**LỜI CAM ĐOAN**

Tôi xin cam đoan rằng toàn bộ nội dung của báo cáo này là kết quả của quá trình học tập, nghiên cứu và thực hiện đồ án một cách nghiêm túc dưới sự hướng dẫn của giảng viên Võ Thành C. Các tài liệu, dữ liệu và hình ảnh sử dụng trong báo cáo đều được trích dẫn rõ ràng nguồn gốc. Tôi hoàn toàn chịu trách nhiệm về tính trung thực và nguyên bản của báo cáo. Nếu phát hiện có sự sao chép, gian lận hay vi phạm bản quyền, tôi xin hoàn toàn chịu trách nhiệm trước hội đồng và quy định của Nhà trường.

*Vĩnh Long, ngày tháng năm 2025*

Sinh viên thực hiện

*(Ký & ghi rõ họ tên)*

**LỜI CẢM ƠN**

Tôi xin gửi lời cảm ơn sâu sắc đến trường đại học Trà Vinh đã tạo điều kiện cho tôi thực hiện dự án này, những người đã đưa ra sự hỗ trợ nhỏ nhất cho tôi dù là một lời gợi ý. Nhờ mọi người mà dự án đã thành công tốt đẹp.

Tôi biết ơn tất cả mọi người đã dành thời gian, công sức và kiến thức của mình để hỗ trợ tôi. Sự giúp đỡ của các bạn là nguồn động lực giúp tôi phát triển bản thân. Các ý kiến đóng góp và sự hợp tác của mọi người đã tạo nên một môi trường làm việc tích cực.

Không có gì quý báu hơn những người thầy, người cô và những người đã góp ý và những người đã hỗ trợ tôi như mọi người, mọi người đã dìu dắt tôi trong dự án đầu tay của mình. Nhờ có những người thầy, người cô và các bạn đã giúp đỡ mà dự án của tôi mới có thể phát triển đến mức này. Tôi hạnh phúc và tự hào về mối quan hệ mà chúng ta đã xây dựng cùng nhau và tin rằng những thành công trong tương lai sẽ tiếp tục phát triển từ sự đồng lòng, cống hiến và sự giúp đỡ của mọi người đối với tôi.

Một lần nữa, xin chân thành cảm ơn mọi người đã giúp đỡ và tôi mong sẽ tiếp tục nhận được sự ủng hộ của mọi người trong những dự án sắp tới.

Trân trọng.

# **MỤC LỤC**

[MỤC LỤC 3](#_Toc206413157)

[DANH MỤC CHỮ VIẾT TẮT 10](#_Toc206413158)

[DANH MỤC BẢNG BIỂU 11](#_Toc206413159)

[DANH MỤC HÌNH ẢNH 12](#_Toc206413160)

[TÓM TẮT 13](#_Toc206413161)

[MỞ ĐẦU 14](#_Toc206413162)

[CHƯƠNG 1: TỔNG QUAN 16](#_Toc206413163)

[CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT 17](#_Toc206413164)

[2.1 Kiến thức cơ bản các ngôn ngữ, thư viện và framework 17](#_Toc206413165)

[*2.1.1 API và Restful API:* 17](#_Toc206413166)

[*2.1.2 Node.js* 18](#_Toc206413167)

[*2.1.3 ReactJs* 18](#_Toc206413168)

[*2.1.4 Bootstrap* 19](#_Toc206413169)

[*2.1.5 Material UI* 19](#_Toc206413170)

[*2.1.6 Express.js* 19](#_Toc206413171)

[*2.1.7 MySQL* 20](#_Toc206413172)

[2.2 Tích hợp giữa Frontend và Backend 20](#_Toc206413173)

[2.3 Giả thiết khoa học và phương pháp nghiên cứu dự án 20](#_Toc206413174)

[Giả thiết khoa học: hiệu suất và sự linh hoạt 20](#_Toc206413175)

[Phương pháp nghiên cứu: phát triển liên tục và tiến hành kiểm thử 21](#_Toc206413176)

[CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU 22](#_Toc206413177)

[3.1 Phân tích và đặt tả yêu cầu hệ thống: 22](#_Toc206413178)

[3.1.1 Phân tích chức năng: 22](#_Toc206413179)

[3.1.2 Yêu cầu phi chức năng 22](#_Toc206413180)

[3.2 Thiết kế hệ thống 23](#_Toc206413181)

[*3.1.1* *Mô hình thực thể kết hợp* 23](#_Toc206413182)

[*3.1.2 Mô hình vật lý* 23](#_Toc206413183)

[*3.1.3* *Sơ đồ Use Case:* 24](#_Toc206413184)

[*3.1.4* *Mô tả các bảng trong lược đồ cơ sở dữ liệu* 24](#_Toc206413185)

[3.2 Cài đặt môi trường 29](#_Toc206413186)

[3.3 Phác thảo giao diện 30](#_Toc206413187)

[CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU 34](#_Toc206413188)

[4.1 Dữ liệu thực nghiệm 34](#_Toc206413189)

[4.2 Các chức năng 36](#_Toc206413190)

[*4.2.1 Đối với admin* 36](#_Toc206413191)

[*4.2.2 Đối với người dùng* 39](#_Toc206413192)

[CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 43](#_Toc206413193)

[5.1 Kết luận 43](#_Toc206413194)

[5.2 Hạn chế 43](#_Toc206413195)

[5.3 Hướng phát triển 43](#_Toc206413196)

[DANH MỤC TÀI LIỆU THAM KHẢO 45](#_Toc206413197)

**NHẬN XÉT**

**(Của giảng viên hướng dẫn trong đồ án, khoá luận của sinh viên)**

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

**Giảng viên hướng dẫn**

(ký và ghi rõ họ tên)

UBND TỈNH TRÀ VINH **CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM**

**TRƯỜNG ĐẠI HỌC TRÀ VINH Độc lập - Tự do - Hạnh phúc**

**BẢN NHẬN XÉT ĐỒ ÁN, KHÓA LUẬN TỐT NGHIỆP**

*(Của giảng viên hướng dẫn)*

Họ và tên sinh viên: Nguyễn Hoàng Nhân MSSV: 110121071

Ngành: Công nghệ Thông tin Khóa: 2021

Tên đề tài: Xây dựng website bán điện thoại di động

Họ và tên Giáo viên hướng dẫn: Võ Thành C

Chức danh: Giảng viên Học vị: Thạc sĩ

**NHẬN XÉT**

1. Nội dung đề tài:

1. Ưu điểm:

1. Khuyết điểm:

1. Điểm mới đề tài:

1. Giá trị thực trên đề tài:

7. Đề nghị sửa chữa bổ sung:

8. Đánh giá:

*Vĩnh Long, ngày tháng năm 2025*

Giảng viên hướng dẫn

*(Ký & ghi rõ họ tên)*

**NHẬN XÉT**

**(Của giảng viên chấm trong đồ án, khoá luận của sinh viên)**

**Giảng viên chấm**

(ký và ghi rõ họ tên)

UBND TỈNH TRÀ VINH **CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM**

**TRƯỜNG ĐẠI HỌC TRÀ VINH Độc lập - Tự do - Hạnh phúc**

**BẢN NHẬN XÉT ĐỒ ÁN, KHÓA LUẬN TỐT NGHIỆP**

*(Của cán bộ chấm đồ án, khóa luận)*

Họ và tên người nhận xét:

Chức danh: Học vị:

Chuyên ngành:

Cơ quan công tác:

Tên sinh viên:

Tên đề tài đồ án, khóa luận tốt nghiệp:

**I. Ý KIẾN NHẬN XÉT**

1. Nội dung:

2. Điểm mới các kết quả của đồ án, khóa luận:

3. Ứng dụng thực tế:

**II. CÁC VẤN ĐỀ CẦN LÀM RÕ**

(Các câu hỏi của giáo viên phản biện)

**III. KẾT LUẬN**

(Ghi rõ đồng ý hay không đồng ý cho bảo vệ đồ án khóa luận tốt nghiệp)

*Vĩnh Long, ngày …… tháng …… năm 2025*

Người nhận xét

*(Ký & ghi rõ họ tên)*

# **DANH MỤC CHỮ VIẾT TẮT**

|  |  |
| --- | --- |
| API | Application Programming Interface (Giao diện lập trình ứng dụng) |
| CSDL | Cơ sở dữ liệu |
| CLI | Command Line Interface (Giao diện dòng lệnh) |
| DOM | Document Object Model |
| HTTP | HyperText Transfer Protocol |
| JWT | JSON Web Token |
| REST | Representational State Transfer |
| URI | Uniform Resource Identifier (Bộ định vị tài nguyên đồng nhất) |

# **DANH MỤC BẢNG BIỂU**

[Bảng 3. 1 Bảng mô tả các bảng trong lược đồ cơ sở dữ liệu 25](#_Toc206413407)

[Bảng 3. 2 Bảng khách hàng 25](#_Toc206413408)

[Bảng 3. 3 Bảng giỏ hàng 25](#_Toc206413409)

[Bảng 3. 4 Bảng chi tiết giỏ hàng 26](#_Toc206413410)

[Bảng 3. 5 Bảng sản phẩm 27](#_Toc206413411)

[Bảng 3. 6 Bảng màu sắc sản phẩm 27](#_Toc206413412)

[Bảng 3. 7 Bảng thương hiệu 28](#_Toc206413413)

[Bảng 3. 8 Bảng đơn hàng 28](#_Toc206413414)

[Bảng 3. 9 Bảng chi tiết đơn 29](#_Toc206413415)

# **DANH MỤC HÌNH ẢNH**

[Hình 3. 1 Mô hình thực thể kết hợp 23](#_Toc206414323)

[Hình 3. 2 Mô hình vật lý 23](#_Toc206414324)

[Hình 3. 3 Use Case khách hàng 24](#_Toc206414325)

[Hình 3. 4 Use Case quản trị viên 24](#_Toc206414326)

[Hình 3. 5 Cấu trúc cây thư mục backend 30](#_Toc206414327)

[Hình 3. 6 Phác thảo giao diện đăng ký 30](#_Toc206414328)

[Hình 3. 7 Phác thảo giao diện đăng nhập 31](#_Toc206414329)

[Hình 3. 8 Phác thảo giao diện quản lý nghiệp vụ của admin 31](#_Toc206414330)

[Hình 3. 9 Phác thảo giao diện trang chủ 32](#_Toc206414331)

[Hình 3. 10 Phác thảo giao diện trang thông tin chi tiết sản phẩm 32](#_Toc206414332)

[Hình 3. 11 Phác thảo giao diện trang giỏ hàng 33](#_Toc206414333)

[Hình 3. 12 Phác thảo giao diện trang thông tin người dùng 33](#_Toc206414334)

[Hình 4. 1 Dữ liệu về khách hàng 34](#_Toc206414302)

[Hình 4. 2 Dữ liệu về sản phẩm 34](#_Toc206414303)

[Hình 4. 3 Dữ liệu về thương hiệu 34](#_Toc206414304)

[Hình 4. 4 Dữ liệu về giỏ hàng 34](#_Toc206414305)

[Hình 4. 5 Dữ liệu về chi tiết giỏ hàng 35](#_Toc206414306)

[Hình 4. 6 Dữ liệu về đơn hàng 35](#_Toc206414307)

[Hình 4. 7 Dữ liệu về chi tiết đơn hàng 35](#_Toc206414308)

[Hình 4. 8 Giao diện đăng nhập của admin 36](#_Toc206414309)

[Hình 4. 9 Giao diện trang quản lý sản phẩm của admin 36](#_Toc206414310)

[Hình 4. 10 Giao diện trang quản lý màu sản phẩm của admin 37](#_Toc206414311)

[Hình 4. 11 Giao diện trang quản lý thương hiệu của admin 37](#_Toc206414312)

[Hình 4. 12 Giao diện trang quản lý tài khoản của admin 38](#_Toc206414313)

[Hình 4. 13 Giao diện trang quản lý tất cả sản phẩm của admin 38](#_Toc206414314)

[Hình 4. 14 Giao diện trang quản lý đơn hàng 39](#_Toc206414315)

[Hình 4. 15 Giao diện đăng ký 39](#_Toc206414316)

[Hình 4. 16 Giao diện đăng nhập 40](#_Toc206414317)

[Hình 4. 17 Giao diện trang chủ 40](#_Toc206414318)

[Hình 4. 18 Giao diện trang thông tin chi tiết sản phẩm 41](#_Toc206414319)

[Hình 4. 19 Giao diện giỏ hàng 41](#_Toc206414320)

[Hình 4. 20 Giao diện thông tin khách hàng 42](#_Toc206414321)

[Hình 4. 21 Giao diện cập nhật thông tin khách hàng 42](#_Toc206414322)

# **TÓM TẮT**

Xây dựng website bán điện thoại di động có các chức năng cơ bản của một website thương mai điện tử thông thường gồm quản lý sản phẩm, giỏ hàng, đơn hàng và quản lý người dùng.

Sử dụng các ngôn ngữ lập trình và thư viện hiện hành như Reactjs cho thiết kế frontend, Nodejs cho thiết kế backend và MySQL cho thiết kế cơ sở dữ liệu. Ngoài ra, một số thư viện dành cho thiết kế giao diện như Bootstrap, Material UI.

