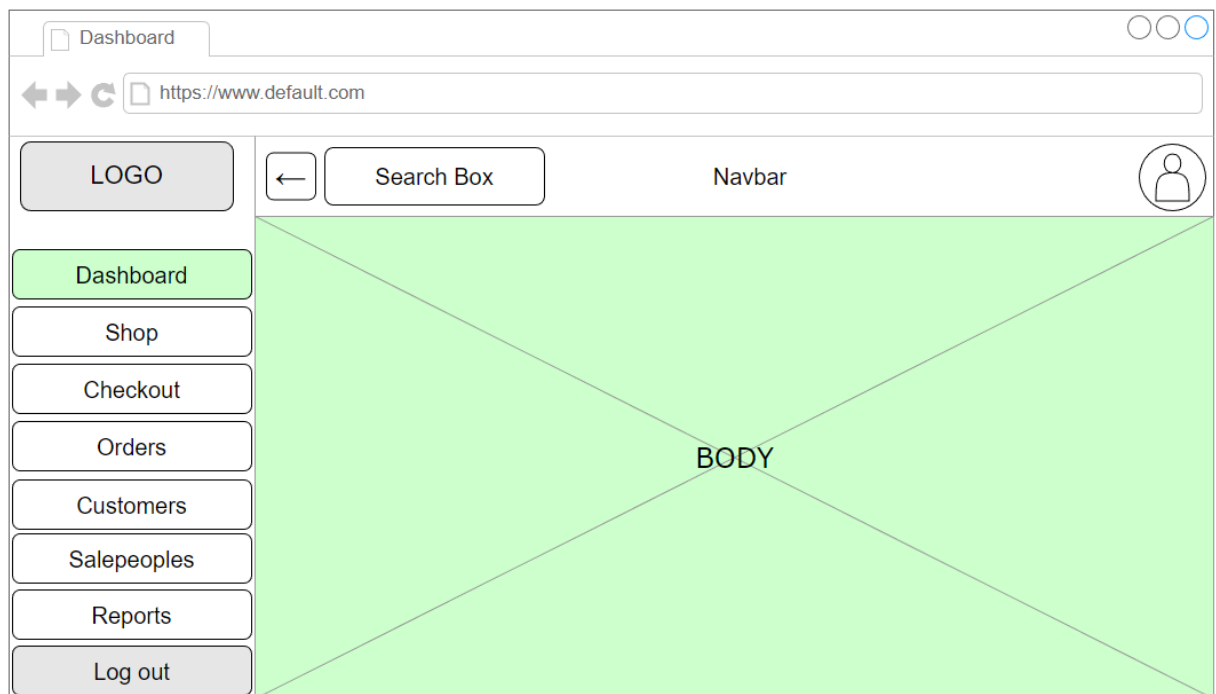
Trong quá trình thiết kế giao diện cho trang web “Tech Hut”, chúng em sẽ tuân theo một số nguyên tắc cơ bản nhằm cải thiện trải nghiệm người dùng như sau:

- **Đơn giản và dễ dùng:** Giao diện cần được thiết kế một cách đơn giản để người dùng dễ sử dụng mà không gặp phải sự phức tạp không cần thiết.
- **Tương tác dễ dàng:** Giao diện cần hỗ trợ các tương tác người dùng thân thiện và tự nhiên, đặc biệt là trong việc thiết lập và tham gia cuộc gọi video mà không cần thiết phải cài đặt gì thêm.
- **Đáp ứng nhiều thiết bị (Responsive):** Giao diện phải có khả năng đáp ứng để tương thích trên nhiều thiết bị khác nhau, bao gồm cả máy tính và thiết bị di động có thể tương tác với nhau.
- **Sắp xếp bố cục hợp lý:** Giúp người dùng dễ dàng tìm thấy thông tin họ cần, đặc biệt là cho nhân viên mới.
- **Cung cấp nội dung chất lượng:** Nội dung chất lượng giúp tăng tính thẩm mỹ và chuyên nghiệp cho trang web.

