**BỘ THÔNG TIN VÀ TRUYỀN THÔNG**

TP. HCM

2023

BÙI TÁ TÂN NGỌC MSSV: N19DCCN122 CHUYÊN NGÀNH: CÔNG NGHỆ PHẦN MỀM 2019-2024 Lớp: D19CQCNPM01-N 2019-2024

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**



**ĐỒ ÁN**

**TỐT NGHIỆP ĐẠI HỌC**

**Đề tài: “Xây dựng web application**

**thi trắc nghiệm”**

**Người hướng dẫn : TH.S NGUYỄN TRUNG HIẾU**

**Sinh viên thực hiện : BÙI TÁ TÂN NGỌC**

**Mã số sinh viên : N19DCCN122**

**Lớp : D19CQCNPM01-N**

**Khóa : 2019 – 2024**

**Hệ : Đại học chính quy**

**TP.HỒ CHÍ MINH, NĂM 2023**

**BỘ THÔNG TIN VÀ TRUYỀN THÔNG**

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**



**ĐỒ ÁN**

**TỐT NGHIỆP ĐẠI HỌC**

**Đề tài: “Xây dựng web application**

**thi trắc nghiệm”**

**Người hướng dẫn : TH.S NGUYỄN TRUNG HIẾU**

**Sinh viên thực hiện : BÙI TÁ TÂN NGỌC**

**Mã số sinh viên : N19DCCN122**

**Lớp : D19CQCNPM01-N**

**Khóa : 2019 – 2024**

**Hệ : Đại học chính quy**

**TP.HỒ CHÍ MINH, NĂM 2023**

# LỜI CẢM ƠN

Em xin được gửi lời cảm ơn chân thành nhất đến các thầy cô trong Khoa Công nghệ Thông tin 2 của Học viện Công nghệ Bưu chính Viễn thông Cơ sở tại Thành phố Hồ Chí Minh đã tạo mọi điều kiện thuận lợi để em có thể hoàn thành đồ án tốt nghiệp một cách tốt nhất. Em xin gửi lời cảm ơn chân thành nhất đến thầy phụ trách hướng dẫn trực tiếp là Thạc sĩ Nguyễn Trung Hiếu. Người đã luôn theo dõi, tích cực hỗ trợ, đưa ra những lời khuyên giá trị trong suốt quá trình thực hiện đề em có thể hoàn thành đồ án thực tập tốt nghiệp này thành công.

Vì thời gian thực hiện đồ án có hạn, phải nghiên cứu một số công nghệ mới trong thời gian ngắn nên sản phẩm sẽ khó tránh khỏi những thiếu sót, những điểm chưa hoàn chỉnh. Rất mong quý thầy cô trong khoa có thể đưa ra nhận xét, đóng góp ý kiến, phản hồi cũng như cung cấp thêm những hướng dẫn cụ thể về mặt chuyên môn để em có thể thực hiện được nội dung được hoàn chỉnh nhất cho quá trình thực tập tốt nghiệp. Đồng thời học hỏi thêm được nhiều kinh nghiệm quý báu để chuẩn bị cho đồ án tốt nghiệp hoặc các dự án em sẽ tham gia trong tương lai.

Cuối cùng, em xin chúc quý thầy cô dồi dào sức khỏe và công tác thật tốt. Em xin chân thành cảm ơn!

Thành phố Hồ Chí Minh, tháng 12/2023

Sinh viên thực hiện

Bùi Tá Tân Ngọc

# MỤC LỤC

[LỜI CẢM ƠN i](#_Toc142869380)

[MỤC LỤC ii](#_Toc142869381)

[DANH MỤC CÁC BẢNG, SƠ ĐỒ, HÌNH iii](#_Toc142869382)

[KÝ HIỆU CÁC CỤM TỪ VIẾT TẮT vi](#_Toc142869383)

[MỞ ĐẦU 1](#_Toc142869384)

[CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI 2](#_Toc142869385)

[1.1. Tổng quan. 2](#_Toc142869386)

[1.2. Tìm hiểu “Xây dựng website kinh doanh mặt hàng điện thoại di động”. 2](#_Toc142869389)

[**1.2.1. Lý thuyết. 2**](#_Toc142869390)

[**1.2.2. Thực hành. 2**](#_Toc142869391)

[1.3. Công cụ sử dụng. 3](#_Toc142869392)

[1.4. Mục tiêu đề tài. 3](#_Toc142869393)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ 4](#_Toc142869394)

[2.1. Cơ sở lý thuyết. 4](#_Toc142869395)

[2.2. Giới thiệu về .NET Core. 4](#_Toc142869396)

[**2.2.1. ASP.NET Core. 4**](#_Toc142869397)

[**2.2.2. Entity framework core. 4**](#_Toc142869398)

[**2.2.3. Identity. 4**](#_Toc142869399)

[2.3. Giới thiệu về Angular. 5](#_Toc142869400)

[**2.3.1. Angular framework. 5**](#_Toc142869401)

[**2.3.2. Thư viện Angular material. 5**](#_Toc142869402)

[2.3. Microsoft SQL Server. 5](#_Toc142869403)

[2.4. RESTful API. 5](#_Toc142869404)

[CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG 6](#_Toc142869405)

[3.1. Mô hình hóa yêu cầu. 6](#_Toc142869406)

[**3.1.1. Tìm hiểu quy trình nghiệp vụ. 6**](#_Toc142869407)

[**3.1.2. Xác định yêu cầu chức năng. 7**](#_Toc142869408)

[**3.1.3. Các tác nhân của hệ thống. 9**](#_Toc142869409)

[3.2. Sơ đồ usecase. 9](#_Toc142869410)

[**3.2.1. Danh sách các usecase: 9**](#_Toc142869411)

[**3.2.2. Sơ đồ usecase tổng quát. 10**](#_Toc142869412)

[**3.2.3. Đặc tả usecase. 11**](#_Toc142869413)

[3.3. Mô hình thực thể ERD. 23](#_Toc142869414)

[**3.3.1. Xác định các thực thể. 23**](#_Toc142869415)

[**3.3.2. Mô hình thực thể ERD (Entity RelationShip Diagram). 24**](#_Toc142869416)

[3.4. Thiết kế cơ sở dữ liệu. 24](#_Toc142869417)

[**3.4.1. Mô hình diagram của cơ sở dữ liệu. 24**](#_Toc142869418)

[**3.4.2. Từ điển dữ liệu. 25**](#_Toc142869419)

[CHƯƠNG 4: XÂY DỰNG CHƯƠNG TRÌNH 33](#_Toc142869420)

[4.1. Lập trình xây dựng ứng dụng. 33](#_Toc142869421)

[4.2. Xây dựng chương trình phía back-end. 33](#_Toc142869422)

[4.3. Xây dựng chương trình phía front-end. 37](#_Toc142869423)

[CHƯƠNG 5: MINH HỌA SẢN PHẨM 38](#_Toc142869424)

[5.1. Giao diện bán hàng. 38](#_Toc142869425)

[5.2. Giao diện trang quản lý. 42](#_Toc142869436)

[CHƯƠNG 6: KẾT LUẬN 46](#_Toc142869443)

[6.1. Kết luận 46](#_Toc142869444)

[**6.1.1. Những kết quả đạt được 46**](#_Toc142869445)

[**6.1.2. Những hạn chế 47**](#_Toc142869446)

[6.2. Hướng phát triển của sản phẩm 47](#_Toc142869447)

[PHỤ LỤC 48](#_Toc142869448)

[DANH MỤC TÀI LIỆU THAM KHẢO 49](#_Toc142869449)

# DANH MỤC CÁC BẢNG, SƠ ĐỒ, HÌNH

[Bảng 3.1. Yêu cầu chức năng nghiệp vụ đối với admin/staff 7](#_Toc142870241)

[Bảng 3.2. Yêu cầu chức năng nghiệp vụ của khách hàng 8](#_Toc142870242)

[Bảng 3.3. Bảng danh sách các tác nhân của hệ thống 9](#_Toc142870243)

[Bảng 3.4. Bảng mô tả danh sách usecase của hệ thống. 9](#_Toc142870244)

[Bảng 3.5. Bảng mô tả usecase đăng ký. 11](#_Toc142870245)

[Bảng 3.6. Bảng mô tả usecase đăng nhập. 12](#_Toc142870246)

[Bảng 3.7. Bảng mô tả usecase quên mật khẩu. 13](#_Toc142870247)

[Bảng 3.8. Bảng mô tả usecase quản lý thông tin cá nhân. 14](#_Toc142870248)

[Bảng 3.9. Bảng mô tả usecase tìm kiếm sản phẩm. 15](#_Toc142870249)

[Bảng 3.10. Bảng mô tả usecase xem chi tiết sản phẩm. 16](#_Toc142870250)

[Bảng 3.11. Bảng mô tả usecase mua hàng. 17](#_Toc142870251)

[Bảng 3.12. Bảng mô tả usecase xem lịch sử mua hàng. 18](#_Toc142870252)

[Bảng 3.13. Bảng mô tả usecase hủy đơn hàng. 19](#_Toc142870253)

[Bảng 3.14. Bảng mô tả usecase quản lý nhân viên. 20](#_Toc142870254)

[Bảng 3.15. Bảng mô tả usecase quản lý sản phẩm. 21](#_Toc142870255)

[Bảng 3.16. Bảng mô tả usecase quản lý kho*.* 22](#_Toc142870256)

[Bảng 3.17. Bảng mô tả usecase quản lý tài khoản khách hàng. 23](#_Toc142870257)

[Bảng 3.18. Mô tả các thuộc tính trong bảng User 25](#_Toc142870258)

[Bảng 3.19. Mô tả các thuộc tính trong bảng Role 25](#_Toc142870259)

[Bảng 3.20. Mô tả các thuộc tính trong bảng RoleUser 26](#_Toc142870260)

[Bảng 3.21. Mô tả các thuộc tính trong bảng Brand 26](#_Toc142870261)

[Bảng 3.22. Mô tả các thuộc tính trong bảng Address 26](#_Toc142870262)

[Bảng 3.23. Mô tả các thuộc tính trong bảng Cart 27](#_Toc142870263)

[Bảng 3.24. Mô tả các thuộc tính trong bảng Homelet 27](#_Toc142870264)

[Bảng 3.25. Mô tả các thuộc tính trong bảng District 27](#_Toc142870265)

[Bảng 3.26. Mô tả các thuộc tính trong bảng Province 28](#_Toc142870266)

[Bảng 3.27. Mô tả các thuộc tính trong bảng Phone 28](#_Toc142870267)

[Bảng 3.28. Mô tả các thuộc tính trong bảng Order 29](#_Toc142870268)

[Bảng 3.29. Mô tả các thuộc tính trong bảng OrderDetail 29](#_Toc142870269)

[Bảng 3.30. Mô tả các thuộc tính trong bảng OrderStatus 30](#_Toc142870270)

[Bảng 3.31. Mô tả các thuộc tính trong bảng PhoneImage 30](#_Toc142870271)

[Bảng 3.32. Mô tả các thuộc tính trong bảng Promote 31](#_Toc142870272)

[Bảng 3.33. Mô tả các thuộc tính trong bảng PromoteDetail 31](#_Toc142870273)

[Bảng 3.34. Mô tả các thuộc tính trong bảng Status 31](#_Toc142870274)

[Bảng 3.35. Mô tả các thuộc tính trong bảng Supplier 31](#_Toc142870275)

[Bảng 3.36. Mô tả các thuộc tính trong bảng WarehouseTicket 32](#_Toc142870276)

[Bảng 3.37. Mô tả các thuộc tính trong bảng TicketDetail 32](#_Toc142870277)

[Hình 3.1. Sơ đồ diagram. 24](#_Toc142870435)

[Hình 3.2. Mô hình diagram của cơ sở dữ liệu 25](#_Toc142870436)

[Hình 4.1. Mô tả lớp AppDbContext. 33](#_Toc142870437)

[Hình 4.2. Mô tả lớp AppDbContext. 34](#_Toc142870438)

[Hình 4.3. Mô tả lớp Address. 34](#_Toc142870439)

[Hình 4.4. Mô tả interface IAddressRepository. 35](#_Toc142870440)

[Hình 4.5. Mô tả class AddressRepository. 35](#_Toc142870441)

[Hình 4.6. Mô tả dependency injection inject service. 36](#_Toc142870442)

[Hình 4.7. Mô tả dependency injection, inject service vào lớp cụ thể. 36](#_Toc142870443)

[Hình 4.8. Mô tả dependency injection, sử dụng service ở lớp cụ thể. 37](#_Toc142870444)

[Hình 5.1. Giao diện đăng ký khách hàng. 38](#_Toc142870445)

[Hình 5.2. Giao diện khách hàng đăng nhập. 38](#_Toc142870446)

[Hình 5.3. Giao diện quên mật khẩu. 39](#_Toc142870447)

[Hình 5.4. Giao diện trang chủ. 39](#_Toc142870448)

[Hình 5.5. Giao diện trang sản phẩm. 40](#_Toc142870449)

[Hình 5.6. Giao diện trang chi tiết sản phẩm. 40](#_Toc142870450)

[Hình 5.7. Giao diện giỏ hàng. 41](#_Toc142870451)

[Hình 5.8. Giao diện trang mua hàng. 41](#_Toc142870452)

[Hình 5.9. Giao diện quản lý đơn mua. 42](#_Toc142870453)

[Hình 5.10. Giao diện quản lý thông tin cá nhân. 42](#_Toc142870454)

[Hình 5.11. Giao diện trang tổng quan. 43](#_Toc142870455)

[Hình 5.12. Giao diện trang quản lý sản phẩm. 43](#_Toc142870456)

[Hình 5.13. Giao diện trang quản lý kho. 44](#_Toc142870457)

[Hình 5.14. Giao diện trang quản lý đơn hàng. 44](#_Toc142870458)

[Hình 5.15. Giao diện trang quản lý nhân viên. 45](#_Toc142870459)

[Hình 5.16. Giao diện trang thống kê. 45](#_Toc142870460)

# KÝ HIỆU CÁC CỤM TỪ VIẾT TẮT

|  |  |
| --- | --- |
| API: Application programming interface | Giao diện lập trình ứng dụng |
| Website | Trang web |
| MSSQL Server | Microsoft SQL Server |
| Interface | Giao diện |
| Class | Lớp |
| SPA | Single page application |

# MỞ ĐẦU

Hiện nay, với sự phát triển một cách bùng nổ của công nghệ, các hoạt động thường ngày của chúng ta cũng đã dần chuyển dịch sang hình thức trực tuyến tích hợp với các công nghệ tiên tiến.

Thiết bị thông minh có thể nói là phổ biến nhất hiện nay có thể nói đến chính là điện thoại thông minh. Hầu như mỗi người trưởng thành hiện nay đều sở hữu ít nhất một chiếc điện thoại thông minh. Nó phục vụ chúng ta từ công việc đến giải trí, .... Chính vì sự đa năng này mà điện thoại thông minh chính là thứ không thể thiếu trong cuộc sống hiện đại ngày nay. Thị trường điện thoại cũng từ đấy mà phát triển, cạnh tranh để tạo ra những sản phẩm với tính năng ngày một hiện đại và giá cả ngày càng cạnh tranh hơn.

Chính vì nhu cầu thị trường lớn nên thị trường mua bán điện thoại cũng đang ngày một phát triển và hình thức bán điện thoại cũng được chuyển dịch rõ rệt. Các nền tảng mua bán trực tuyến dần trở thành sự lựa chọn yêu thích của người tiêu dùng vì sự tiện lợi, đơn giản của nó. Người dùng có thể ngồi tại nhà đặt hàng và sản phẩm sẽ được giao đến tận nơi mà họ sinh sống.

Đề tài “Website bán điện thoại trực tuyến” được xây dựng để có thể bắt kịp xu thế kinh doanh trực tuyến đang phát triển bùng nổ hiện nay.

# CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI

## 1.1. Tổng quan.

### 1.1.1. Tổng quan về ứng dụng.

Web application thi trắc nghiệm là một ứng dụng cho phép người dùng có thể tham gia làm các bài thi trắc nghiệm trực tuyến kinh doanh mặt hàng điện thoại di động là một ứng dụng cho giúp cho khách hàng có thể dễ dàng tiếp cận và tham khảo, tìm mua được chiếc điện thoại mà mình yêu thích, giúp việc mua bán trở nên thuận tiện và thân thiện hơn. Website cũng giúp cho những người điều hành việc kinh doanh có thể dễ dàng quản lý, kiểm soát thật tốt tình hình kinh doanh của “cửa hàng điện thoại trực tuyến” của mình.

### 1.1.2. Khảo sát yêu cầu nghiệp vụ.

* Qua quá trình tìm hiểu và khảo sát thông tin về nghiệp vụ kinh doanh và quản lý của một website kinh doanh mặt hàng điện thoại di động trực tuyến có thể thấy một số yêu cầu nghiệp vụ chính của website như sau:
  + Nghiệp vụ mua hàng:
    - Khách hàng tìm kiếm sản phẩm yêu thích.
    - Quá trình thanh toán.
  + Nghiệp vụ quản lý sản phẩm:
    - Quản lý sản phẩm điện thoại trong kho.
    - Quản lý thông tin của những chiếc điện thoại.
  + Nghiệp vụ quản trị:
    - Quản lý nhân viên của website.

## 1.2. Tìm hiểu “Xây dựng website kinh doanh mặt hàng điện thoại di động”.

### 1.2.1. Lý thuyết.

* Tìm hiểu nghiệp vụ liên quan đến website bán hàng (giỏ hàng, đặt hàng, ...).
* Tìm hiểu mô hình client server.
* Front-end: Angular, angular material.
* Back-end: ASP.NET core, Entity framework core.
* Tìm hiểu xác thực với JWT kết hợp Identity.
* Tìm hiểu dịch vụ gửi mail, upload ảnh lên cloudinary.

### 1.2.2. Thực hành.

Tính năng chung:

* + Đăng ký, đăng nhập, quên mật khẩu, quản lý thông tin cá nhân.

Module quản trị:

* + Được sử dụng chính bởi người chủ của website.
  + Dùng để quản lý tình hình chung của cửa hàng và các nhân viên của cửa hàng.

Module quản lý sản phẩm:

* + Được sử dụng chính bởi các nhân viên của website.
  + Quản lý thông tin về những chiếc điện thoại đang được bán trong cửa hàng.
  + Quản lý về số lượng của các điện thoại hiện đang còn lại.
  + Quản lý thông tin về các đợt khuyến mãi dựa trên sự chấp thuận của cấp trên.
  + Quản lý trạng thái của các đơn hàng, xác nhận đơn hàng cũng như việc thanh toán.
* Module bán hàng:
  + Được các khách hàng sử dụng để tìm mua các chiếc điện thoại mà mình yêu thích.
  + Khách hàng có thể quản lý thông tin cá nhân, thông tin địa chỉ nhận hàng của mình và thông tin về đơn hàng mà mình đã đặt.

## 1.3. Công cụ sử dụng.

* Visual studio.
* Android studio.
* Postman.

## 1.4. Mục tiêu đề tài.

Đảm bảo hoạt động của cửa hàng diễn ra nhanh chóng, chính xác:

* Giúp người quản lý và nhân viên có thể quản lý hoạt động kinh doanh của cửa hàng một các thuận tiện.
* Giúp khách hàng có được trải nghiệm mua hàng thật nhanh chóng và đơn giản.

# CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ

## 2.1. Cơ sở lý thuyết.

## 2.2. Giới thiệu về NodeJs và ExpressJs.

### 2.2.1. Node.js.

Node.js là một một trường thực thi javascript đa nền tảng mã nguồn mở được xây dựng trên V8 Javascript engine, Node.js giúp xậy dựng các ứng dụng web một cách nhanh chóng và hiệu quả. Các ứng dụng được viết ở đồ án này sử dụng các công nghệ như ExpressJs, React, React native đều chạy trên nền Node.js.

Các ứng dụng Node.js chạy đơn luồng, tức là không tạo thêm luồng cho các request mới. Thay vào đó Node.js có cơ chế xử lý bất đồng bộ giúp có thể xử lý nhiều request đồng thời một cách hiệu quả. Khi thực hiện một thao tác, thay vì chặn luồng thực hiện và lãng phí CPU thì Node.js sẽ tiếp tục các hoạt động khác trong lúc chờ phản hồi. Điều này giúp Node.js có thể xử lý hàng nghìn kết nối đồng thời mà không gây ra gánh nặng quản lý luồng đồng thời, điều mà có thể tiềm ẩn nhiều rủi ro đáng kể.

Vì có cả các framework, thư viện front-end và back-end đều chạy trên Node.js mà các nhà phát triển front-end có thể dễ dàng học thêm các kiến thức mới về back-end và ngược lại mà không gặp trở ngại vì sự thay đổi ngôn ngữ lập trình.

### 2.2.2. Express.js.

Express.js là một framework phía máy chủ (server-side) được viết bằng javascript và được xây dựng trên cơ sở Node.js. Nó được sử dụng để xây dựng các ứng dụng web và cả Restful API.

Với cú pháp đơn giản và dễ tiếp cận, Express.js cho phép các nhà phát triển xây dựng nhanh chóng các ứng dụng web phức tạp, từ các trang web tĩnh đơn giản đến các ứng dụng đa người dùng phức tạp.

Expressjs cho phép bạn xây dựng các ứng dụng backend như các trang web động, các ứng dụng web đơn trang (Single-Page Applications - SPAs), và các API (Application Programming Interface).

Với Express.js, bạn có thể xử lý các yêu cầu HTTP, quản lý định tuyến (routing), xử lý middleware, và tạo các template động để phục vụ dữ liệu cho các trình duyệt hoặc các ứng dụng di động. Nó cung cấp các phương thức và hàm tiện ích để xử lý các yêu cầu GET, POST, PUT, DELETE và các phương thức HTTP khác.

Ngoài ra, Express.js cũng hỗ trợ việc tạo và quản lý các phiên làm việc (sessions), xử lý các lỗi (error handling), và tích hợp với các cơ sở dữ liệu (database) như MongoDB hoặc MySQL để lưu trữ và truy vấn dữ liệu. Với sự linh hoạt và khả năng mở rộng của nó, Express.js đã trở thành một trong những lựa chọn phổ biến cho việc phát triển phía máy chủ trong cộng đồng JavaScript.

### 2.2.3. Sequeqlize.

Identity là một API cung cấp các tính năng hỗ trợ cho việc xác thực, phân quyền trong ASP.NET core.

Identity cũng có thể được tùy chỉnh để tích hợp với các công nghệ xác thực như JWT để hỗ trợ việc xác thực người dùng thông qua json web token, ngoài ra identity còn hỗ trợ đăng nhập bằng nền tảng thứ hai như thông qua tài khoản Facebook, Google, Twitter, ...

Để tài lần này sẽ sử dụng kết hợp Identity và Json web token cho việc xác thực và phân quyền người dùng.

## 2.3. Giới thiệu về React.

React là một thư viện javascript mạnh mẽ được Facebook phát triển. React cho phép các developer có thể sử dụng để tạo ra những ứng dụng phía máy khách một cách nhanh chóng và hiệu quả.

React được sử dụng để xây dựng các ứng dụng web theo hướng “SPA (single page application)”, SPA là một kiểu thiết kế website mà trang tại đấy các thông tin của ứng dụng web chỉ tải một lần duy nhất lúc ban đầu. Sau đó ứng dụng sẽ cập nhật lại những phần thay đổi nếu có yêu cầu mới từ người dùng mà không tải lại toàn bộ ứng dụng như các cách thiết kế trước đây. Điều này giúp ứng dụng xây dựng theo kiểu SPA trở nên thân thiện với người sử dụng hơn.

Một ứng dụng React được hợp thành từ những đơn vị thành phần (component). Chúng sử dụng jsx (Javascript XML), một cú pháp mở rộng cho phép nhúng code jsvascript vào file HTML một cách dễ dàng. Các component sẽ kết nối với nhau tạo nên một ứng dụng web. Khi có những yêu cầu thay đổi, ứng dụng sẽ chỉ thay đổi những thứ cần thay đổi, từ đấy tăng tốc đáng kể hiệu năng của ứng dụng. Đây chính là những ưu điểm chính mà chúng ta nên chọn sử dụng React.

## 2.3. React native.

Microsoft SQL Server là một trong các hệ quản trị cơ sở dữ liệu quan hệ phổ biến nhất trên thế giới hiện nay được phát triển bởi Microsoft. Cung cấp các giải pháp quản lý dữ liệu cho các ứng dụng dưới dạng dữ liệu quan hệ.

Đề tài sẽ sử dụng SQL Server làm cơ sở dữ liệu chính để lưu trữ dữ liệu cho ứng dụng.

## 2.4. MySQL Server.

REST là viết tắt của Representational State Transfer. RESTful API là một tiêu chuẩn dùng trong việc thiết kế các API cho các ứng dụng web để quản lý các resource. RESTful là một trong những kiểu thiết kế API được sử dụng phổ biến ngày nay để cho các ứng dụng (web, mobile…) khác nhau giao tiếp với nhau.

Những tiêu chuẩn phổ biến nhất của REST như là quy định cách sử dụng các HTTP method (như GET, POST, PUT, DELETE…) và cách định dạng các URL cho ứng dụng web để quản các resource. RESTful không quy định logic code ứng dụng và không giới hạn bởi ngôn ngữ lập trình ứng dụng, bất kỳ ngôn ngữ hoặc framework nào cũng có thể sử dụng để thiết kế một RESTful API.

Website mua bán điện thoại sẽ xây dựng các API dựa trên tiêu chuẩn của RESTful API.

# CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## 3.1. Mô hình hóa yêu cầu.

### 3.1.1. Tìm hiểu quy trình nghiệp vụ.

Website kinh doanh mặt hàng điện thoại di động được sử dụng để quản lý việc kinh doanh của một cửa hàng bán điện thoại vừa và nhỏ. Đối tượng sử dụng của website là nhân viên và khách hàng.

* Đối với nhân viên, bao gồm có quản lý (admin) và các nhân viên của cửa hàng (staff):
  + Quản lý (admin): Là người quản lý có quyền lớn nhất, tham gia vào quản lý tình hình kinh doanh cửa hàng và quản lý các nhân viên của cửa hàng
  + Nhân viên (staff): Gồm nhiều nhân viên sử dụng phần mềm để thực hiện các công việc quản lý danh mục, sản phẩm, đơn hàng, ....
* Đối với khách hàng(customer):
  + Khách hàng (customer): Khách hàng có thể vào và xem thông tin về các sản phẩm hiện đang được bày bán trong cửa hàng. Để có thể mua hàng, khách hàng cần đăng ký tài khoản và thêm một số thông tin các nhân như địa chỉ giao hàng, ... để có thể thực hiện việc mua hàng. Sau khi mua hàng, khách hàng có thể theo dõi đơn hàng của mình, ngoài ra còn có thể xem lại lịch sử mua hàng và chỉnh sửa thông tin cá nhân.

Quy trình đặt hàng tại website được thực hiện như sau:

* **Bước 1:** Truy cập website và tìm kiếm sản phẩm mình muốn mua, khách hàng có thể tìm kiếm dễ dàng bằng bộ lọc cũng như thanh tìm kiếm.
* **Bước 2:** Click vào sản phẩm muốn mua, hệ thống sẽ chuyển đến màn hình chi tiết sản phẩm:

+ Nếu khách hàng mua hàng: Chọn số lượng và bấm nút “Thêm vào giỏ hàng” để thêm sản phẩm vào giỏ hàng.