Sản phẩm cuối cùng là một website thương mại điện tử bán điện thoại có các chức năng như đăng nhập, đăng ký, mua sản phẩm, thêm giỏ hàng và thống kê doanh thu.

# **MỞ ĐẦU**

**Lý do chọn đề tài**

Em chọn đề tài "Xây dựng website bán điện thoại di động" vì em có đam mê công nghệ đồng thời với ước mơ trở thành lập trình viên website nên em đã chọn đề tài. Trong môi trường đầy tính cạnh tranh hiện nay thì tôi cần phải tự mình phát triển những kĩ năng cần có của một lập trình viên website và việc chọn một dự án website thương mại điện tử sẽ cung cấp cơ hội tuyệt vời để khám phá và thách thức bản thân trong việc xây dựng một ứng dụng hoàn chỉnh từ đầu đến cuối.

Trong hành trình trở thành một lập trình viên website, việc nắm vững cả hai phía frontend và backend là quan trọng nhất để có khả năng xây dựng và duy trì hệ thống website toàn diện. Chính vì lẽ đó, dự án nghiên cứu này không chỉ là một nỗ lực để hiểu rõ về quy trình và cách thức phát triển của một website thương mại điện tử mà còn là cơ hội để rèn luyện, phát triển kĩ năng và cách áp dụng kiến thức của mình để phát triển kiến thức trong môi trường cạnh tranh thực tế.

**Đối tượng nghiên cứu**

Dự án hướng đến việc nghiên cứu cách sử dụng các framework như Nodejs để thiết kế backend và Reactjs để thiết kế frontend. Ngoài ra dự án cũng nghiên cứu cách mà các framework này hoạt động nhằm nâng cao kiến thức cho bản thân người sử dụng và những người đang tìm hiểu và có ý định sử dụng.

**Mục tiêu nghiên cứu**

Mục đích của dự án không chỉ là xây dựng một trang web bán điện thoại hiệu quả mà còn là việc phát triển khả năng website bằng cách làm việc với Node.js và React.js. Điều này sẽ giúp tôi hiểu rõ hơn về quá trình tích hợp giữa frontend và backend, đồng thời nắm vững quản lý dữ liệu và tương tác người dùng.

**Phương pháp nghiên cứu**

Phương pháp khảo sát và thu thập quy trình:

- Tìm hiểu quy trình mua hàng trên các website thương mại điện nhằm xác định những thông tin tối thiểu cần thiết cho quy trình mua hàng.

- Xác định đặc điểm của các nhóm người dùng chính: quản trị viên, khách hàng từ đó xây dựng các luồng chức năng và phân quyền phù hợp.

Phương pháp phân tích và thiết kế hệ thống:

- Phân tích chức năng: xây dựng sơ đồ use case để xác định đầy đủ các chức năng của hệ thống như: đăng nhập/đăng ký, quản lý sản phẩm, quản lý người dùng, quản lý đơn hàng, mua hàng, thêm giỏ hàng.

- Phân tích yêu cầu phi chức năng: bao gồm khả năng mở rộng, hiệu năng, bảo mật (JWT, phân quyền), tính thân thiện với người dùng.

Phương pháp thiết kế:

- Thiết kế giao diện người dùng: giao diện hiển thị rõ ràng không phức tạp, màu sắc đơn giản, hài hòa.

- Thiết kế hệ thống backend: xây dựng backend với NodeJS, phân tách các lớp controller, model, route dễ dàng mở rộng và bảo trì hệ thống.

- Thiết kế cơ sở dữ liệu: sử dụng MySQL để thiết kế CSDL.

Phương pháp triển khai: sử dụng ReactJS cùng với Boostrap + MUI để xây dựng giao diện website. Xây dựng API RESTful với NodeJs.

# **CHƯƠNG 1: TỔNG QUAN**

Dự án này tập trung vào việc tạo và phát triển một website thương mại điện tử sử dụng Node.js và Express.js cho phần backend và React.js cho frontend. Sự kết hợp này không chỉ cung cấp một cơ sở hạ tầng mạnh mẽ và linh hoạt, mà còn tạo ra một giao diện người dùng đáng chú ý với trải nghiệm mua sắm trực tuyến thú vị và thuận tiện.

Chúng ta sẽ tìm hiểu cách Node.js và Express.js với khả năng xử lý không đồng bộ và mã nguồn mở mạnh mẽ, đóng vai trò quan trọng trong việc xây dựng các API và xử lý dữ liệu của hệ thống. Mặt khác, React.js sẽ giúp chúng ta xây dựng giao diện người dùng động, tương tác giữa người dùng với server và dễ bảo trì dù trong bất kì hoàn cảnh nào.

Dự án không chỉ hướng đến mục tiêu cung cấp một website bán điện thoại trực tuyến, mà còn là việc nghiên cứu và thử nghiệm các phương pháp và công nghệ mới nhằm cải thiện hiệu suất, tính tiện nghi và bảo mật, đồng thời cũng như trải nghiệm người dùng. Thông qua dự án này, tôi hy vọng mình sẽ phát triển được khả năng tư duy cách một website thương mại điện tử hoạt động cũng như là hy vọng mình sẽ đóng góp vào sự phát triển của lĩnh vực thương mại điện tử và mang lại giải pháp hiệu quả cho những thách thức hiện đại trong ngành, đặc biệt là ngành công nghệ thông tin.

# 

# **CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT**

## **2.1 Kiến thức cơ bản các ngôn ngữ, thư viện và framework**

### ***2.1.1 API và Restful API:***

API (Application Programming Interface)là mộtgiao diện lập trình ứng dụng**,** cho phép các hệ thống hoặc thành phần phần mềm giao tiếp với nhau thông qua một tập hợp các quy tắc và giao thức nhất định. API không nhất thiết phải là web – nó có thể là một thư viện trong hệ điều hành, một interface trong chương trình C/C++, hoặc một RESTful service trên Internet. Trong ngữ cảnh Web API, API được hiểu là một HTTP-based interface cho phép client (ví dụ: frontend, mobile app, system khác...) gửi request đến backend/server để thực hiện một hành động nào đó (lấy dữ liệu, thêm, xóa, sửa...).[5]

REST (Representational State Transfer) là là một kiến trúc phần mềm được giới thiệu bởi Roy Fielding trong luận án tiến sĩ năm 2000, REST không phải là một giao thức hay công nghệ, mà là tập hợp các ràng buộc giúp xây dựng hệ thống phân tán linh hoạt, mở rộng, dễ bảo trì. Sáu ràng buộc bao gồm:[4]

* Client-Server: Phân tách giao diện người dùng (client) và lưu trữ/logic (server).
* Stateless: Mỗi request từ client phải tự chứa đủ thông tin, server không lưu trạng thái giữa các lần gọi.
* Cacheable: Các response phải cho phép hoặc không cho phép cache rõ ràng để cải thiện hiệu suất.
* Uniform Interface: Giao diện thống nhất – điều này quan trọng nhất, bao gồm:

+ Sử dụng đúng HTTP verbs (GET, POST…)

+ Tài nguyên được định danh bằng URI

+ Dữ liệu đại diện (representation) dùng JSON/XML/etc

+ HATEOAS (Hypermedia as the Engine of Application State)

* Layered System: Kiến trúc gồm nhiều lớp – client không cần biết backend có qua proxy/gateway/load balancer nào.
* Code on Demand: Server có thể cung cấp mã (JS...) cho client thực thi động.

RESTful là các API tuân thủ đúng (hoặc gần đúng) các ràng buộc của REST.[4]

### ***2.1.2 Node.js***

Node.js là một môi trường thực thi (runtime environment) cho JavaScipt với mã nguồn mở và đa nền tảng.

Node.js là mã nguồn mở: Mã nguồn của Node.js là công khai. Và nó được duy

trì bởi những người đóng góp từ khắp nơi trên thế giới.

Node.js là môi trường thực thi JavaScript.

Các trình duyệt như Chorme và FireFox có môi trường thực thi. Đó là lí do tại sao các trình duyệt có thể chạy mã JavaScript. Trước khi Node.js được tạo ra thì JavaScript chỉ có thể được chạy trên trình duyệt. Và chỉ được sử dụng để xây dựng các ứng dụng front-end.

Node.js cung cấp một môi trường chạy JavaScript ngoài trình duyệt. Nó được xây dựng trên công cụ JavaScript Chrome V8. Điều này cho phép xây dựng các ứng dụng back-end bằng cách sử dụng cùng một ngôn ngữ lập trình JavaScript.[3] [6]

### ***2.1.3 ReactJs***

React, về bản chất, là một thư viện tập trung vào việc xây dựng các thành phần UI. Thay vì thao tác trực tiếp với DOM của trình duyệt, React sử dụng một "Virtual DOM" (DOM ảo). Khi có sự thay đổi dữ liệu, React sẽ so sánh Virtual DOM hiện tại với Virtual DOM mới, và chỉ cập nhật những phần thực sự khác biệt lên DOM thực. Cơ chế này giúp tối ưu hóa hiệu năng ứng dụng, giảm thiểu số lượng thao tác DOM tốn kém và mang lại trải nghiệm người dùng mượt mà hơn.[1][6]

Một trong những điểm mạnh của React là tính "component-based". Ứng dụng React được xây dựng từ các thành phần độc lập, có thể tái sử dụng ở nhiều nơi khác nhau. Mỗi thành phần có thể chứa logic riêng, dữ liệu riêng và giao diện riêng. Cách tiếp cận này giúp đơn giản hóa việc quản lý mã nguồn, tăng tính bảo trì và khả năng mở rộng của ứng dụng.[1][2]

JSX (JavaScript XML) là một phần không thể thiếu của React. Đây là một cú pháp cho phép viết mã HTML trực tiếp trong JavaScript. JSX giúp mã nguồn trở nên dễ đọc và dễ hiểu hơn, đồng thời cho phép các nhà phát triển tận dụng sức mạnh của JavaScript để tạo ra các giao diện người dùng động và linh hoạt.[1][3]

Tuy nhiên, việc thiết lập một dự án React truyền thống có thể gặp phải một số thách thức. Các công cụ như Create React App (CRA) đã giúp đơn giản hóa quá trình này, nhưng vẫn còn một số hạn chế. CRA sử dụng Webpack để đóng gói (bundle) tất cả các module JavaScript thành một hoặc một vài file lớn. Quá trình này có thể tốn thời gian, đặc biệt đối với các ứng dụng lớn. Ngoài ra, CRA cũng khá "cứng nhắc" trong việc tùy chỉnh cấu hình bên dưới.

### ***2.1.4 Bootstrap***

Bootstrap là một framework HTML, CSS và JavaScript cho phép người dùng dễ dàng thiết kế website theo một chuẩn nhất định, tạo các website thân thiện với các thiết bị cầm tay như mobile, ipad, tablet,…

Bootstrap bao gồm những cái cơ bản có sẵn như: typography, forms, buttons, tables, navigation, modals, image carousels và nhiều thứ khác. Trong bootstrap có thêm nhiều Component, Javascript hỗ trợ cho việc thiết kế reponsive của bạn dễ dàng, thuận tiện và nhanh chóng hơn.

### ***2.1.5 Material UI***

[Material UI](http://material-ui.com/) là một thư viện các React Component đã được tích hợp thêm cả Google's Material Design. Theo như giới thiệu trên trang chủ thì được xây dựng nhờ tình cảm với React và Google's Material Design. Do đó mà phần hướng dẫn trên trang chủ của Material UI cũng đã nói nên sử dụng Material UI với React.

Material UI đem đến cho bạn và trang web của bạn một giao diện hoàn toàn mới với những button, textfield, toogle... được design theo một phong cách mới lạ, thay vì việc nhà nhà người người dùng Bootstrap như hiện nay.

### ***2.1.6 Express.js***

* Expressjs là một framework được xây dựng trên nền tảng của Nodejs. Nó cung cấp các tính năng mạnh mẽ để phát triển web hoặc mobile. Expressjs hỗ trợ các method HTTP và midleware tạo ra API vô cùng mạnh mẽ và dễ sử dụng.
* Tổng hợp một số chức năng chính của Expressjs như sau:
* Thiết lập các lớp trung gian để trả về các HTTP request.
* Define router cho phép sử dụng với các hành động khác nhau dựa trên phương thức HTTP và URL.
* Cho phép trả về các trang HTML dựa vào các tham số.

### ***2.1.7 MySQL***

MySQL là một hệ thống quản trị cơ sở dữ liệu mã nguồn mở (Relational Database Management System, viết tắt là RDBMS) hoạt động theo mô hình client-server. RDBMS là một phần mềm hay dịch vụ dùng để tạo và quản lý các cơ sở dữ liệu (Database) theo hình thức quản lý các mối liên hệ giữa chúng.

MySQL là một trong số các phần mềm RDBMS. RDBMS và MySQL thường được cho là một vì độ phổ biến quá lớn của MySQL. [Các ứng dụng web lớn nhất](https://stackshare.io/mysql) như Facebook, Twitter, YouTube, Google, và Yahoo! đều dùng MySQL cho mục đích lưu trữ dữ liệu. Kể cả khi ban đầu nó chỉ được dùng rất hạn chế nhưng giờ nó đã tương thích với nhiều hạ tầng máy tính quan trọng như Linux, macOS, Microsoft Windows, và Ubuntu.