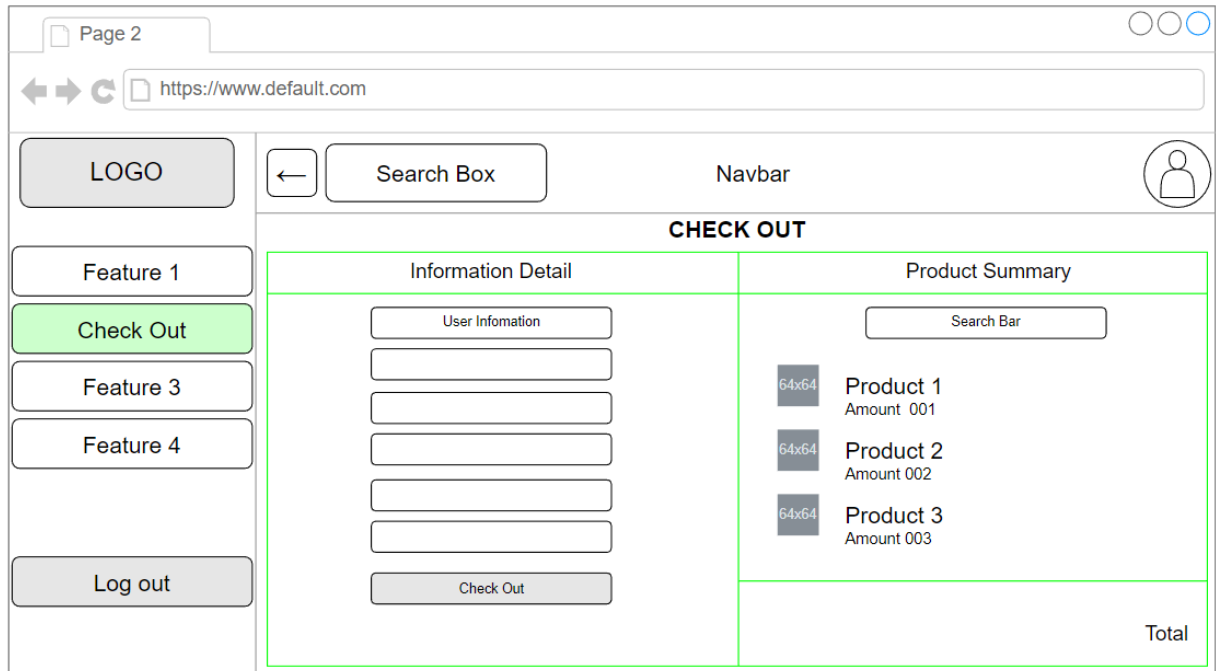
## 4.2 Wireframes



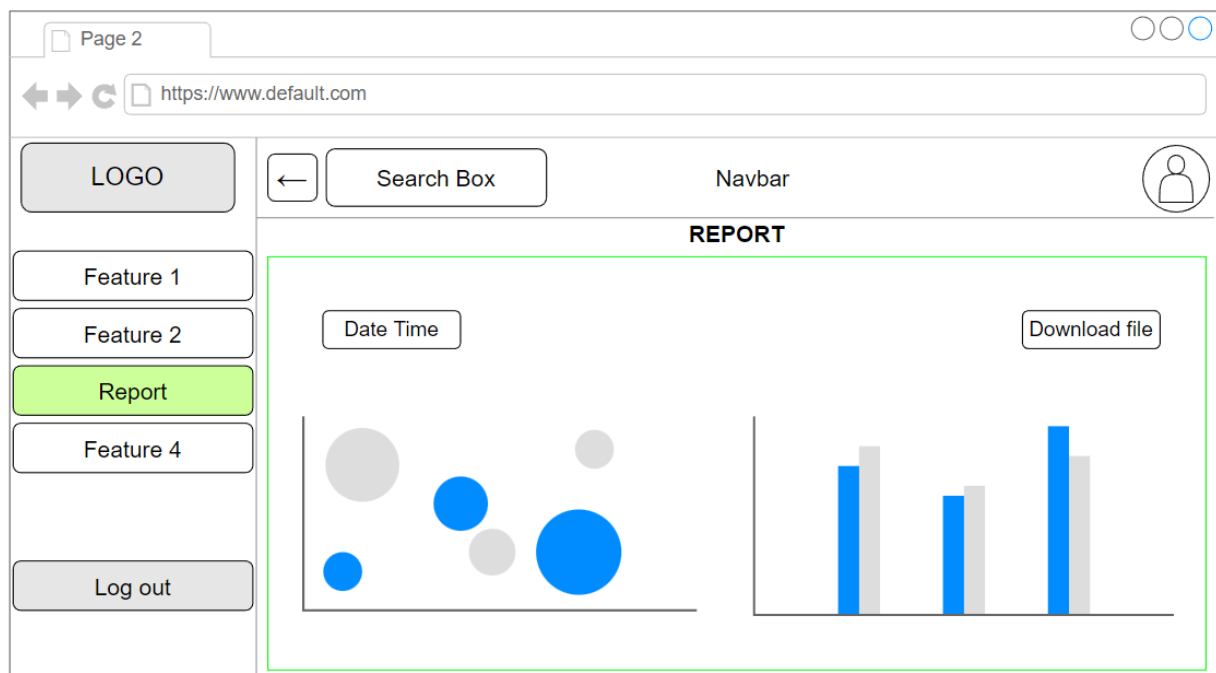
Hình 4.1 Khung sườn cho trang đăng nhập



Hình 4.2 Khung sườn cho trang chủ

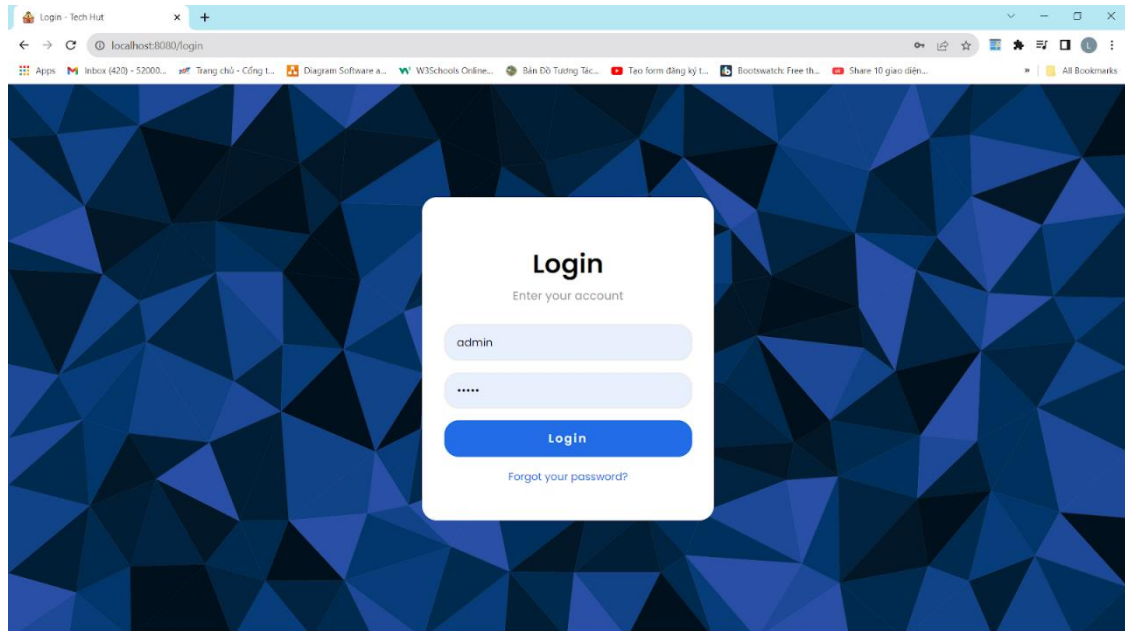


Hình 4.3 Khung sườn cho trang Check Out

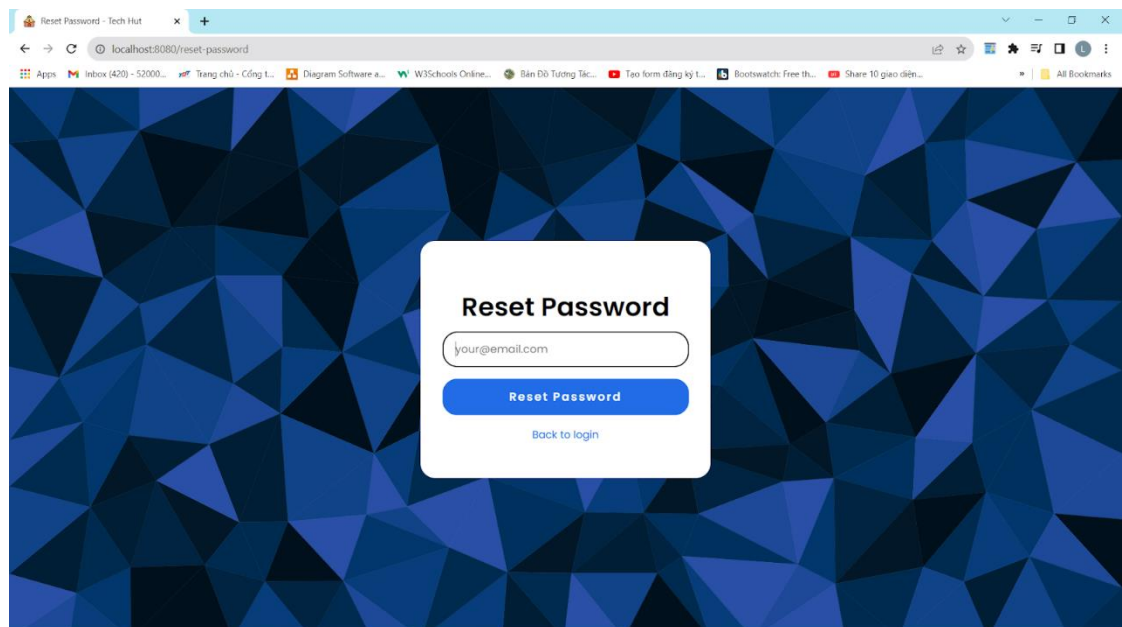


Hình 4.4 Khung sườn cho trang Report

### 4.3 Giao diện Người dùng (UI) và mô tả chức năng của từng trang

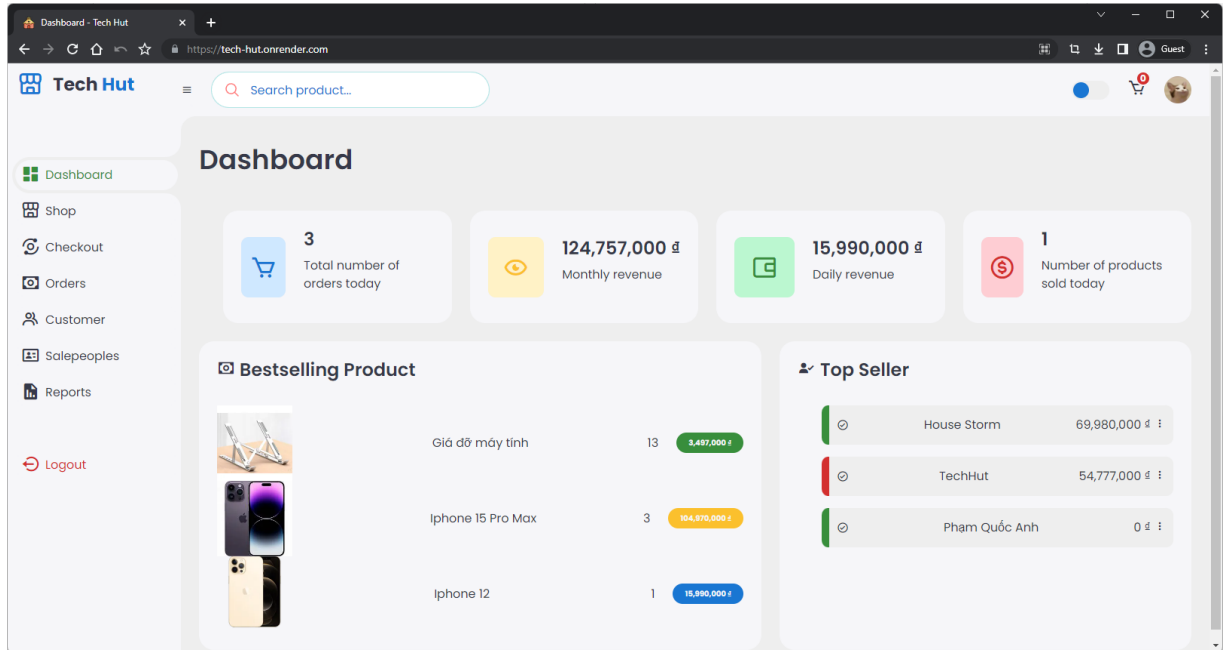


Hình 4.5 Giao diện trang đăng nhập

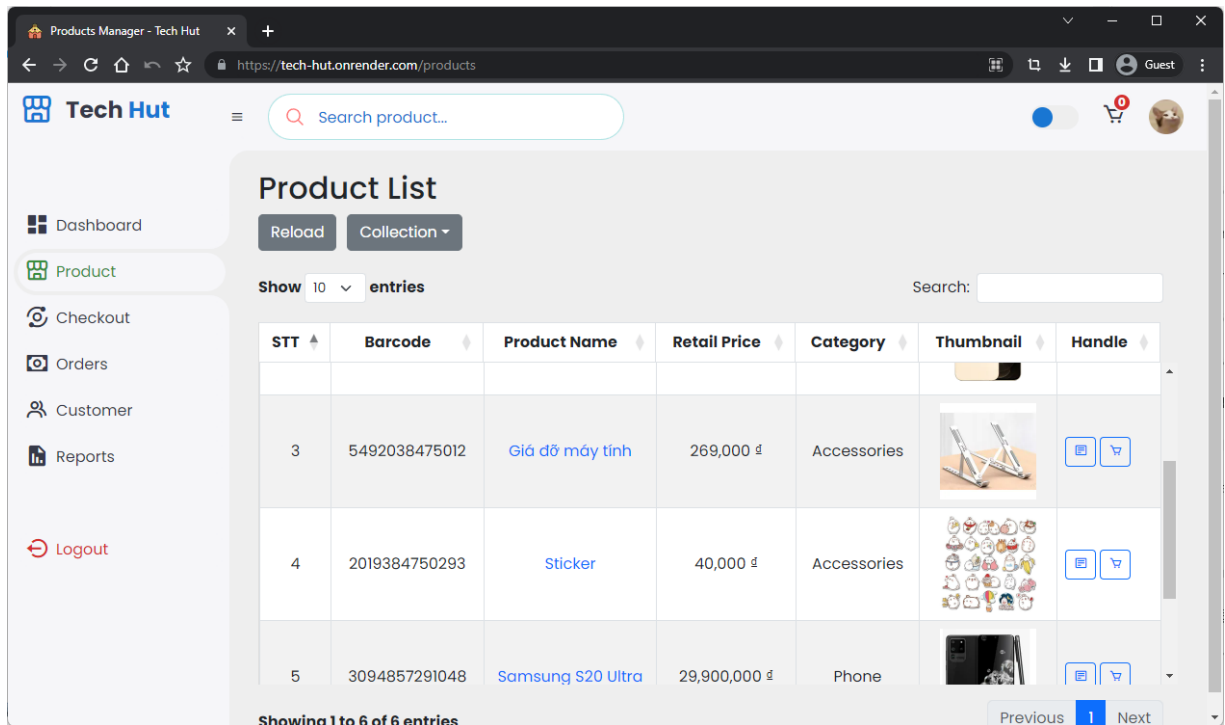


Hình 4.6 Giao diện trang quên mật khẩu

**Quên mật khẩu:** Nếu người dùng quên tài khoản hay mật khẩu thì có thể lấy lại bằng cách nhập email và xác nhận.

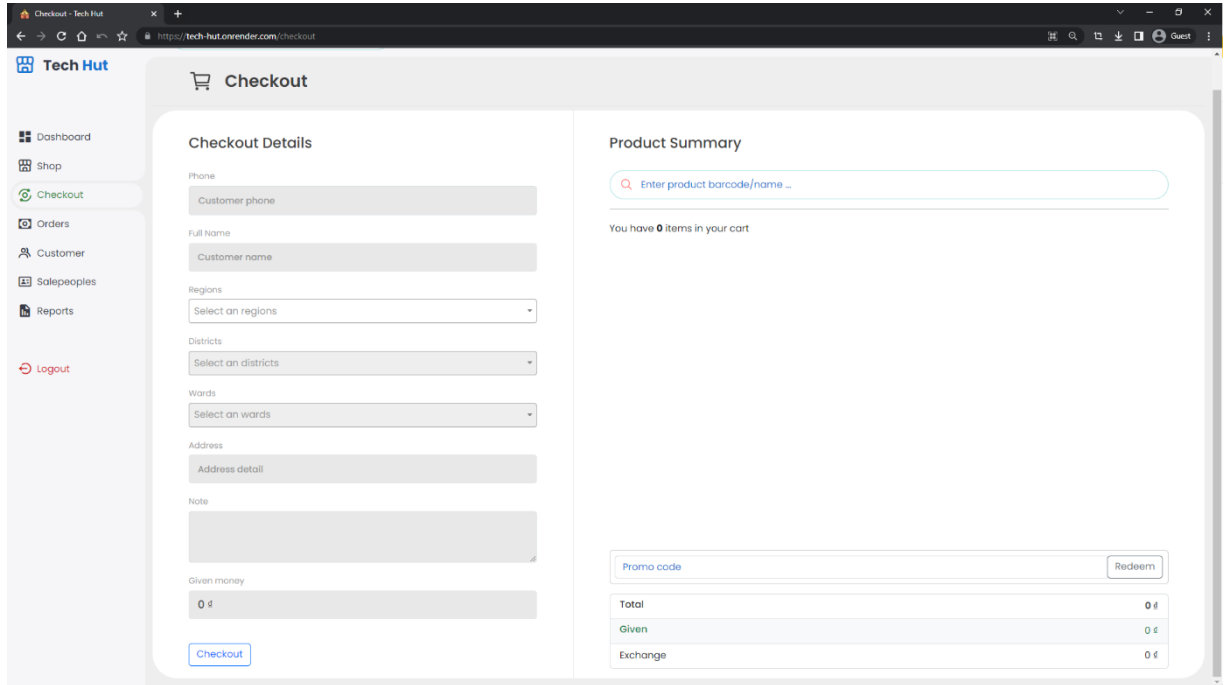


Hình 4.7 Giao diện trang chủ Tech Hut



Hình 4.8 Giao diện trang danh sách sản phẩm

**Lưu ý:** Nhân viên (Sale) sẽ không thấy được danh sách nhân viên.



**Checkout**

**Checkout Details**

Phone  
Customer phone

Full Name  
Customer name

Regions  
Select an regions

Districts  