+ Nếu khách hàng muốn xem giỏ hàng để đặt hàng hoặc cập nhật sản phẩm: Bấm vào giỏ hàng để xem các sản phẩm hiện đang có trong giỏ hàng.

+ Nếu khách hàng muốn đặt hàng và thanh toán cho sản phẩm: Chọn địa chỉ nhận hàng và phương thức thanh toán sau đấy bấm nút “Đặt hàng”.

Để có thể thêm sản phẩm vào giỏ hàng cũng như đặt hàng, khách hàng phải thực hiện đăng nhập trước, nếu chưa có tài khoản thì phải đăng ký.

* **Bước 3:** Nếu hình thức thanh toán là online, khách hàng cần điền thông tin về việc đã thanh toán (ảnh hóa đơn chuyển khoản).
* **Bước 4:** Sau khi nhận được đơn hàng, nhân viên sẽ liên hệ bằng cách gọi điện lại để xác nhận đơn hàng cũng như xác nhận việc thanh toán (nếu khách hàng chọn thanh toán online), sau đó sẽ thay đổi trạng thái của đơn hàng cũng như trạng thái thanh toán của đơn hàng.
* **Bước 5:** Khi đơn hàng chưa được xác nhận, khách hàng có thể hủy đơn hàng, trạng thái đơn hàng lúc này sẽ chuyển về “đã hủy”.

### 3.1.2. Xác định yêu cầu chức năng.

Bảng 3.. Yêu cầu chức năng nghiệp vụ đối với admin/staff

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TT** | **Công việc** | **Loại công việc** | **Quy định/Công thức liên quan** | **Biểu mẫu liên quan** |
| **Nhóm yêu cầu chức năng cho cả nhân viên và admin** | | | | |
| ***Quản lý thương hiệu sản phẩm*** | | | | |
| 1 | Thêm thương hiệu | Lưu trữ | Ràng buộc sửa đủ và đúng kiểu dữ liệu ràng buộc. Tên thương hiệu là duy nhất. |  |
| 2 | Sửa thương hiệu | Lưu trữ | Ràng buộc sửa đủ và đúng kiểu dữ liệu ràng buộc. Tên thương hiệu là duy nhất. |  |
| 3 | Xóa thương hiệu | Lưu trữ | Chỉ có thể xóa thương hiệu chưa có sản phẩm sử dụng |  |
| ***Quản lý sản phẩm*** | | | | |
| 4 | Thêm sản phẩm | Lưu trữ | Ràng buộc nhập đủ và đúng kiểu dữ liệu ràng buộc về số lượng, hình ảnh, giá cả. |  |
| 5 | Sửa sản phẩm | Lưu trữ | Ràng buộc sửa đủ và đúng kiểu dữ liệu ràng buộc về số lượng, hình ảnh, giá cả. |  |
| 6 | Xóa sản phẩm | Lưu trữ | Không xóa sản phẩm đã lập đơn hàng |  |
| 7 | Tìm kiếm sản phẩm | Tra cứu | Tìm kiếm theo tên sản phẩm / giá cả / tên thương hiệu / tính năng |  |
| ***Quản lý tài khoản khách hàng*** | | | | |
| 8 | Xem thông tin tài khoản khách hàng | Lưu trữ | Xem thông tin tài khoản khách hàng. |  |
| 9 | Khóa tài khoản khách hàng | Lưu trữ | Khi khách hàng vi phạm chính sách cửa hàng. |  |
| 10 | Tìm kiếm khách hàng | Tra cứu | Tìm kiếm khách hàng theo tên / số điện thoại / địa chỉ / email |  |
| ***Quản lý đơn hàng*** | | | | |
| 11 | Tìm kiếm và xem đơn hàng | Tra cứu | Tìm kiếm theo mã hóa đơn, trạng thái. |  |
| 12 | Cập nhật trạng thái đơn hàng | Lưu trữ | Chỉ có thể sửa trạng thái đơn hàng khi đơn hàng chưa bị hủy. |  |
| ***Nhóm chức năng đăng nhập*** | | | | |
| 13 | Đăng nhập trang thương mại điện tử | Lưu trữ | Tài khoản được cấp bởi admin |  |
| 14 | Đặt lại mật khẩu | Lưu trữ | Địa chỉ email để đặt lại mật khẩu phải là địa chỉ email dùng để đăng ký tài khoản. |  |
| 15 | Quản lý thông tin tài khoản | Lưu trữ | Cho phép thay đổi thông tin cá nhân. |  |
| ***Yêu cầu chức năng nghiệp vụ của admin*** | | | | |
| ***Quản lý tài khoản quản trị*** | | | | |
| 16 | Thêm mới tài khoản | Lưu trữ | Ràng buộc nhập đủ và đúng kiểu dữ liệu ràng buộc về tên đăng nhập, email, hình ảnh, số điện thoại, địa chỉ, mật khẩu. |  |
| 17 | Sửa tài khoản | Lưu trữ | Ràng buộc sửa đủ và đúng kiểu dữ liệu ràng buộc về tên đăng nhập, email, hình ảnh, số điện thoại, địa chỉ, mật khẩu. |  |
| 18 | Xóa tài khoản | Lưu trữ | Không thể xóa tài khoản đang đăng nhập |  |
| 19 | Tìm kiếm tài khoản | Tra cứu | Tìm kiếm tài khoản theo tên đăng nhập, email, số điện thoại, địa chỉ. |  |
| 20 | Cấp quyền cho tài khoản | Lưu trữ | Cấp quyền cho tài khoản |  |

Bảng 3.. Yêu cầu chức năng nghiệp vụ của khách hàng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TT** | **Công việc** | **Loại công việc** | **Quy định/ Công thức liên quan** | **Biểu mẫu liên quan** |
| **Yêu cầu chức năng nghiệp vụ cho khách hàng** | | | | |
| 1 | Tìm kiếm sản phẩm theo tên sản phẩm/danh mục sản phẩm | Tra cứu |  |  |
| 2 | Xem thông tin chi tiết sản phẩm | Tra cứu |  |  |
| 3 | Thêm sản phẩm vào giỏ hàng | Lưu trữ | Chỉ có thể thêm vào giỏ hàng sản phẩm còn kinh doanh |  |
| **Yêu cầu chức năng nghiệp vụ cho khách hàng thành viên** | | | | |
| 4 | Đặt mua hàng | Lưu trữ | Chỉ có thể đặt mua những sản phẩm còn kinh doanh |  |
| 5 | Hủy đơn hàng | Lưu trữ | Khi đơn hàng chưa được xác nhận. |  |
| 5 | Đăng nhập, đăng ký trang thương mại điện tử | Lưu trữ |  |  |
| 6 | Đặt lại mật khẩu | Lưu trữ | Địa chỉ email để đặt lại mật khẩu phải là địa chỉ email dùng để đăng ký tài khoản. |  |
| 7 | Quản lý thông tin cá nhân | Lưu trữ | Cho phép thay đổi thông tin cá nhân. |  |

### 3.1.3. Các tác nhân của hệ thống.

Bảng 3.. Bảng danh sách các tác nhân của hệ thống

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Tác nhân** | **Ý nghĩa** | **Mô tả** |
| 1 | Quản lý  (Admin) | Người chịu trách nhiệm quản lý điều hành hệ thống. | - Quản lý tài khoản.  - Quản lý phân quyền.  - Quản lý sản phẩm.  - Quản lý đơn hàng.  - Quản lý tài khoản khách hàng. |
| 2 | Nhân viên  (Staff) | Người chịu trách nhiệm quản lý và điều hành trực tiếp các hoạt động kinh doanh của hệ thống. | - Quản lý tài khoản khách hàng.  - Quản lý sản phẩm.  - Quản lý đơn hàng.  - Quản lý hóa đơn. |
| 4 | Khách hàng  (Customer) | Người sử dụng website bán hàng. Tài khoản đăng nhập là địa chỉ email của khách hàng. | - Tìm kiếm sản phẩm.  - Xem chi tiết sản phẩm.  - Thêm sản phẩm vào giỏ hàng.  - Đặt hàng.  - Theo dõi đơn hàng.  - Xem lịch sử mua hàng. |

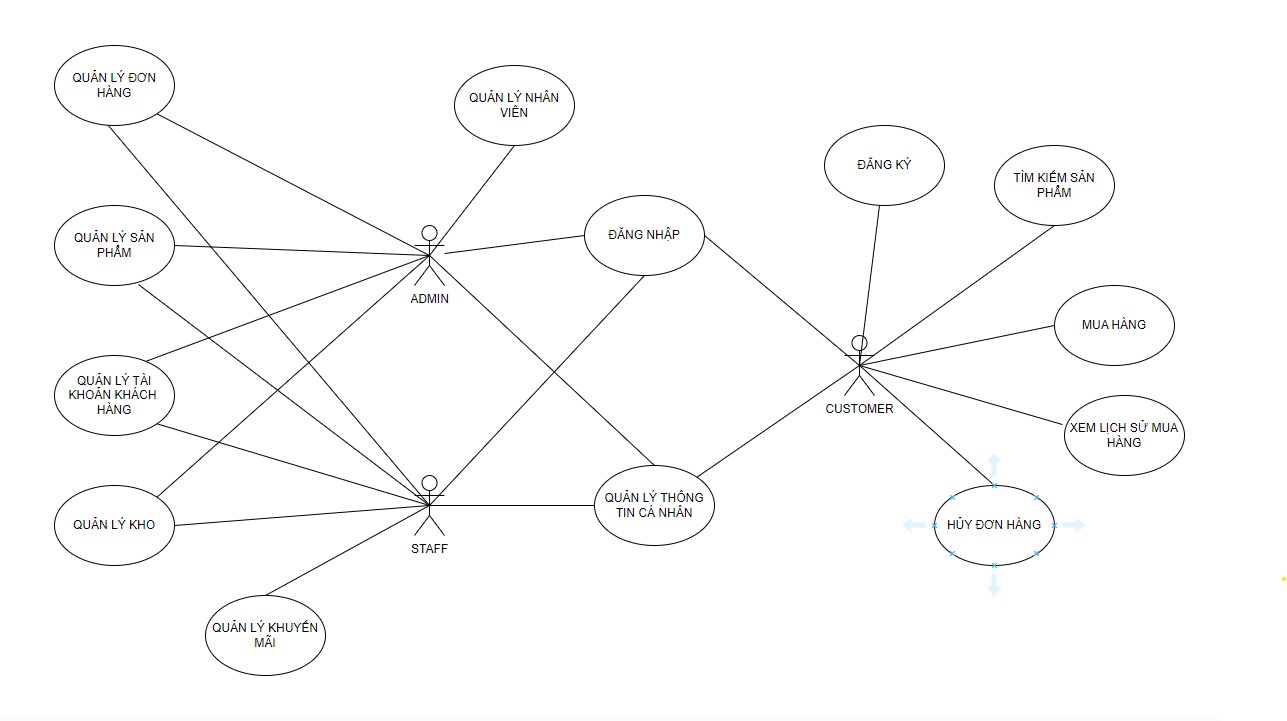
## 3.2. Sơ đồ usecase.

### 3.2.1. Danh sách các usecase:

Bảng 3.. Bảng mô tả danh sách usecase của hệ thống.

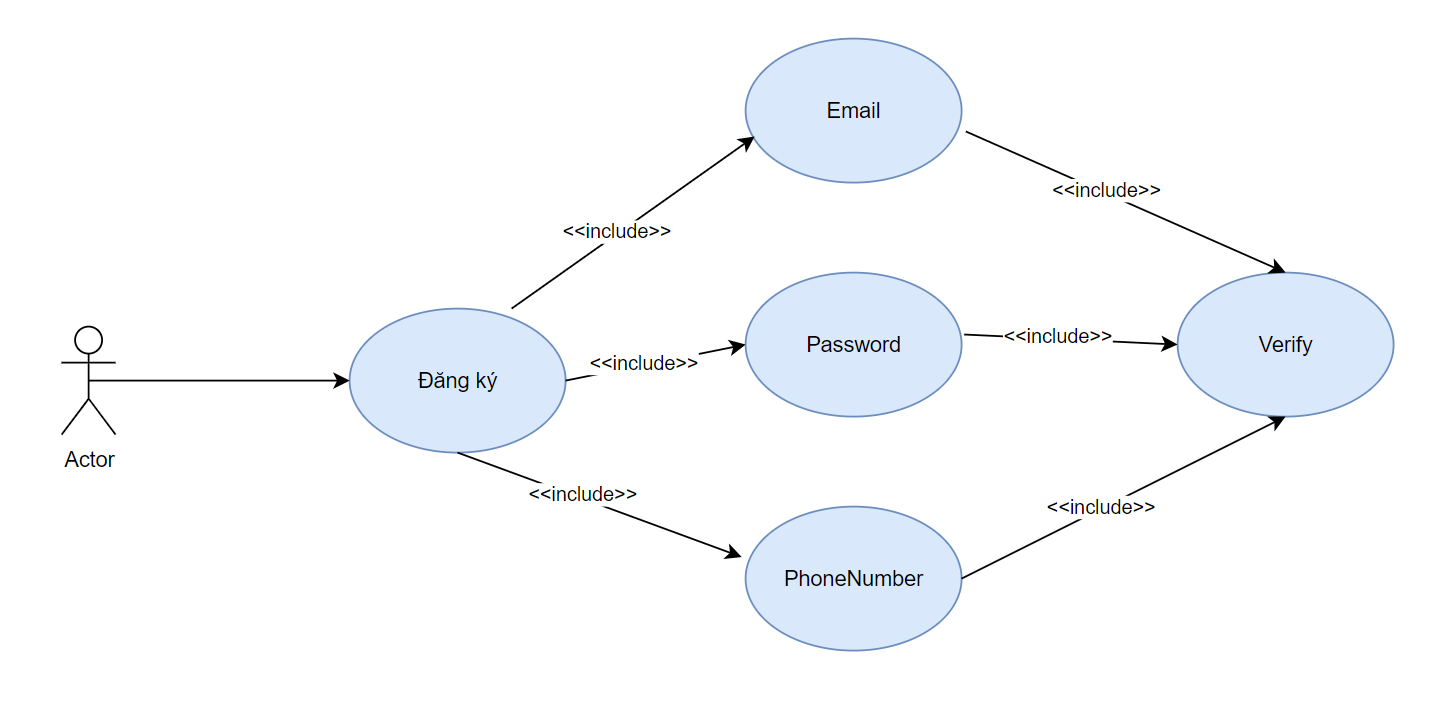
|  |  |  |
| --- | --- | --- |
| **TT** | **Tên Use case** | **Mô tả** |
| UC01 | Đăng nhập | Use case này mô tả các bước đăng nhập của các actor vào hệ thống trang quản trị. |
| UC02 | Đăng ký | Khách hàng sử dụng Use case “Đăng ký” để tạo tài khoản cho mình trên website. |
| UC03 | Quên mật khẩu | Use case này sử dụng khi actor quên mật khẩu, actor sẽ đổi lại mật khẩu mới bằng việc gửi email. |
| UC04 | Quản lý lịch sử mua hàng đơn hàng | Use case này cho phép khách hàng quản lý đơn hàng mình đã mua |
| UC05 | Quản lý tài khoản khách hàng | Use case này cho phép admin/nhân viên xem/xóa tài khoản khách hàng, |
| UC06 | Chỉnh sửa thông tin cá nhân | Use case này sử dụng khi actor muốn thay đổi các thông tin cá nhân của mình. |
| UC07 | Tìm kiếm sản phẩm | Use case này cho phép khách hàng tìm kiếm sản phẩm theo nhiều field. |
| UC08 | Xem chi tiết sản phẩm | Use case này cho phép khách hàng xem chi tiết sản phẩm. |
| UC10 | Hủy đơn hàng | Use case này cho phép khách hàng hủy đơn hàng đã đặt. |
| UC11 | Xem lịch sử mua hàng | Use case này cho phép admin/nhân viên xem/xóa tài khoản khách hàng, |
| UC12 | Quản lý sản phẩm | Use case này cho phép nhân viên/admin thêm/sửa/xóa/cập nhật số lượng của sản phẩm. |
| UC13 | Quản lý đơn hàng | Use case này cho phép nhân viên/admin xem/cập nhật trạng thái hóa đơn. |
| UC14 | Mua hàng | Usecase cho phép khách hàng mua hàng |

### 3.2.2. Sơ đồ usecase tổng quát.



### 3.2.3. Đặc tả usecase.

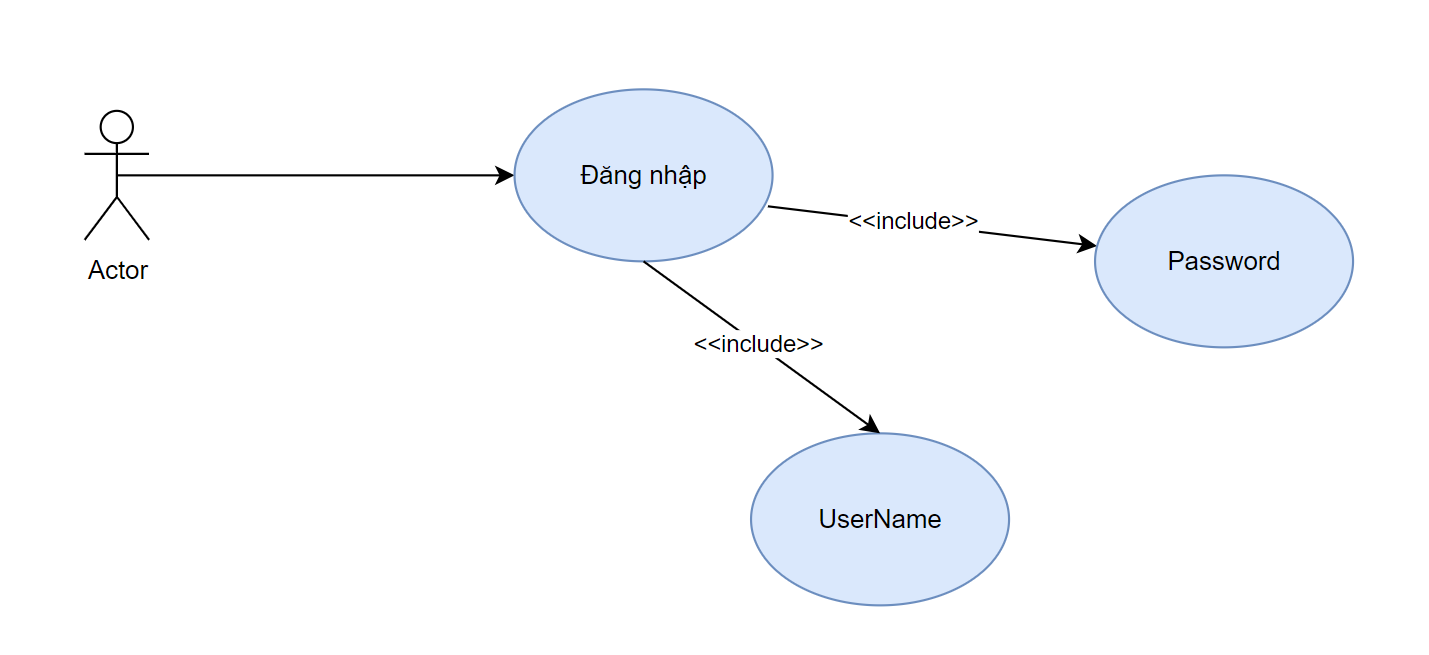
#### 3.2.3.1. Usecase “Đăng ký”



Bảng 3.. Bảng mô tả usecase đăng ký.

|  |  |
| --- | --- |
| **Tên use-case** | Đăng ký |
| **Tác nhân** | Khách hàng (customer) |
| **Tiền điều kiện** | Người dùng chưa có tài khoản. |
| **Hậu điều kiện** | Đăng ký tài khoản thành công |
| **Mô tả** | Đăng ký tài khoản mới |
| **Kịch bản:**  1. Click button Đăng ký ở trang đăng nhập.  2. Nhập các thông tin vào form đăng ký (username, password, ...).  3. Click Đăng ký  4. Hệ thống kiểm tra thông tin hợp lệ, đầy đủ hay chưa.  5.Hệ thống xử lý và trả về kết quả | |
| **Ngoại lệ**:  4.1. Các trường chưa được điền đầy đủ thông tin, không đúng định dạng  5.1. Địa chỉ email đã được sử dụng.  5.2. Địa chỉ email không tồn tại. | |

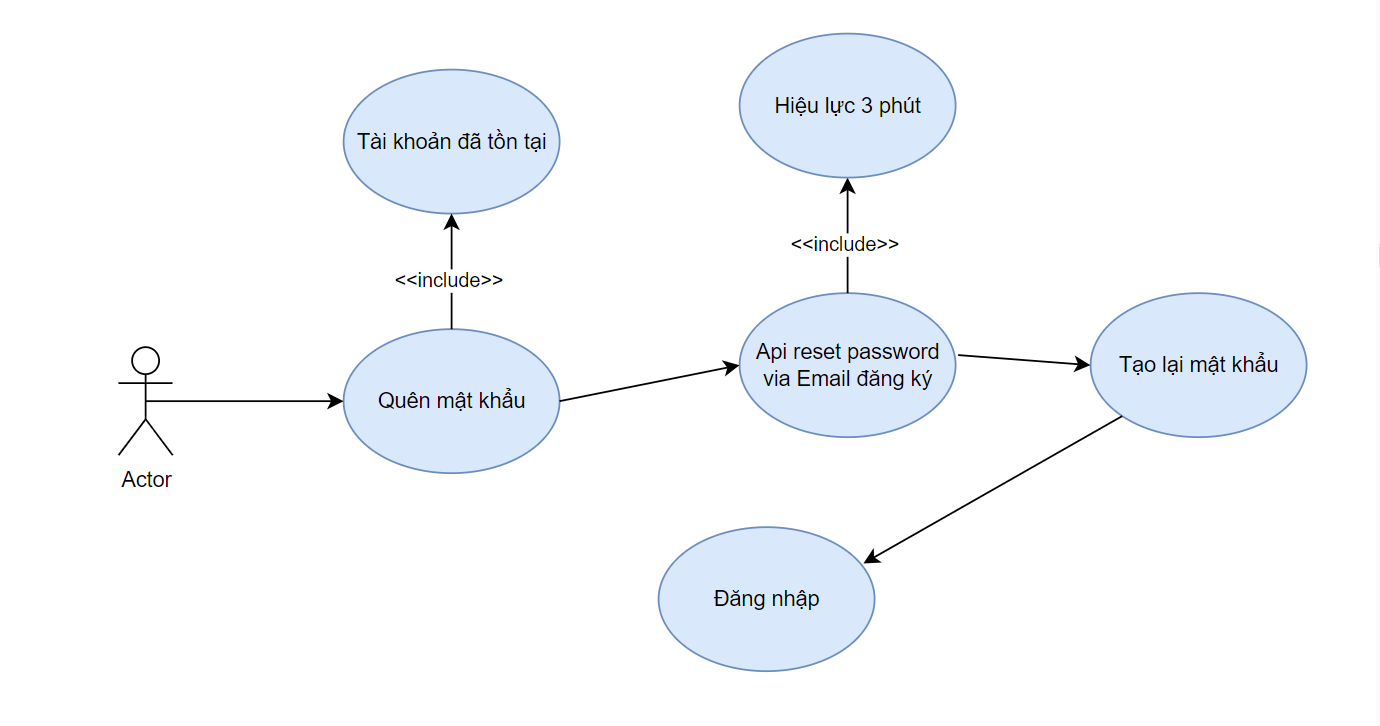
#### 3.2.3.2. Usecase “Đăng nhập”



Bảng 3.. Bảng mô tả usecase đăng nhập.

|  |  |
| --- | --- |
| **Tên use-case** | Đăng nhập |
| **Tác nhân** | User (admin, staff, customer) |
| **Tiền điều kiện** | Đã có tài khoản để truy cập hệ thống |
| **Hậu điều kiện** | Đăng nhập thành công. |
| **Mô tả** | Đăng nhập vào hệ thống. |
| **Kịch bản:**  1. Click button Đăng nhập ở trang đăng nhập.  2. Nhập các thông tin vào form đăng nhập (email, password).  3. Click Đăng nhập  4. Hệ thống kiểm tra thông tin hợp lệ, đầy đủ hay chưa.  5.Hệ thống xử lý và trả về kết quả | |
| **Ngoại lệ**:  4.1. Các trường chưa được điền đầy đủ thông tin, không đúng định dạng  5.1. Địa chỉ email đã được sử dụng.  5.2. Địa chỉ email không tồn tại. | |

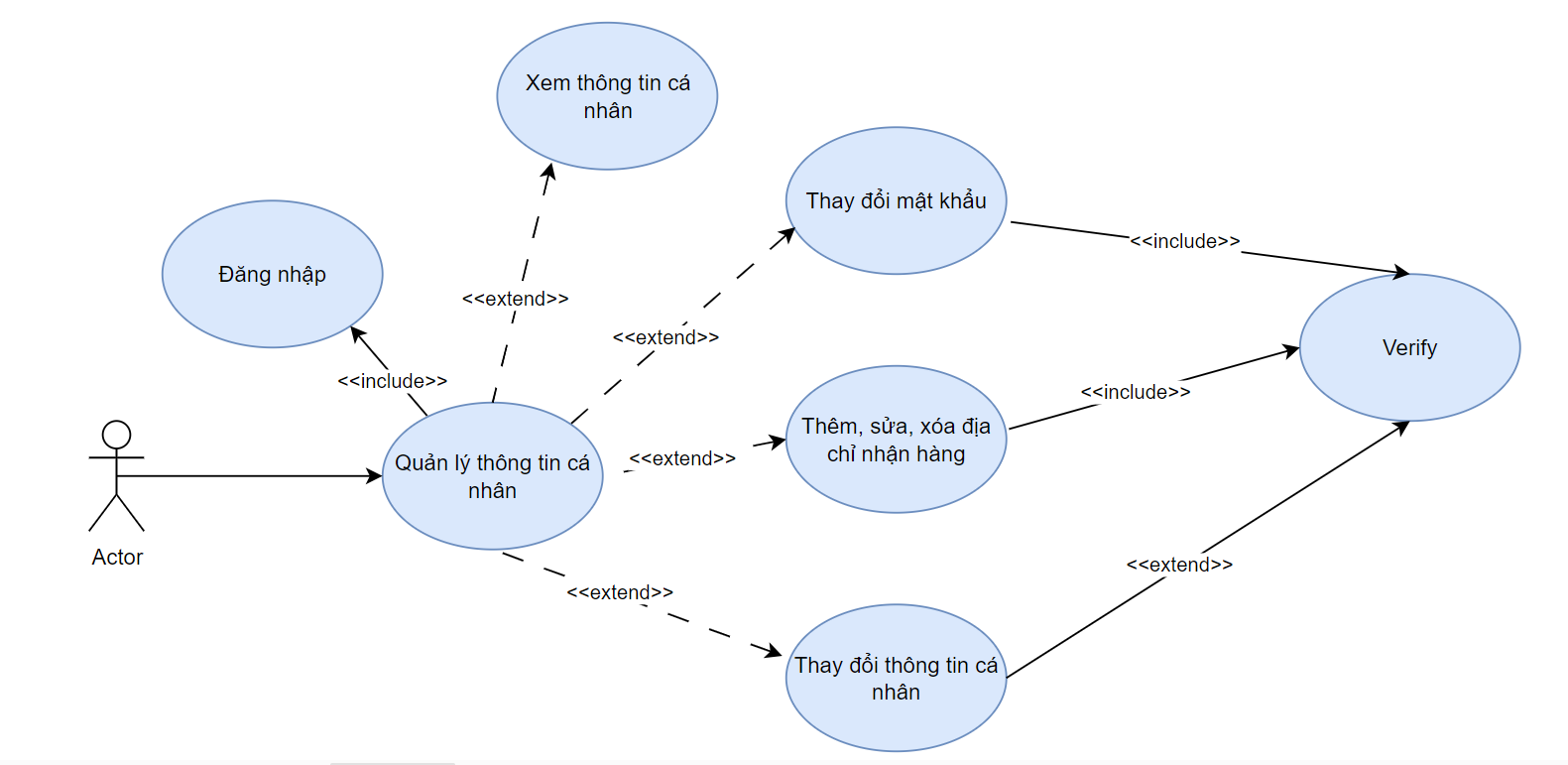
#### 3.2.3.3. Usecase “Quên mật khẩu”