## **2.2 Tích hợp giữa Frontend và Backend**

Việc tích hợp giữa phần frontend và backend là chìa khóa để xây dựng một ứng dụng hoàn chỉnh. Truyền thông dữ liệu hiệu quả, quản lý trạng thái và xử lý yêu cầu đòi hỏi sự hiểu biết sâu sắc về cả hai phía của ứng dụng.

Việc tích hợp hiệu quả yêu cầu sự hiểu biết về RESTful APIs và MySQL. RESTful APIs cung cấp một cách tiếp cận đơn giản và linh hoạt, trong khi MySQL cho phép truy vấn dữ liệu theo nhu cầu, giúp tối ưu hóa việc truyền thông.

* + 1. ***Kết nối dữ liệu***
* Backend: cung cấp các API xử lý nghiệp vụ từ frontend.
* Frontend: gọi các API từ backend để nhận và hiển thị dữ liệu.

***2.2.2 Bảo mật*:** sử dụng JWT (JSON Web Token) để bảo vệ dữ liệu.

## **2.3 Giả thiết khoa học và phương pháp nghiên cứu dự án**

### Giả thiết khoa học: hiệu suất và sự linh hoạt

Giả thiết cơ bản của dự án là sự kết hợp giữa Node.js và React.js sẽ tạo ra một hệ thống website thương mại điện tử mạnh mẽ với hiệu suất cao và trải nghiệm người dùng đáng chú ý. Điều này dựa trên khả năng của Node.js xử lý đa nhiệm và sự linh hoạt của React.js trong quản lý trạng thái.

### Phương pháp nghiên cứu: phát triển liên tục và tiến hành kiểm thử

Phương pháp nghiên cứu sẽ áp dụng tích hợp liên tục để đảm bảo sự linh hoạt trong quá trình phát triển và kiểm thử hiệu suất để đánh giá khả năng chịu tải và ổn định của hệ thống trong môi trường thực tế.

# **CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU**

## **3.1 Phân tích và đặt tả yêu cầu hệ thống:**

### **3.1.1 Phân tích chức năng:**

Đối với người dùng (khách hàng):

- Đăng nhập, đăng xuất khổi hệ thống: đăng nhập đăng ký bằng tài khoản thông thường và bằng email.

- Mua hàng: khách hàng có thể mua hàng bằng cách thêm vào giỏ hàng và mua hàng.

- Thông tin cá nhân: người dùng có thể xem thông tin của bản thân và cập nhật thông tin.

- Sản phẩm: xem thông tin chi tiết của sản phẩm.

Đối với quản trị viên:

- Quản lý sản phẩm: xem danh sách, thêm, sửa và xóa sản phẩm.

- Quản lý danh mục: xem danh sách, thêm, sửa và xóa danh mục.

- Quản lý người dùng: xem danh sách, cập nhật người dùng.

- Quản lý đơn hàng: xem danh sách, cập nhật trạng thái đơn hàng.

### **3.1.2 Yêu cầu phi chức năng**

- Khả năng mở rộng: Dễ nâng cấp, thêm tính năng mới trong tương lai.

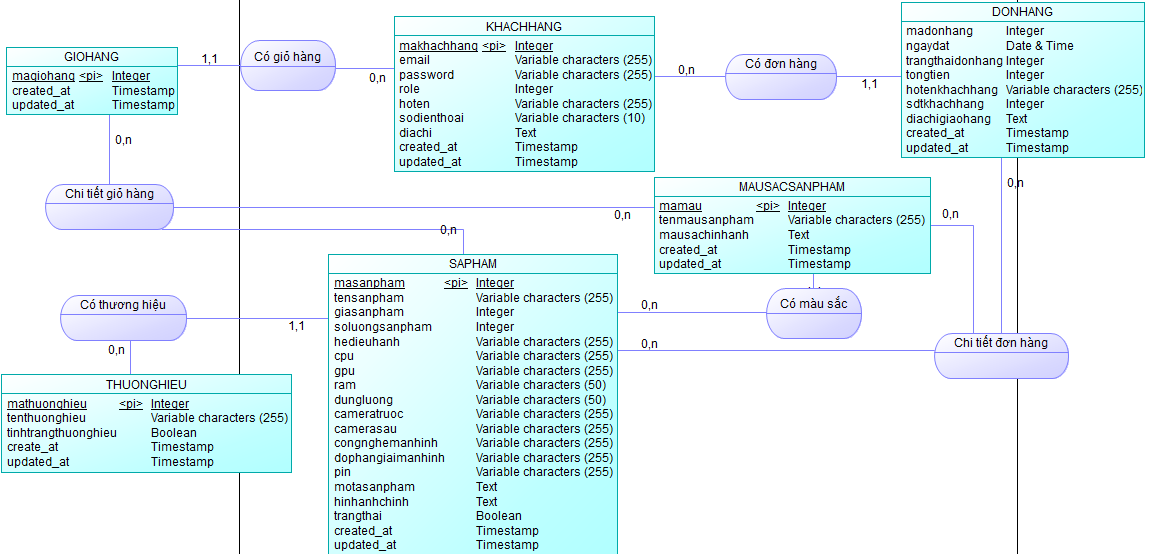
- Bảo mật: Sử dụng JWT để xác thực, phân quyền rõ ràng giữa user và admin.

- Hiệu năng: Giao diện tải nhanh, truy vấn dữ liệu tối ưu.

- Dễ sử dụng: Giao diện thân thiện với khách hàng và quản trị viên.

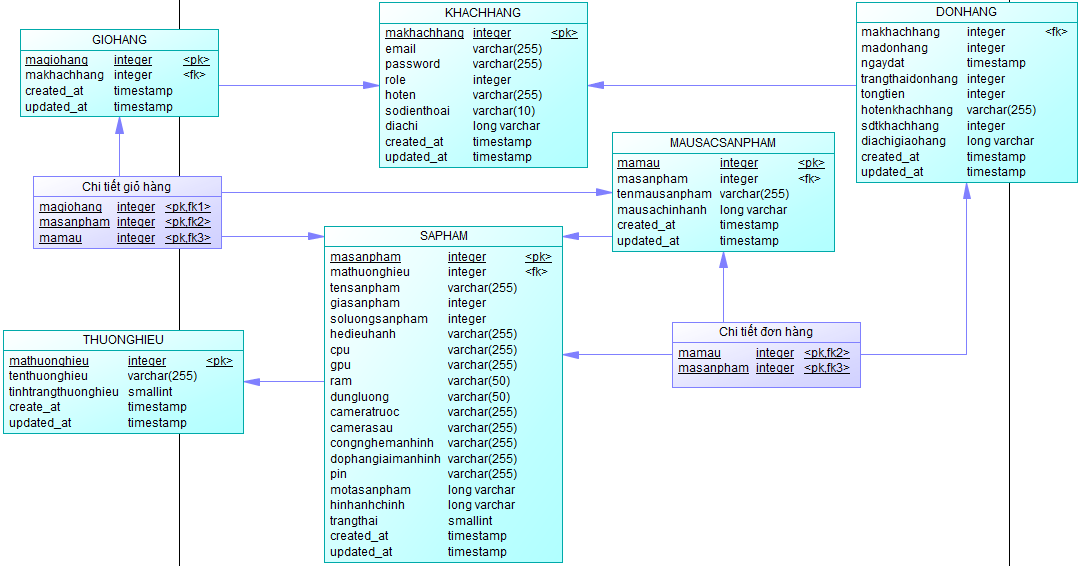
## **Thiết kế hệ thống**

### ***Mô hình thực thể kết hợp***



Hình 3. 1 Mô hình thực thể kết hợp

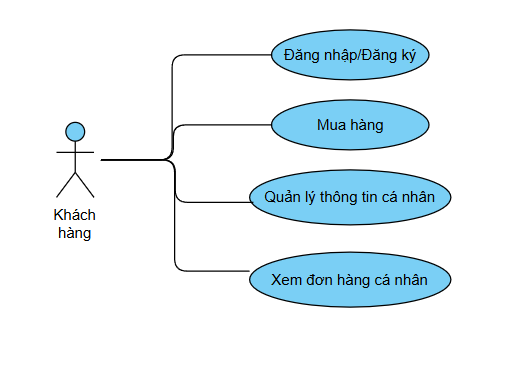
### ***Mô hình vật lý***



Hình 3. 2 Mô hình vật lý

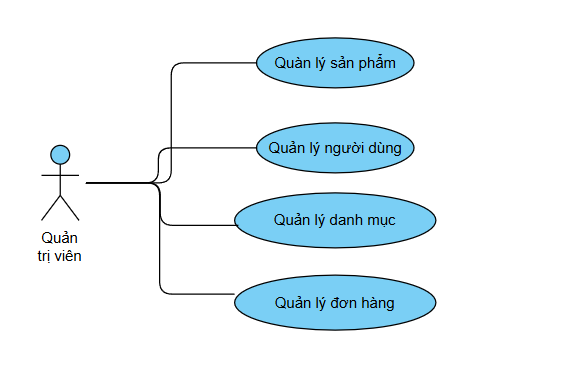
### ***Sơ đồ Use Case:***

- Đối với khách hàng



Hình 3. 3 Use Case khách hàng

- Đối với quản trị viên



Hình 3. 4 Use Case quản trị viên

### ***Mô tả các bảng trong lược đồ cơ sở dữ liệu***

|  |  |
| --- | --- |
| **Tên bảng** | **Diễn giải** |
| KHACHHANG | Khách hàng |
| GIOHANG | Giỏ hàng |
| CHITIETGIOHANG | Chi tiết giỏ hàng |
| SANPHAM | Sản phẩm |
| MAUSACSANPHAM | Màu sắc sản phẩm |
| THUONGHIEU | Thương hiệu |
| DONHANG | Đơn hàng |
| CHITIETDONHANG | Chi tiết đơn hàng |

Bảng 3. 1 Bảng mô tả các bảng trong lược đồ cơ sở dữ liệu

**Bảng KHACHHANG (Khách hàng)**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **TT** | **Thuộc tính** | **Diễn giải** | **Loại giá trị** | **Kiểu dữ liệu** | **Miền giá trị** | **Chiều dài** | **Ghi chú** |
| 1. | makhachhang | Mã khách hàng | Bắt buộc | Int | Khóa chính |  |  |
| 2. | email | Email |  | Varchar |  | 255 |  |
| 3. | password | Password |  | Varchar |  | 255 |  |
| 4. | role | Role |  | Int |  |  |  |
| 5. | hoten | Họ tên |  | Varchar |  | 255 |  |
| 6. | sodienthoai | Số điện thoại |  | Varchar |  | 10 |  |
| 7. | diachi | Địa chỉ |  | Text |  |  |  |

Bảng 3. 2 Bảng khách hàng

**Bảng GIOHANG (Giỏ hàng)**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **TT** | **Thuộc tính** | **Diễn giải** | **Loại giá trị** | **Kiểu dữ liệu** | **Miền giá trị** | **Chiều dài** | **Ghi chú** |
| 1. | magiohang | Mã giỏ hàng | Bắt buộc | Int | Khóa chính |  |  |
| 2. | makhachhang | Mã khách hàng | Bắt buộc | Int | Khóa ngoại |  |  |

Bảng 3. 3 Bảng giỏ hàng

**Bảng CHITIETGIOHANG (Chi tiết giỏ hàng)**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **TT** | **Thuộc tính** | **Diễn giải** | **Loại giá trị** | **Kiểu dữ liệu** | **Miền giá trị** | **Chiều dài** | **Ghi chú** |
| 1. | magiohang | Mã giỏ hàng | Bắt buộc | Int | Khóa chính, khóa ngoại |  |  |
| 2. | masanpham | Mã sản phẩm | Bắt buộc | Int | Khóa chính, khóa ngoại |  |  |
| 3. | mamau | Mã màu | Bắt buộc | Int | Khóa chính, khóa ngoại |  |  |
| 4. | soluong | Số lượng |  | Int |  |  |  |
| 5. | gia | Giá |  | Decimal (18,2) |  |  |  |