Select an districts

Wards  
Select an wards

Address  
Address detail

Note

Given money  
0 ₫

**Product Summary**

Enter product barcode/name ...

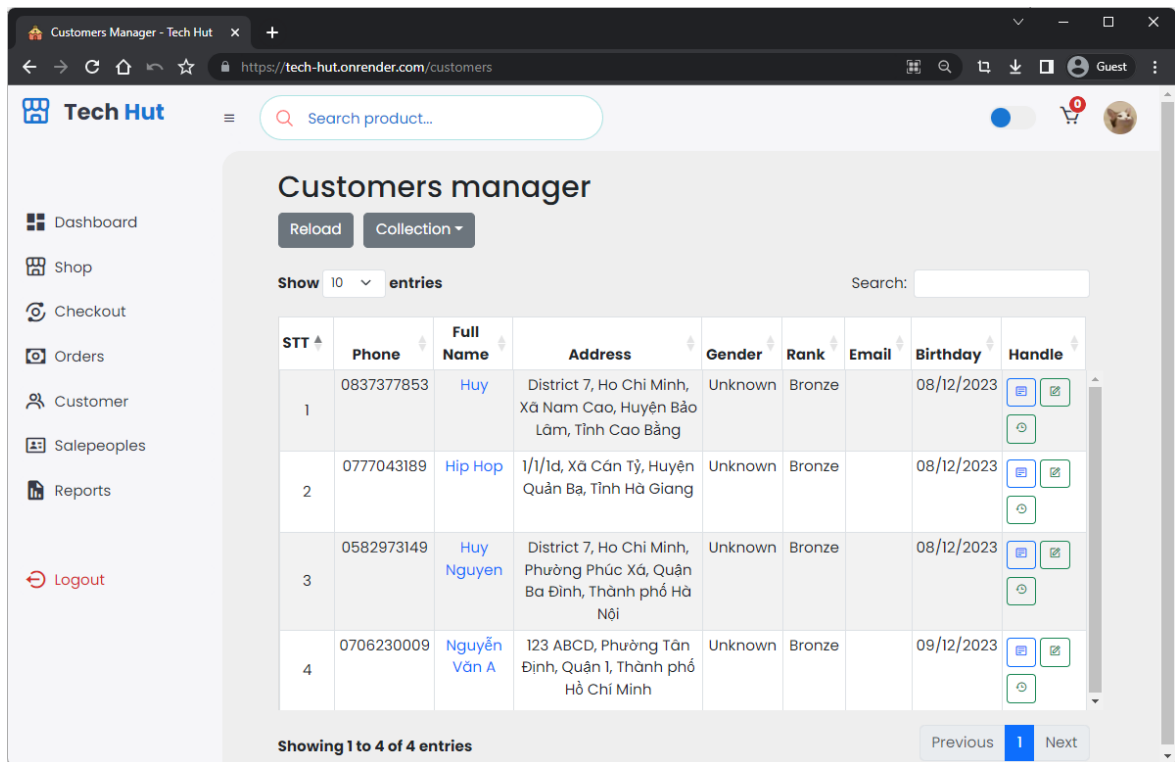
You have 0 items in your cart

Promo code Redeem

Total	0 ₫
Given	0 ₫
Exchange	0 ₫

Checkout

Hình 4.9 Giao diện trang thanh toán



**Customers manager**

Reload Collection

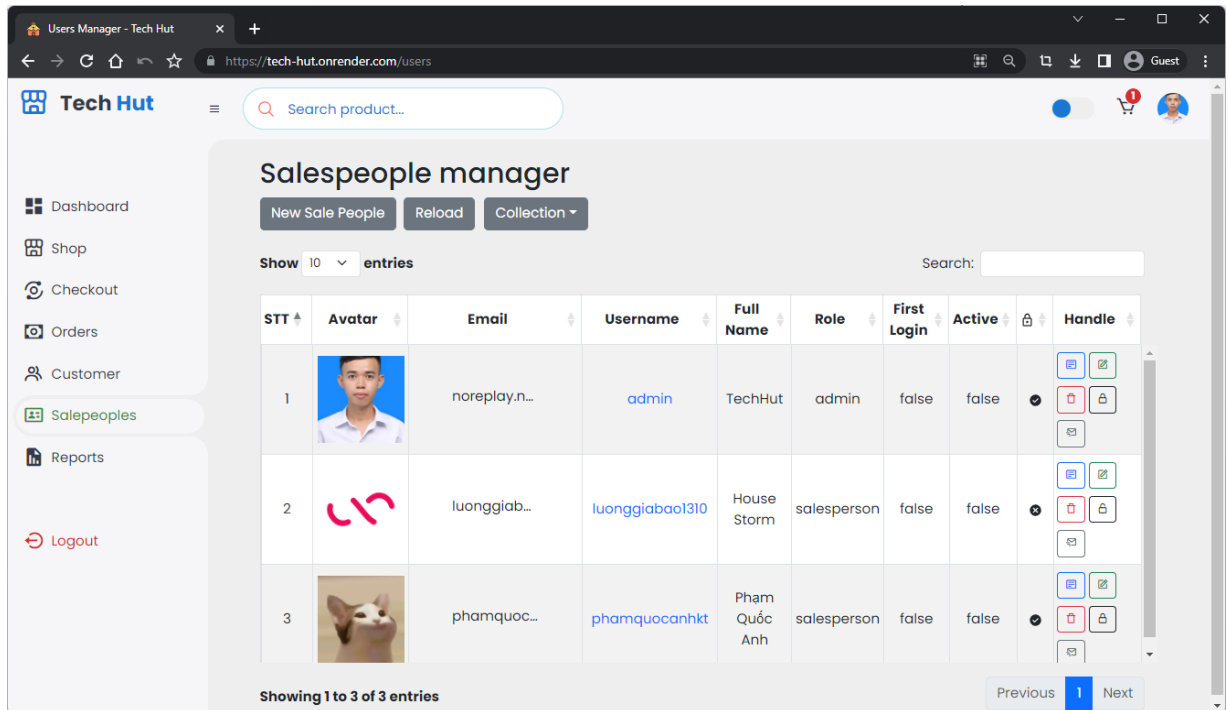
Show 10 entries Search:

STT	Phone	Full Name	Address	Gender	Rank	Email	Birthday	Handle
1	0837377853	Huy	District 7, Ho Chi Minh, Xã Nam Cao, Huyện Bảo Lâm, Tỉnh Cao Bằng	Unknown	Bronze		08/12/2023	[Edit] [Delete] [Add]
2	0777043189	Hip Hop	1/1/1d, Xã Cán Tỷ, Huyện Quận Ba, Tỉnh Hà Giang	Unknown	Bronze		08/12/2023	[Edit] [Delete] [Add]
3	0582973149	Huy Nguyen	District 7, Ho Chi Minh, Phường Phúc Xá, Quận Ba Đình, Thành phố Hà Nội	Unknown	Bronze		08/12/2023	[Edit] [Delete] [Add]
4	0706230009	Nguyễn Văn A	123 ABCD, Phường Tân Đình, Quận 1, Thành phố Hồ Chí Minh	Unknown	Bronze		09/12/2023	[Edit] [Delete] [Add]