Bảng 3.. Bảng mô tả usecase quên mật khẩu.

|  |  |
| --- | --- |
| **Tên use-case** | Quên mật khẩu |
| **Tác nhân** | User (staff, customer) |
| **Tiền điều kiện** | Đã có tài khoản để truy cập hệ thống |
| **Hậu điều kiện** | Reset mật khẩu thành công. |
| **Mô tả** | Tạo lại mật khẩu khi quên mật khẩu |
| **Kịch bản:**  1. Click button quên mật khẩu.  2. Nhập email đăng ký.  3. Hệ thống kiểm tra và gửi mail đặt lại mật khẩu mới.  4. User truy cập và đặt lại mật khẩu.  5.Hệ thống xử lý và trả về kết quả | |
| **Ngoại lệ**:  2.1 Email không tồn tại.  3.1. Mật khẩu mới không đúng định dạng. | |

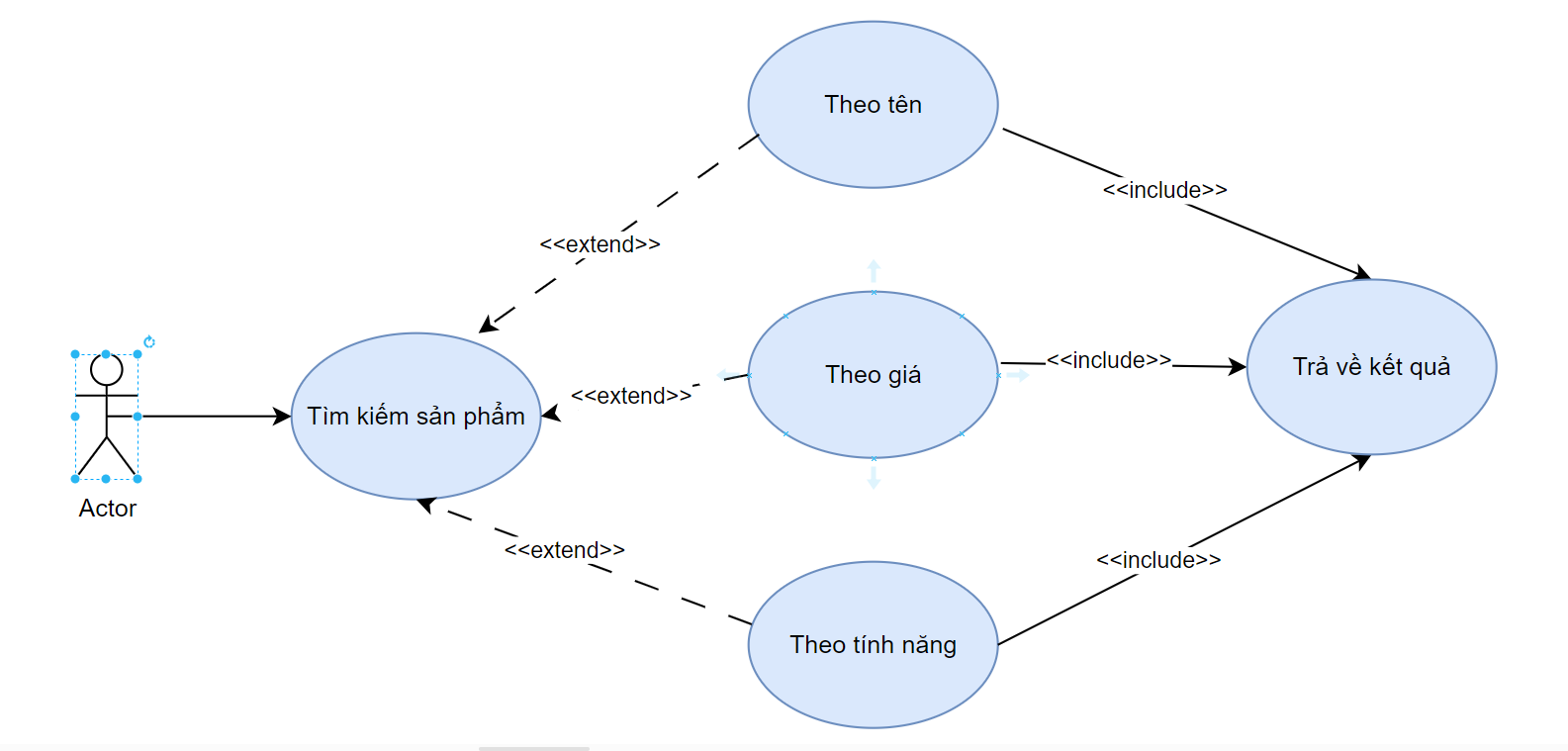
#### 3.2.3.4. Usecase “Quản lý thông tin cá nhân”



Bảng 3.. Bảng mô tả usecase quản lý thông tin cá nhân.

|  |  |
| --- | --- |
| **Tên use-case** | Quản lý thông tin cá nhân |
| **Tác nhân** | User (admin, staff, customer) |
| **Tiền điều kiện** | Đăng nhập thành công |
| **Hậu điều kiện** | Xem là quản lý được thông tin các nhân |
| **Mô tả** | Quản lý thông tin cá nhân |
| **Kịch bản:**  1. Click button thông tin user  2. Chọn thêm, sửa thông tin cá nhân  3. Nhập thông tin (nếu được yêu cầu).  4.Hệ thống xử lý và trả về kết quả | |
| **Ngoại lệ**:  3.1. Thông tin nhập không đúng định dạng. | |

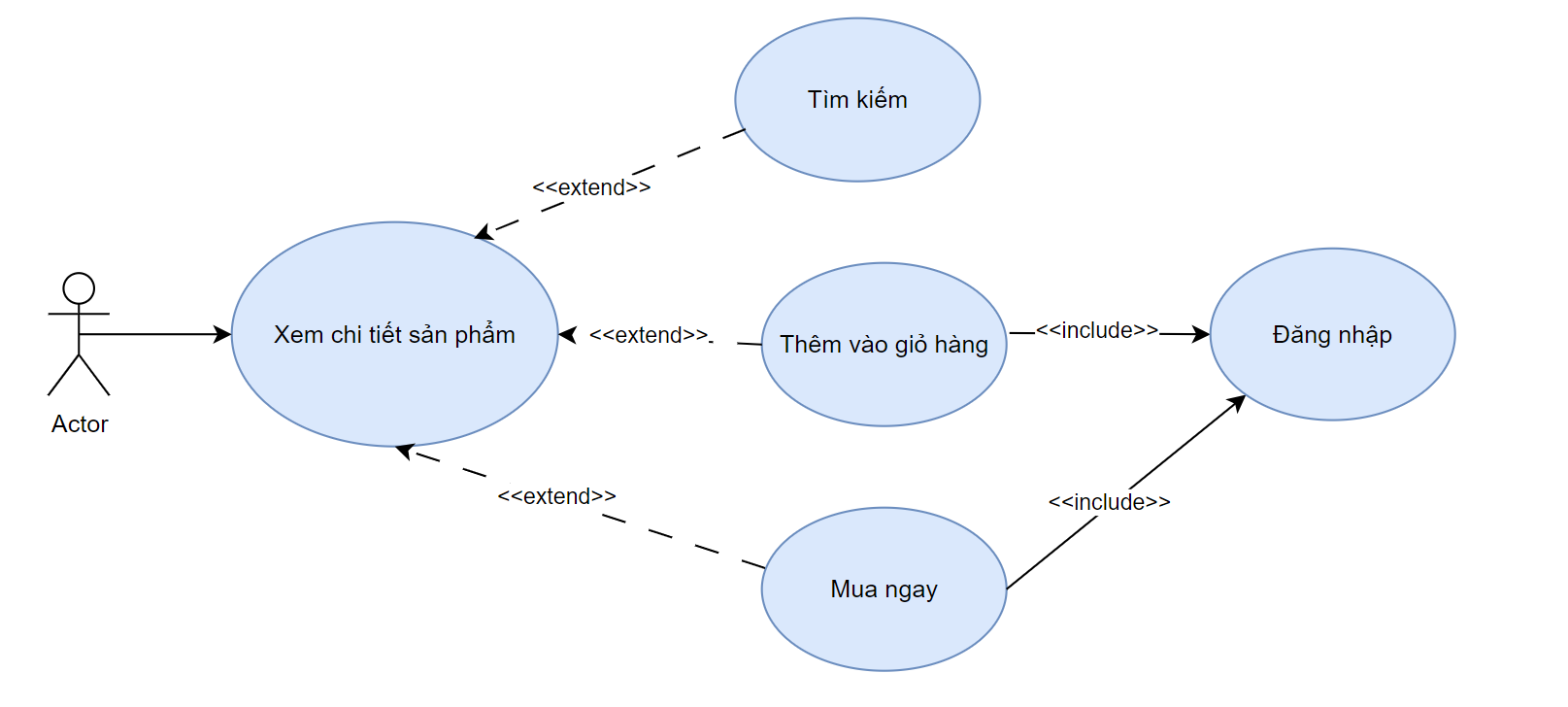
#### 3.2.3.5. Usecase “Tìm kiếm sản phẩm”



Bảng 3.. Bảng mô tả usecase tìm kiếm sản phẩm.

|  |  |
| --- | --- |
| **Tên use-case** | Tìm kiếm sản phẩm |
| **Tác nhân** | Khách hàng |
| **Tiền điều kiện** |  |
| **Hậu điều kiện** | Xem được danh sách sản phẩm cần tìm |
| **Mô tả** | Tìm kiếm danh sách sản phẩm |
| **Kịch bản:**  1. Click xem danh sách sản phẩm  2. Chọn các tiêu chí lọc, tìm kiếm.  3.Hệ thống xử lý và trả về kết quả | |
| **Ngoại lệ**: | |

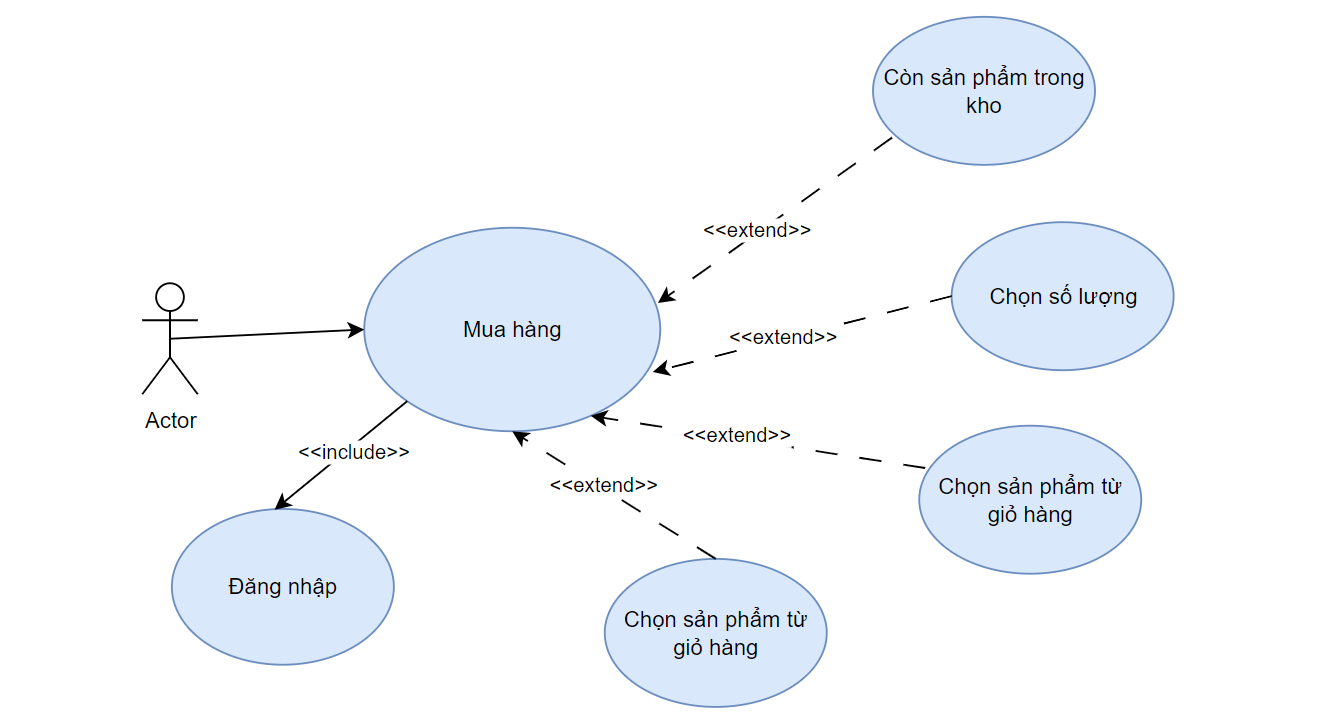
#### 3.2.3.6. Usecase “Xem chi tiết sản phẩm”



Bảng 3.. Bảng mô tả usecase xem chi tiết sản phẩm.

|  |  |
| --- | --- |
| **Tên use-case** | Xem chi tiết sản phẩm |
| **Tác nhân** | Khách hàng |
| **Tiền điều kiện** |  |
| **Hậu điều kiện** | Xem là quản lý được thông tin sản phẩm cụ thể. |
| **Mô tả** | Xem chi tiết thông tin sản phẩm. |
| **Kịch bản:**  1. Click vào thẻ sản phẩm cụ thể  2.Hệ thống xử lý và trả về kết quả | |
| **Ngoại lệ**: | |

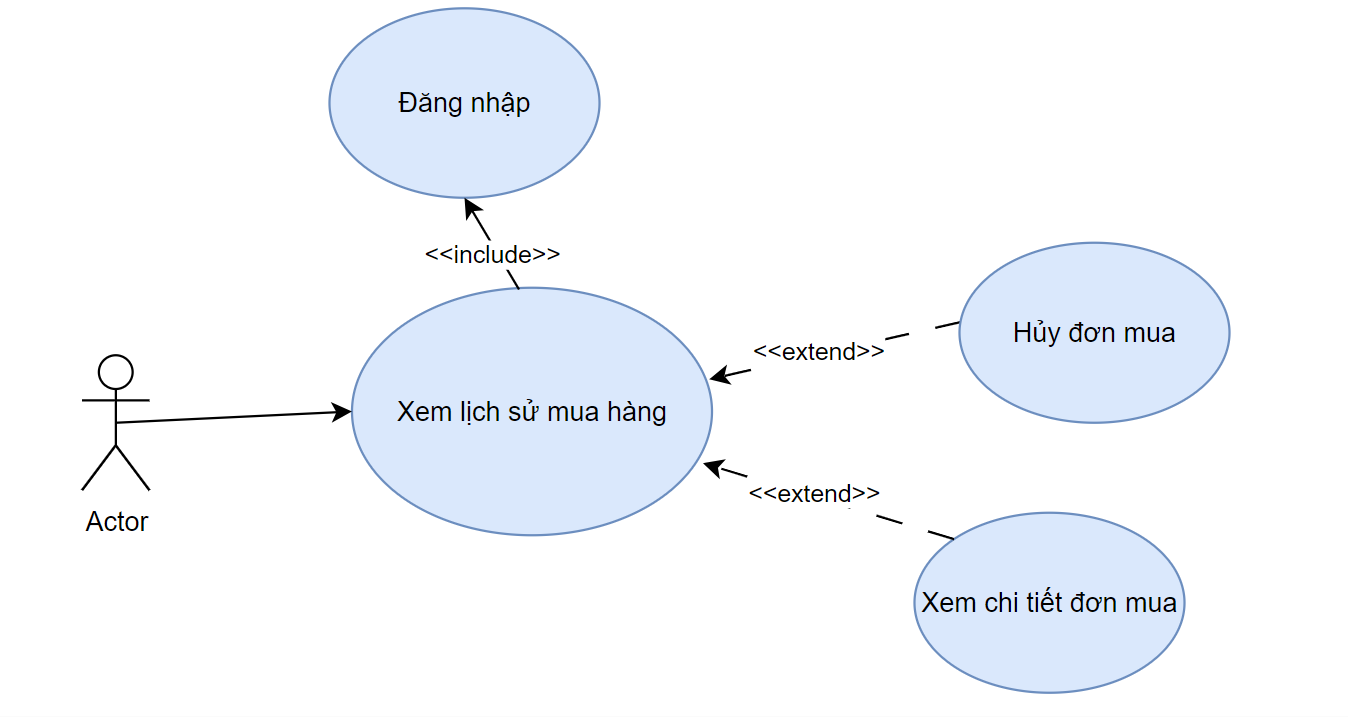
#### 3.2.3.7. Usecase “Mua hàng”



Bảng 3.. Bảng mô tả usecase mua hàng.

|  |  |
| --- | --- |
| **Tên use-case** | Mua hàng |
| **Tác nhân** | Khách hàng |
| **Tiền điều kiện** | Đăng nhập thành công |
| **Hậu điều kiện** | Mua hàng thành công |
| **Mô tả** | Mua hàng |
| **Kịch bản:**  1.Chọn sản phẩm cần mua từ giỏ hàng hoặc chi tiết sản phẩm.  2.Nhấn mua hàng.  3. Chọn phương thức thanh toán.  4. Hệ thống xử lý và trả về kết quả. | |
| **Ngoại lệ**:  1.1. Sản phẩm trong kho không đủ. | |

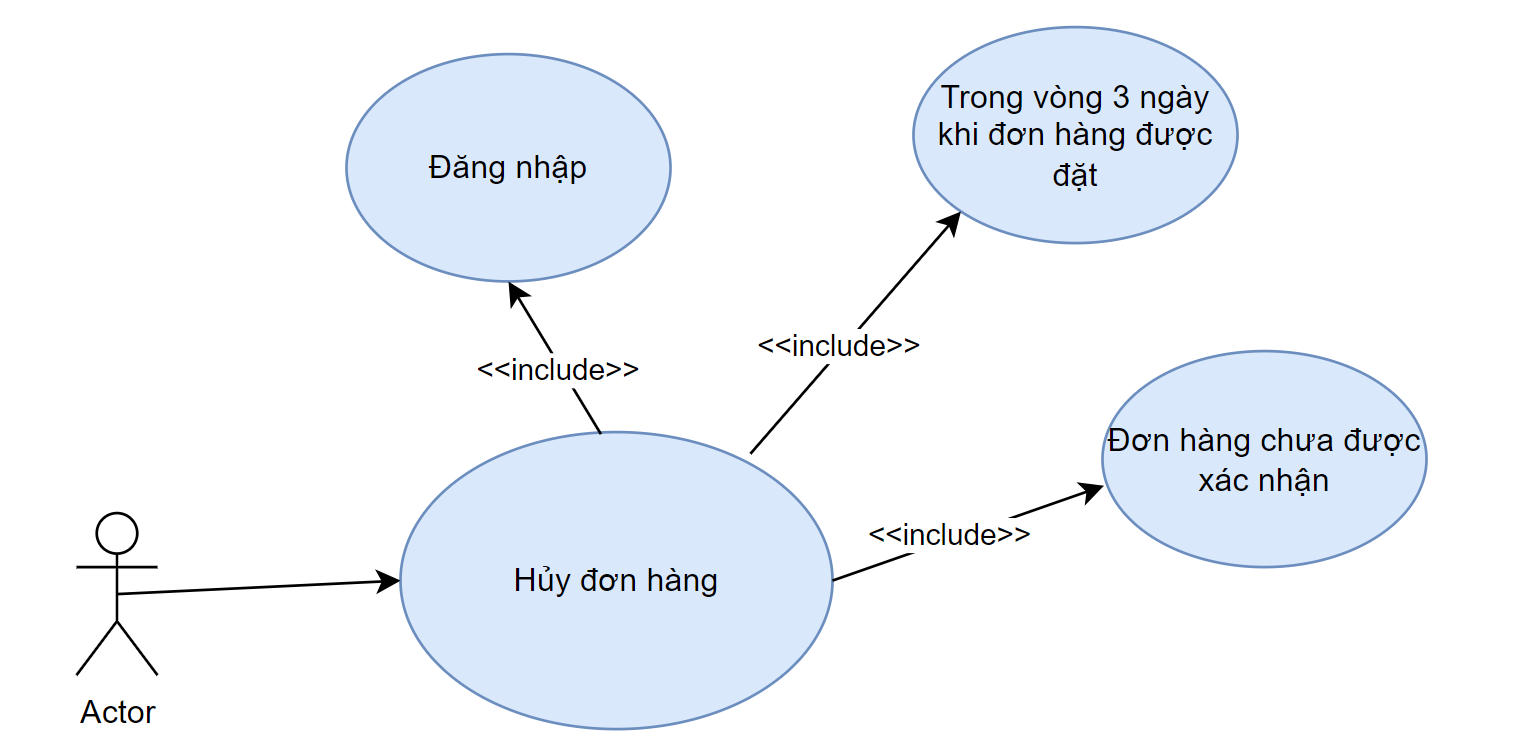
#### 3.2.3.8. Usecase “Xem lịch sử mua hàng”



Bảng 3.. Bảng mô tả usecase xem lịch sử mua hàng.

|  |  |
| --- | --- |
| **Tên use-case** | Xem lịch sử mua hàng |
| **Tác nhân** | Khách hàng |
| **Tiền điều kiện** | Đăng nhập thành công. |
| **Hậu điều kiện** | Xem được lịch sử mua hàng |
| **Mô tả** | Xem lịch sử những đơn hàng đã mua. |
| **Kịch bản:**  1. Click vào nút xem lịch sử đơn hàng  2. Click vào một đơn hàng cụ thể để xem chi tiết đơn hàng.  3.Hệ thống xử lý và trả về kết quả | |
| **Ngoại lệ**: | |

#### 3.2.3.9. Usecase “Hủy đơn hàng”



Bảng 3.. Bảng mô tả usecase hủy đơn hàng.

|  |  |
| --- | --- |
| **Tên use-case** | Hủy đơn hàng |
| **Tác nhân** | Khách hàng |
| **Tiền điều kiện** | Đăng nhập thành công |
| **Hậu điều kiện** | Hủy đơn hàng thành công |
| **Mô tả** | Hủy đơn hàng |
| **Kịch bản:**  1. Click vào xem lịch sử mua hàng.  2.Click vào hủy đơn hàng.  3. Hệ thống xử lý và trả về kết quả. | |
| **Ngoại lệ**:  2.1 Đơn hàng không còn hợp lệ để hủy (quá ba ngày, đã được xác nhận, ...). | |

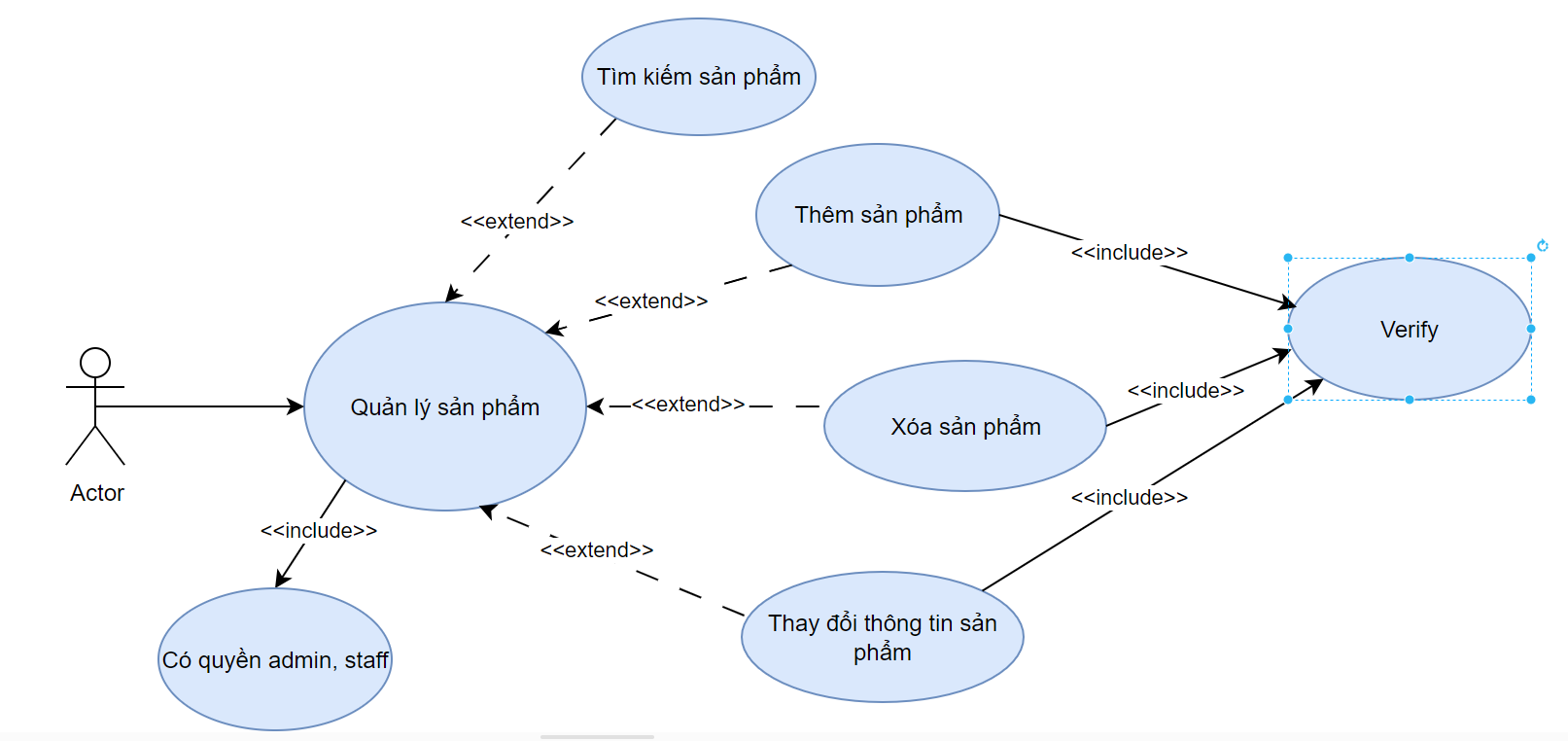
#### 3.2.3.10. Usecase “Quản lý nhân viên”