Bảng 3. 4 Bảng chi tiết giỏ hàng

**Bảng SANPHAM (Sản phẩm)**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **TT** | **Thuộc tính** | **Diễn giải** | **Loại giá trị** | **Kiểu dữ liệu** | **Miền giá trị** | **Chiều dài** | **Ghi chú** |
| 1. | masanpham | Mã sản phẩm | Bắt buộc | Int | Khóa chính |  |  |
| 2. | mathuonghieu | Mã thương hiệu | Bắt buộc | Int | Khóa ngoại |  |  |
| 3. | tensanpham | Tên sản phẩm |  | Varchar |  | 255 |  |
| 4. | giasanpham | Giá sản phẩm |  | Int |  | 255 |  |
| 5. | soluongsanpham | Số lượng sản phẩm |  | Int |  | 255 |  |
| 6. | hedieuhanh | Hệ điều hành |  | Varchar |  | 255 |  |
| 7. | cpu | CPU |  | Varchar |  | 255 |  |
| 8. | gpu | GPU |  | Varchar |  | 255 |  |
| 9. | ram | RAM |  | Varchar |  | 50 |  |
| 10. | dungluong | Dung lượng |  | Varchar |  | 50 |  |
| 11. | cameratruoc | Camera trước |  | Varchar |  | 255 |  |
| 12. | camerasau | Camera sau |  | Varchar |  | 255 |  |
| 13. | congnghemanhinh | Công nghệ phần mềm |  | Varchar |  | 255 |  |
| 14. | dophangiaimanhinh | Độ phân giải màn hình |  | Varchar |  | 255 |  |
| 15. | pin | Pin |  | Varchar |  | 255 |  |
| 16. | motasanpham | Mô tả sản phẩm |  | Text |  |  |  |
| 17. | hinhanhchinh | Hình ảnh chính |  | Text |  |  |  |
| 18. | trangthai | Trạng thái |  | Boolean |  |  |  |

Bảng 3. 5 Bảng sản phẩm

**Bảng MAUSACHINHANH (Màu sắc sản phẩm)**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **TT** | **Thuộc tính** | **Diễn giải** | **Loại giá trị** | **Kiểu dữ liệu** | **Miền giá trị** | **Chiều dài** | **Ghi chú** |
| 1. | mamau | Mã màu | Bắt buộc | Int | Khóa chính |  |  |
| 2. | masanpham | Mã sản phẩm | Bắt buộc | Int | Khóa ngoại |  |  |
| 3. | tenmausanpham | Tên màu sản phẩm |  | Varchar |  | 50 |  |
| 4. | mausachinhanh | Màu sắc hình ảnh |  | Text |  |  |  |

Bảng 3. 6 Bảng màu sắc sản phẩm

**Bảng THUONGHIEU (Thương hiệu)**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **TT** | **Thuộc tính** | **Diễn giải** | **Loại giá trị** | **Kiểu dữ liệu** | **Miền giá trị** | **Chiều dài** | **Ghi chú** |
| 1. | mathuonghieu | Mã thương hiệu | Bắt buộc | Int | Khóa chính |  |  |
| 2. | tenthuonghieu | Tên thương hiệu |  | Varchar |  | 255 |  |
| 3. | trangthaithuonghieu | Trạng thái thương hiệu |  | Boolean |  |  |  |

Bảng 3. 7 Bảng thương hiệu

**Bảng DONHANG (Đơn hàng)**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **TT** | **Thuộc tính** | **Diễn giải** | **Loại giá trị** | **Kiểu dữ liệu** | **Miền giá trị** | **Chiều dài** | **Ghi chú** |
| 1. | madonhang | Mã đơn hàng | Bắt buộc | Int | Khóa chính |  |  |
| 2. | makhachhang | Mã khách hàng | Bắt buộc | Int | Khóa ngoại |  |  |
| 3. | ngaydat | Ngày đặt |  | Date & Time |  |  |  |
| 4. | trangthaidonhang | Trạng thái đơn hàng |  | Int |  |  |  |
| 5. | tongtien | Tổng tiền |  | Int |  |  |  |
| 6. | hotenkhachhang | Họ tên khách hàng |  | Varchar |  | 255 |  |
| 7. | sdtkhachhang | Số điện thoại khách hàng |  | Int |  |  |  |
| 8. | diachigiaohang | Địa chỉ giao hàng |  | Text |  |  |  |

Bảng 3. 8 Bảng đơn hàng

**Bảng CHITIETDONHANG (Chi tiết đơn hàng)**

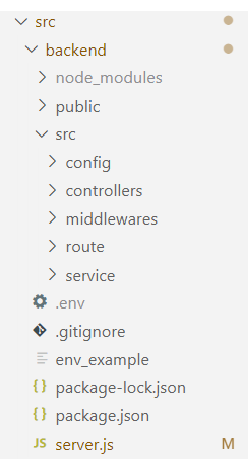
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **TT** | **Thuộc tính** | **Diễn giải** | **Loại giá trị** | **Kiểu dữ liệu** | **Miền giá trị** | **Chiều dài** | **Ghi chú** |
| 1. | madonhang | Mã đơn hàng | Bắt buộc | Int | Khóa chính, khóa ngoại |  |  |
| 2. | masanpham | Mã sản phẩm | Bắt buộc | Int | Khóa chính, khóa ngoại |  |  |
| 3. | mamau | Mã màu | Bắt buộc | Int | Khóa chính, khóa ngoại |  |  |
| 4. | giatien | Giá tiền |  | Int |  |  |  |
| 5. | soluong | Số lượng |  | Int |  |  |  |

Bảng 3. 9 Bảng chi tiết đơn

## **Cài đặt môi trường**

Tải NodeJs về, sau đó vào Visual Studio Code bật **Terminal** lên và gõ câu lệnh “npm init”, điền các thông tin cần thiết vào như Nodejs version, npm version, author, link Github,… sau đó tạo file server.js và tiến hành tổ chức cây thư mục.

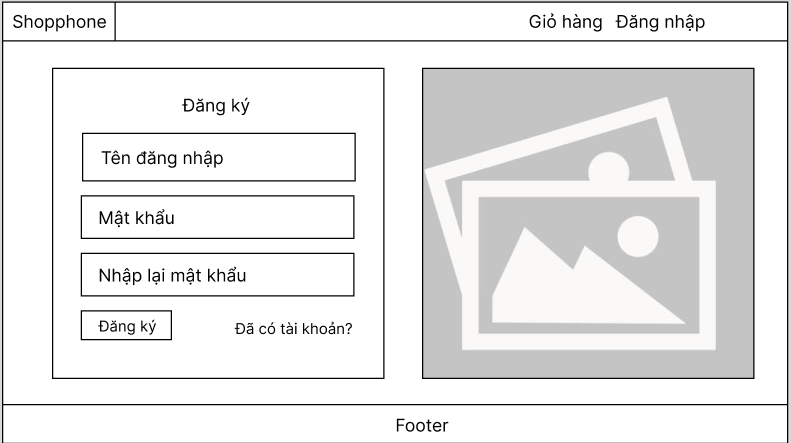
Tiến hành tạo dự án Nodejs theo Cây thư mục trên, từ Thư mục server.js bắt đầu thêm một số thư viện cần thiết cho dự án như nodemon, express, path, mysql2,… sau đó chạy thử dự án trên môi trường Nodejs từ file server.js sau đó tiến hành viết dự án theo mô hình MVC với điều kiện: các route hoặc các đường link viết bên route, Controller.js sẽ chứa các hàm thực thi câu lệnh liên quan đến việc thêm sửa xóa và hiển thị dữ liệu. Middlewares là thư mục dùng để xác thực bảo mật. Config sẽ là thư mục giúp định nghĩa các đường dẫn cũng như là file cơ sở dữ liệu cho thư mục.



Hình 3. 5 Cấu trúc cây thư mục backend

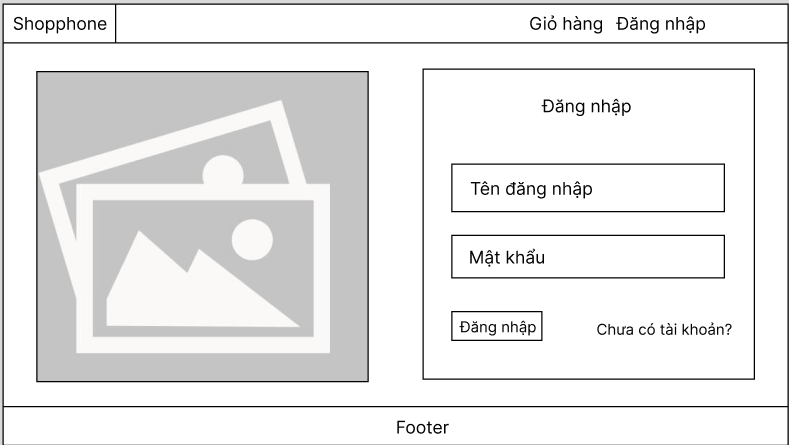
## **Phác thảo giao diện**

Giao diện trang đăng ký



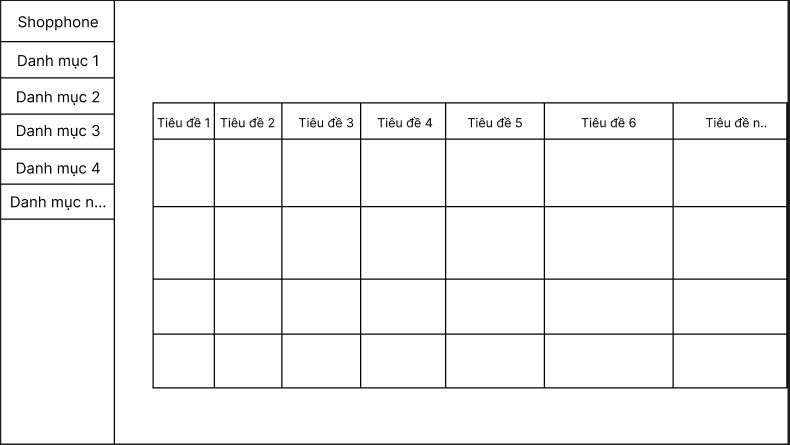
Hình 3. 6 Phác thảo giao diện đăng ký

Giao diện trang đăng nhập



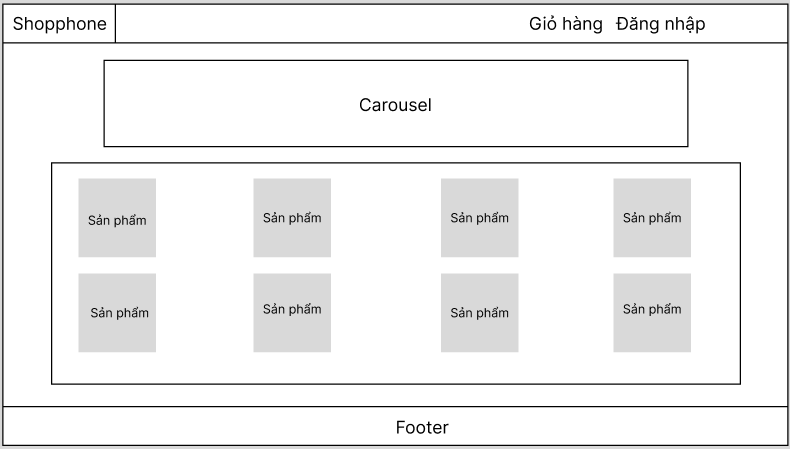
Hình 3. 7 Phác thảo giao diện đăng nhập

Giao diện trang quản lý nghiệp vụ của admin



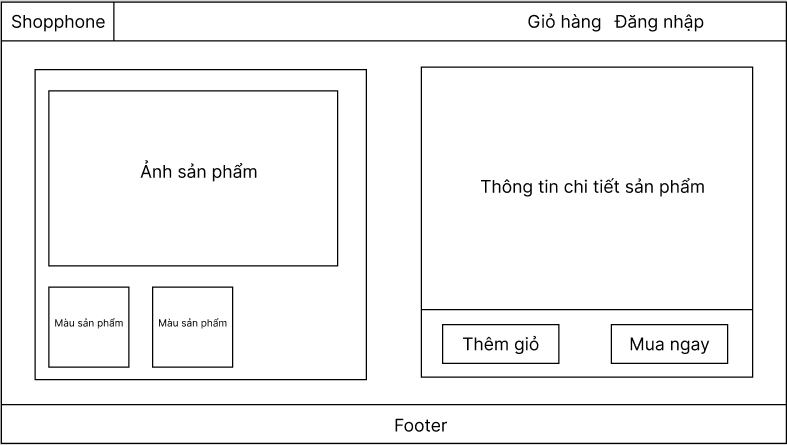
Hình 3. 8 Phác thảo giao diện quản lý nghiệp vụ của admin