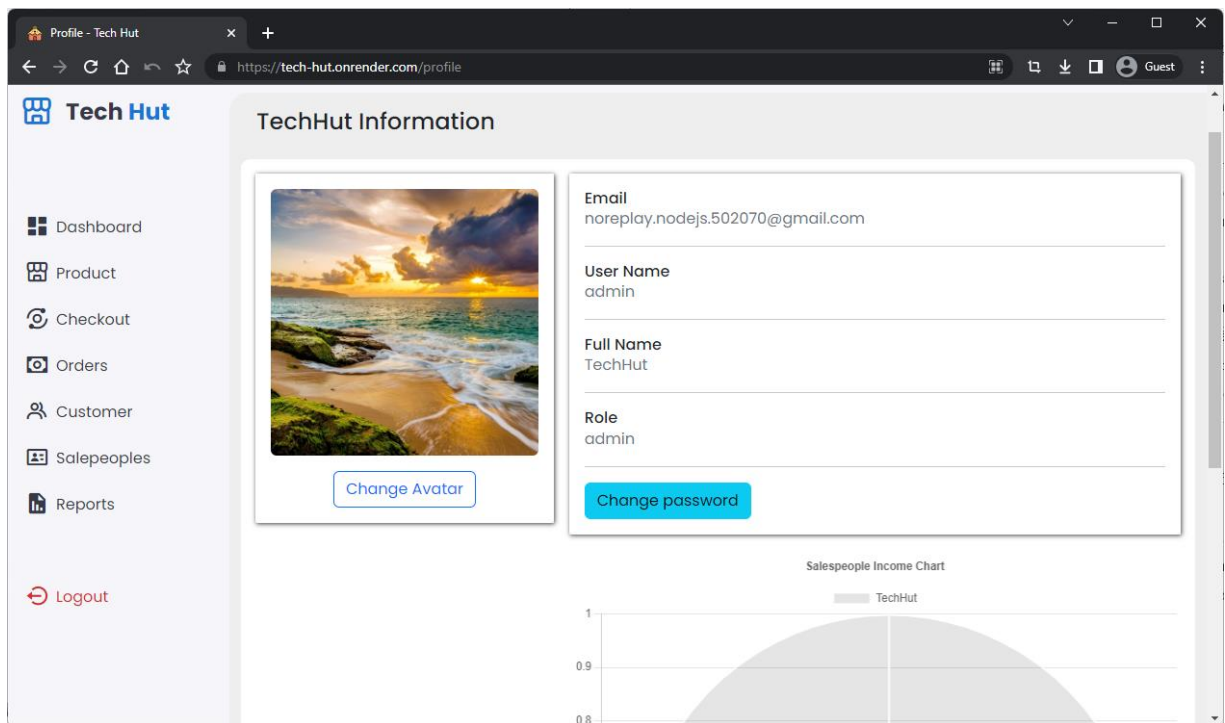
Showing 1 to 4 of 4 entries Previous 1 Next

Hình 4.10 Giao diện quản lý khách hàng

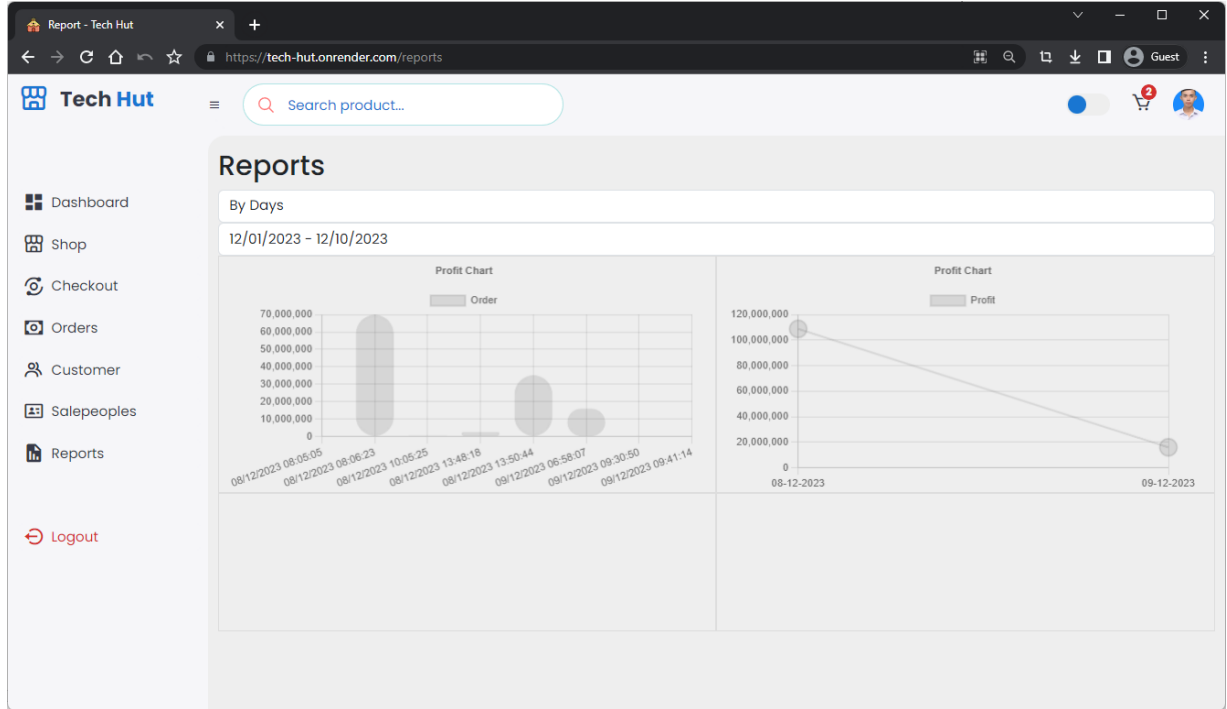




Hình 4.11 Giao diện trang quản lý nhân viên



Hình 4.12 Giao diện trang thông tin cá nhân



Hình 4.13 Giao diện báo cáo doanh thu

## 4.4 Phần quan trọng (code)

### 4.4.1 Phân quyền hệ thống

```
router
  .use(limiter)
  // auth router
  .use(authRoutes)
  .use(ensureAuthenticated)
  .get("/search/address", searchController.searchAddress)
  .use(updateCurrentUser)
  .get("/change-password", authController.getChangePassword)
  .post("/change-password", validationChangePassword,
    authController.postChangePassword)
```

*Giải thích:* Trong đoạn code trên, quá trình xử lý yêu cầu được phân chia một cách tổ chức và an toàn. Đặc biệt, đoạn quan trọng nhất là việc sử dụng middleware **ensureAuthenticated**. Điều này đảm bảo rằng chỉ những người dùng đã xác thực (là

nhân viên hoặc admin) mới được phép truy cập vào các tính năng chính của trang web. Nếu người dùng chưa đăng nhập, hệ thống sẽ chuyển hướng người dùng đến trang đăng nhập. Điều này làm tăng cường bảo mật của ứng dụng và giúp duy trì tính toàn vẹn của dữ liệu và chức năng.

#### 4.4.2 Biểu đồ doanh thu

```
fetch(/reports/statistical?from=${startDate}&to=${endDate})
  .then(response => response.json())
  .then(response => {
    const labels = response.data.map(entry => entry.purchaseDate);
    const data = response.data.map(entry => entry.totalAmount);
    const ctx1 = document.getElementById('salesChart').getContext('2d');

    new Chart(ctx1, { type: 'bar', data: { labels, datasets: [{ label:
'Order', data }] }, options });
  });
```

*Giải thích:* Đoạn code trên là một phần quan trọng của quá trình hiển thị biểu đồ thống kê trong ứng dụng. Nó thực hiện việc gửi yêu cầu tới endpoint `/reports/statistical` với các tham số `from` và `to`, đại diện cho khoảng thời gian bắt đầu và kết thúc. Khi nhận được phản hồi từ server, nó trích xuất dữ liệu cần thiết như ngày mua hàng và tổng giá trị đơn hàng.

Sau đó, nó sử dụng thư viện Chart.js <sup>12</sup> để tạo biểu đồ cột, trong đó các nhãn là các ngày mua hàng và dữ liệu là tổng giá trị của mỗi đơn hàng. Việc này giúp hiển thị một cách trực quan sự phân phối và xu hướng của doanh số bán hàng trong khoảng thời gian được chọn. Biểu đồ này không chỉ mang lại cái nhìn tổng quan về hiệu suất bán hàng mà còn hỗ trợ quá trình ra quyết định và phân tích chi tiết về các ngày quan trọng.

---

<sup>12</sup> "Chart.js."

#### 4.4.3 In hóa đơn

```
// HTML
<iframe src="" id="frMauIn" class="d-none"
name="frMauIn"></iframe>
// JavaScript
$('#frMauIn').attr('src', '/checkout/recipe/${response.order._id}');
setTimeout(function () {
    window.frames["frMauIn"].focus();
    window.frames["frMauIn"].print();
}, 3000);
```

*Giải thích:* Đoạn code trên là một phần quan trọng của quá trình in hóa đơn trong trang web. Khi nhân viên hoặc admin thực hiện thanh toán thành công, mã nguồn sử dụng một thẻ iframe để tạo ra một nơi chứa để hiển thị và in hóa đơn. Thông qua JavaScript, src của iframe được cập nhật để trỏ đến đường dẫn `/checkout/recipe/${response.order._id}`, nơi server sẽ xử lý và trả về một trang HTML của hóa đơn tương ứng với đơn hàng vừa được thanh toán.

## CHƯƠNG 5 – HIỆN THỰC HỆ THỐNG

### 5.1 Triển khai và chạy hệ thống

#### 5.1.1 Yêu cầu hệ thống

Trước khi thực hiện triển khai thì hãy đảm bảo rằng máy tính phải đáp ứng các yêu cầu sau đây:

Cài đặt Node.js phiên bản 18.18.0 hoặc mới hơn. Có thể tải phiên bản này tại trang web chính thức của [Node.js](https://nodejs.org/).

Dùng lệnh ``npm install -g npm`` để cập nhật phiên bản mới nhất cho npm.

Yêu cầu Yarn phiên bản 1.22.19 hoặc mới hơn. Có thể cài đặt nó bằng cách sử dụng lệnh: ``npm install -g yarn``

Đảm bảo có kết nối mạng.