Bảng 3.. Bảng mô tả usecase quản lý nhân viên.

|  |  |
| --- | --- |
| **Tên use-case** | Quản lý nhân viên |
| **Tác nhân** | Admin |
| **Tiền điều kiện** | Đã đăng nhập thành công |
| **Hậu điều kiện** | Quản lý được danh sách nhân viên |
| **Mô tả** | Quản lý nhân viên |
| **Kịch bản:**  1. Click vào quản lý nhân viên  2.Click vào một nhân viên cụ thể.  3. Xem hoặc sửa, xóa thông tin của nhân viên.  4. Hệ thống xử lý và trả về kết quả | |
| **Ngoại lệ**: | |

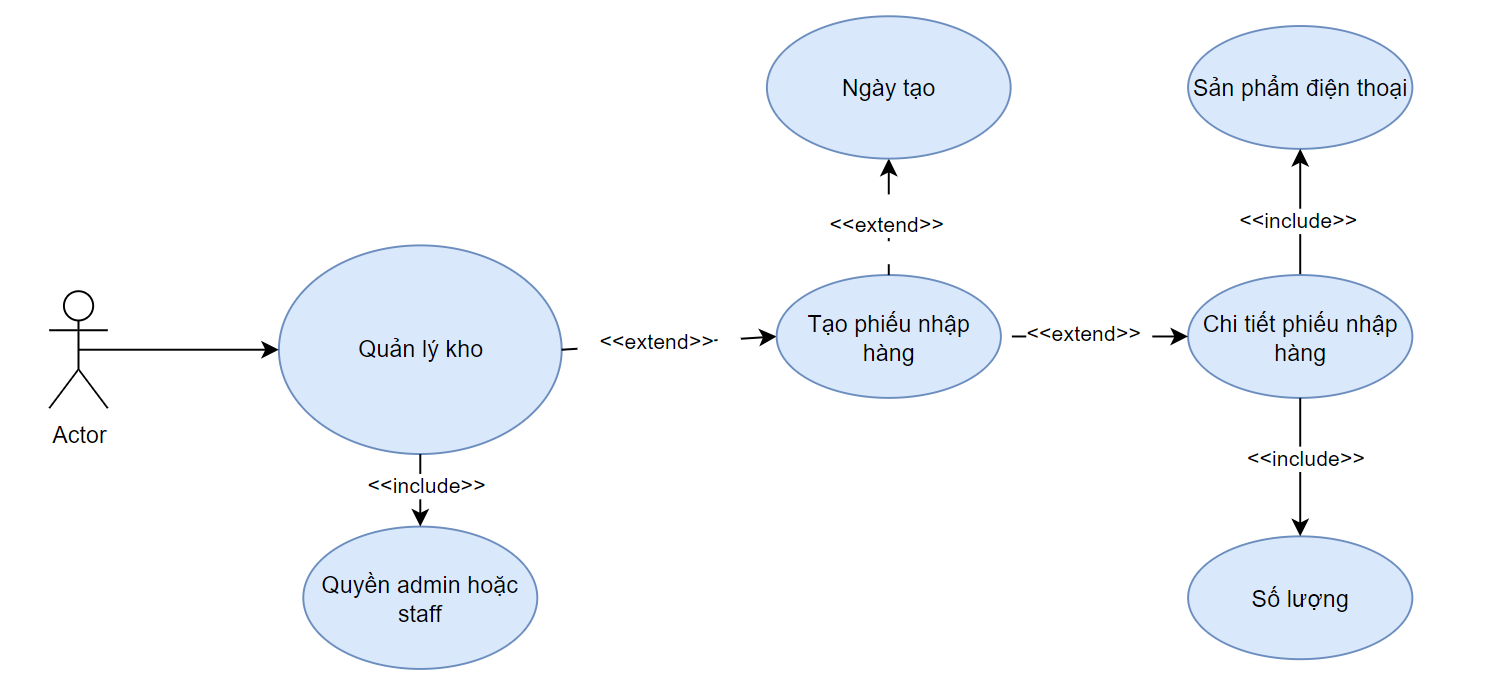
#### 3.2.3.11. Usecase “Quản lý sản phẩm”



Bảng 3.. Bảng mô tả usecase quản lý sản phẩm.

|  |  |
| --- | --- |
| **Tên use-case** | Quản lý sản phẩm |
| **Tác nhân** | Admin, staff |
| **Tiền điều kiện** | Đã đăng nhập, có quyền staff hoặc admin |
| **Hậu điều kiện** | Quản lý được thông tin sản phẩm hiện có trong cửa hàng |
| **Mô tả** | Quản lý thông tin sản phẩm |
| **Kịch bản:**  1. Click vào quản lý sản phẩm.  2. Click vào sản phẩm cụ thể để sửa thông tin hoặc xem chi tiết.  3. Hệ thống xử lý và trả về kết quả. | |
| **Ngoại lệ**:  2.1 Thông tin sản phẩm khi thay đổi không hợp lệ. | |

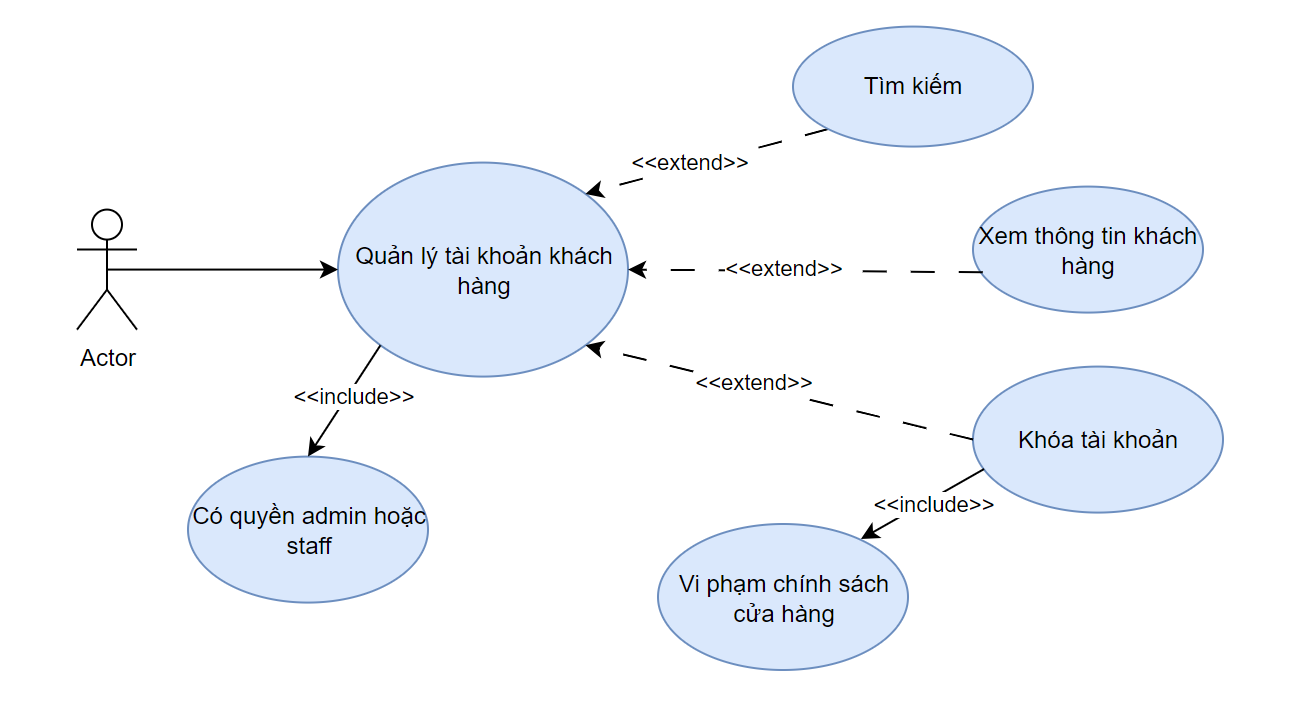
#### 3.2.3.12. Usecase “Quản lý kho”



Bảng 3.. Bảng mô tả usecase quản lý kho.

|  |  |
| --- | --- |
| **Tên use-case** | Quản lý kho |
| **Tác nhân** | Admin, staff |
| **Tiền điều kiện** | Đã đăng nhập, có quyền staff hoặc admin |
| **Hậu điều kiện** | Xem là quản lý được thông tin kho. |
| **Mô tả** | Quản lý kho |
| **Kịch bản:**  1. Click vào quản lý kho.  2. Tạo phiếu nhập hàng hoặc xem thông tin kho.  3. Nhập thông tin cần thiết (nếu nhập hàng).  4. Hệ thống xử lý và trả về kết quả. | |
| **Ngoại lệ**:  3.1 Thông tin nhập không hợp lệ. | |

#### 3.2.3.13. Usecase “Quản lý tài khoản khách hàng”



Bảng 3.. Bảng mô tả usecase quản lý tài khoản khách hàng.

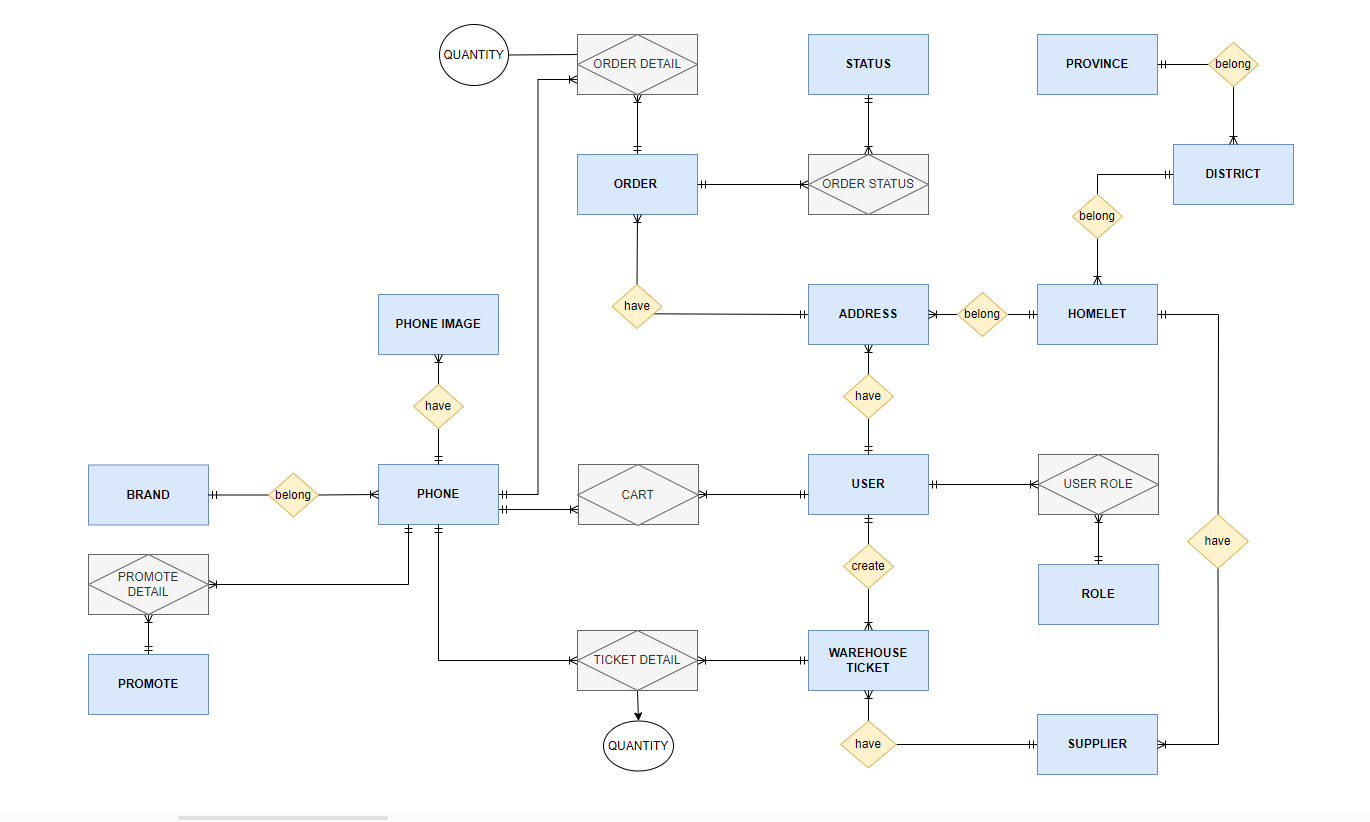
|  |  |
| --- | --- |
| **Tên use-case** | Quản lý tài khoản khách hàng |
| **Tác nhân** | Admin, staff |
| **Tiền điều kiện** | Đăng nhập, có quyền admin, staff |
| **Hậu điều kiện** | Quản lý tài khoản khách hàng thành công |
| **Mô tả** | Quản lý tài khoản khách hàng hiên có trong cửa hàng. |
| **Kịch bản:**  1. Click vào quản lý khách hàng.  2. Click vào chi tiết khách hàng.  3. Click vào khóa tài khoản khách hàng.  4.Hệ thống xử lý và trả về kết quả. | |
| **Ngoại lệ**: | |

## 3.3. Mô hình thực thể ERD.

### 3.3.1. Xác định các thực thể.

* Address (Id, DetailAddress, Type).
* Role (Id, Name, NormalizedName).
* Brand (Id, Name, Logo, Description).
* Phone (Id, Name, Description, Operation, CPU, RAM, ROM, PIN, ScreenSize, ScreenResolution, FrontCamera, BehindCamera, ScreenTouch, OtherBenefit, BrandId, Quantity, ImportPrice, SoldPrice, Slug).
* PhoneImage (Id, Link).
* Promote (Id, Name, Description, FromDate, ToDate).
* Order (Id, CreateDate, PaymentMethod, PaymentStatus, PaymentOnlineReceipt).
* Homelet (Id, Name, Type).
* District (Id, Name, Type).
* Province (Id, Name, Type).
* User (Id, FirstName, LastName, Age, Avatar, ResetPasswordToken, RefreshToken, IsBlock, UserName, Email, PasswordHash, PhoneNumber).
* WarehouseTicket (Id, Created).
* Supplier (Id, Name, Description).
* Status (Id, StatusType).

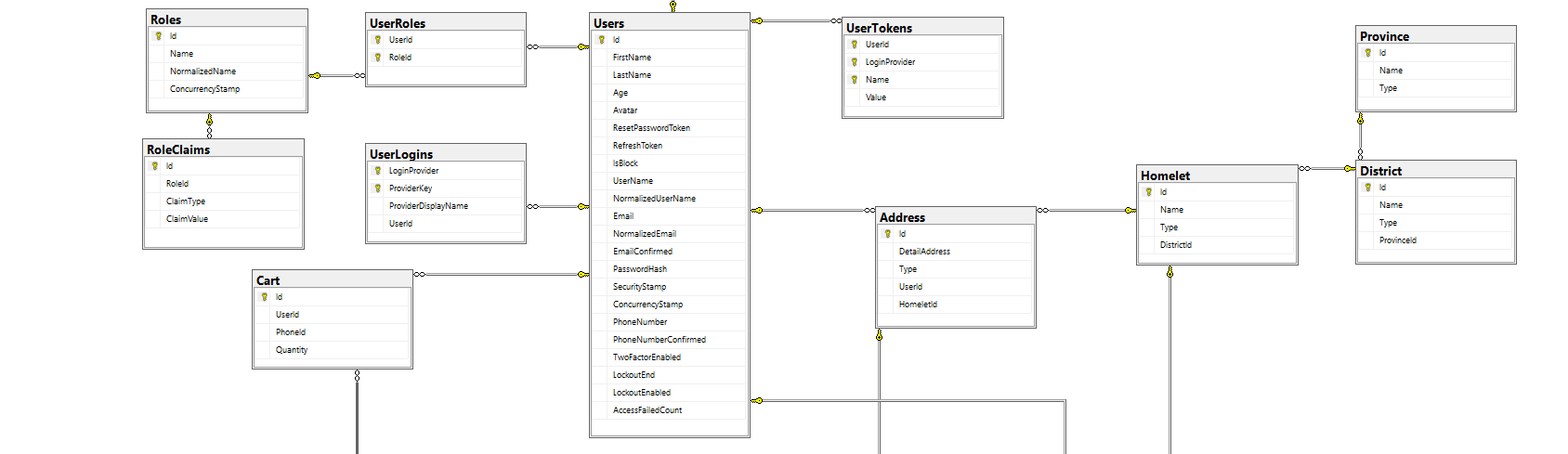
### 3.3.2. Mô hình thực thể ERD (Entity RelationShip Diagram).

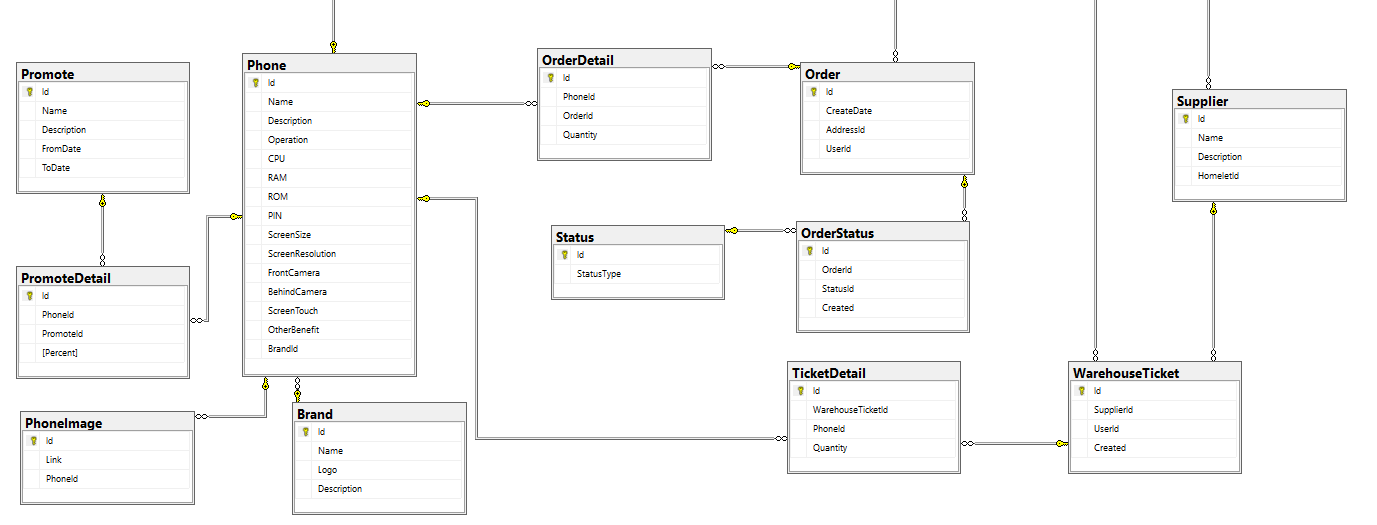


Hình 3.. Sơ đồ diagram.

## 3.4. Thiết kế cơ sở dữ liệu.

### 3.4.1. Mô hình diagram của cơ sở dữ liệu.





Hình 3.. Mô hình diagram của cơ sở dữ liệu

### 3.4.2. Từ điển dữ liệu.

#### 3.4.2.1. Bảng User

Bảng 3.. Mô tả các thuộc tính trong bảng User

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | nvarchar | 255 | Mã Id | Not null  PrimaryKey |
| 2 | FirstName | nvarchar | 255 | Họ | Not null |
| 3 | LastName | nvarchar |  | Tên | Not null |
| 4 | Email | nvarchar | 45 | Địa chỉ email | Not null |
| 5 | Avatar | nvarchar | 255 | Hình ảnh |  |
| 6 | PasswordHash | nvarchar | 45 | Mật khẩu được mã hóa | Not null |
| 7 | PhoneNumber | nvarchar | 10 | Số điện thoại | Not null |
| 8 | RefeshToken | nvarchar | 1000 |  |  |
| 9 | UserName | nvarchar | 45 | Tên đăng nhập | Not null |
| 10 | IsBlock | Bit | 1 | Khóa tài khoản | Not null |
| 11 | Resetpassword Token | nvarchar | 255 | Token đặt lại mật khẩu |  |
| 12 | Age | int |  | Tuổi |  |

#### 3.4.2.2. Bảng Role

Bảng 3.. Mô tả các thuộc tính trong bảng Role

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | nvarchar |  | Id | Not null  PrimaryKey |
| 2 | Name | varchar | 255 | Mã vai trò | Not null |
| 3 | NormalizedName | varchar | 255 | Mã vai trò in hoa | Not null |

#### 3.4.2.3. Bảng UserRole

Bảng 3.. Mô tả các thuộc tính trong bảng RoleUser

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | UserId | nvarchar | 255 | Mã người dùng | Primary Key, Foreign Key,  Not null |
| 2 | RoleId | nvarchar | 255 | Mã vai trò |

#### 3.4.2.4. Bảng Brand

Bảng 3.. Mô tả các thuộc tính trong bảng Brand

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | int |  | Id | Not null  PrimaryKey |
| 2 | Name | varchar | 255 | Tên hãng | Not null |
| 3 | Logo | varchar | 255 | Logo |  |
| 4 | Description | Nvarchar | Max | Mô tả |  |

#### 3.4.2.5. Bảng Address

Bảng 3.. Mô tả các thuộc tính trong bảng Address

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | uniqueidentifỉer |  | Mã người dùng | Primary key |
| 2 | DetailAddress | nvarchar | 1000 | Chi tiết | Not null |
| 3 | Type | varchar | 45 | Địa chỉ email | Not null |
| 4 | UserId | nvarchar | 255 | Mã user | Foreign Key  Not null |
| 5 | HomeletId | nvarchar | 10 | Mã xã | Foreign Key  Not null |

#### 3.4.2.6. Bảng Cart

Bảng 3.. Mô tả các thuộc tính trong bảng Cart

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | nvarchar |  | Id | Not null  PrimaryKey |
| 2 | UserId | nvarchar | 255 | Mã user | Foreignkey  Not null |
| 3 | PhoneId | uniqueidentifỉer |  | Mã điện thoại | Foreignkey  Not null |
| 4 | Quantity | Int |  | Số lượng | Not null |

#### 3.4.2.7. Bảng Homelet

Bảng 3.. Mô tả các thuộc tính trong bảng Homelet

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | nvarchar | 255 | Id | Not null  PrimaryKey |
| 2 | Name | nvarchar | 255 | Tên Xã | Not null |
| 3 | Type | varchar | 255 | Loại (xã, phường, thị trấn) | Not null |
| 4 | DistrictId | BigDecimal | 255 | Mã huyện | Foreign Key Not null |

#### 3.4.2.8. Bảng District

Bảng 3.. Mô tả các thuộc tính trong bảng District

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | nvarchar | 255 | Id | Not null  PrimaryKey |
| 2 | Name | nvarchar | 255 | Tên huyện | Not null |
| 3 | Type | varchar | 255 | Loại (huyện, quận, thị xã, thành phố) | Not null |
| 4 | ProvinceId | nvarchar | 255 | Mã tỉnh | Foreign Key Not null |

#### 3.4.2.9. Bảng Province

Bảng 3.. Mô tả các thuộc tính trong bảng Province

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | nvarchar | 255 | Id | Not null  PrimaryKey |
| 2 | Name | nvarchar | 255 | Tên huyện | Not null |
| 3 | Type | varchar | 255 | Loại (tỉnh, thành phố) | Not null |

#### 3.4.2.10. Bảng Phone

Bảng 3.. Mô tả các thuộc tính trong bảng Phone

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | uniqueidentifỉer |  | Mã Id | Not null  PrimaryKey |
| 2 | Name | nvarchar | 11 | Tên điện thoại | Not null |
| 3 | Description | nvarchar |  | Mô tả | Not null |
| 4 | Operation | nvarchar | 255 | Hệ điều hành | Not null |
| 5 | CPU | nvarchar | 255 | Bộ xử lý | Not null |
| 6 | RAM | nvarchar | 255 |  | Not null |
| 7 | ROM | nvarchar | 255 | Bộ nhớ trong | Not null |
| 8 | PIN | nvarchar | 255 | Thời lượng pin | Not null |
| 9 | ScreenSize | float |  | Độ rộng màn hình | Not null |
| 10 | ScreenResolution | nvarchar | 255 | Độ phân giải | Not null |
| 11 | FrontCamera | nvarchar | 255 | Camera trước | Not null |
| 12 | BehindCamera | nvarchar | 255 | Camera sau | Not null |
| 13 | ScreenTouch | nvarchar | 255 | Cảm ứng | Not null |
| 14 | OtherBenefit | nvarchar | Max | Tiện ích khác | Not null |
| 15 | BrandId | uniqueidentifỉer |  | Mã thương hiệu | Foreignkey  Not null |
| 16 | Quantity | int |  | Số lượng trong kho | Not null |
| 17 | ImportPrice | float |  | Giá nhập | Not null |
| 18 | SoldPrice | float |  | Giá bán | Not null |
| 19 | Slug | nvarchar | 255 | Đường dẫn duy nhất | Unique  Not null |

#### 3.4.2.11. Bảng Order

Bảng 3.. Mô tả các thuộc tính trong bảng Order

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | uniqueidentifier |  | Id | Not null  PrimaryKey |
| 2 | CreateDate | datetime2(7) | 255 | Ngày tạo | Not null |
| 3 | AddressId | uniqueidentifier | 255 | Địa chỉ giao hàng | Foreign Key, Not null |
| 4 | UserId | nvarchar | 255 | Id | Foreign Key, Not null |
| 5 | PaymentMethod | nvarchar | 255 | Phương thức thanh toán | Not null |
| 6 | PaymentStatus | nvarchar | 255 | Trạng thái | Not null |
| 7 | PaymentOnlineReceipt | nvarchar | max | Hóa đơn | Not null |

#### 3.4.2.12. Bảng OrderDetail

Bảng 3.. Mô tả các thuộc tính trong bảng OrderDetail

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | uniqueidentifier |  | Id | Not null  PrimaryKey |
| 2 | PhoneId | uniqueidentifier |  | Mã điện thoại | Foreign Key, Not null |
| 3 | OrderId | uniqueidentifier |  | Mã đơn đặt hàng | Foreign Key, Not null |
| 4 | Quantity | int |  | Số lượng | Not null |

#### 3.4.2.13. Bảng OrderStatus

Bảng 3.. Mô tả các thuộc tính trong bảng OrderStatus

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | uniqueidentifier |  | Mã Id | Not null  PrimaryKey |
| 2 | OrderId | uniqueidentifier |  | Mã đơn đặt hàng | Foreign Key  Not null |
| 3 | StatusId | uniqueidentifier |  | Mã trạng thái | Foreign Key  Not null |
| 4 | Created | datetime2(7) | 255 | Ngày tạo | Foreign Key Not null |