Giao diện trang chủ của người dùng



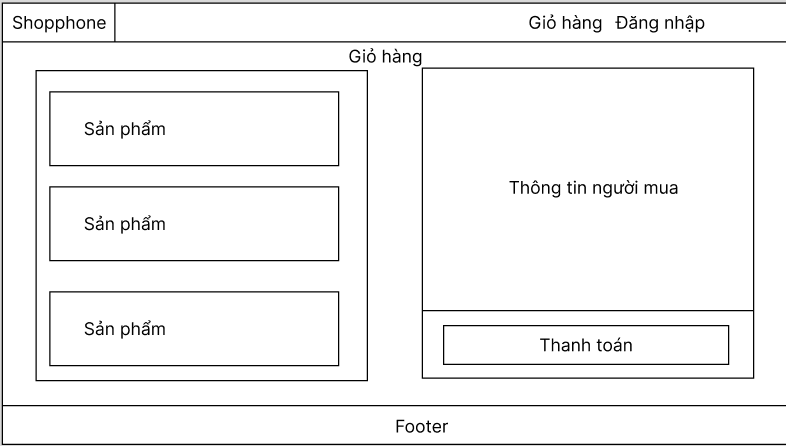
Hình 3. 9 Phác thảo giao diện trang chủ

Giao diện chi tiết sản phẩm



Hình 3. 10 Phác thảo giao diện trang thông tin chi tiết sản phẩm

Trang giao diện giỏ hàng



Hình 3. 11 Phác thảo giao diện trang giỏ hàng

Giao diện trang thông tin cá nhân của khách hàng



Hình 3. 12 Phác thảo giao diện trang thông tin người dùng

# **CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU**

## **Dữ liệu thực nghiệm**

Dữ liệu về khách hàng



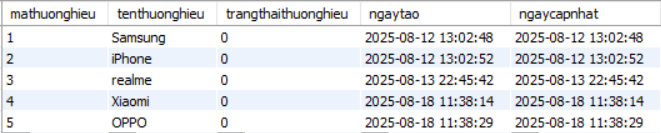
Hình 4. 1 Dữ liệu về khách hàng

Dữ liệu về sản phẩm



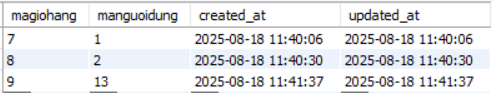
Hình 4. 2 Dữ liệu về sản phẩm

Dữ liệu về thương hiệu



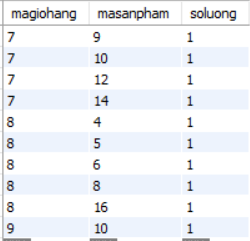
Hình 4. 3 Dữ liệu về thương hiệu

Dữ liệu về giỏ hàng



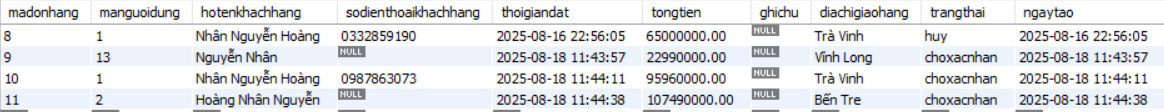
Hình 4. 4 Dữ liệu về giỏ hàng

Dữ liệu về chi tiết giỏ hàng



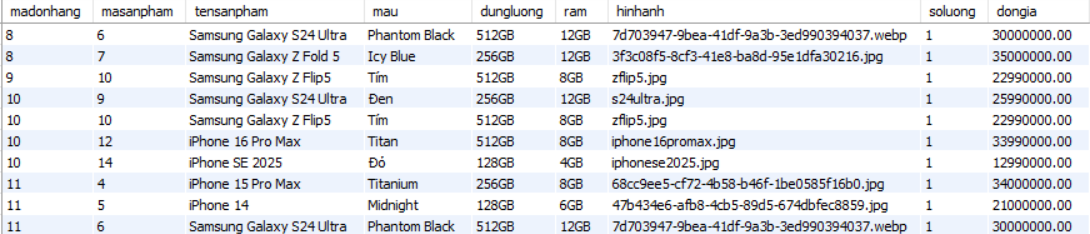
Hình 4. 5 Dữ liệu về chi tiết giỏ hàng

Dữ liệu về đơn hàng



Hình 4. 6 Dữ liệu về đơn hàng

Dữ liệu về chi tiết đơn hàng

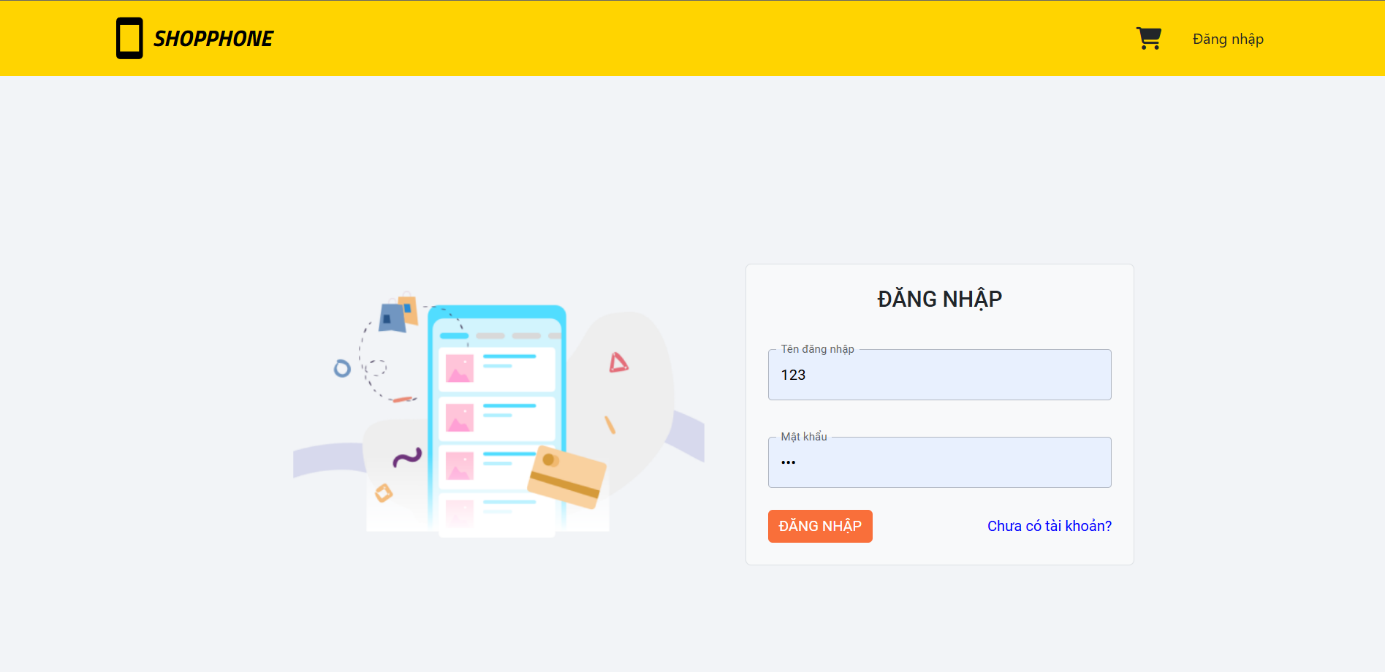


Hình 4. 7 Dữ liệu về chi tiết đơn hàng

## **Các chức năng**

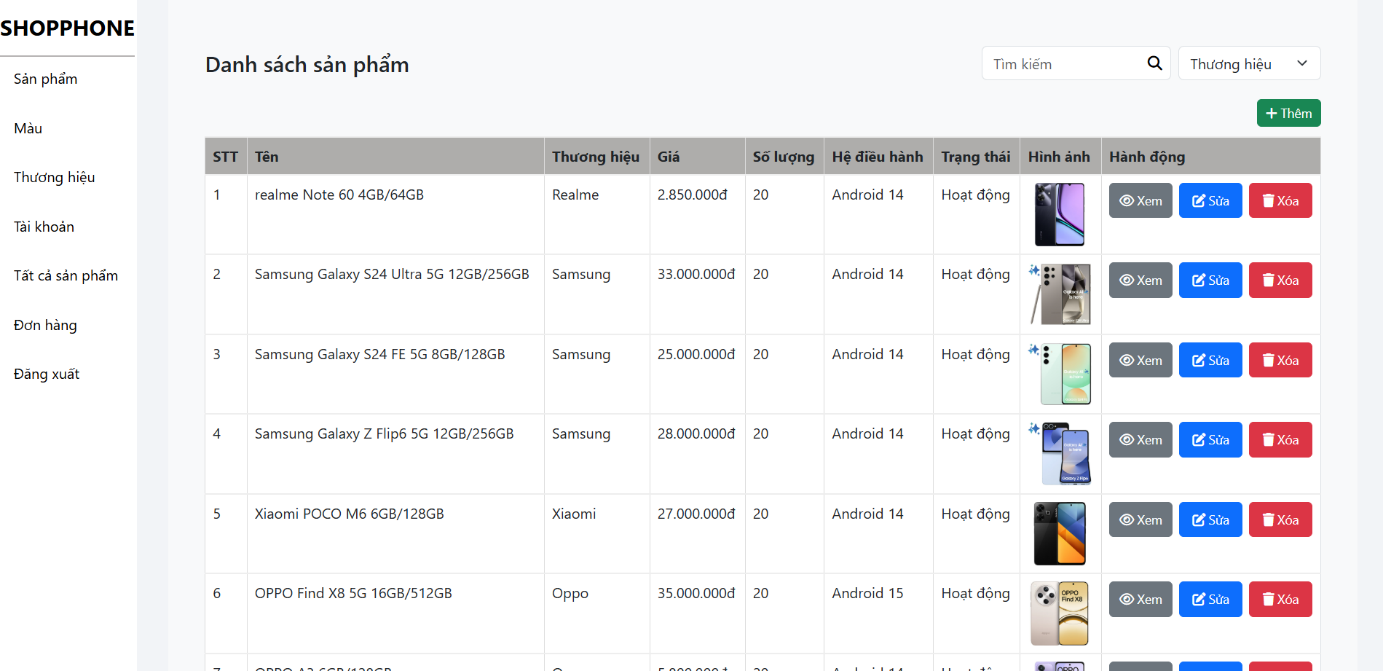
### ***4.2.1 Đối với admin***

Xây dựng giao diện đăng nhập



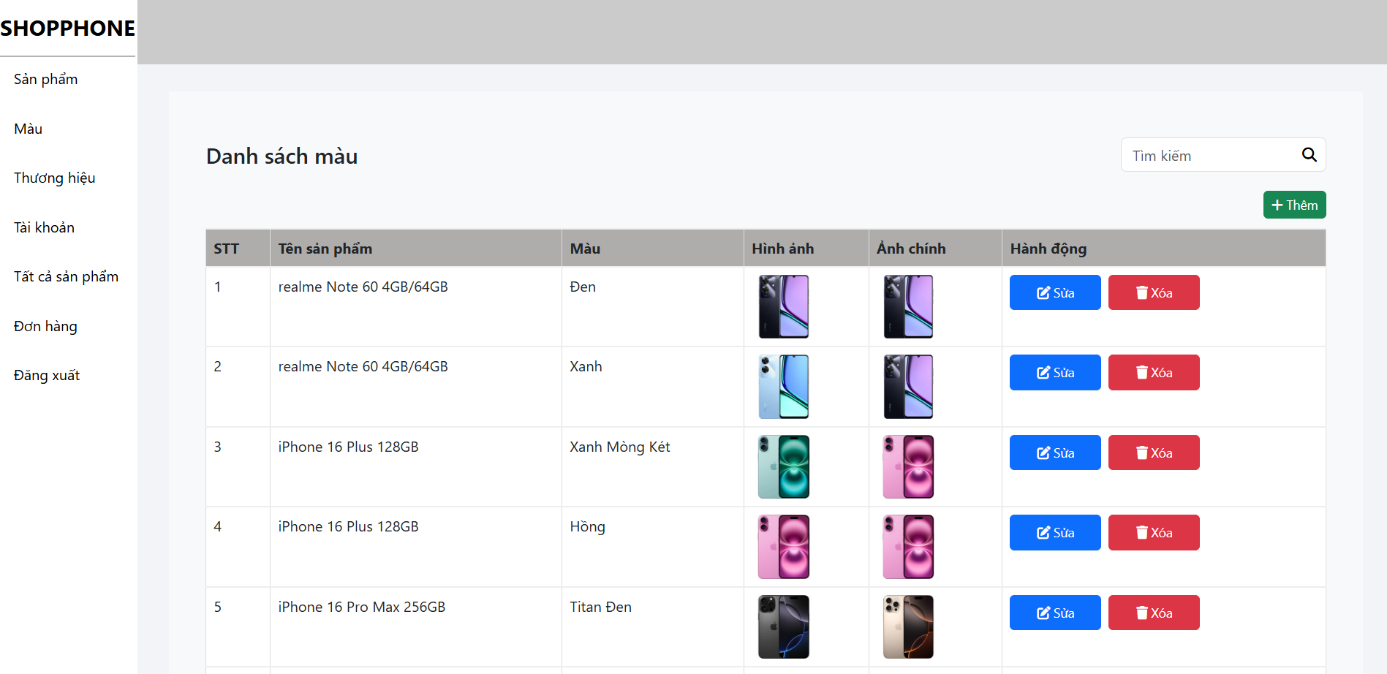
Hình 4. 8 Giao diện đăng nhập của admin

Xây dựng giao diện và chức năng nghiệp vụ của sản phẩm (admin có thể lọc sản phẩm, xem, thêm, sửa và xóa sản phẩm)