#### 5.1.2 Triển khai mã nguồn

Để triển khai mã nguồn của dự án, hãy làm theo các bước sau đây:

```
# Sao chép repository
git clone https://gitlab.duthu.net/S52000668/nodejs-502070-2324-final-project
# hoặc
git clone https://github.com/nguyenhuy158/nodejs-502070-2324-final-project

# Di chuyển đến thư mục dự án
cd nodejs-502070-2324-final-project/

# Cài đặt các thư viện phụ thuộc
yarn

# Xây dựng và chạy dự án
yarn dev
```

### 5.1.3 *Kết nối cơ sở dữ liệu*

Để chạy dự án thành công, bạn cần đảm bảo máy tính của mình có kết nối mạng, vì dự án sử dụng MongoDB URI trực tuyến để kết nối cơ sở dữ liệu. Nếu không có mạng thì sẽ không kết nối được đến dữ liệu của trang web và không sử dụng được các tính năng cơ bản.

```
MONGODB_URI=mongodb+srv://noreplaynodejs502070:9bmdioNxz8UeylCQ@techhutgc.fo
ofgxp.mongodb.net/techhut
```

## 5.2 Hướng dẫn sử dụng và lưu ý

Tài khoản mặc định của admin cho Website là:

USERNAME: admin

PASSWORD: admin

Tài khoản mặc định để đăng nhập vào **cloud.mongodb.com** là:

USERNAME = noreplay.nodejs.502070@gmail.com

PASSWORD = noreplay.nodejs.502070

Để tạo tài khoản cho nhân viên hãy làm theo các bước sau:

1. Người dùng cần đăng nhập vào tài khoản admin.
2. Truy cập đến trang quản lý người dùng “Salepeoples”, chọn `New Sale People`.
3. Nhập các thông tin cơ bản cho nhân viên (Tên, Email,...).
4. Nhân viên nhận mail và nhấn vào link xác nhận.
5. Đổi mật khẩu hoặc mật khẩu sẽ là tên của email.

## 5.3 Video demo

Chi tiết hơn sẽ được chúng em trình bày trong video demo ( để tránh cho bài báo cáo bị dài và thừa thông tin):

- Link drive: [Link](#)
- Link youtube: [Link](#)

## CHƯƠNG 6 – KẾT LUẬN

### 6.1 Tóm tắt

Sau cùng, chúng ta cũng đã tìm hiểu được toàn bộ hệ thống web POS TechHut. Với các chức năng hiện có, nhân viên bán hàng sẽ có thể quản lý các sản phẩm trong cửa hàng và giúp người dùng thanh toán sản phẩm một cách dễ dàng. Trong trường hợp nhân viên không tìm được sản phẩm vì lượng sản phẩm lớn, trang tìm kiếm sẽ là một công cụ lý tưởng để nhân viên nâng cao hiệu suất bán hàng của mình. Bên cạnh đó, nhân viên cũng có thể truy xuất lại danh sách đơn hàng và khách hàng, phòng trường hợp có vấn đề bất chính xảy ra. Hơn nữa, nhân viên cũng có thể xem doanh thu của cửa hàng trong khoảng thời gian nhất định, cùng với đó là các sản phẩm bán chạy và nhân viên ưu tú nhất. Doanh thu cũng được hiển thị bằng biểu đồ, giúp cho Admin có thể dễ dàng xem xét và thống kê cho hệ thống:

### 6.2 Học được

- **Node.js và Express:** Hiểu rõ về cấu trúc của Node.js và cách nó hoạt động, cùng với sự linh hoạt và hiệu suất của nó trong xây dựng các ứng dụng web. Bên cạnh đó ta còn có thể biết cách sử dụng các module và thư viện phổ biến để giải quyết các vấn đề trong dự án. Hơn nữa chúng ta cũng sẽ nắm vững về xử lý bất đồng bộ và callback trong Node.js.

Phát triển kỹ năng sử dụng Express, một framework Node.js phổ biến, để quản lý tạo và quản lý các route, middleware, và xử lý yêu cầu HTTP. Mặc dù việc sử dụng Express.js có thể làm cho ứng dụng trở nên phức tạp hơn so với việc sử dụng thuần Node.js, nhưng nó mang lại nhiều lợi ích về hiệu suất, sự tổ chức và tiết kiệm thời gian trong quá trình phát triển hệ thống web.

- **MongoDB:** Học cách sử dụng MongoDB, một hệ quản trị cơ sở dữ liệu NoSQL, để lưu trữ và truy vấn dữ liệu. Ngoài ra ta cũng biết cách thiết kế cơ sở dữ liệu MongoDB sao cho hiệu quả và phù hợp với yêu cầu của dự án.

Phát triển kỹ năng thiết kế cơ sở dữ liệu NoSQL

### - APIs:

Xây dựng và quản lý APIs để tương tác với cơ sở dữ liệu và cung cấp dữ liệu cho phía máy khách.

### - Quản lý Tài Nguyên:

Học cách quản lý tài nguyên như người dùng, sản phẩm và đơn hàng thông qua API.

Nhận ra rằng quản lý tài nguyên để đảm bảo hiệu suất và tính ổn định của ứng dụng.

Phát triển kỹ năng về xác thực và ủy quyền để đảm bảo an toàn thông tin.

Theo dõi và ghi log các hoạt động của web để phát hiện và giải quyết các vấn đề liên quan đến tài nguyên.

### - Quản lý source code:

Sử dụng công cụ quản lý như Git để theo dõi sự thay đổi và quản lý mã nguồn của dự án. giúp các thành viên có thể làm việc chung đẩy nhanh tốc độ phát triển dự án

### - Sử Dụng Các Package Nâng Cao:

- **cors:** xử lý Cross-Origin Resource Sharing (CORS) trong ứng dụng Node.js
- **dotenv:** quản lý biến môi trường trong ứng dụng Node.js
- **bcryptjs:** cung cấp các phương thức mã hóa và so sánh mật khẩu dựa trên thuật toán bcrypt.
- **nodemon:** ông cụ phát triển được sử dụng để theo dõi và tự động khởi động lại ứng dụng Node.js khi có thay đổi trong mã nguồn.
- **nodemailer:** gửi email từ ứng dụng Node.js.
- **sharp:** thư viện xử lý hình ảnh trong Node.js.

## 6.3 Ưu điểm và nhược điểm

### - Ưu điểm:

- Các chức năng cơ bản hoạt động tốt và có thêm một số tính năng phụ kèm theo.



- *Phân quyền*: chỉ cho phép admin thực hiện các thao tác như thêm, xóa, sửa, trong khi nhân viên chỉ có thể xem và thanh toán (checkout). Điều này giúp đảm bảo tính bảo mật và quản lý dữ liệu hiệu quả.
- *Tích hợp tiền tệ VND*: cho phép người dùng thực hiện giao dịch trong đơn vị tiền tệ quốc gia. Điều này giúp tạo sự thuận tiện và đáp ứng nhu cầu của người dùng Việt Nam.
- *Thanh toán nhanh chóng bằng tiền mặt*: cho phép người dùng thực hiện thanh toán một cách nhanh chóng và tiện lợi. Điều này phù hợp với những trường hợp người dùng muốn trả tiền trực tiếp khi mua hàng hoặc sử dụng dịch vụ.
- *Nhân viên được tự động tạo tài khoản*: khi trở thành nhân viên thì ta sẽ được admin tạo giùm tài khoản, sau đó sẽ dùng email để kích hoạt tài khoản và sử dụng trang web.

- Nhược điểm:

- *Chưa tích hợp thanh toán online*: chưa có các phương thức thanh toán bằng momo, shopeePay, zaloPay,... hoặc quét mã QR.
- *UI chưa hoàn thiện*: màu sắc, bố cục, hình ảnh, biểu đồ, và các thành phần giao diện khác chưa được tối ưu và hấp dẫn.
- Thông tin hiển thị chưa được rõ ràng, đầy đủ, có thể gây khó hiểu cho nhân viên.
- *Tốc độ tải trang chưa được tối ưu*: tốc độ tải trang còn chậm, làm mất thời gian, tạo trải nghiệm không tốt

#### 6.4 Hướng phát triển thêm trong tương lai

Để phát triển thêm cho ứng dụng của chúng em trong tương lai thì sau đây là một số ý tưởng mà chúng em muốn thêm vào để làm cho nó trở nên phong phú và hấp dẫn hơn.

Thứ nhất, chúng em muốn tích hợp WebSocket để làm cho ứng dụng hoạt động "real time", nghĩa là có thể cập nhật thông tin ngay lập tức, chẳng hạn như doanh số bán hàng. Điều này sẽ giúp người dùng có cái nhìn chính xác và nhanh chóng về hoạt động kinh doanh.

Thứ hai, chúng em muốn thêm tính năng thanh toán trực tuyến để người dùng có thể dễ dàng thanh toán bằng các ví điện tử như Momo, ShopeePay, ZaloPay hoặc thậm chí là qua ngân hàng. Việc này sẽ làm cho quá trình thanh toán trở nên thuận tiện hơn và nhanh chóng.

Chúng em cũng muốn thêm chức năng quản lý tồn kho để người dùng có thể kiểm tra số lượng sản phẩm còn trong kho, đã bán được bao nhiêu và còn lại bao nhiêu. Điều này sẽ giúp doanh nghiệp duy trì quản lý hiệu quả và đảm bảo luôn đủ hàng cho khách hàng.

Một tính năng khác là trí tuệ nhân tạo (AI) để đề xuất sản phẩm dựa trên đánh giá của người dùng và gợi ý sản phẩm tương tự khi họ tìm kiếm. Điều này sẽ làm cho trải nghiệm mua sắm trở nên thú vị và mang lại nhiều lựa chọn hơn.

Chúng em cũng muốn thêm tính năng giảm giá, phát hành mã voucher và phiếu giảm giá để khuyến khích người dùng mua sắm và tăng doanh số bán hàng. Việc này có thể tạo ra niềm đam mê săn sale và tăng sự hứng thú từ phía khách hàng.

Cuối cùng là phát triển ứng dụng di động để nhân viên có thể tương tác với khách hàng một cách linh hoạt hơn. Điều này bao gồm việc đặt hàng, tra cứu thông tin sản phẩm và hỗ trợ khách hàng trực tiếp từ điện thoại di động.

Bằng cách tích hợp các loại tiền tệ khác nhau, chúng em hy vọng rằng sẽ thu hút được nhiều người dùng từ khắp nơi trên thế giới và mở rộng cơ hội kinh doanh cho phía bên người dùng.