#### 3.4.2.14. Bảng PhoneImage

Bảng 3.. Mô tả các thuộc tính trong bảng PhoneImage

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** |  | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | uniqueidentifier |  |  | Id | PrimaryKey  Not null |
| 2 | Link | nvarchar | max |  | Đường dẫn ảnh | Not null |
| 3 | PhoneId | uniqueidentifier |  |  | Mã điện thoại | Foreignkey  Not null |

#### 3.4.2.15. Bảng Promote

Bảng 3.. Mô tả các thuộc tính trong bảng Promote

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | uniqueidentifier |  | Id | PrimaryKey  Not null |
| 2 | Name | nvarchar | max | Đường dẫn ảnh | Not null |
| 3 | Description | nvarchar |  | Mô tả | Foreignkey  Not null |
| 4 | FromDate | datetime2(7) |  | Ngày bắt đầu | Not null |
| 5 | ToDate | datetime2(7) |  | Ngày kết thúc | Not null |

#### 3.4.2.16. Bảng PromoteDetail

Bảng 3.. Mô tả các thuộc tính trong bảng PromoteDetail

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | uniqueidentifier |  | Id | PrimaryKey  Not null |
| 2 | PhoneId | uniqueidentifier | max | Mã điện thoại | Foreignkey  Not null |
| 3 | PromoteId | uniqueidentifier |  | Mã đợt khuyến mãi | Foreignkey  Not null |
| 4 | Percent | float |  | Tỉ lệ giảm | Not null |

#### 3.4.2.17. Bảng Status

Bảng 3.. Mô tả các thuộc tính trong bảng Status

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | uniqueidentifier |  | Id | PrimaryKey  Not null |
| 2 | StatusType | nvarchar | 255 | Tên trạng thái | Not null |

#### 3.4.2.18. Bảng Supplier

Bảng 3.. Mô tả các thuộc tính trong bảng Supplier

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | uniqueidentifier |  | Id | PrimaryKey  Not null |
| 2 | Name | nvarchar | max | Tên nhà phân phối | Not null |
| 3 | Description | nvarchar |  | Mô tả |  |
| 4 | HomeletId | nvarchar |  | Mã xã | Foreignkey  Not null |

#### 3.4.2.19. Bảng WarehouseTicket

Bảng 3.. Mô tả các thuộc tính trong bảng WarehouseTicket

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | uniqueidentifier |  | Id | PrimaryKey  Not null |
| 2 | SupplierId | uniqueidentifier | max | Mã nhà phân phối | Foreignkey  Not null |
| 3 | UserId | nvarchar |  | Mã nhân viên | Foreignkey  Not null |
| 4 | Created | datetime2(7) |  | Ngày tạo | Not null |

#### 3.4.2.20. Bảng TicketDetail

Bảng 3.. Mô tả các thuộc tính trong bảng TicketDetail

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | uniqueidentifier |  | Id | PrimaryKey  Not null |
| 2 | WarehouseTicketId | uniqueidentifier | max | Mã nhà phân phối | Foreignkey  Not null |
| 3 | PhoneId | uniqueidentifier |  | Mã điện thoại | Foreignkey  Not null |
| 4 | Quantity | int |  | Số lượng | Not null |

# CHƯƠNG 4: XÂY DỰNG CHƯƠNG TRÌNH

## 4.1. Lập trình xây dựng ứng dụng.

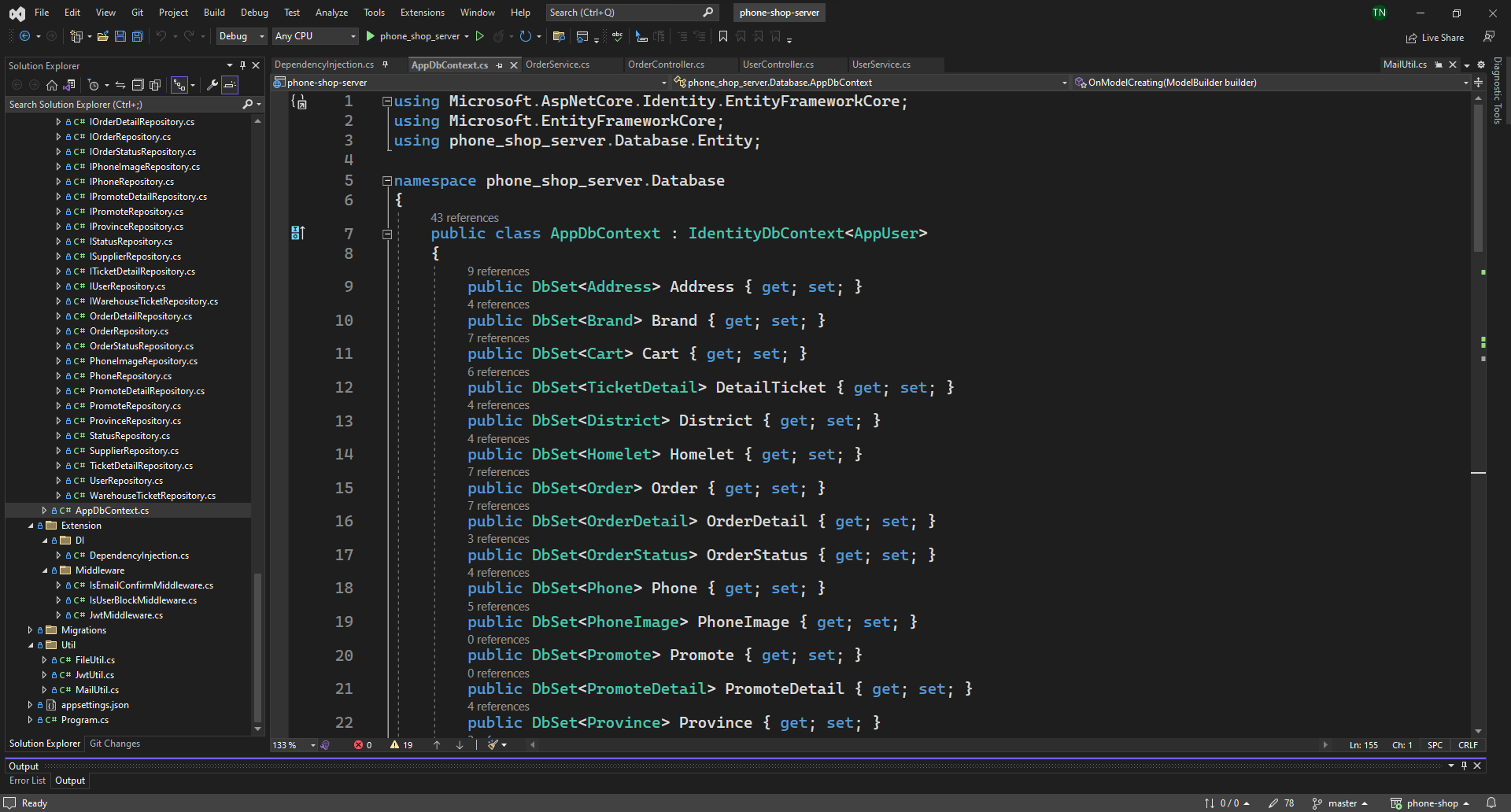
Công nghệ được sử dụng để xây dựng chương trình là ASP.NET Core để viết API cho phần back-end của webbsite và Angular cho phẩn front-end của website. Sử dụng hệ quản trị cơ sở dữ liệu MSSQL Server để lưu trữ dữ liệu.

## 4.2. Xây dựng chương trình phía back-end.

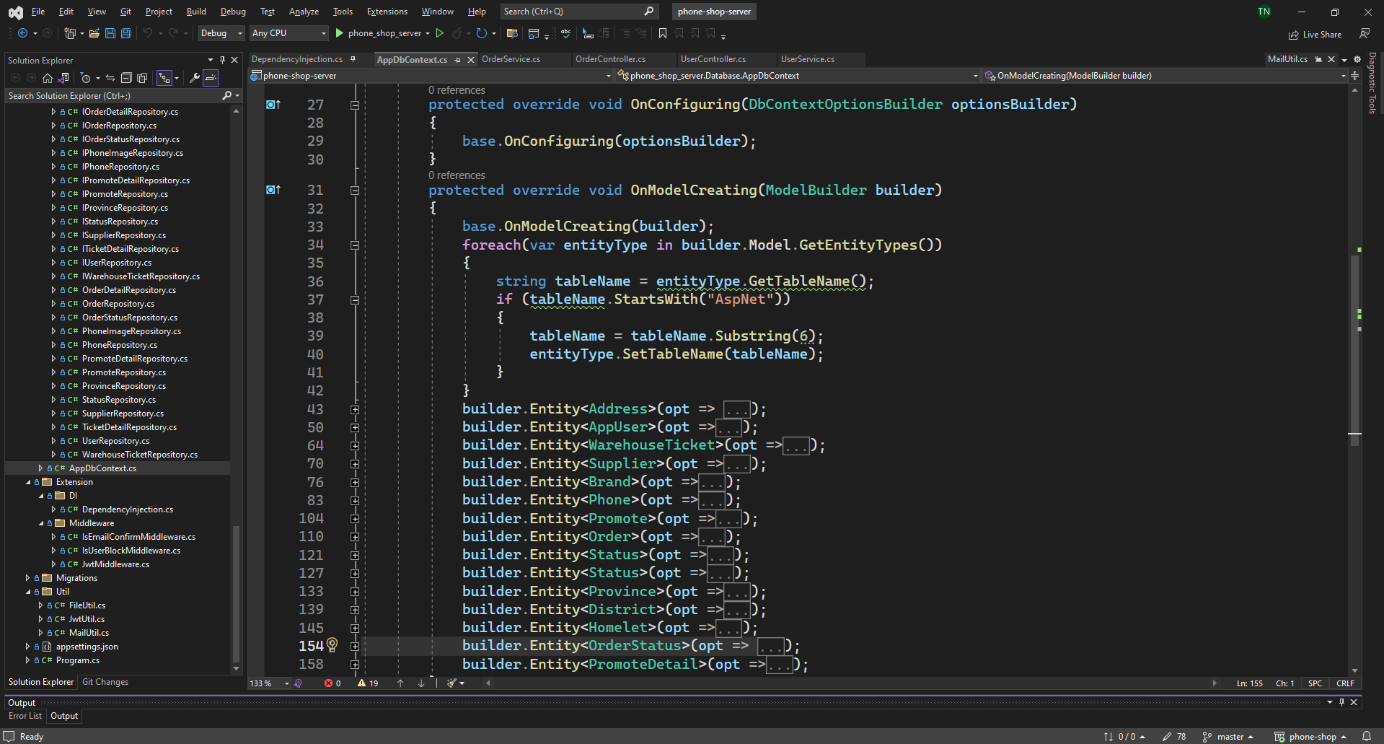
Phẩn back-end của website sử dụng ASP.NET Core để viết các API cung cấp cho phía front-end sử dụng để tạo nên một sản phẩm hoàn chỉnh.

**4.2.1. EntifyFramework Core.**

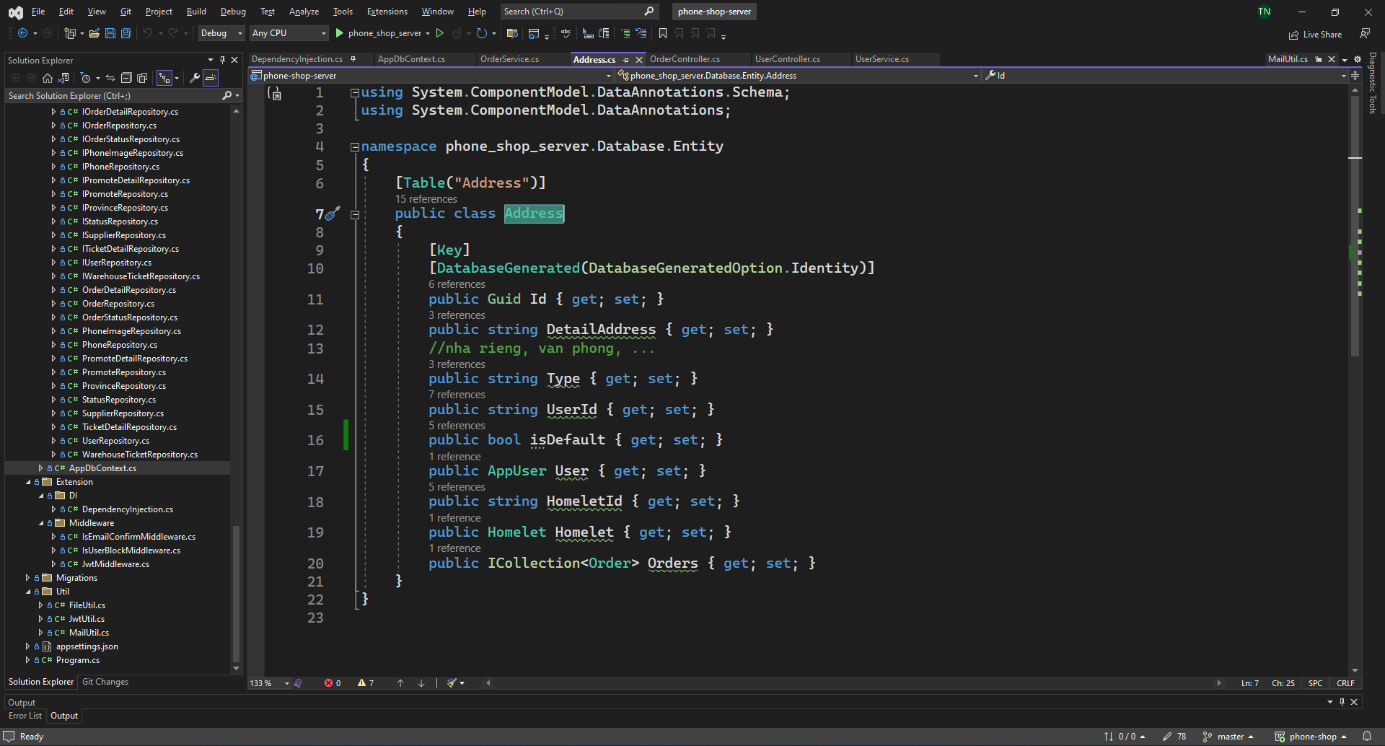
Đề tài sử dụng Entity Framework Core để kết nối đến cơ sở dữ liệu cũng như tạo cơ sở dữ liệu theo hướng code first (sử dụng code để xây dựng database). Dưới đây là AppDbbContext, nơi khai báo thiết lập các đối tượng có trong cơ sở dữ liệu. Mỗi DbSet tương ứng một bảng trong cơ sở dữ liệu.



Hình 4.. Mô tả lớp AppDbContext.

  
Hình 4.. Mô tả lớp AppDbContext.

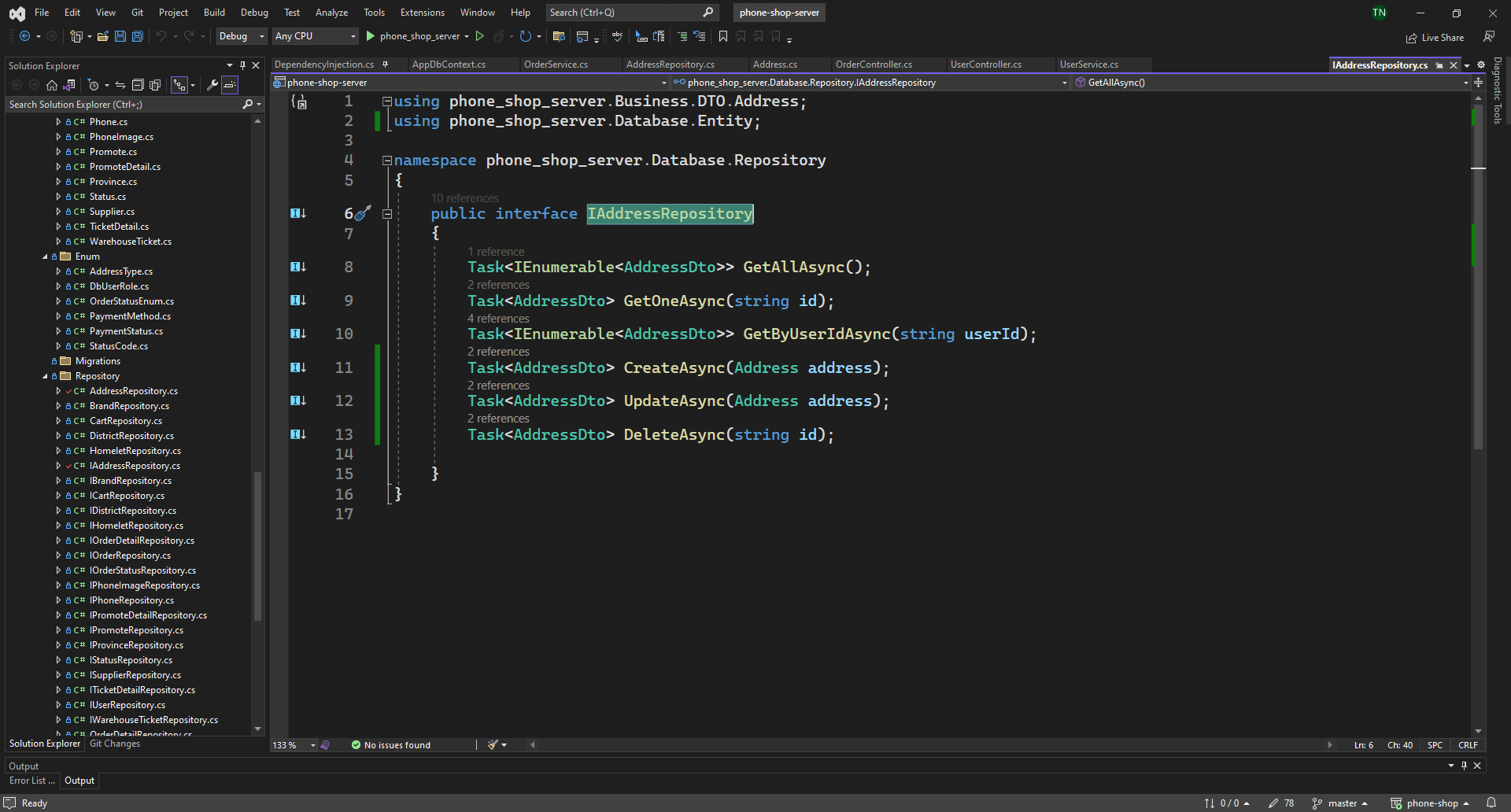
Ví dụ: Bảng địa chỉ (Address) tương ứng sẽ được khai báo ở lớp Address như sau:

  
Hình 4.. Mô tả lớp Address.

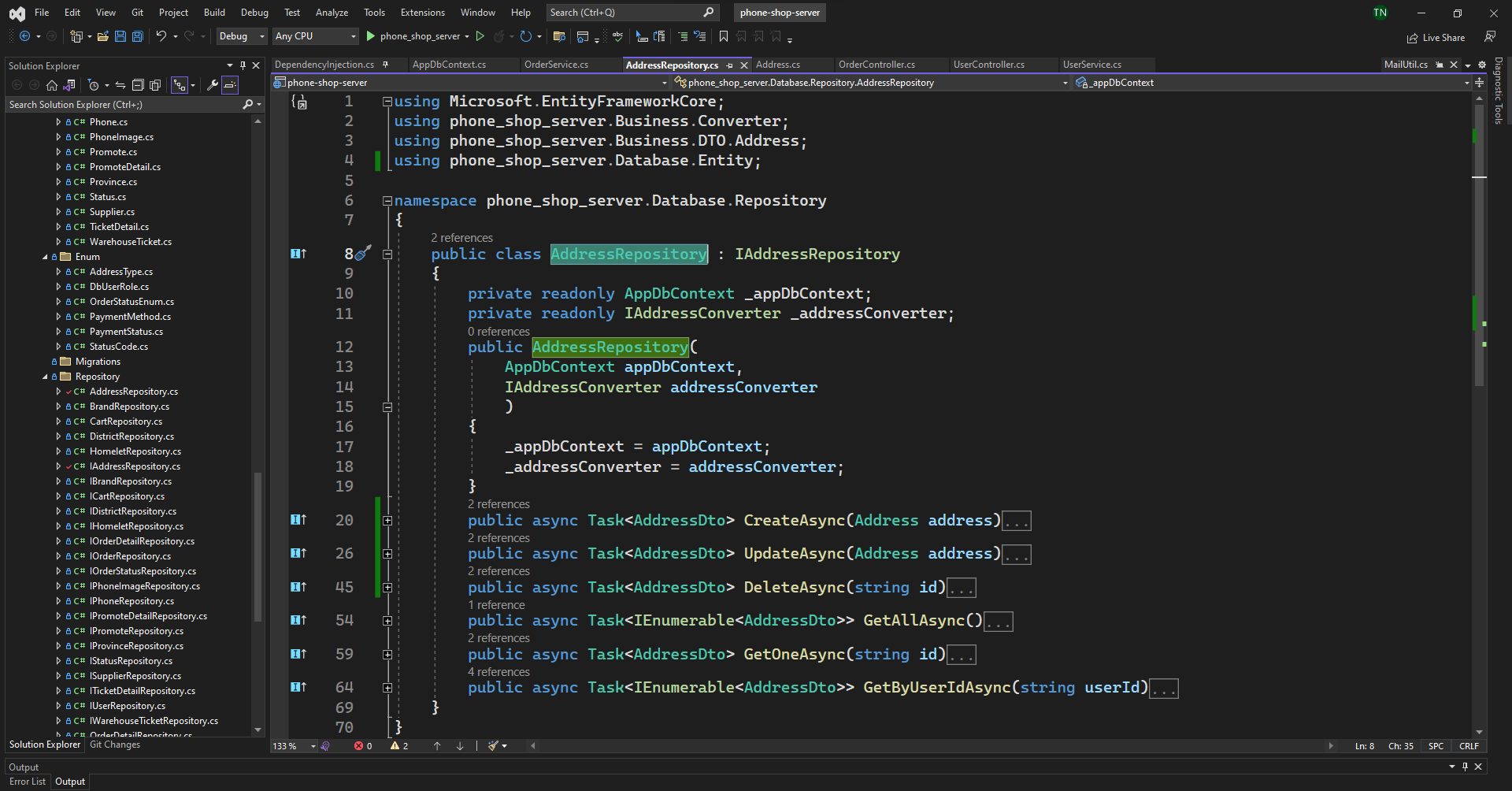
Đề tài cũng sử dụng một số design pattern để phát triển sản phẩm, trong đó có repository pattern và dependency injection pattern. Giúp việc viết code trở nên dễ bảo trì và mở rộng sau này.

**4.2.2. Repository pattern.**

Đối với repository pattern. Các repository được tạo nên kế thừa interface tương ứng, triển khai một số phương thức truy vấn dữ liệu liên quan đến một bảng. Các phương thức này sẽ dễ dàng sử dụng ở những nơi khác, việc sửa chữa sẽ dễ dàng hơn khi chỉ cần sửa ở một nơi, giúp tiết kiệm nhiều tài nguyên trong việc phát triển sản phẩm. Dưới đây là ví dụ về AddressRepository. Một repository giúp truy vấn thông tin liên quan đến bảng Address. Lớp AddressRepository kế thừa và triển khai các phương thức truy vấn đến bảng Address. Những phương thức này được Inject vào chương trình thông qua Dependency Injection và được sử dụng bằng cách nạp vào lớp cần nó thông qua phương thức khởi tạo.



Hình 4.. Mô tả interface IAddressRepository.

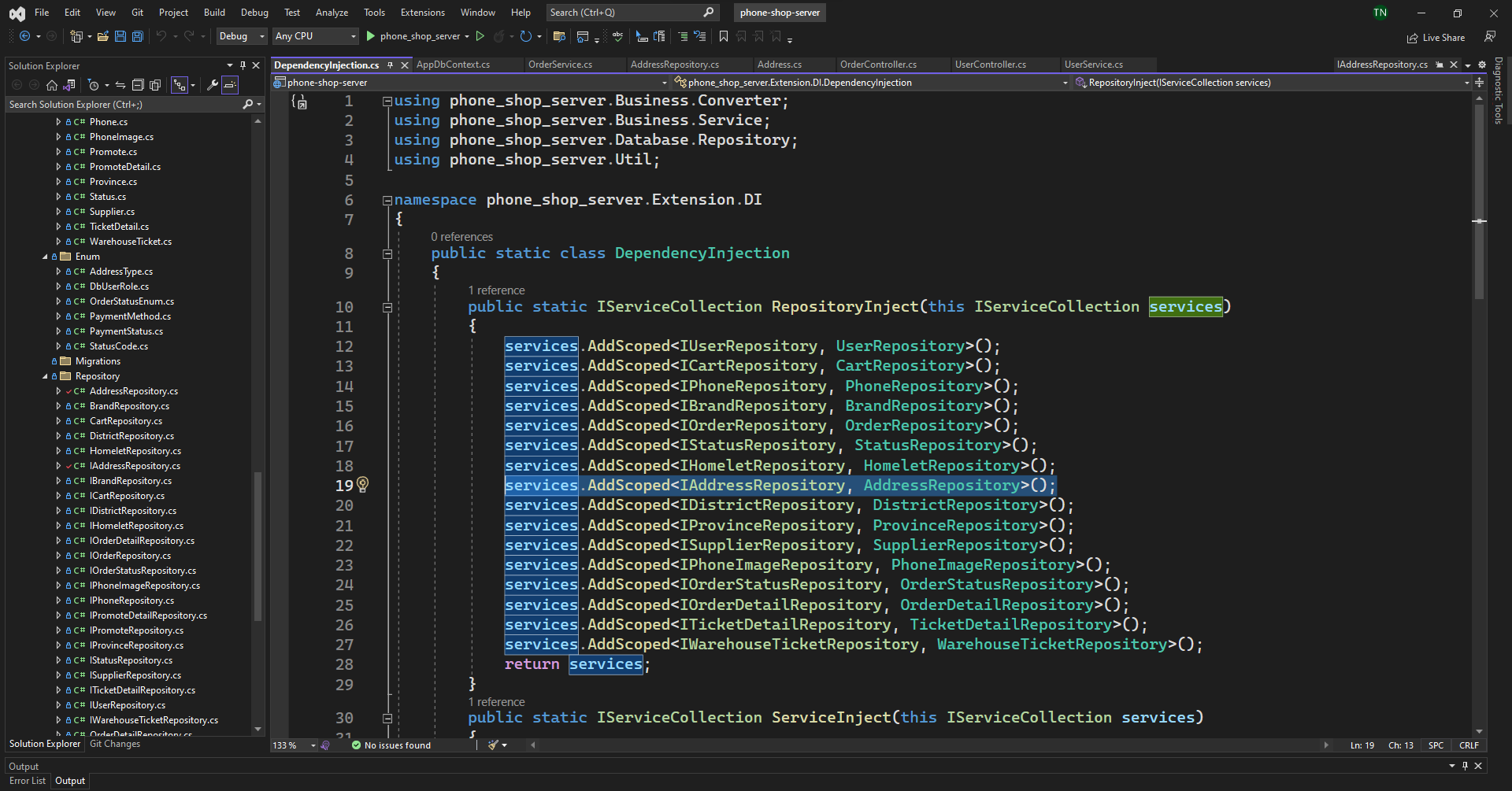


Hình 4.. Mô tả class AddressRepository.