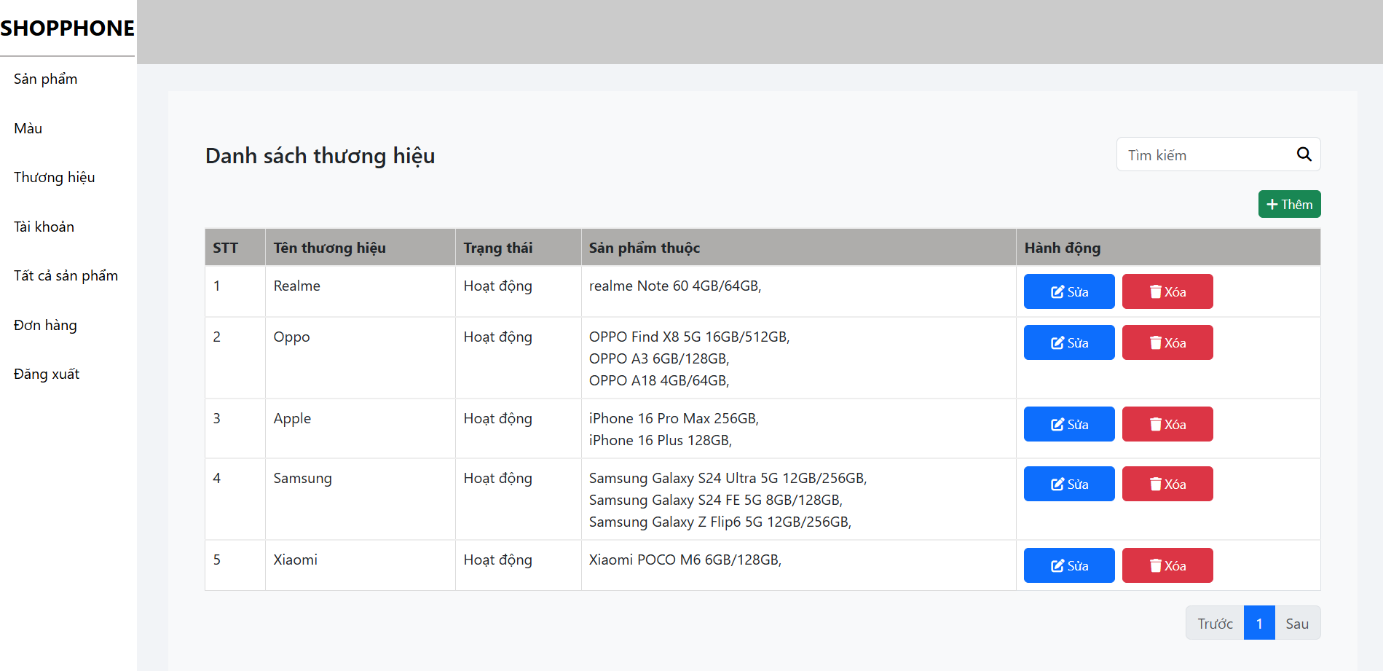
Hình 4. 9 Giao diện trang quản lý sản phẩm của admin

Xây dựng giao diện quản lý màu sản phẩm (admin có thể thêm màu sản phẩm cho sản phẩm muốn thêm, sửa và xóa màu sản phẩm)



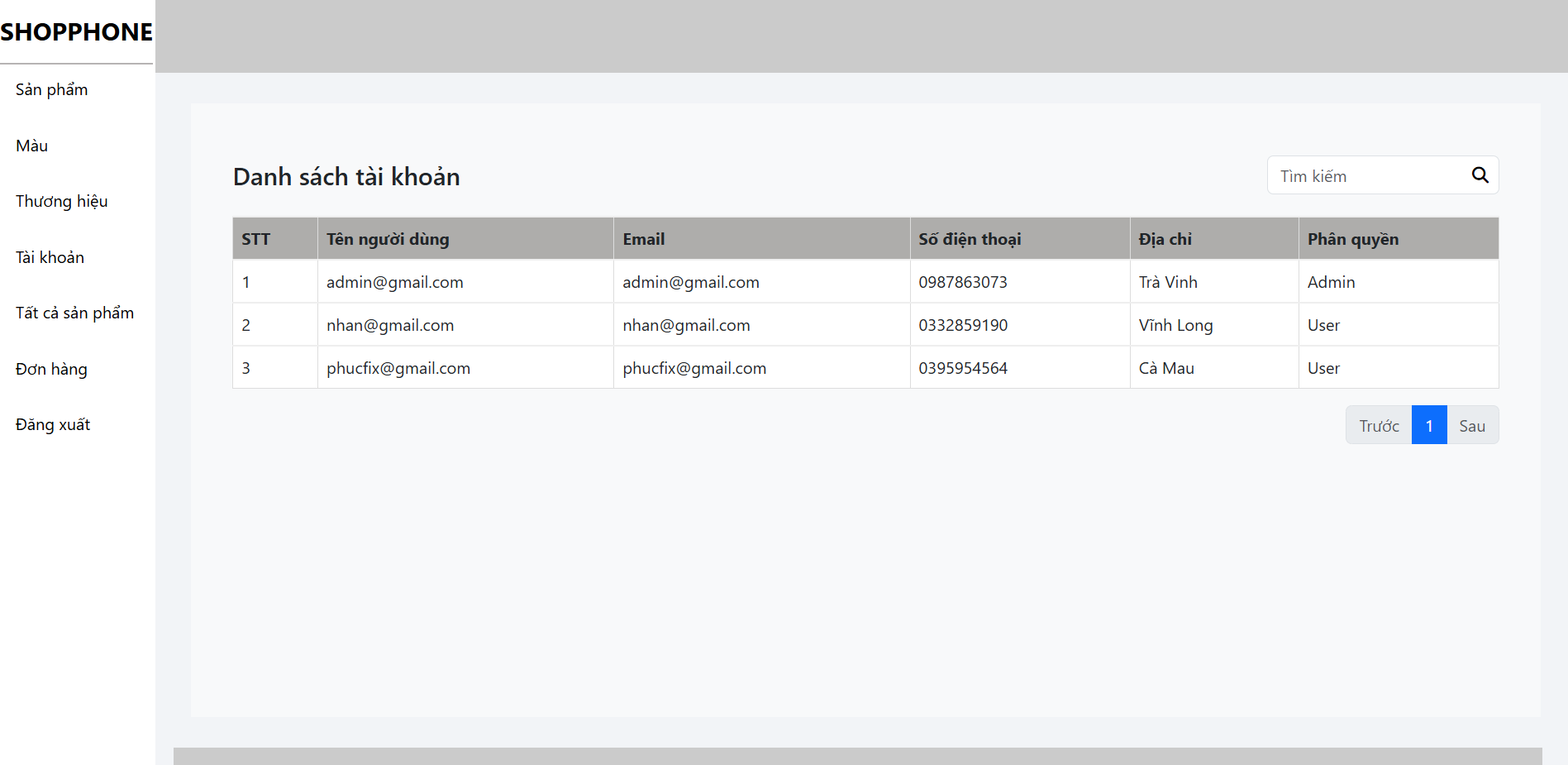
Hình 4. 10 Giao diện trang quản lý màu sản phẩm của admin

Xây dựng giao diện quản lý thương hiệu (admin có thể tìm kiếm tên thương hiệu, xem, thêm, sửa và xóa thương hiệu)



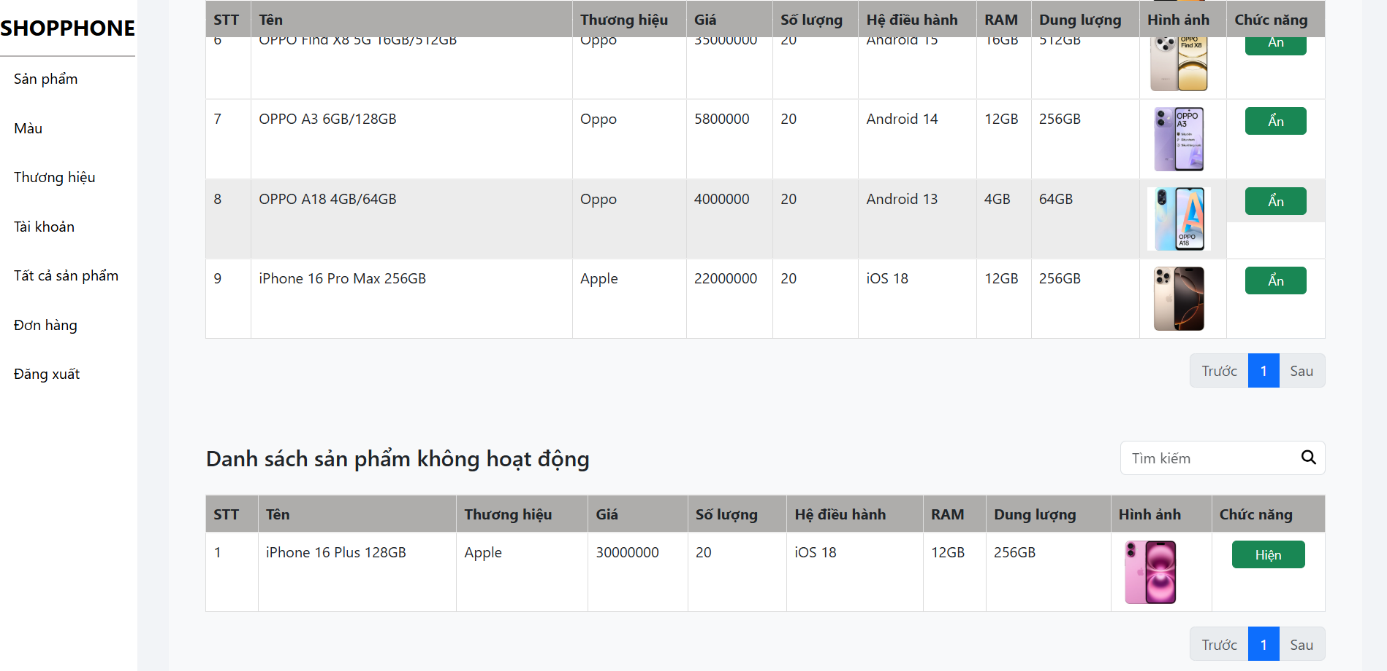
Hình 4. 11 Giao diện trang quản lý thương hiệu của admin

Xây dựng giao diện quản lý tài khoản (admin có thể xem thông tin của các tài khoản mà hệ thống đang có)



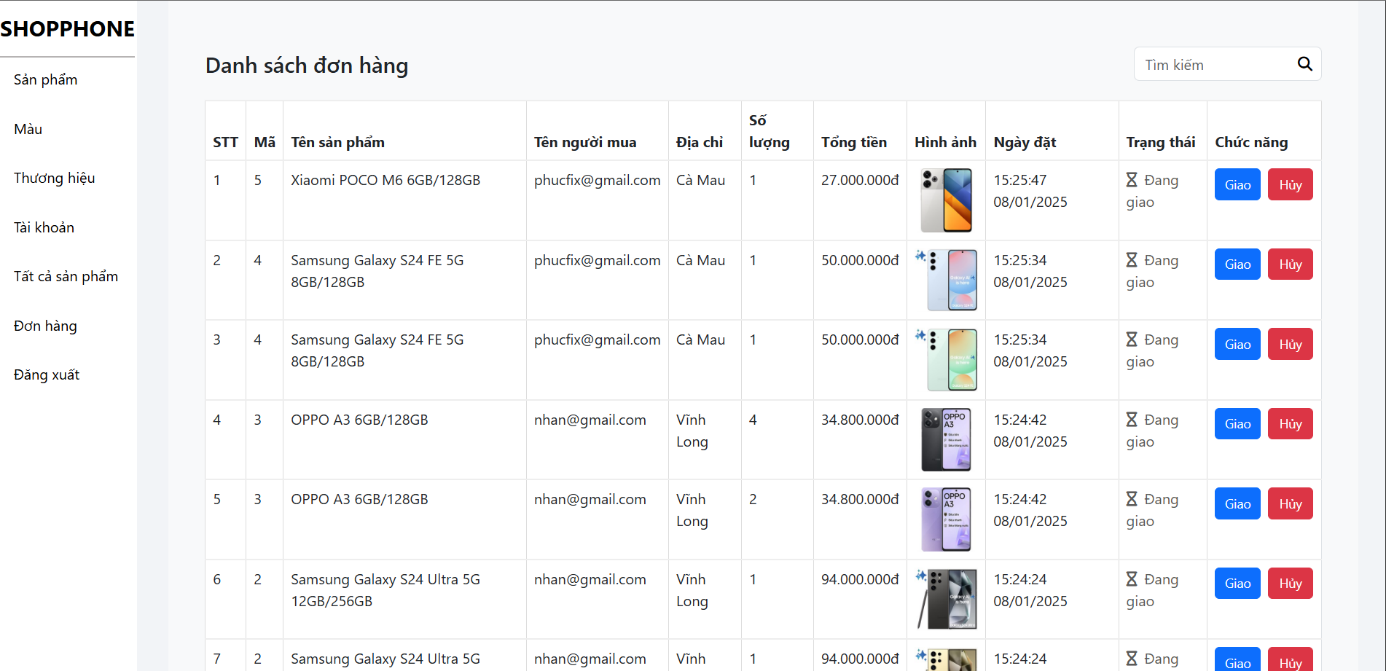
Hình 4. 12 Giao diện trang quản lý tài khoản của admin

Xây dựng giao quản lý tất cả sản phẩm bao gồm các sản phẩm đang hoạt động và các sản phẩm không hoạt động