## TÀI LIỆU THAM KHẢO

- [1]. “Chart.js.” Accessed December 10, 2023. <https://www.chartjs.org/>.
- [2]. Cloudinary. “Image and Video Upload, Storage, Optimization and CDN.” Accessed December 1, 2023. <https://cloudinary.com/>.
- [3]. “EJS -- Embedded JavaScript Templates.” Accessed December 1, 2023. <https://ejs.co/>.
- [4]. “Express - Node.js Web Application Framework.” Accessed December 2, 2023. <https://expressjs.com/>.
- [5]. “MongoDB: The Developer Data Platform | MongoDB.” Accessed December 1, 2023. <https://www.mongodb.com/>.
- [6]. “Node.js.” Accessed December 2, 2023. <https://nodejs.org/en>.
- [7]. “Nodemailer :: Nodemailer.” Accessed December 1, 2023. <https://nodemailer.com/>.
- [8]. npm. “Passport,” November 27, 2023. <https://www.npmjs.com/package/passport>.
- [9]. npm. “Sharp,” November 29, 2023. <https://www.npmjs.com/package/sharp>.
- [10]. npm. “Toastr,” December 8, 2017. <https://www.npmjs.com/package/toastr>.
- [11]. npm. “Winston,” October 7, 2023. <https://www.npmjs.com/package/winston>.
- [12]. “Sass: Syntactically Awesome Style Sheets.” Accessed December 1, 2023. <https://sass-lang.com/>.

## PHỤ LỤC

### 1. Bootstrap Cheat Sheet

- <https://bootstrap-cheatsheet.themeselection.com/>
- Bảng tóm tắt các class, component và utilities trong Bootstrap, giúp tăng hiệu suất trong thiết kế giao diện và tối ưu hóa trải nghiệm người dùng.

### 2. Express Quick Reference

- <https://quickref.me/express.html>
- Tài liệu tham khảo nhanh về Express, cung cấp thông tin chi tiết về các chức năng, middleware và cách sử dụng framework này để phát triển ứng dụng web hiệu quả.

### 3. Express Validator

- <https://express-validator.github.io/>
- Hướng dẫn sử dụng thư viện Express Validator để kiểm tra và xác thực dữ liệu nhập vào từ người dùng, giúp bảo vệ ứng dụng khỏi các lỗ hổng bảo mật.

### 4. jQuery Documentation

- <https://oscarotero.com/jquery/>
- Tài liệu chính thức của thư viện jQuery, cung cấp thông tin về các phương thức, sự kiện và API để thực hiện các thao tác DOM và tương tác người dùng một cách linh hoạt.

### 5. Toastr

- <https://github.com/CodeSeven/toastr>

- Trang GitHub chính thức của thư viện Toastr, cung cấp thông tin về cách sử dụng và tùy chỉnh thông báo pop-up hiển thị ở góc màn hình, thích hợp cho các thông báo và cảnh báo trong ứng dụng web.

#### 6. DataTables

- <https://datatables.net/>
- DataTables là một plugin dựa trên thư viện jQuery, được thiết kế để cải thiện và mở rộng khả năng tương tác với bảng HTML. Plugin này mang lại nhiều tính năng như phân trang, sắp xếp, tìm kiếm và thậm chí là chức năng AJAX để tải dữ liệu bảng từ máy chủ.

#### 7. Select2

- <https://select2.org/>
- Nó giúp tối ưu hóa và làm đẹp các hộp chọn (dropdown) trong trang web. Nó cung cấp nhiều tính năng mở rộng so với hộp chọn thông thường, bao gồm: tìm kiếm, lọc, tùy chỉnh, đa lựa chọn. Trong bài thì dùng cho việc chọn thành phố, quận, huyện dễ dàng.

#### 8. SweetAlert2

- <https://sweetalert2.github.io/>
- Nó cung cấp thiết kế và nhiều kiểu thông báo (cảnh báo, lỗi, thành công), tùy chỉnh nội dung, và hỗ trợ Promise để xử lý sự kiện sau khi người dùng tương tác giúp tăng cường trải nghiệm người dùng khi thông báo về sự kiện hoặc yêu cầu xác nhận.

#### 9. Tippyjs

- <https://atomiks.github.io/tippyjs/>
- Dễ dàng thêm các hộp chú thích (tooltip) hiện đại và dễ định hình cho các phần tử trên trang. Thư viện này hỗ trợ nhiều tùy chọn tùy chỉnh, hiệu ứng, và tính năng linh hoạt như tooltip động, tooltip theo chuột, và nhiều loại hiển thị khác nhau.

#### 10. Currency.js

- <https://currency.js.org/>
- Hỗ trợ các format tiền tệ, xử lý số tiền theo định dạng tiền tệ trong ứng dụng web và làm việc với các đơn vị tiền tệ khác nhau một cách dễ dàng.

#### 11. Date Range Picker

- <https://www.daterangepicker.com/>
- Hỗ trợ bổ sung cho chart, giúp cho việc lựa chọn giữa ngày tháng năm dễ dàng, kể cả từ ngày cụ thể đến thời điểm hiện tại.

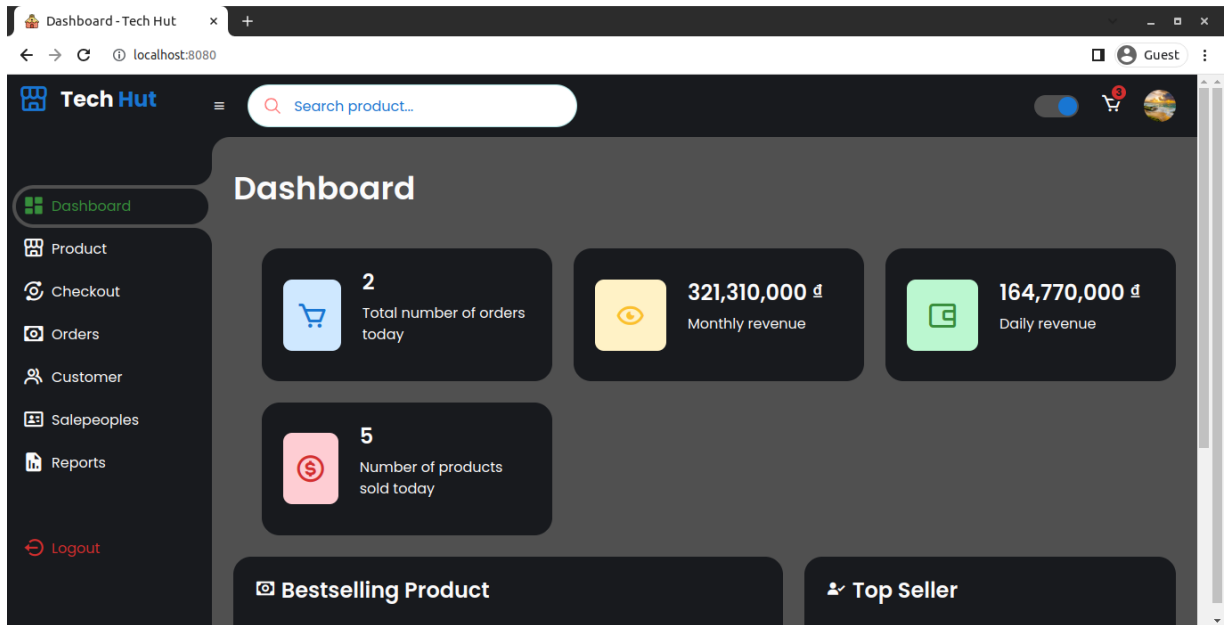
#### 12. Fotorama

- <https://fotorama.io/>
- Cung cấp một giao diện đẹp mắt và dễ sử dụng, hỗ trợ tính năng chuyển đổi linh hoạt giữa các ảnh, chế độ xem toàn màn hình, và nhiều tùy chọn tùy chỉnh khác giúp cho hiển thị hình ảnh một cách mượt mà và thú vị.

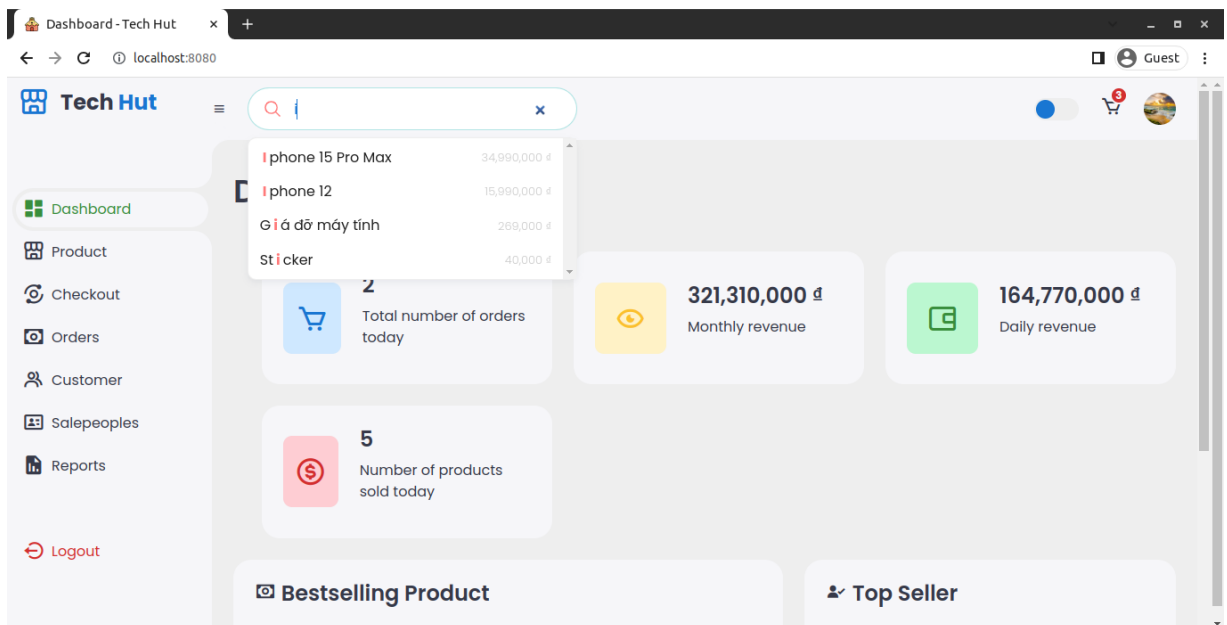
#### 13. JsBarcode

- <https://lindell.me/JsBarcode/>
- Được sử dụng để tạo mã vạch (barcode) trong ứng dụng web. Với JsBarcode, ta có thể dễ dàng tạo các loại mã vạch phổ biến như EAN-13, CODE128, và nhiều loại khác.