**4.2.2. Dependency Injection pattern.**

Dependency Injection pattern cho phép đăng ký các dịch vụ cần thiết vào lúc khởi tạo ứng dụng và chúng ta có thể dễ dàng gọi và sử dụng chúng ở lớp triển khai mà không cần phải khởi tạo lại nó mỗi khi sử dụng.

Ví dụ với AddressRepository, nó được Inject vào ứng dụng trong lúc khởi tạo.

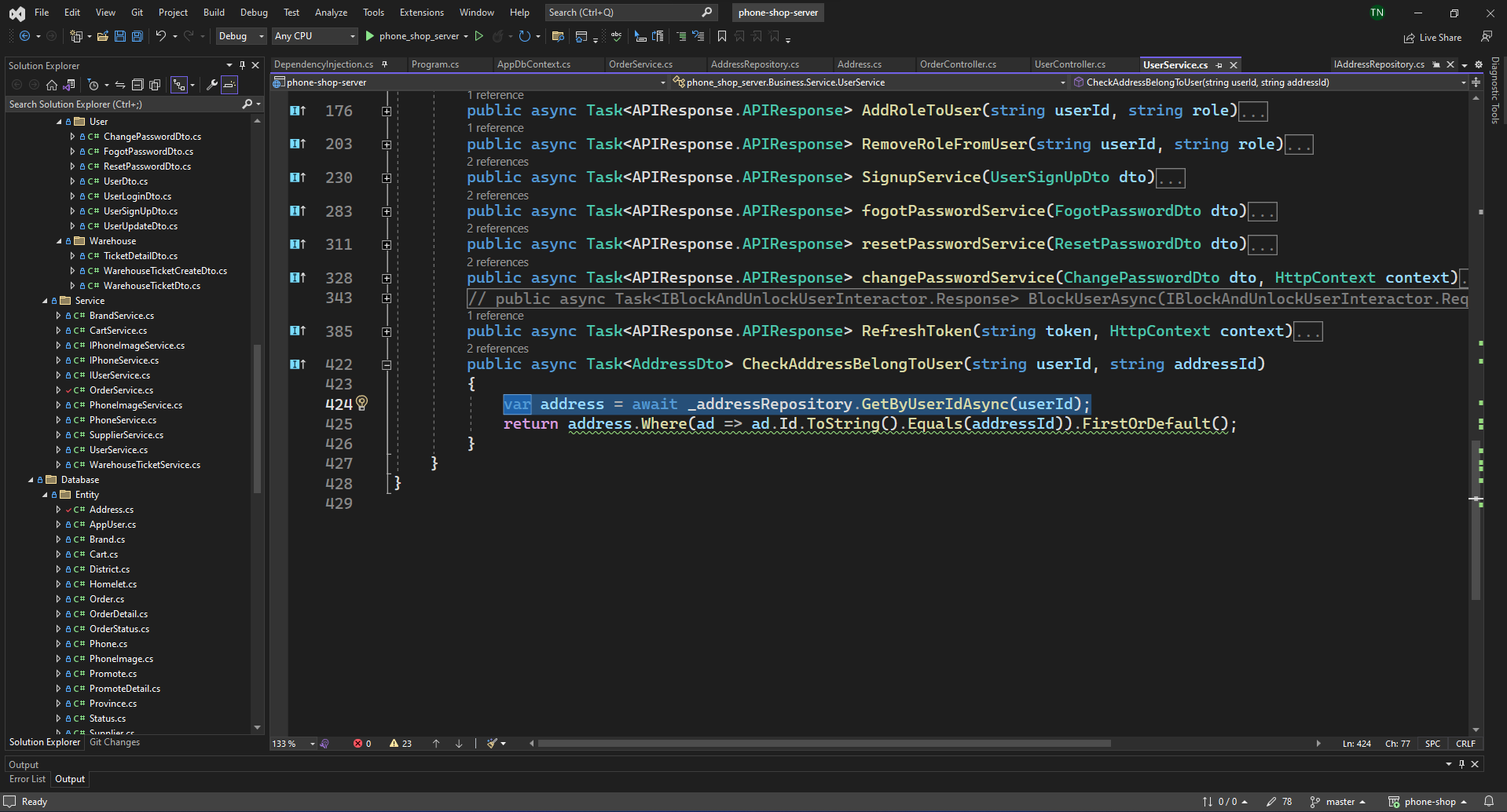
**

Hình 4.. Mô tả dependency injection inject service.

Khi cần sử dụng các phương thức của AddressRepository, chúng ta đơn giản chỉ cần Inject nó vào thông qua phương thức khởi tạo của lớp cần nó.



Hình 4.. Mô tả dependency injection, inject service vào lớp cụ thể.



Hình 4.. Mô tả dependency injection, sử dụng service ở lớp cụ thể.

## 4.3. Xây dựng chương trình phía front-end.

Ứng dụng phía client sử dụng framework Angular và các thư viện được cung cấp liên quan để xây dựng như Angular Material, Bootstrap, RxJs, ...

Ứng dụng sử dụng lớp HttpClient để gọi các API đến back-end để nhận, gửi các dữ liệu liên quan.

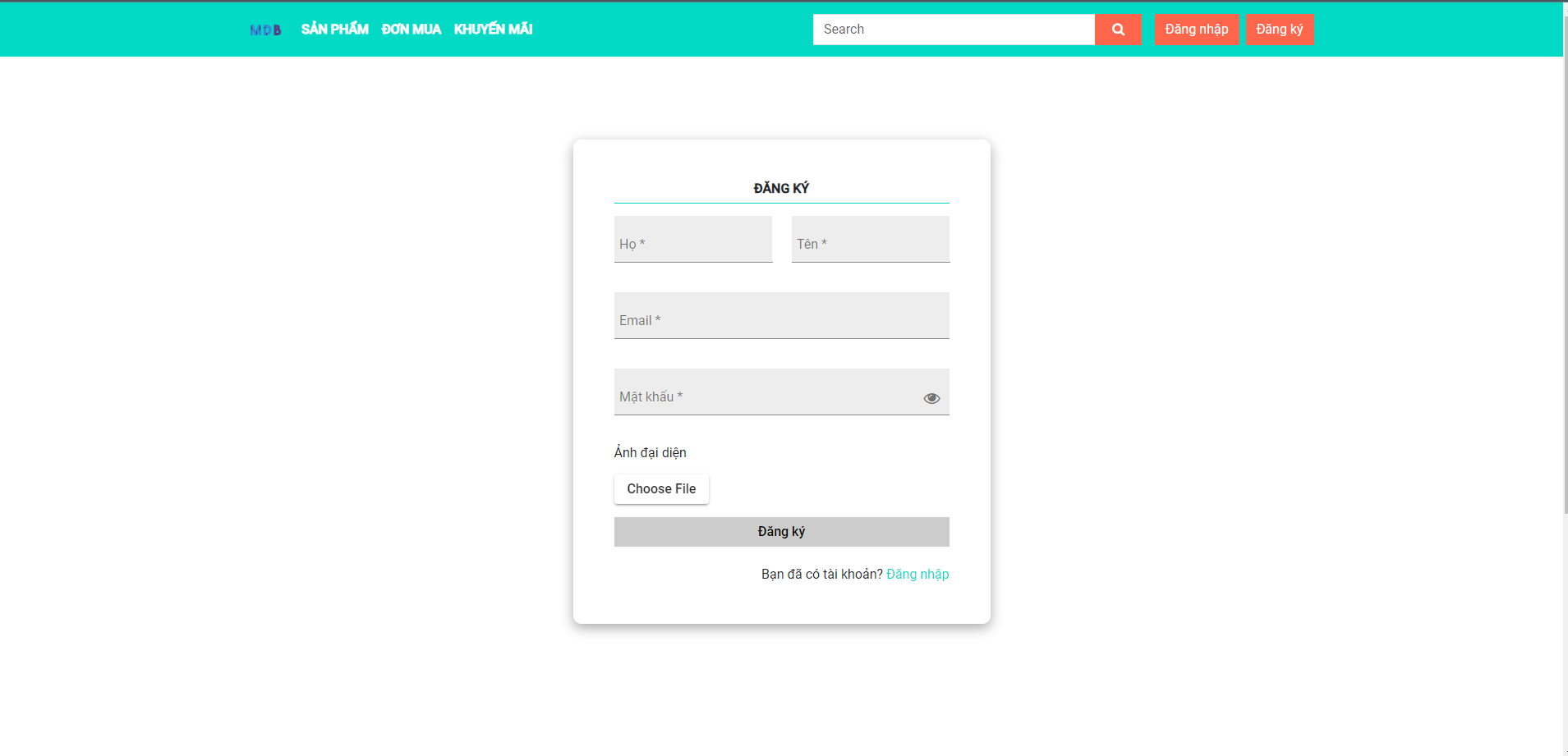
# CHƯƠNG 5: MINH HỌA SẢN PHẨM

## 5.1. Giao diện bán hàng.

Giao diện để khách hàng sử dụng để mua hàng.

### 5.1.1. Giao diện đăng ký tài khoản.

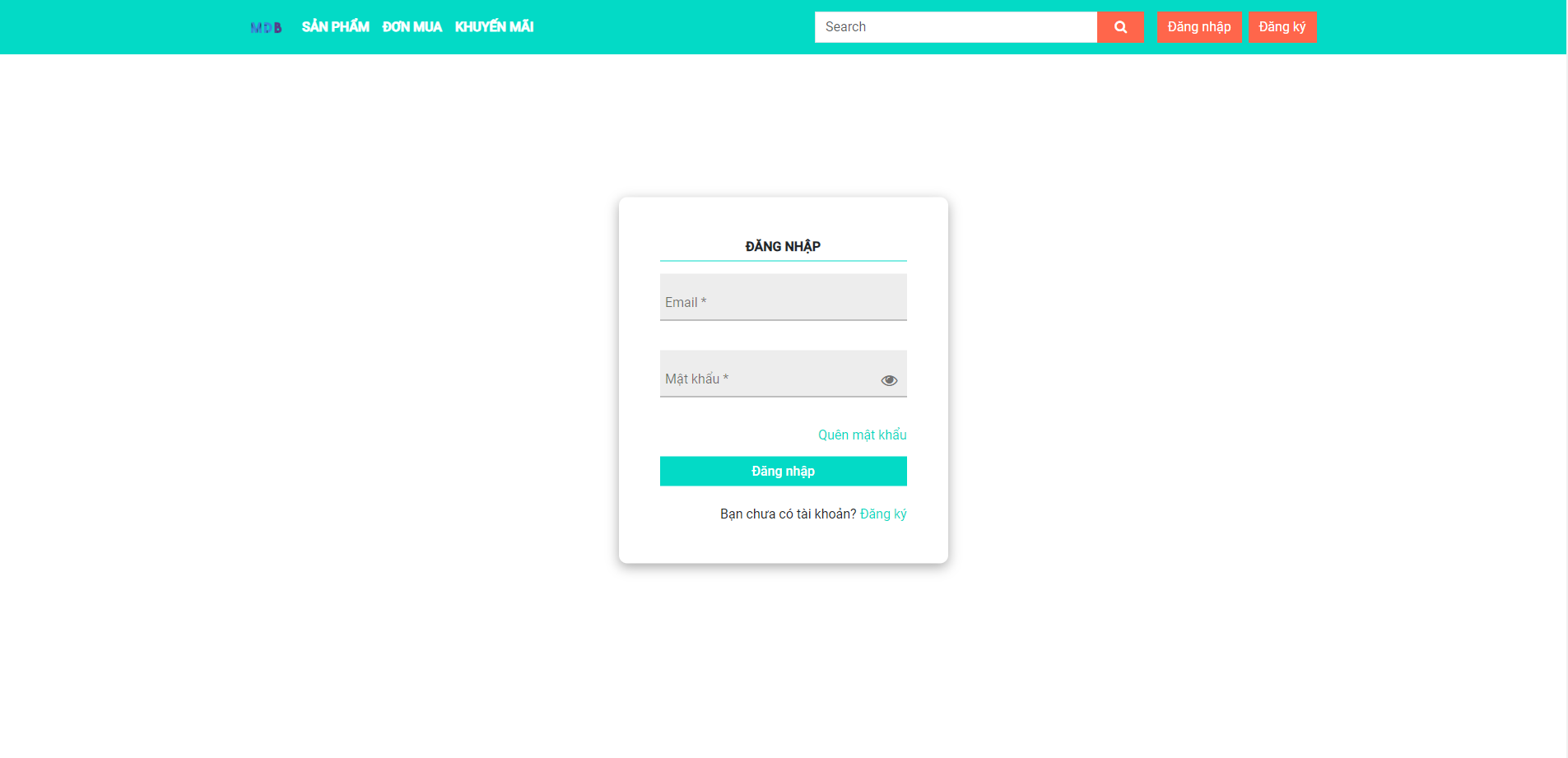
Khách hàng đăng ký tài khoản mới của mình bằng cách nhấn vào nút đăng ký trên website sau đấy nhập các thông tin cần thiết để đăng ký tài khoản mới.



Hình 5.. Giao diện đăng ký khách hàng.

### 5.1.2. Giao diện đăng nhập.

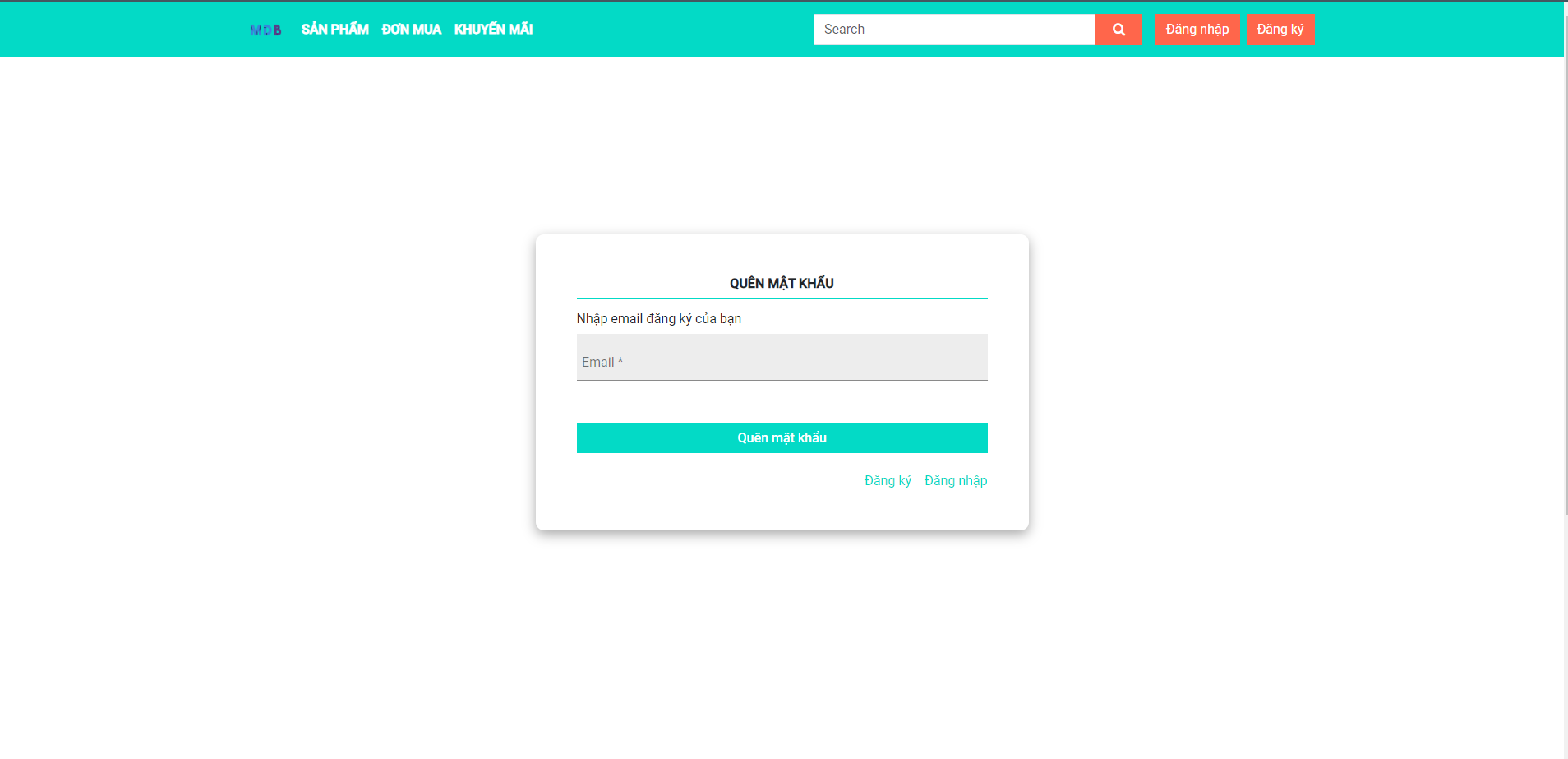
Khách hàng đăng nhập vào website để có thể mua hàng cũng như theo dõi đơn hàng của mình.



Hình 5.. Giao diện khách hàng đăng nhập.

### 5.1.3. Giao diện quên mật khẩu.

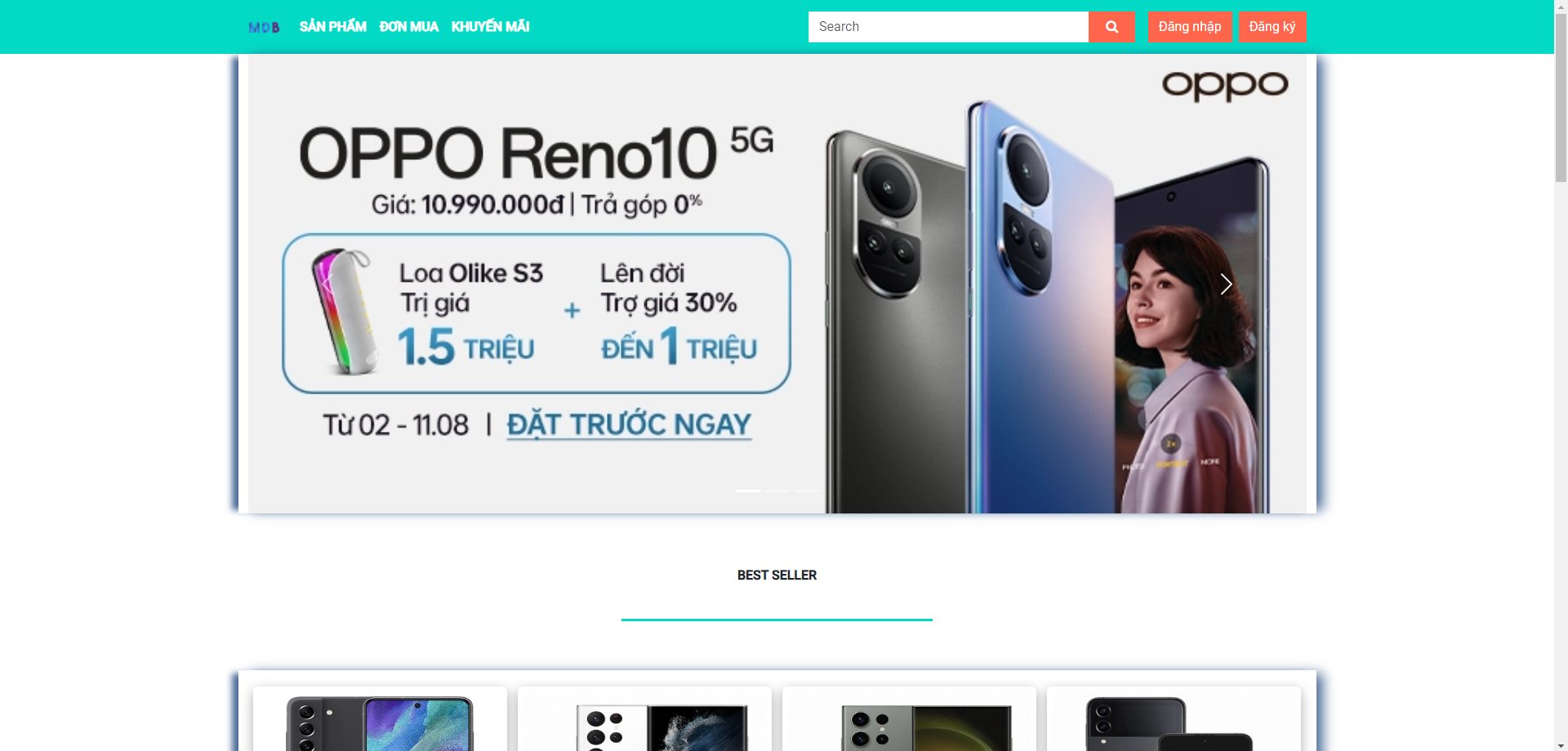
Khách hàng khi quên mật khẩu sẽ được yêu cầu nhập email đăng ký để có thể tạo lại mật khẩu mới. Sau khi nhấn quên mật khẩu, một url đặt lại mật khẩu sẽ được gửi đến email đăng ký tài khoản, sau khi nhấn vào url sẽ chuyển đến trang đặt lại mật khẩu mới.



Hình 5.. Giao diện quên mật khẩu.

### 5.1.4. Giao diện trang chủ.

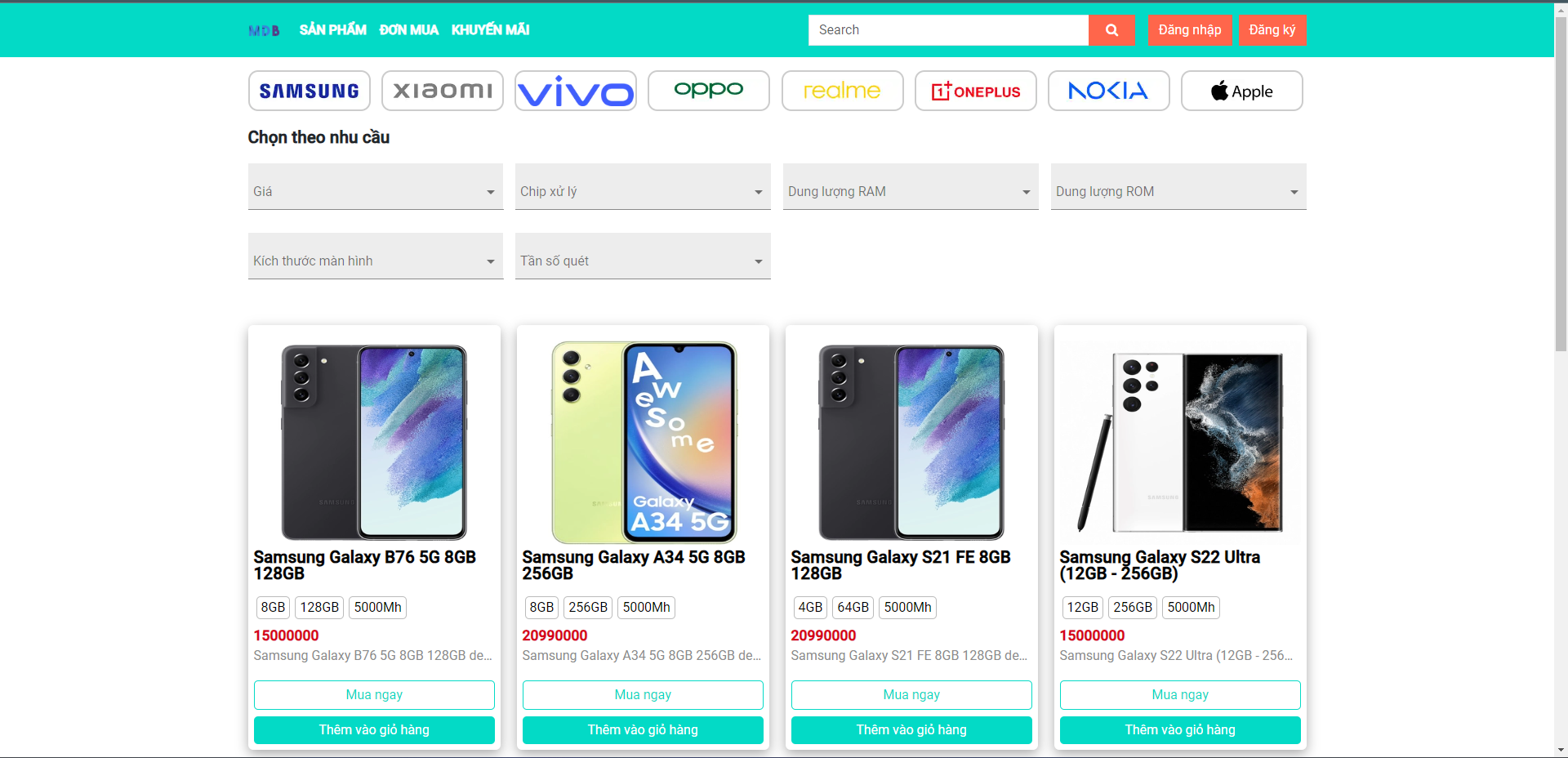
Trang chủ có hiển thị thông tin về các sản phẩm có trong cửa hàng, top sản phẩm bán chạy, ... để khách hàng lựa chọn.



Hình 5.. Giao diện trang chủ.