Hình 4. 13 Giao diện trang quản lý tất cả sản phẩm của admin

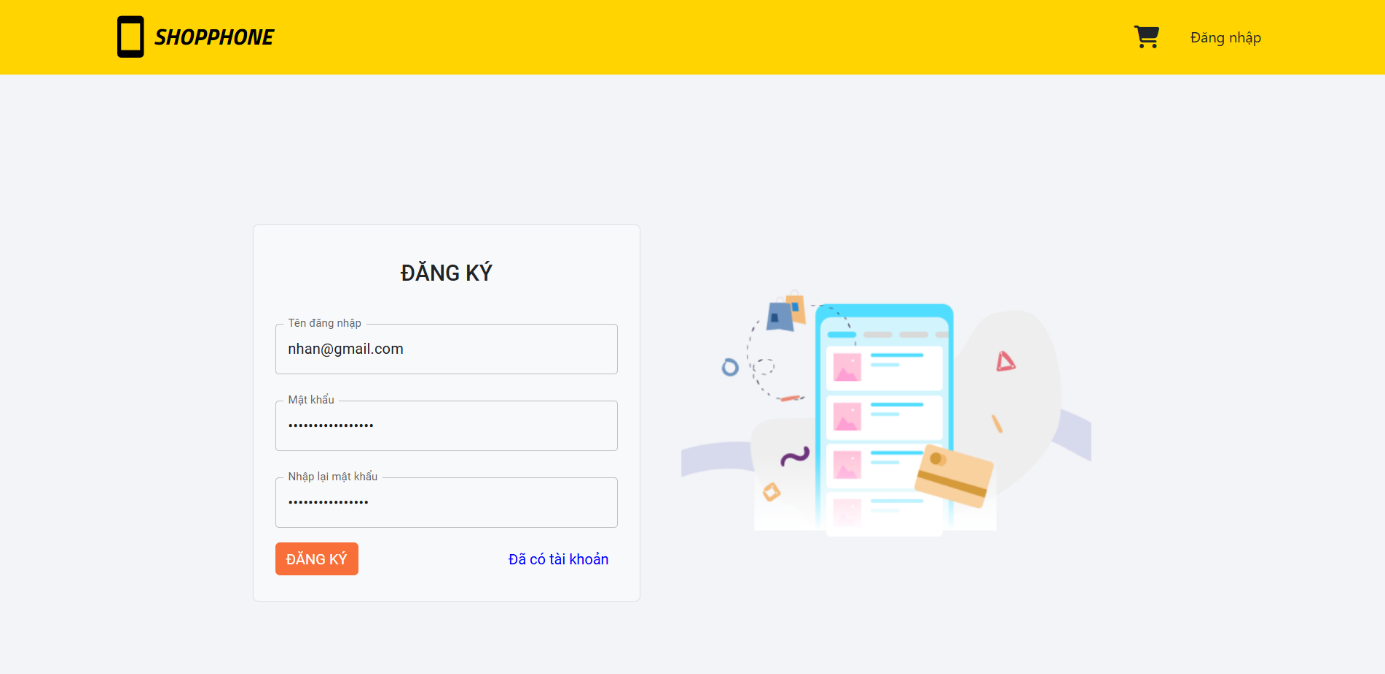
Xây dựng giao diện quản lý đơn hàng (admin có thể xem đơn hàng, tìm kiếm đơn hàng, thao tác với đơn hàng như giao hàng và hủy đơn hàng)



Hình 4. 14 Giao diện trang quản lý đơn hàng

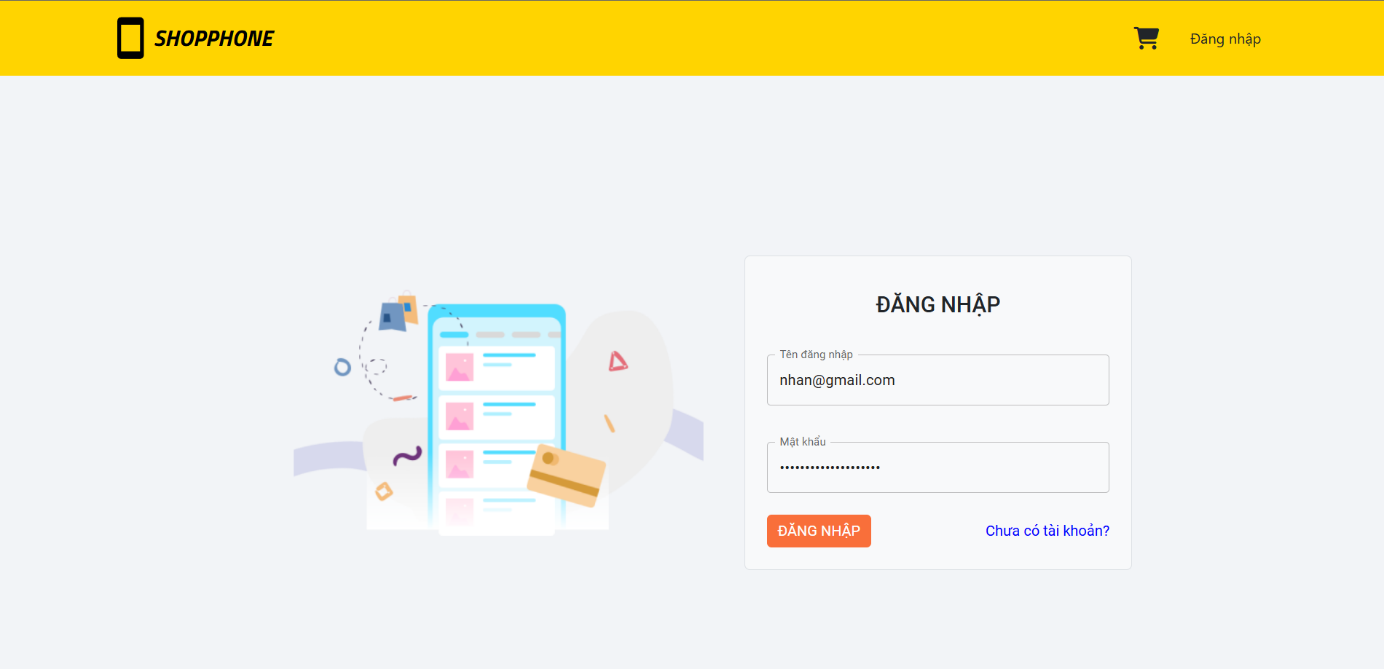
### ***4.2.2 Đối với người dùng***

Xây dựng giao diện trang đăng ký



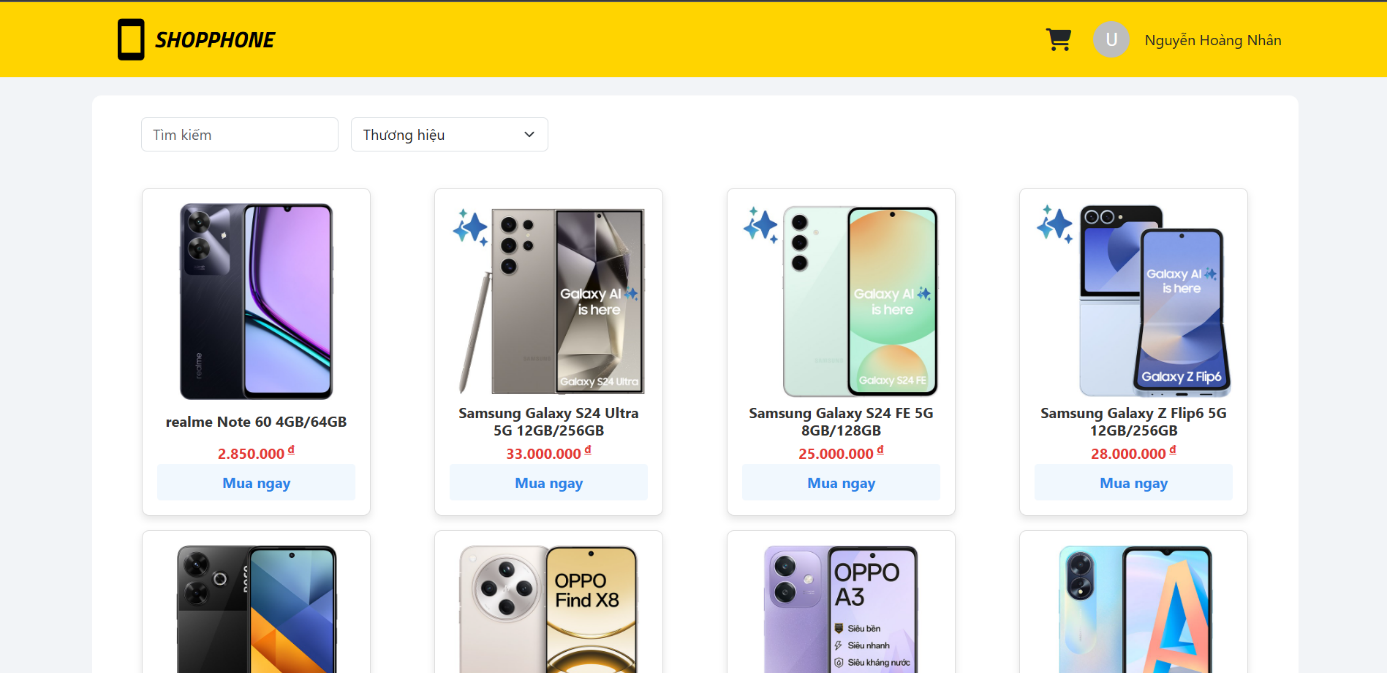
Hình 4. 15 Giao diện đăng ký

Xây dựng giao diện trang đăng nhập



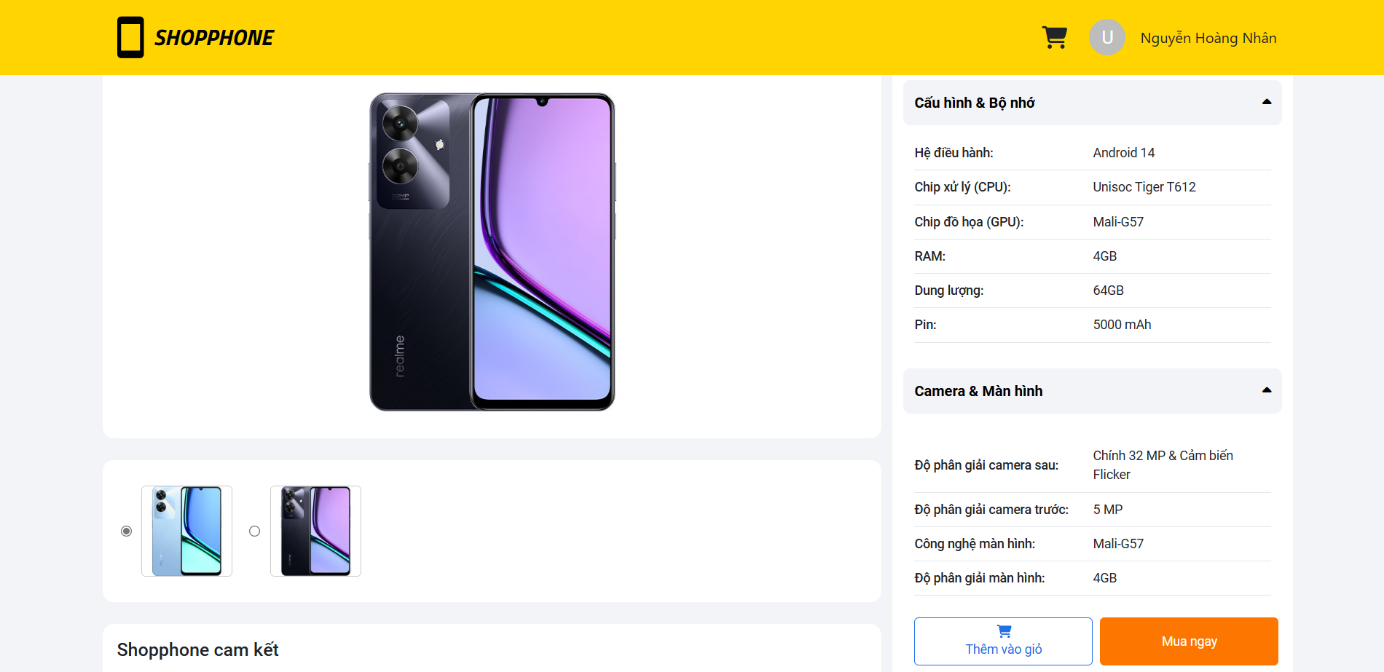
Hình 4. 16 Giao diện đăng nhập

Xây dựng giao diện trang chủ gồm: lọc sản phẩm theo tên và thương hiệu, xem các sản phẩm, giỏ hàng