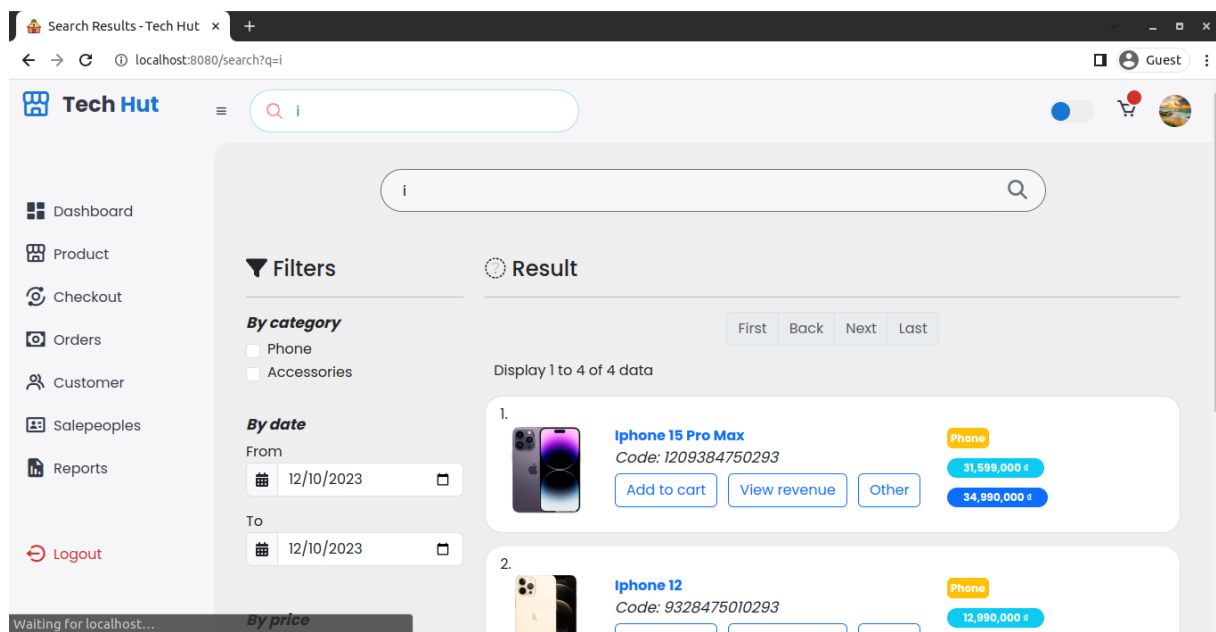
## Một số hình ảnh khác của trang web



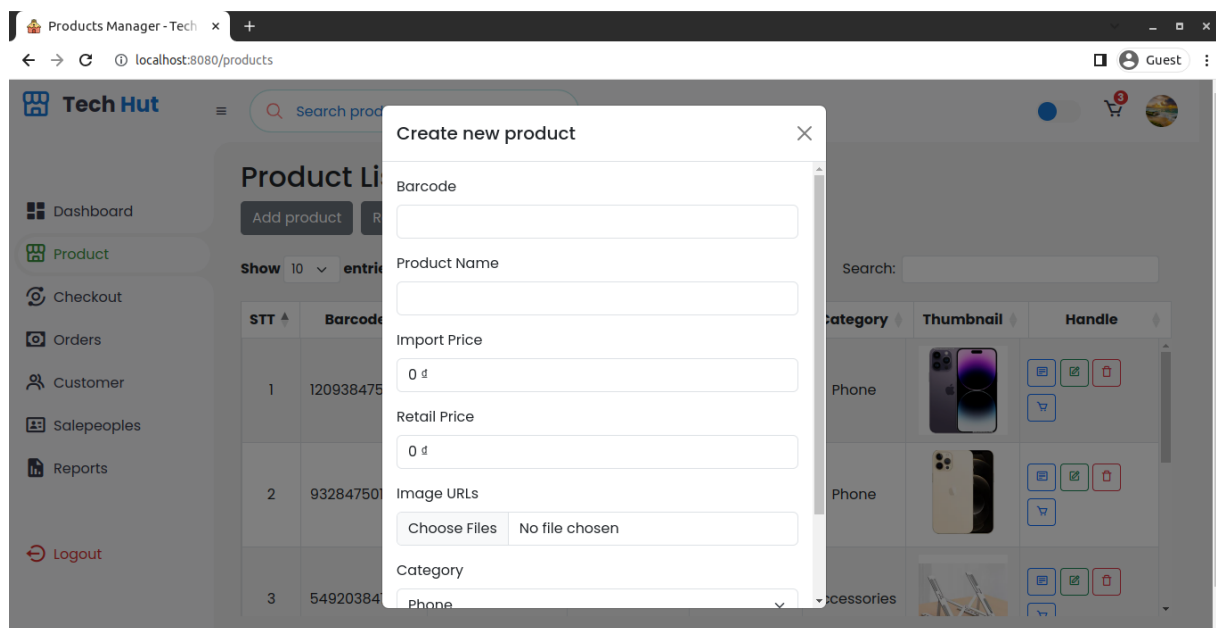
Trang web có hỗ trợ giao diện darkmode



Tính năng search tìm kiếm sản phẩm

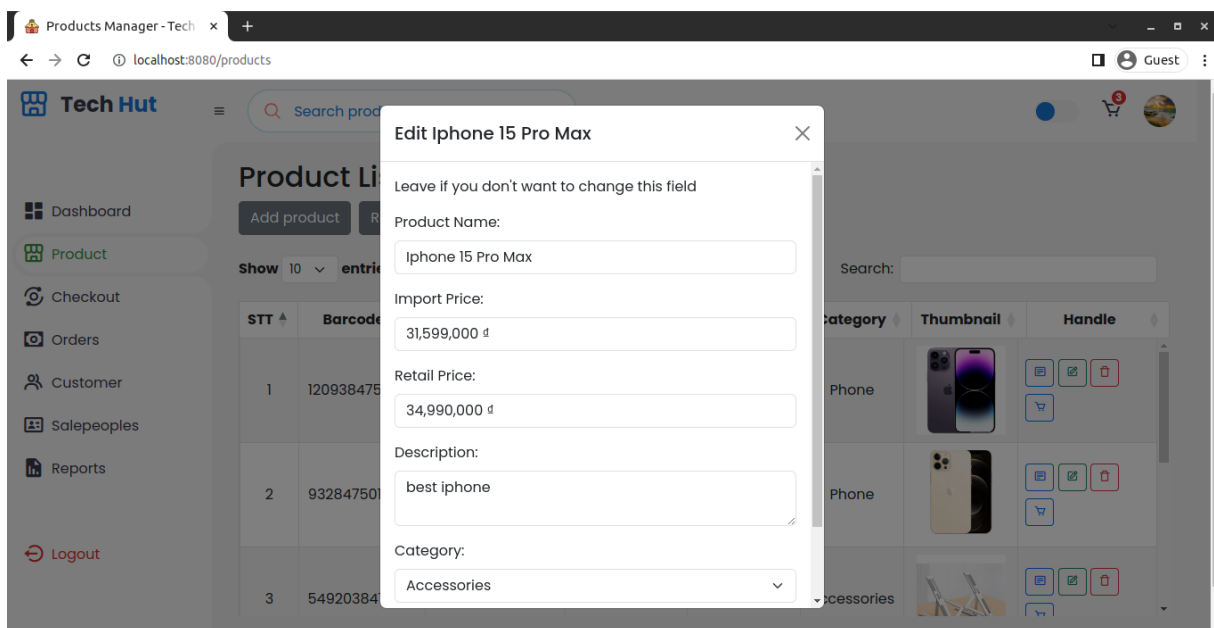


Trang kết quả search tìm kiếm

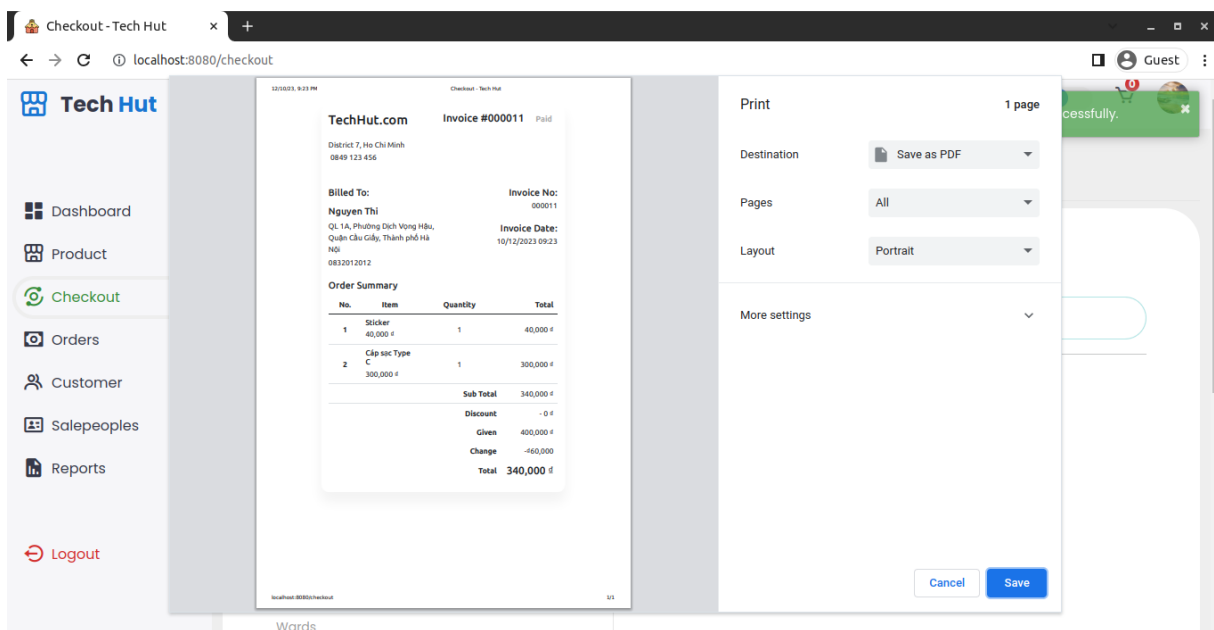


Tạo sản phẩm mới





Cập nhật sản phẩm mới



In ra hóa đơn

Order detail - Tech Hut

localhost:8080/orders/6575c9cdefc9359ec186ebaf

Guest

**Tech Hut**


District 7, Ho Chi Minh  
0849 123 456

Billed To: **Nguyen Thi**  
0832012012

Invoice No: 000011

Invoice Date: 10/12/2023 09:23

**Order Summary**

No.	Item	Quantity	Total
1	<b>Sticker</b> 40,000 đ 	1	40,000 đ

Cáp sạc Type C

Xem lại hóa đơn, lịch sử

Order detail - Tech Hut

localhost:8080/orders/6575c9cdefc9359ec186ebaf

Guest

**Tech Hut**

Cáp sạc Type C  
300,000 đ

2

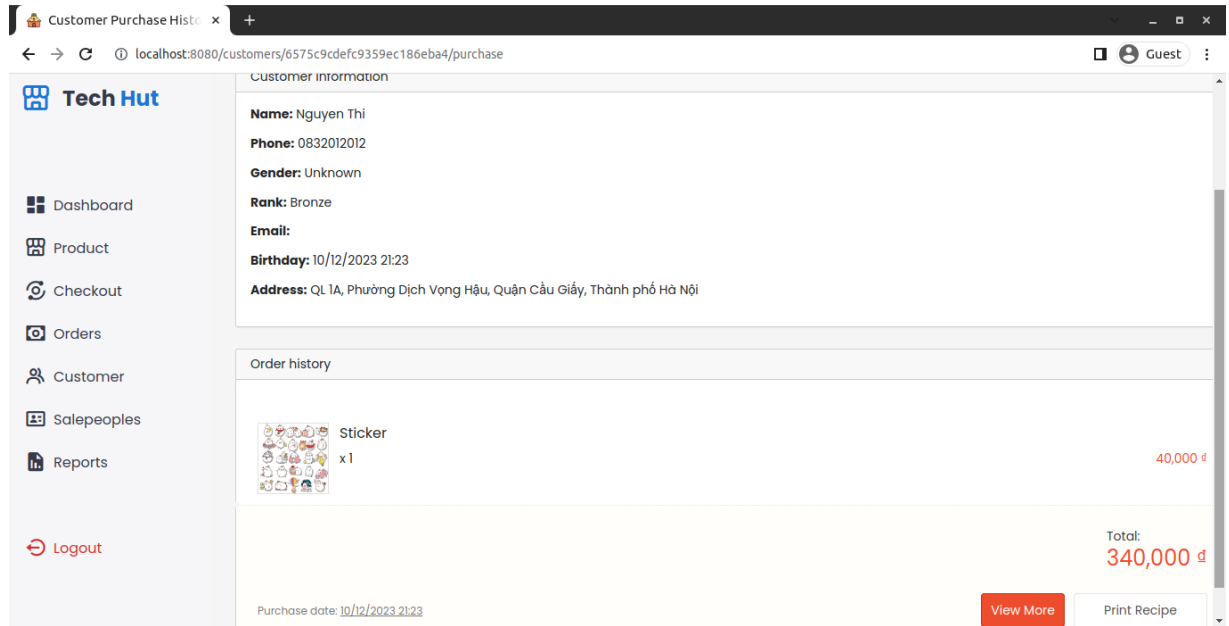
1

300,000 đ

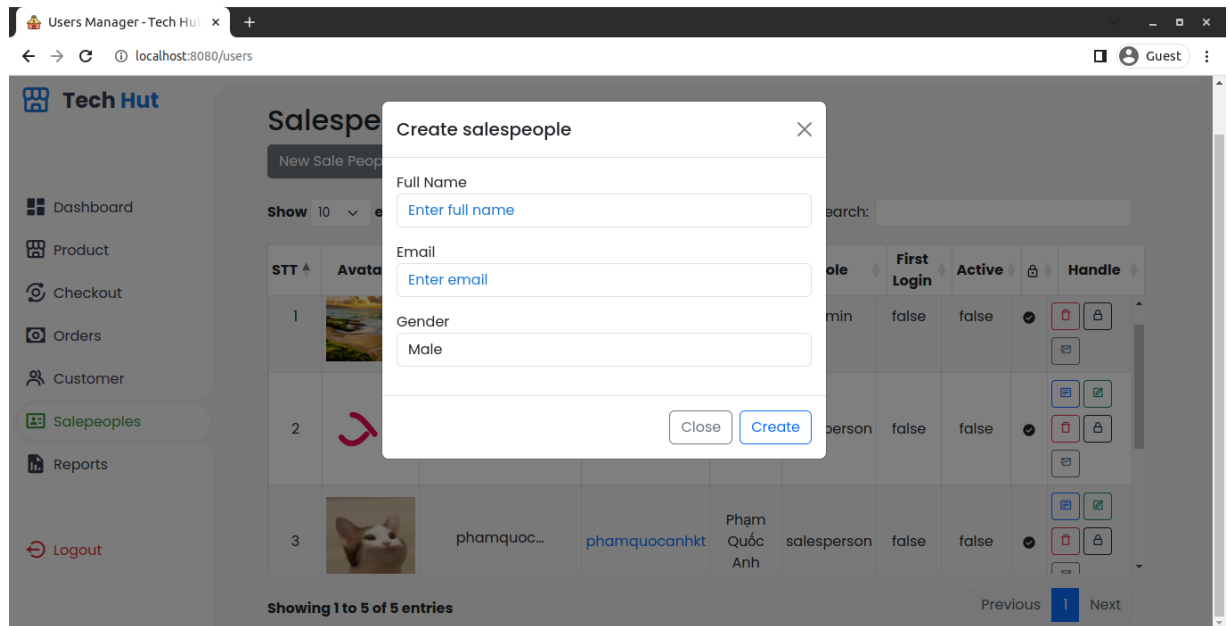
<b>Sub Total</b>	340,000 đ
<b>Discount</b>	- 0 đ
<b>Given</b>	400,000 đ
<b>Change</b>	-đ60,000
<b>Total</b>	<b>340,000 đ</b>

Print

Xem chi tiết hoá đơn



Xem lại thông tin khách hàng



Tạo tài khoản mới cho nhân viên (Admin)