### 5.1.5. Giao diện trang sản phẩm.

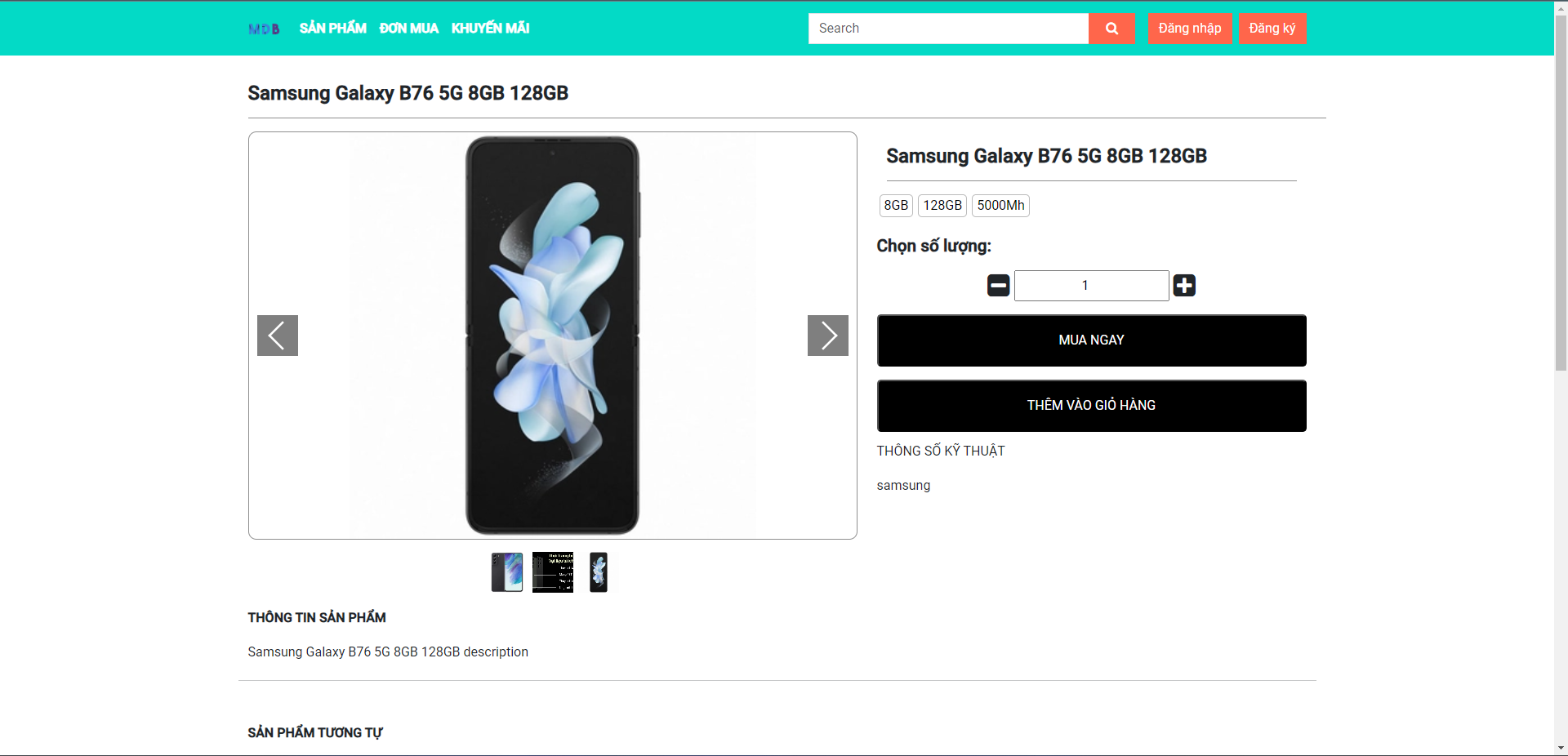
Trang sản phẩm hiển thị các sản phẩm hiện có trong cửa hàng, được phân trang. Khách hàng có thể kết hợp tìm kiếm là lọc theo danh mục để tìm được sản phẩm điện thoại mình muốn tìm. Khách hàng có thể nhấn mua ngay hoặc thêm vào giỏ hàng để mua sản phẩm, hoặc nhấn vào sản phẩm để xem chi tiết sản phẩm.



Hình 5.. Giao diện trang sản phẩm.

### 5.1.6. Giao diện trang chi tiết sản phẩm.

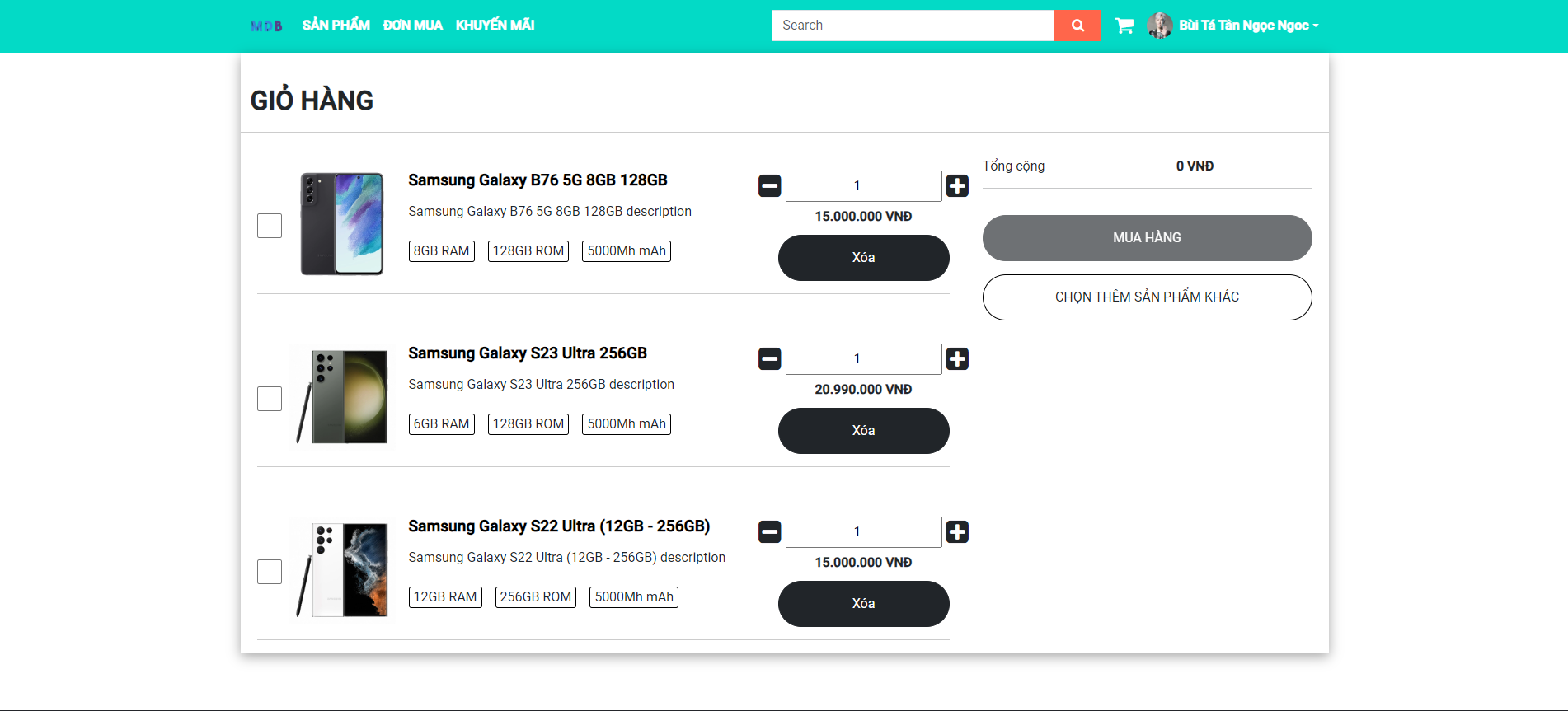
Trang chi tiết sản phẩm hiển thị chi tiết thông tin về sản phẩm cụ thể. Khách hàng có thể chọn số lượng để mua hoặc thêm vào giỏ hàng tại đây. Trang cũng sẽ hiển thị các sản phẩm liên quan cho khách hàng chọn lựa.



Hình 5.. Giao diện trang chi tiết sản phẩm.

### 5.1.7. Giao diện trang giỏ hàng.

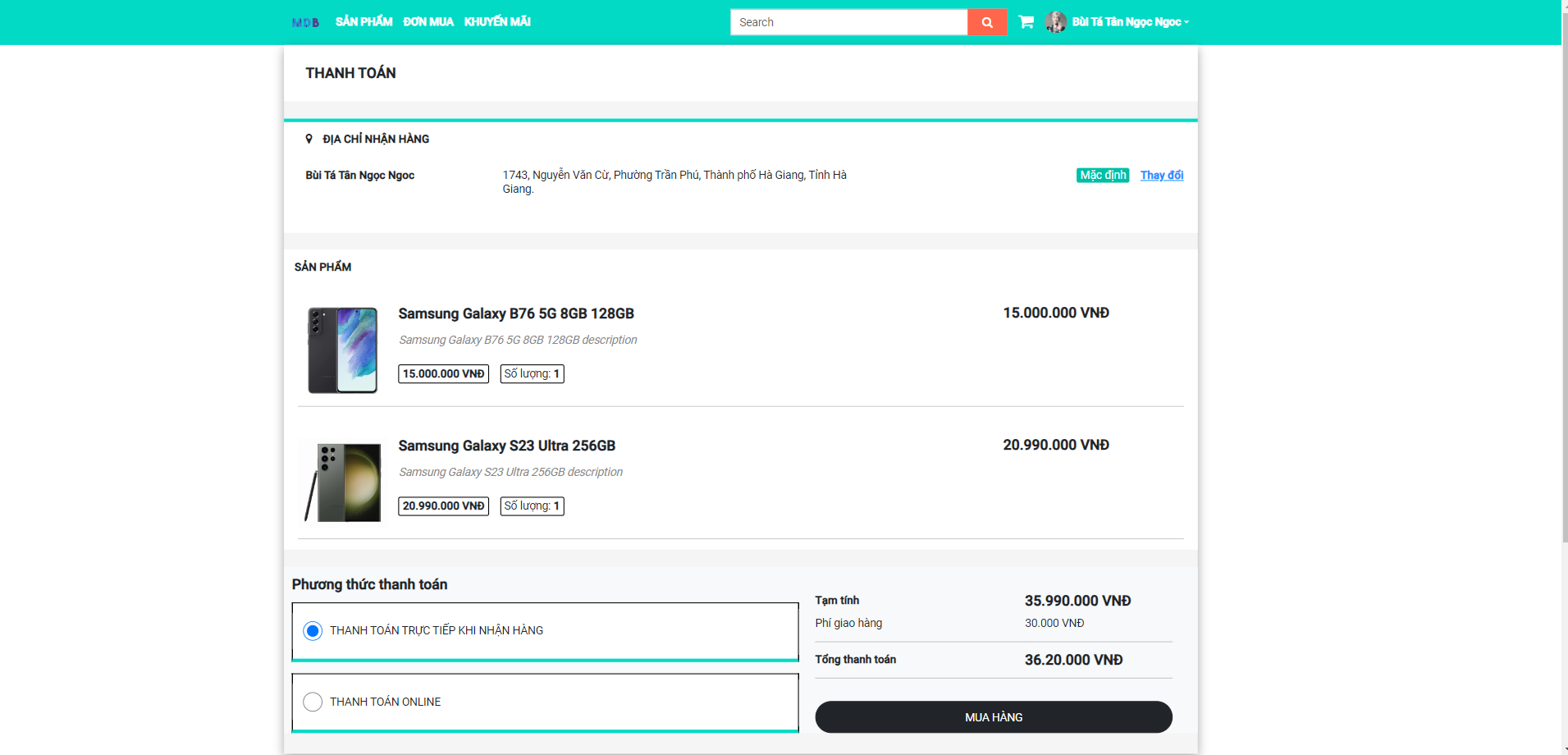
Trang giỏ hàng chứa các sản phẩm được khách hàng thêm vào giỏ. Tại đây khách hàng có thể xóa sản phẩm khỏi giỏ hàng hoặc chọn các sản phẩm muốn mua để mua hàng.



Hình 5.. Giao diện giỏ hàng.

### 5.1.8. Giao diện trang mua hàng.

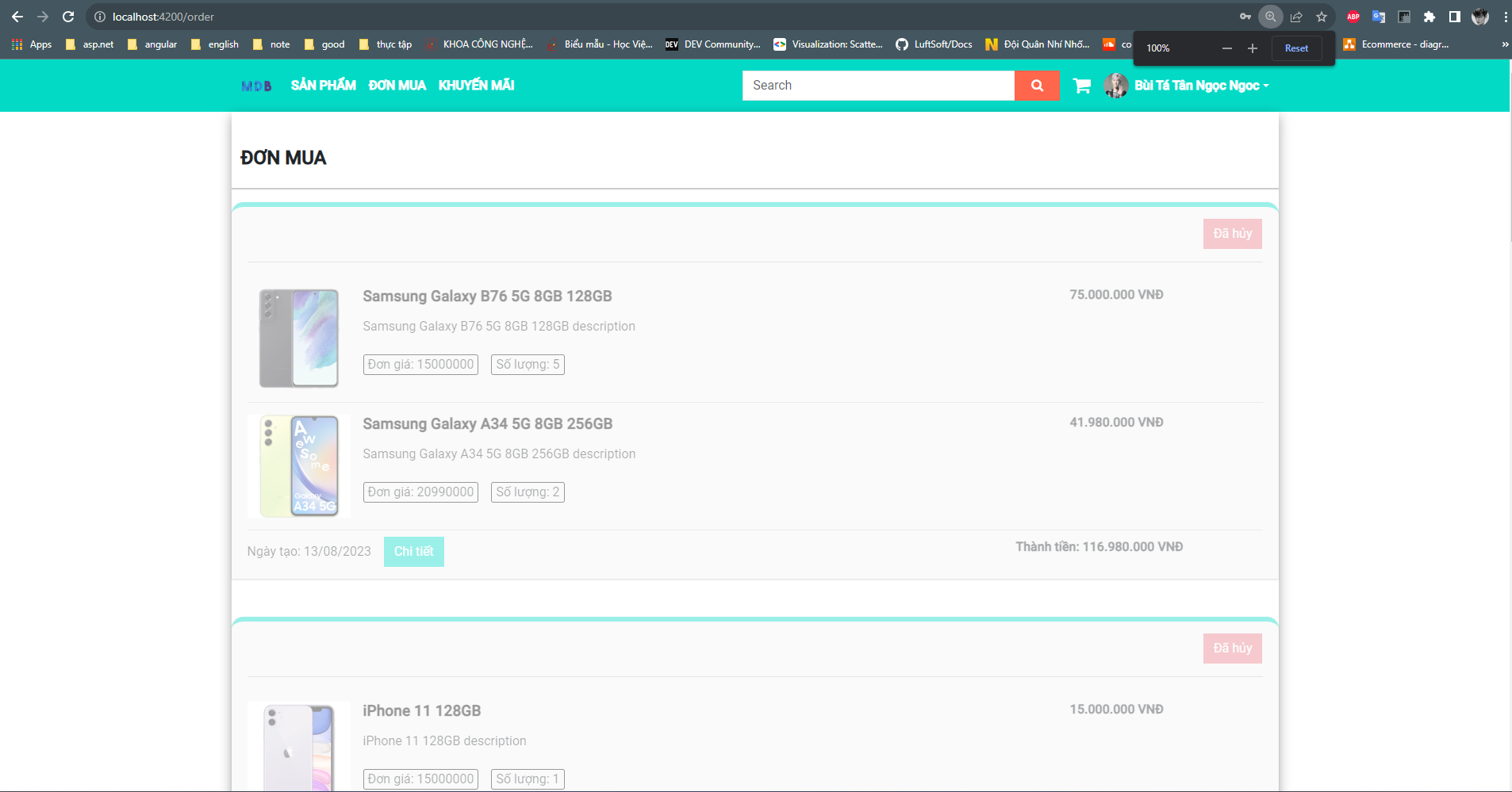
Trang mua hàng chứa các sản phẩm được khách hàng chọn để mua. Tại đây khách hàng có thể chọn địa chỉ nhận hàng để giao và phương thức thanh toán cho đơn hàng.



Hình 5.. Giao diện trang mua hàng.

### 5.1.9. Giao diện trang quản lý đơn hàng.

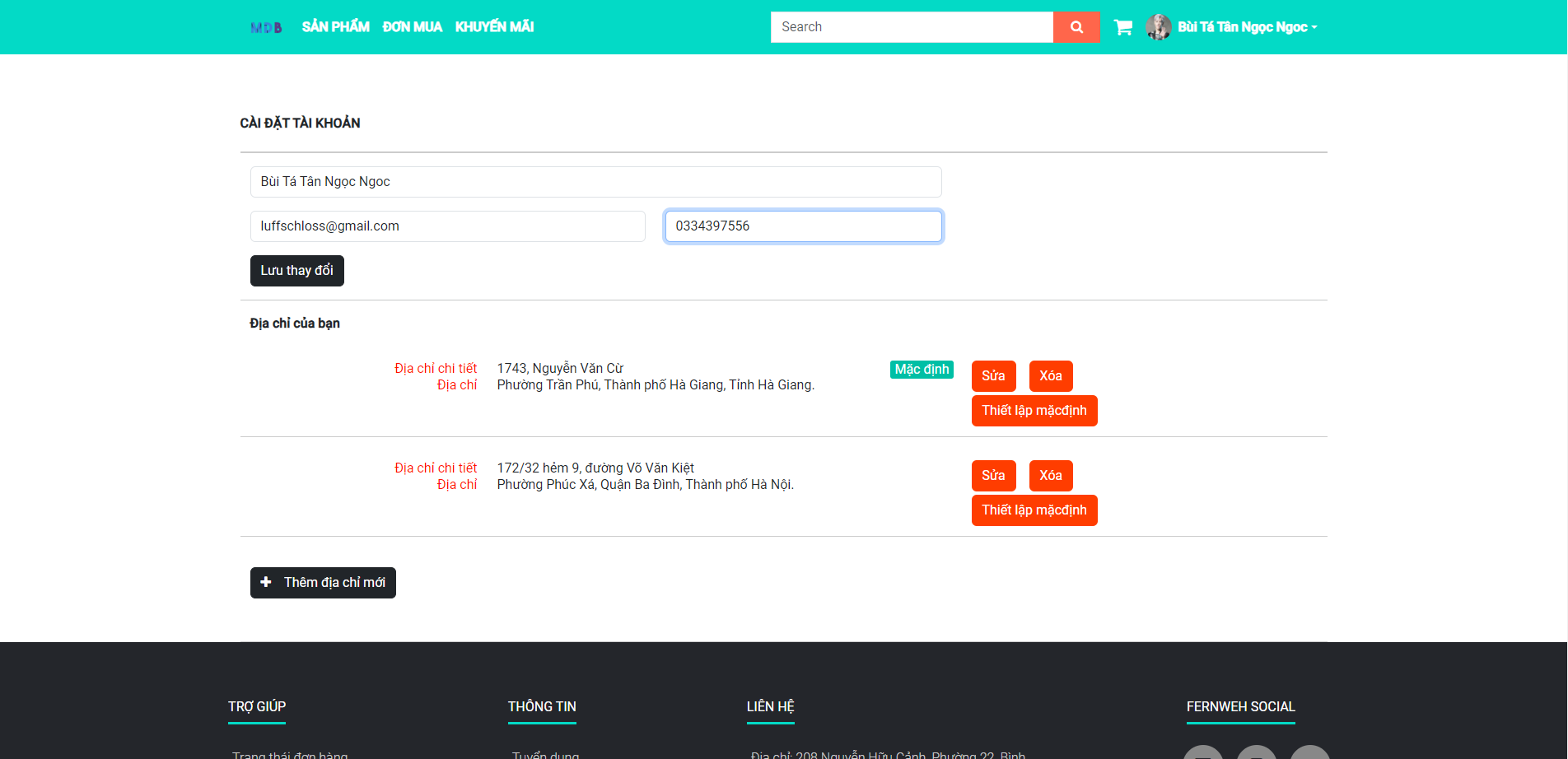
Trang quản lý đơn hàng chứa các đơn hàng đã được khách hàng đặt và các sản phẩm nằm trong đấy. Nếu sản phẩm đang được xác nhận, khách hàng có thể hủy đơn hàng. Sau khi đơn hàng được giao khách hàng có thể xác nhận đã nhận được hàng.



Hình 5.. Giao diện quản lý đơn mua.

### 5.1.10. Giao diện trang quản lý thông tin cá nhân.

Tại trang quản lý thông tin cá nhân, khách hàng có thể cập nhật thông tin cá nhân của mình như tên hoặc số điện thoại. Ngoài ra khách hàng còn có thể quản lý danh sách các địa chỉ nhận hàng mà mình đang có.



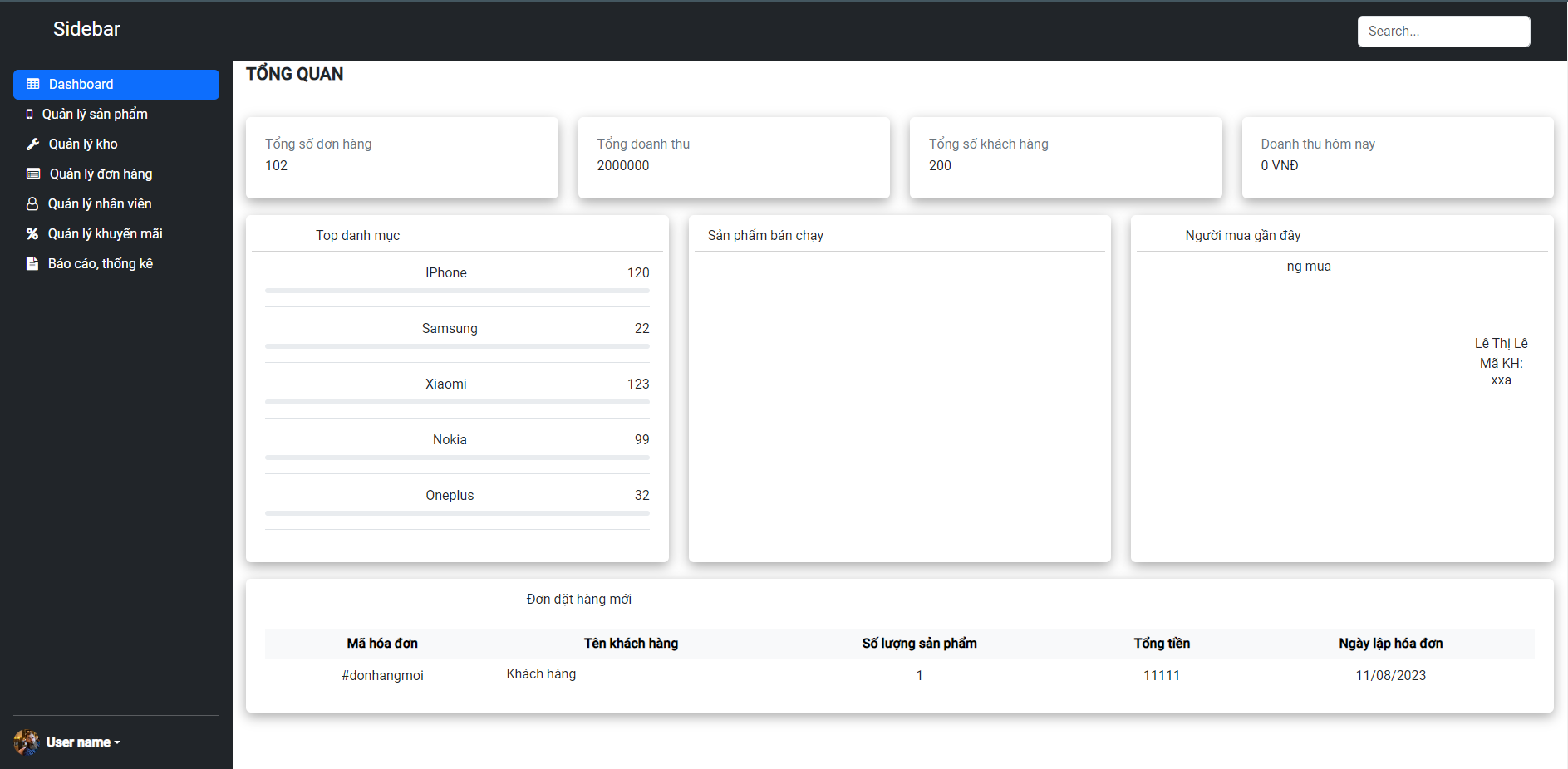
Hình 5.. Giao diện quản lý thông tin cá nhân.

## 5.2. Giao diện trang quản lý.

Giao diện được dùng để nhân viên (STAFF) và quản lý (ADMIN) của cửa hàng quản lý và điều phối hoạt động kinh doang của website.

### 5.2.1. Giao diện trang tổng quan.

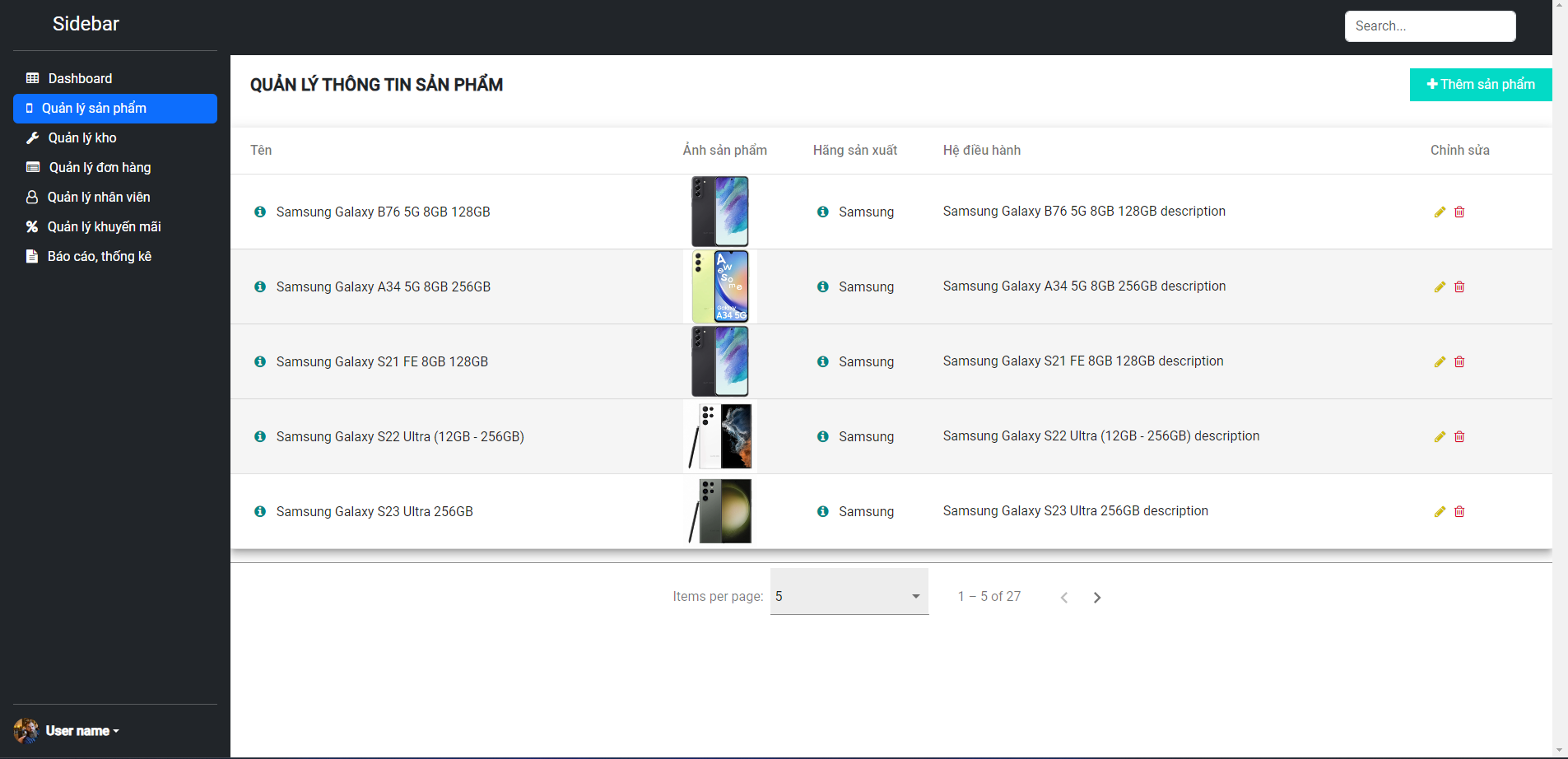
Khái quát tổng quan tình hình kinh doanh của cửa hàng, như sản phẩm bán chạy, doanh thu trong ngày, khách hàng thân thiết, ...



Hình 5.. Giao diện trang tổng quan.