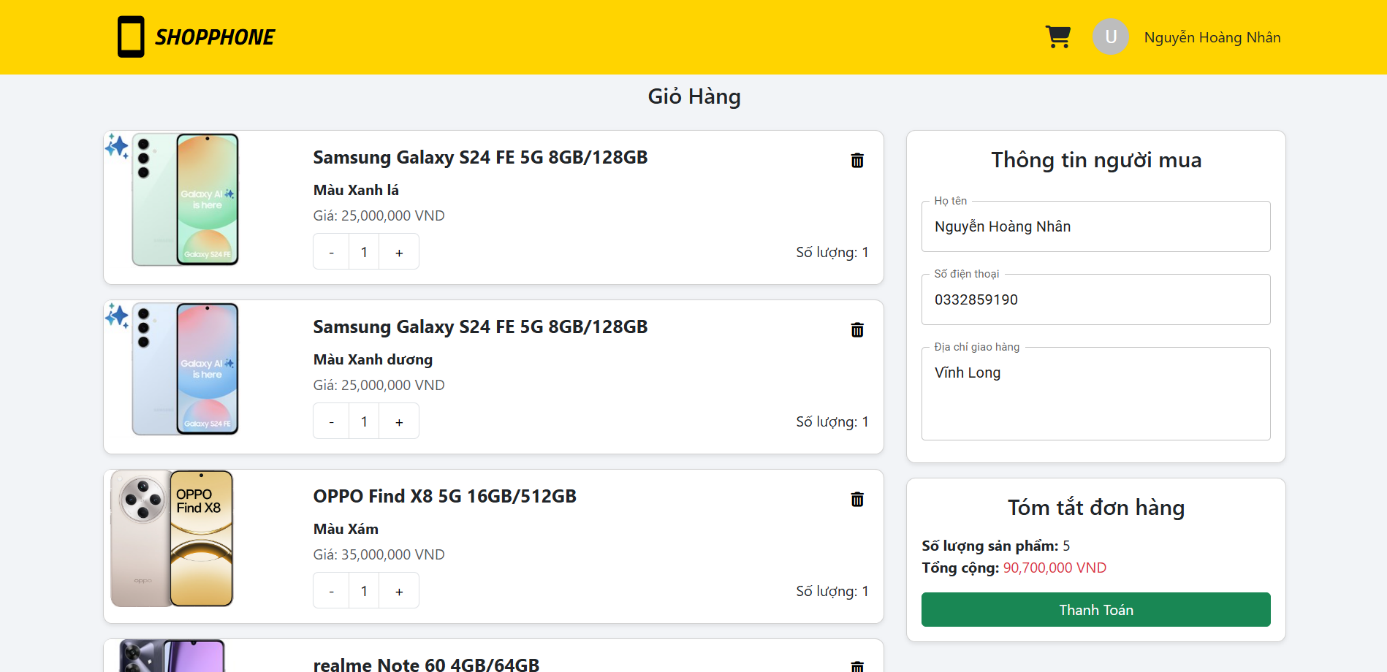
Hình 4. 17 Giao diện trang chủ

Thiết kế giao diện thông tin chi tiết sản phẩm gồm thông tin sản phẩm, thêm giỏ hàng và mua hàng



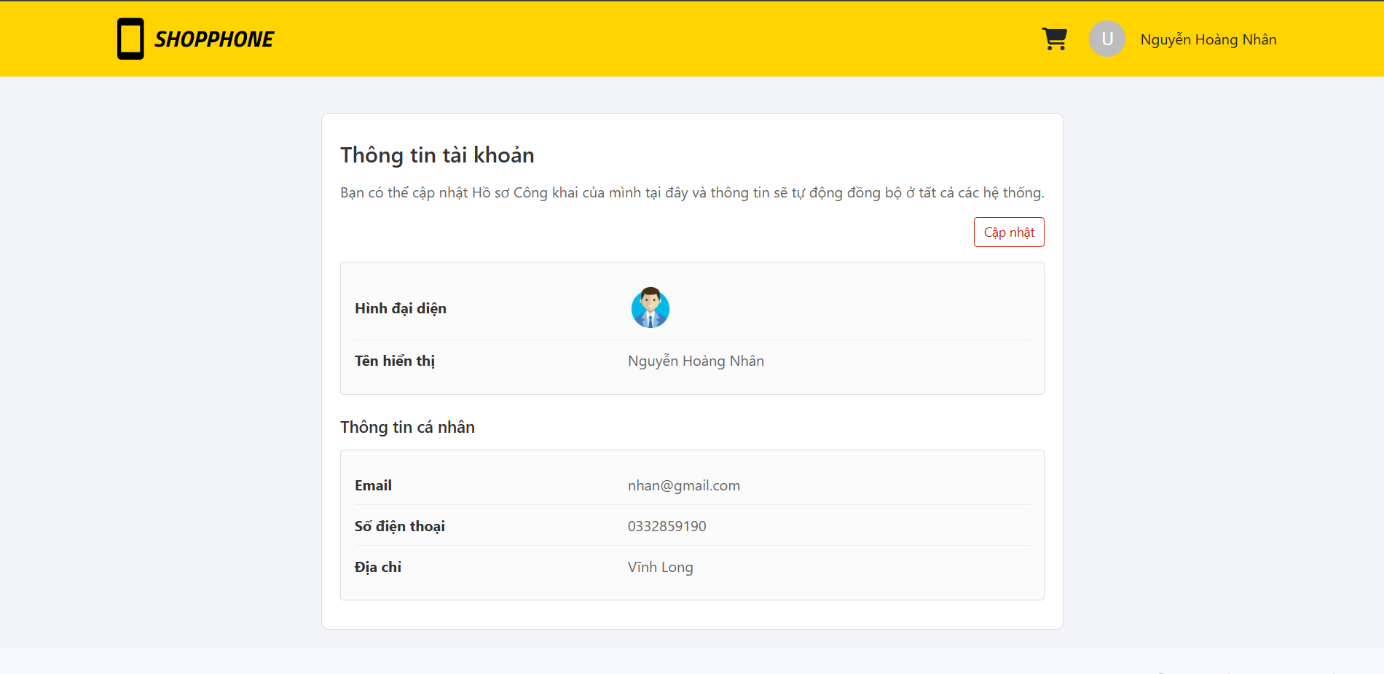
Hình 4. 18 Giao diện trang thông tin chi tiết sản phẩm

Thiết kế trang giao diện giỏ hàng gồm thông tin các sản phẩm trong giỏ hàng của khách hàng, thông tin khách hàng và thanh toán đơn hàng

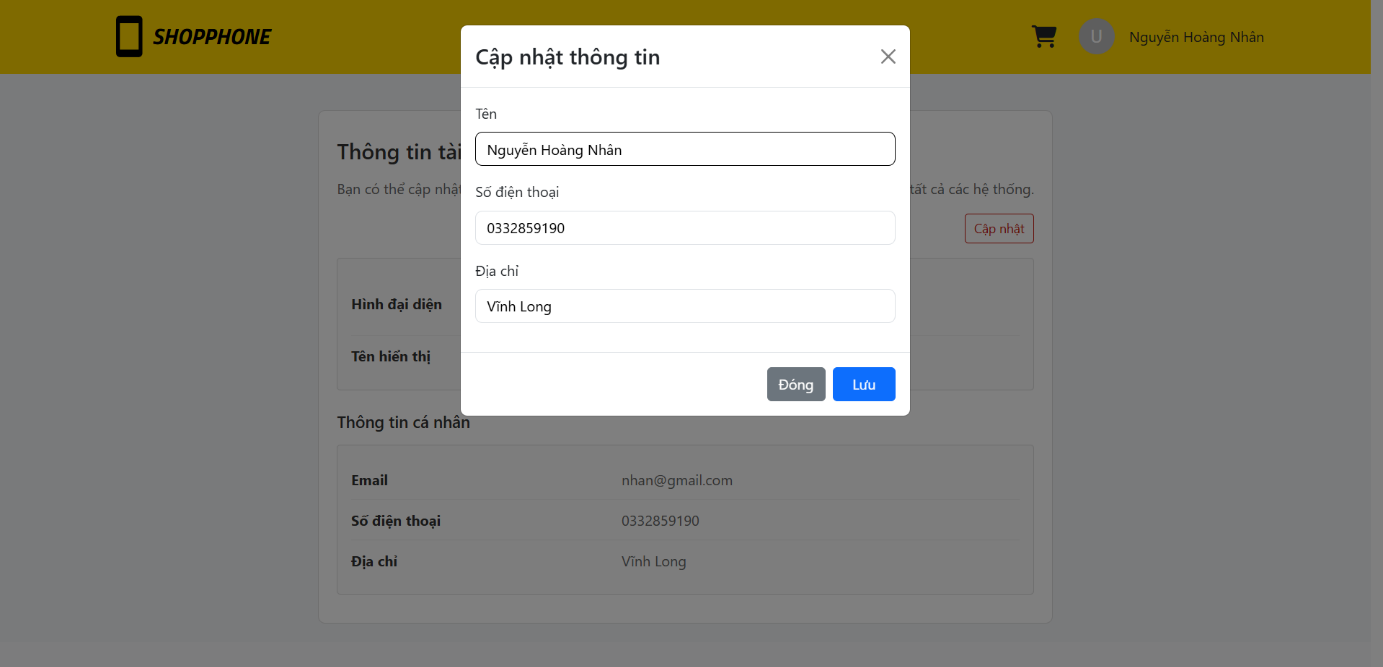


Hình 4. 19 Giao diện giỏ hàng

Thiết kế trang giao diện thông tin người dùng gồm thông tin cá nhân của khách hàng và cập nhật thông tin



Hình 4. 20 Giao diện thông tin khách hàng



Hình 4. 21 Giao diện cập nhật thông tin khách hàng

# **CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

## **5.1 Kết luận**

Dự án xây dựng website bán điện thoại đã đạt được những kết quả tích cực và mang lại những đóng góp quan trọng trong quá trình triển khai và phát triển. Em đã thành công trong việc tạo ra một nền tảng thân thiện người dùng, giúp người tiêu dùng dễ dàng tìm kiếm và mua sắm sản phẩm.

Đối với những đóng góp mới, em đã áp dụng các phương pháp và công nghệ mới nhất để cải thiện trải nghiệm người dùng. Giao diện thân thiện, quản lý đơn hàng hiệu quả, và khả năng tương tác cao là những điểm nổi bật mà em nghĩ rằng mình đã đem đến.

## **5.2 Hạn chế**

Bên cạnh sự thành công thì hiện tại dự án cũng có một vài hạn chế:

* Hiện tại, dự án vẫn chưa có chức năng thanh toán online và bình luận sản phẩm, sự thiếu xót này sẽ ảnh hưởng đến trải nghiệm mua sắm của người dùng, đây là thứ cần phải thực hiện để có thể tạo ra dự án hoàn chỉnh và tiện lợi.
* Giao diện vẫn chưa được được tối ưu và chưa được chăm sóc kĩ càng, điều này sẽ làm giảm giá trị thẩm mỹ của website.
* Những hạn chế trên cần được xử lý để tạo ra một trang web bán điện thoại hoàn chỉnh, thuận tiện và hấp dẫn đối với người dùng. Cải thiện được những hạn chế này sẽ nâng cao chất lượng của trang web, sẽ góp phần tăng cường trải nghiệm mua sắm trực tuyến của người dùng.

## **5.3 Hướng phát triển**

Để nâng cao chất lượng và khả năng cạnh tranh, em nghĩ rằng mình cần một số hướng phát triển tiếp theo như:

* Tối ưu hóa trải nghiệm người dùng: tăng cường phát triển, mở rộng thêm các chức năng mới để tạo ra trải nghiệm mua sắm trực tuyến dễ dàng và thuận lợi hơn.
* Mở rộng danh mục sản phẩm: thêm vào danh mục sản phẩm những loại mới, thêm chức năng thanh toán trực tuyến, bình luận và đánh giá sản phẩm để đáp ứng nhu cầu của khách hàng.
* Cải thiện những hạn chế của website: tìm hiểu và cải thiện những khuyết điểm trên của website.
* Học hỏi và tìm hiểu: tăng cường sự hiểu biết của bản thân từ việc học thêm những kiến thức cần thiết trong việc xây dựng và phát triển website.
* Những hướng phát triển này sẽ giúp em không chỉ phát triển năng lực bản thân mà còn giúp em có thể nhận ra được sự cạnh tranh khốc liệt trong thời gian tới.

Em cam kết tiếp tục nỗ lực và cải thiện để mang đến trải nghiệm mua sắm tốt nhất cho khách hàng trong thời gian tới.

## **DANH MỤC TÀI LIỆU THAM KHẢO**

1. Accomazzo, A., Murray, N., Lerner, C., & Zepernick, T. (2017). *Fullstack React: The Complete Guide to ReactJS and Friends*. Fullstack.io.
2. Banks, A., & Porcello, E. (2020). *Learning React: Functional Web Development with React and Redux* (2nd ed.). O’Reilly Media.
3. Greg Lim (2019). *Beginning Node.js, Express & MongoDB Development*. Apress.
4. Leonard Richardson, Mike Amundsen & Sam Ruby (2013). What Is REST?, Your First RESTful Service. *RESTful Web APIs,* O'Reilly Media, Sebastopol, California.
5. JJ Geewax (2021). Introduction to APIs. *API Design Patterns,* Shelter Island, New York.
6. Mario Casciaro & Luciano Mammino (2020). *Node.js Design Patterns* (3rd ed.).
7. Mardan, A. (2017). *React Quickly: Painless web apps with React, JSX, Redux, and GraphQL*. Manning Publications.