The screenshot shows the 'Phạm Quốc Anh Information' page in the Tech Hut application. On the left is a sidebar with navigation links: Dashboard, Product, Checkout, Orders, Customer, Reports, and Logout. The main content area displays a user profile for 'Phạm Quốc Anh', a salesperson, with a circular profile picture of a cat. To the right of the profile is a table of user details.

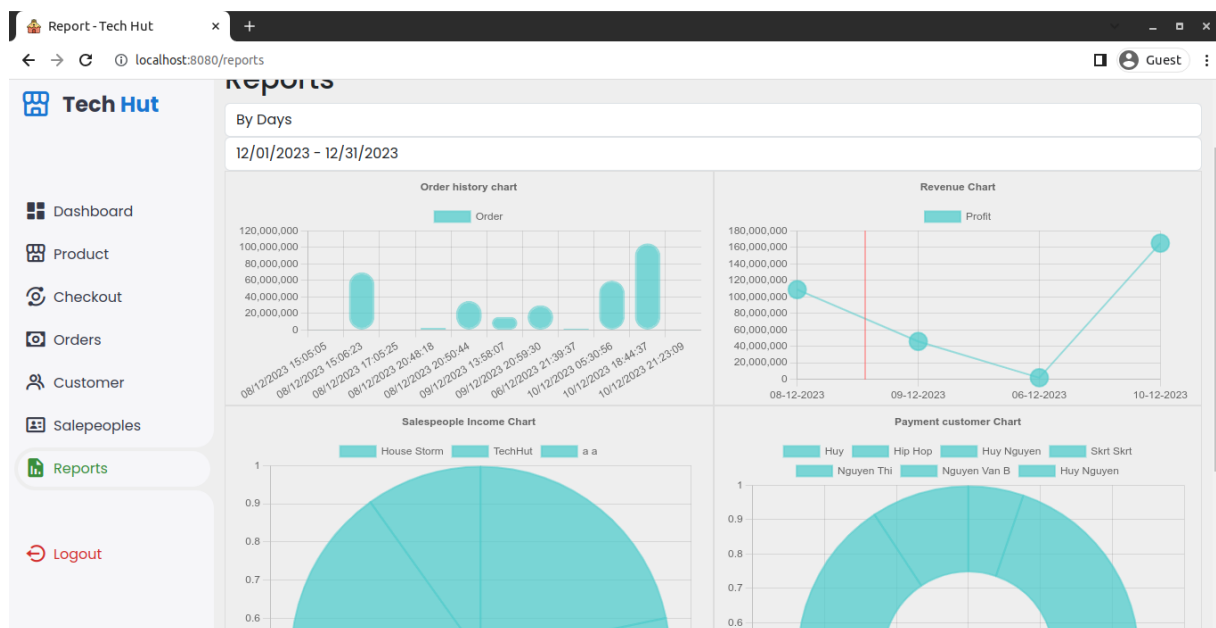
Full Name	Phạm Quốc Anh
Email	phamquocanhkt@gmail.com
Gender	Unknown
Role	salesperson
Username	phamquocanhkt
Account has been activated	✓
Lock/Unlock	🔒
Online/Offline	🔒
Created at	10/12/2023 21:24

Below the profile picture are four buttons: Unlock, Lock, Resent mail, and Delete.

Xem thông tin của nhân viên

The screenshot shows the 'Reports' page in the Tech Hut application. The sidebar on the left includes links for Dashboard, Product, Checkout, Orders, Customer, Salepeoples, Reports, and Logout. The main content area features a 'By Days' date range selector set to '12/10/2023 - 12/11/2023'. Below this is a calendar view for December 2023 and January 2024. A dropdown menu is open, showing various time range options: Today, Yesterday, Last 7 Days, Last 30 Days, This Month, Last Month, This Year, Last Year, and Custom Range. The 'Custom Range' option is selected, and the date range '12/10/2023 - 12/11/2023' is displayed at the bottom right of the calendar.

Xem biểu đồ doanh thu ở nhiều mốc thời gian



Có 4 loại biểu đồ