### 5.2.2. Giao diện trang quản lý sản phẩm.

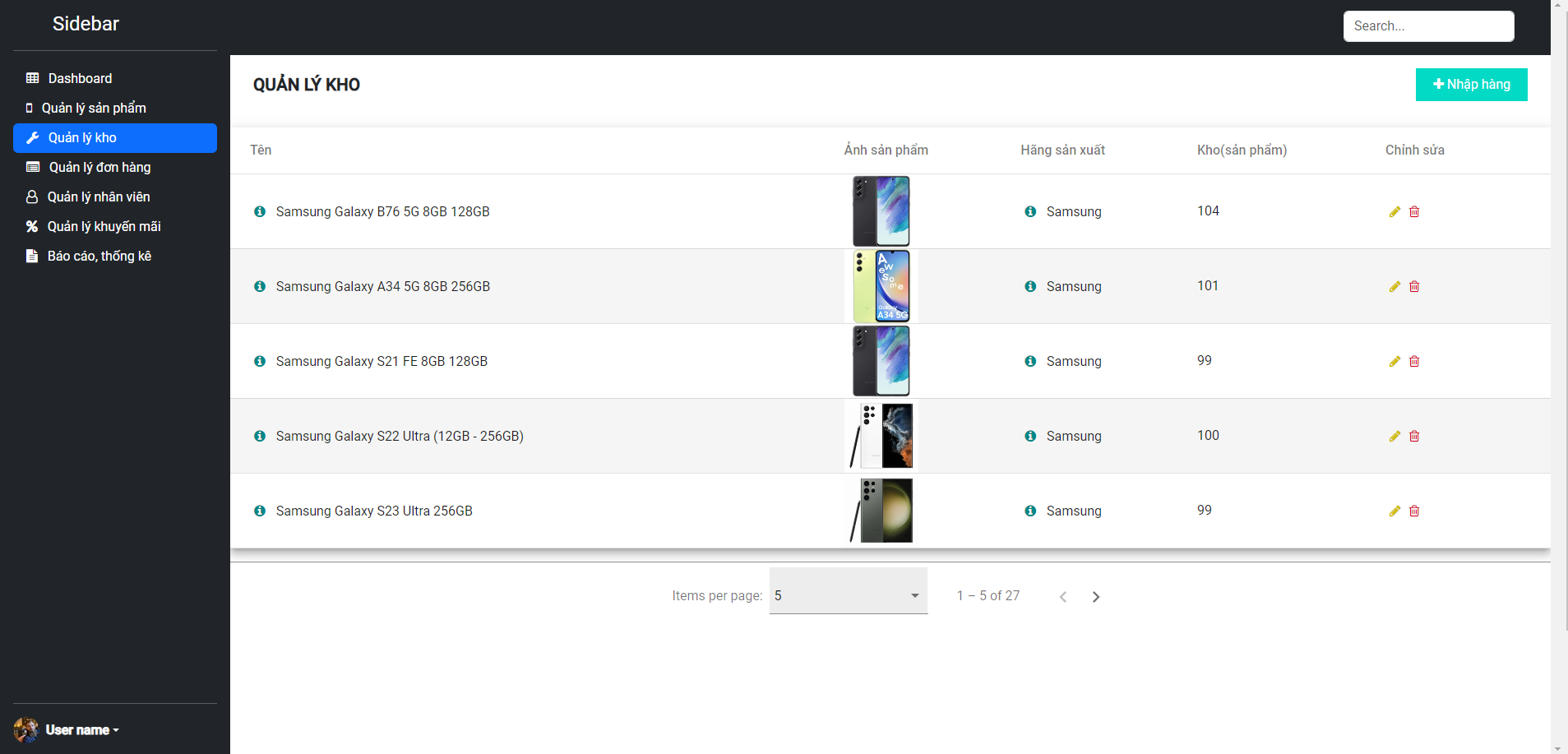
Nhân viên và quản lý có thể quản lý thông tin các sản phẩm hiện có trong cửa hàng, thêm mới và xóa các sản phẩm không còn được bán trong cửa hàng.



Hình 5.. Giao diện trang quản lý sản phẩm.

### 5.2.3. Giao diện trang quản lý kho.

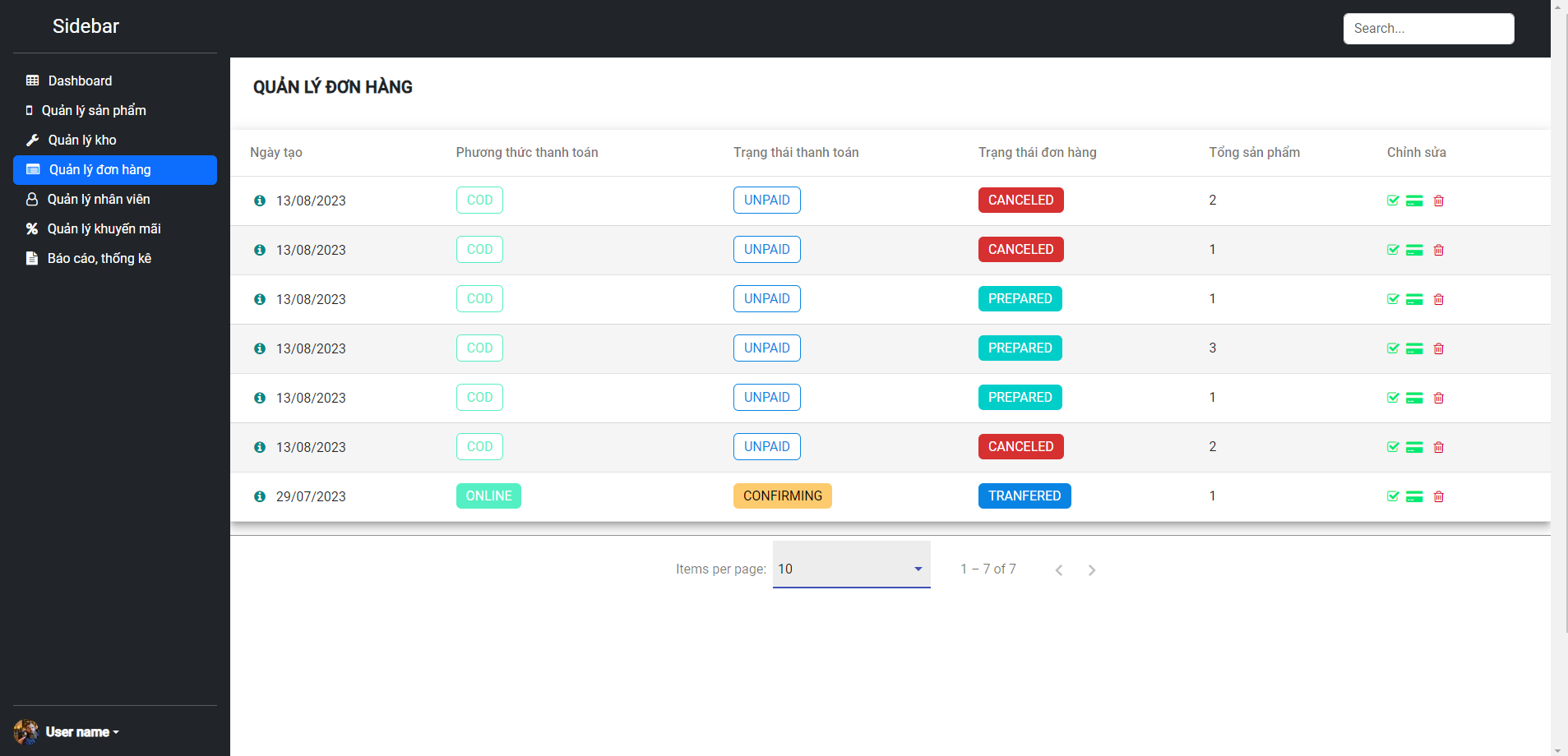
Nhân viên và quản lý có thể quản lý thông tin số lượng sản phẩm của từng mặt hàng hiện có trong cửa hàng, nhập hàng.



Hình 5.. Giao diện trang quản lý kho.

### 5.2.4. Giao diện trang quản lý đơn hàng.

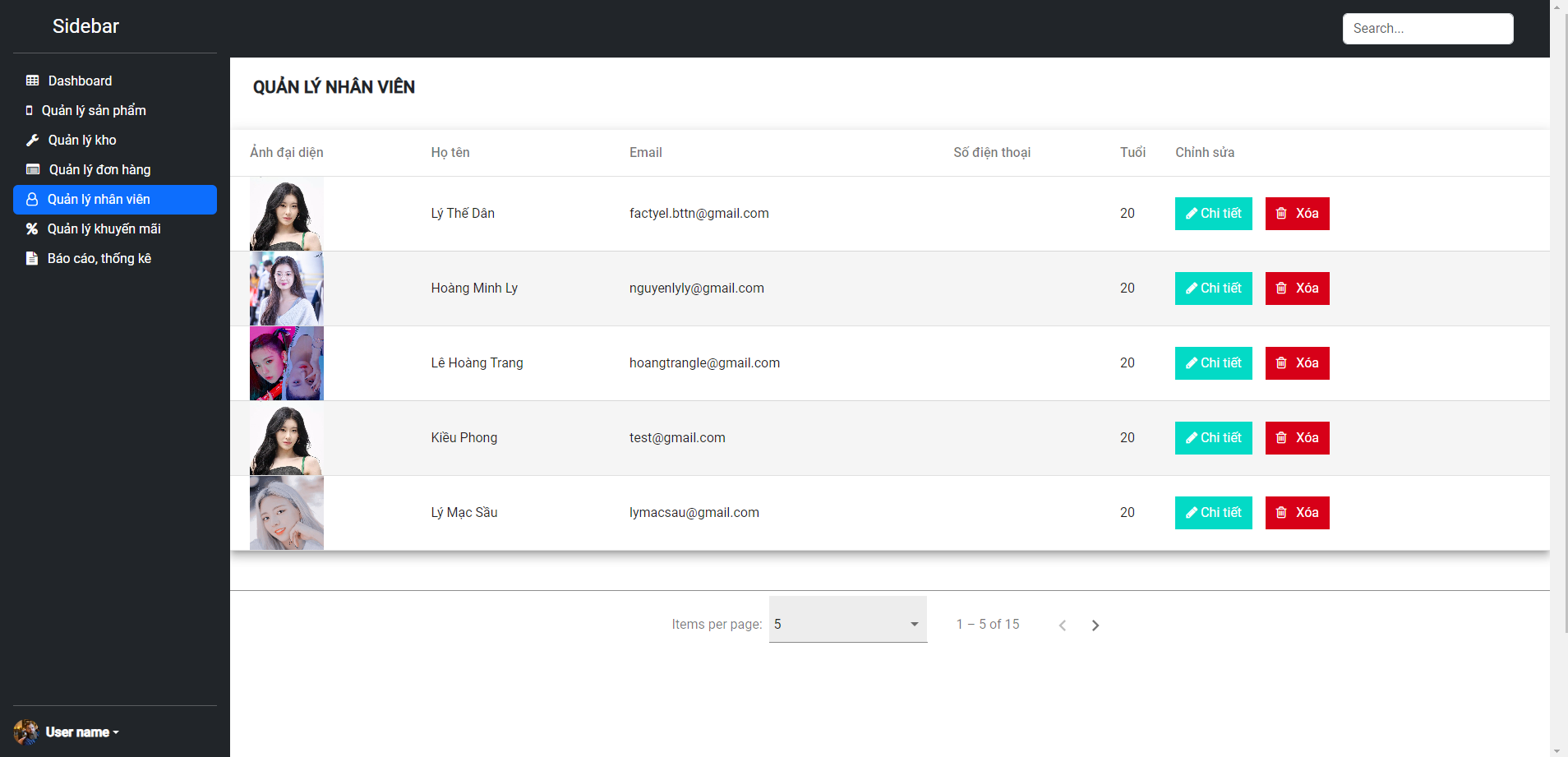
Giao diện để quản lý các đơn hàng trong cửa hàng và tình trạng của chúng. Nhân viên có thể xác nhận tình trạng thanh toán, vận chuyển của đơn hàng tại đây.



Hình 5.. Giao diện trang quản lý đơn hàng.

### 5.2.5. Giao diện trang quản lý nhân viên.

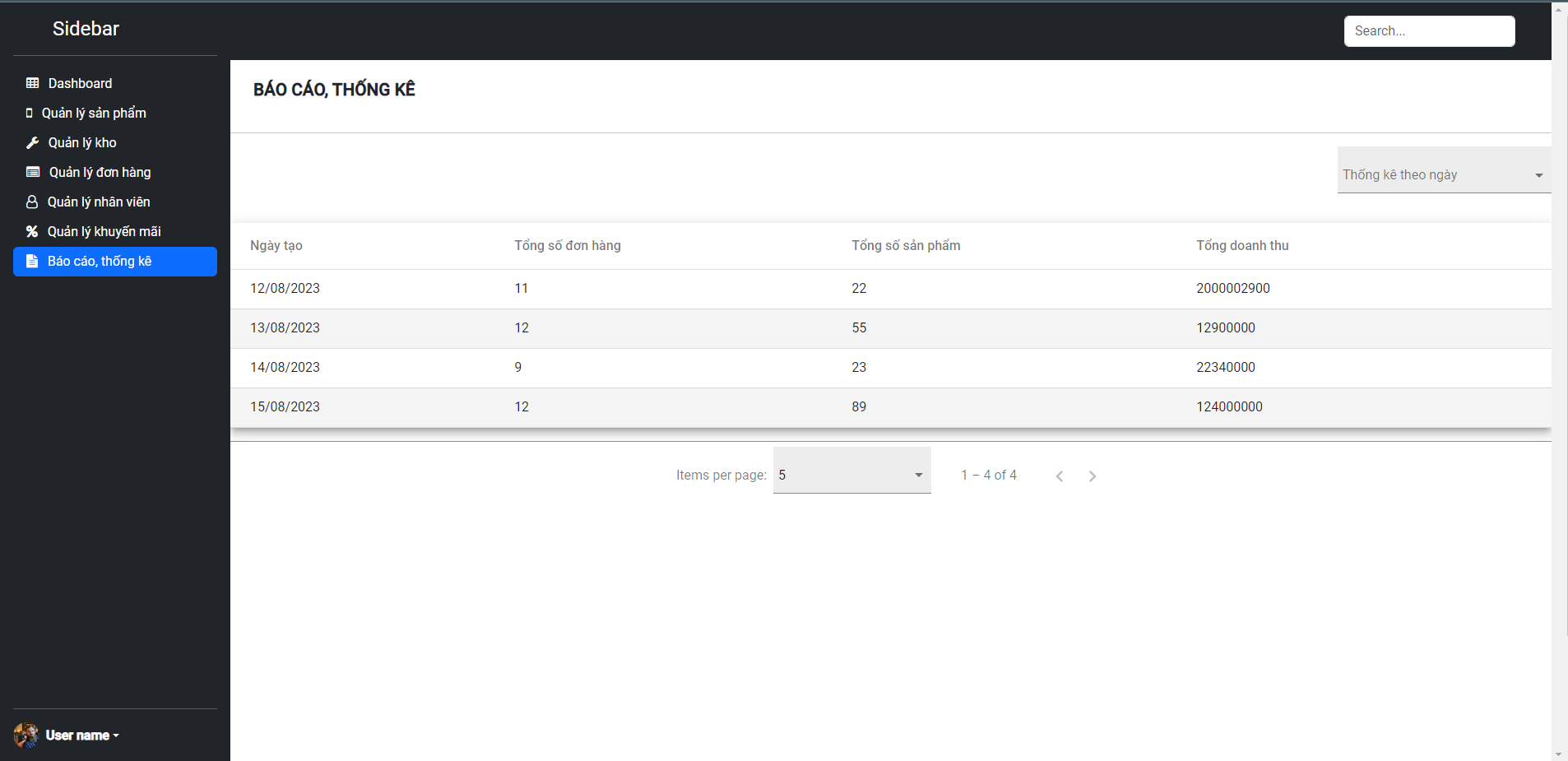
Trang quản lý nhân viên cho phép người quản lý (ADMIN) quản lý các nhân viên hiện đang làm việc trong cửa hàng của mình.



Hình 5.. Giao diện trang quản lý nhân viên.

### 5.2.6. Giao diện trang thống kê.

Trang thống kê cho phép nhân viên (STAFF) và quản lý (ADMIN) xem các thống kê về tình hình kinh doanh của cửa hàng.



Hình 5.. Giao diện trang thống kê.

# CHƯƠNG 6: KẾT LUẬN

## 6.1. Kết luận

Xây dựng một website kinh doanh điện thoại di động tuyến không phải là một đề tài quá mới mẻ, tuy nhiên với sự phát triển của công nghệ hiện nay thì đề tài này có tính ứng dụng cao cũng như giúp trang bị cho em nhiều kiến thức mới khi hoàn thành nó. Điện thoại di động đã và đang là một thị trường phát triển cực kỳ mạnh mẽ ở cả trong nước và thế giới, để giúp các cửa hàng vừa và nhỏ có thể dễ dàng đem các sản phẩm điện thoại của mình đến với khách hàng thông qua website. Em đã thực hiện đề tài “Website kinh doanh mặt hàng điện thoại di động” nhằm đưa ra một giải pháp quản lý phù hợp cho những cửa hàng bán điện thoại đang có mong muốn mở rộng thị trường kinh doanh của mình từ truyền thống sang cả thương mại điện tử, đồng thời website cũng cung cấp các giải pháp quản trị nhân viên cũng như sản phẩm trong cửa hàng, bên cạnh việc tạo một nơi cho khách hàng có thể vào tìm mua sản phẩm yêu thích.

Do thời gian có giới hạn trong kỳ thực tập, đồng thời việc thực hiện đồ án và tham gia thực tập thực tế ở công ty và kinh nghiệm thực tế chưa cao nên sản phẩm hiện chỉ mới đáp ứng được những yêu cầu cơ bản của người sử dụng, việc đưa sản phẩm vào thực tế chắc chắn cần phải nâng cấp thêm nhiều tính năng, hiệu năng, vấn đề bảo mật, ... hơn nữa để đáp ứng nhiều hơn thị hiếu của người dùng ngày càng cao cũng như để phòng chống sự tấn công của các hacker. Trong phạm vi đề tài, em đã cố gắng hoàn thành các module chính được đặt ra trong phiếu giao đề tài. Tuy nhiên vẫn sẽ còn những thiếu sót, rất mong nhận được những góp ý của quý thầy cô để sản phẩm trở nên hoàn thiện hơn.

### 6.1.1. Những kết quả đạt được

Tìm hiểu được nghiệp vụ của một website kinh doanh điện thoại di động trực tuyến.

Tìm hiểu Framework ASP.NET Core, Angular và các thư viện liên quan của hai framework này kết hợp với hệ quản trị cơ sở dữ liệu Microsoft SQL Server để áp dụng xây dựng trang website kinh doanh điện thoại di động trực tuyến.

Khảo sát yêu cầu để nắm được cơ bản quy trình kinh doanh nghiệp vụ của một website kinh doanh mặt hàng điện thoại di động trực tuyến.

Phân tích và thiết kế cơ sở dữ liệu, chức năng hệ thống.

Xây dựng API cho hệ thống thông qua framework ASP.NET Core.

Xây dựng giao diện cho hệ thống bằng Angular framework.

Xây dựng cơ sở dữ liệu cho hệ thống bằng MSSQL Server.

Các chức năng quản trị website:

- Quản lý sản phẩm.

- Quản lý kho

- Quản lý người dùng.

- Quản lý nhân viên

- Quản lý đơn đặt hàng.

- Quản lý thống kê.

Các chức năng dành cho khách hàng:

- Đăng ký, đăng nhập tài khoản thành viên.

- Quên mật khẩu.

- Xem danh sách các sản phẩm hiện đang bán.

- Xem chi tiết sản phẩm.

- Tìm kiếm sản phẩm theo từ khóa kết hợp với danh mục.

- Quản lý giỏ hàng (Thêm, xóa, sửa giỏ hàng).

- Quản lý tài khoản.

- Đặt hàng.

- Quản lý đơn đặt hàng.

### 6.1.2. Những hạn chế

- Còn chưa đa dạng sản phẩm cũng như thông tin của sản phẩm.

- Phương thức thanh toán còn hạn chế, chưa thân thiện với người dùng.

## 6.2. Hướng phát triển của sản phẩm

- Tìm hiểu thêm công nghệ đang xử dụng để tối ưu hóa tốc độ xử lý, nâng cao trải nghiệm của người dùng.

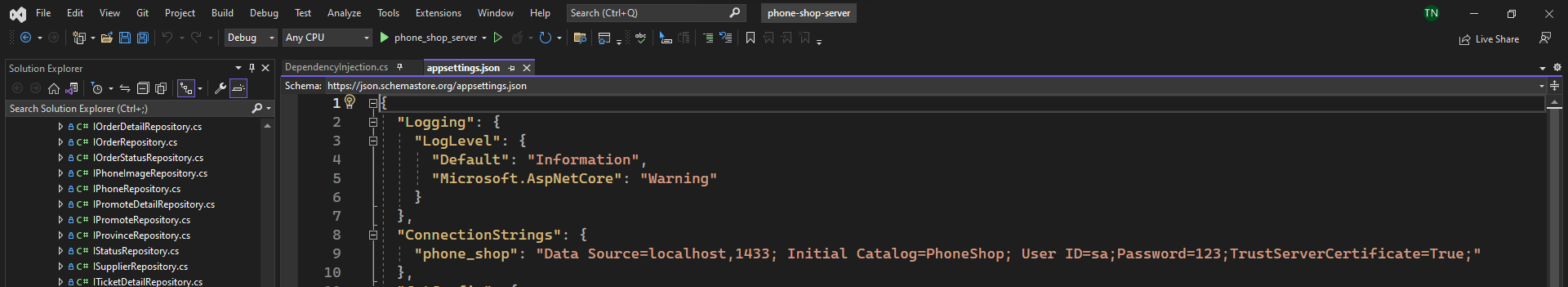
- Đa dạng thêm sản phẩm, danh mục kinh doanh.

- Tích hợp da dạng thêm nhiều phương thức thanh toán cho khách hàng lựa chọn (Ví Momo, ngân hàng, PayPal).

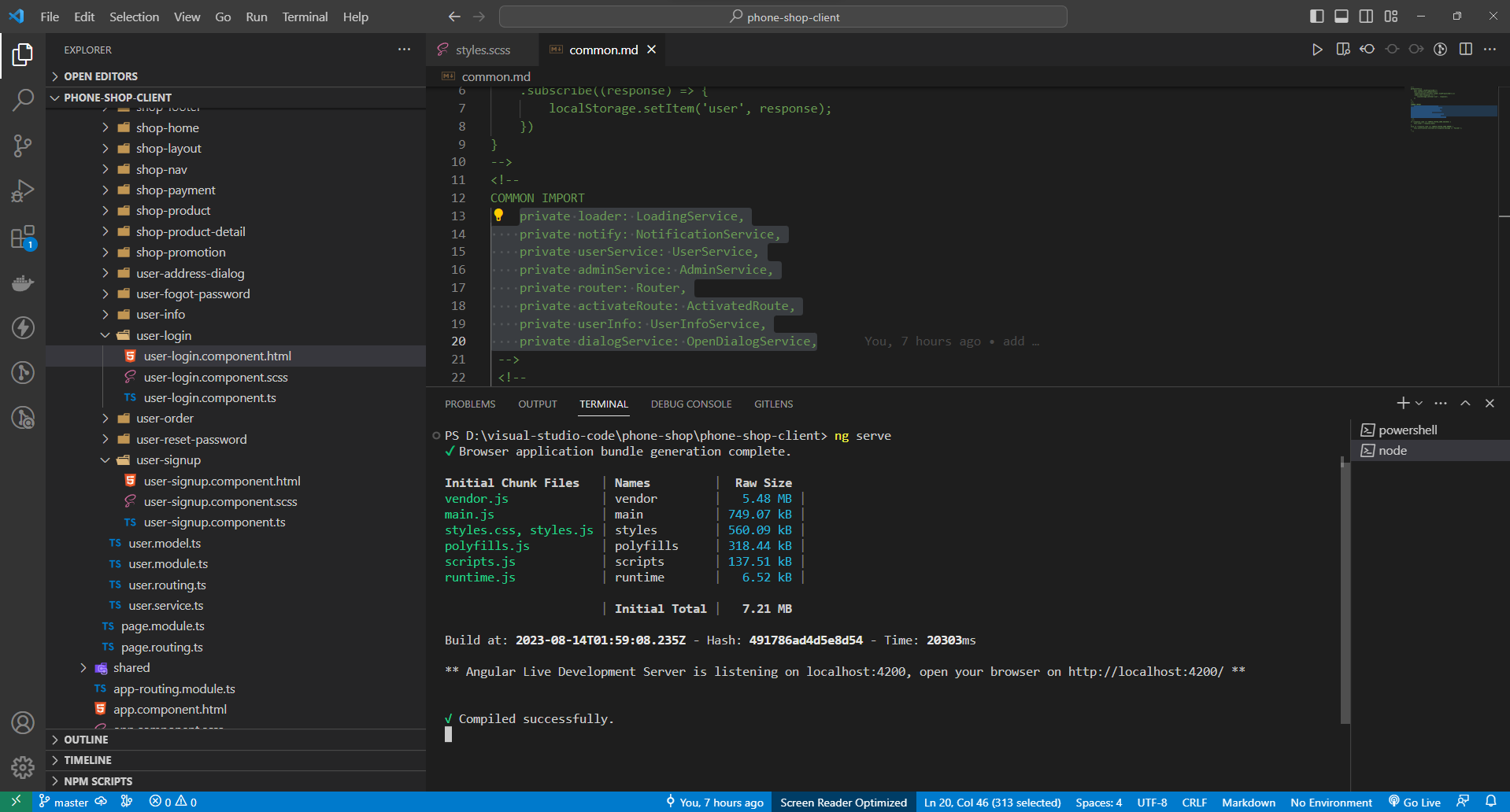
# PHỤ LỤC

(Hướng dẫn cài đặt)

* Ứng dụng phía back-end chạy bằng visual studio, database sẽ được tạo nếu chưa tồn tại, tên và nơi lưu trữ databse được cấu hình trong file config “appsetting.json”



* Để chạy ứng dụng phía front-end, sau khi mở source code trong Visual studio, mở command line bằng tổ hợp phím “Crtl+`(~)” sau đó chạy lệnh “npm i” để tải về các thư viện cần thiết.
* Để chạy chương trình chạy lệnh “ng serve”.



# DANH MỤC TÀI LIỆU THAM KHẢO

**Tiếng Việt:**

**Tiếng Anh:**

**Danh mục các Website tham khảo:**

1. Angular documentation: <https://angular.io/docs>
2. Angular university: <https://angular-university.io/>
3. Angular Material: <https://material.angular.io/>
4. ASP.NET core documentation: <https://learn.microsoft.com/en-us/aspnet/core>
5. C sharp corner: <https://www.c-sharpcorner.com/>
6. Stackoverflow: <https://stackoverflow.com/>
7. SQL Server documentation: <https://learn.microsoft.com/en-us/sql/sql-server/?view=sql-server-ver16>