TRƯỜNG ĐẠI HỌC ĐỒNG THÁP

**KHOA CÔNG NGHỆ VÀ KỸ THUẬT**



0021413407 – Nguyễn Thị Cẩm Trân

**ĐỒ ÁN MÔN HỌC 2**

**XÂY DỰNG WEBSITE**

**QUẢN LÝ CỬA HÀNG TẠP HÓA TRÂN TRÂN**

Mã đề tài: CNPM21-55

CHUYÊN NGÀNH KHOA HỌC MÁY TÍNH

LỚP: ĐHCNTT21B

**Đồng tháp, ngày tháng năm 2024**

TRƯỜNG ĐẠI HỌC ĐỒNG THÁP

**KHOA CÔNG NGHỆ VÀ KỸ THUẬT**



0021413407 – Nguyễn Thị Cẩm Trân

**ĐỒ ÁN MÔN HỌC 2**

**XÂY DỰNG WEBSITE**

**QUẢN LÝ CỬA HÀNG TẠP HÓA TRÂN TRÂN**

Mã đề tài: CNPM21-55

CHUYÊN NGÀNH KHOA HỌC MÁY TÍNH

LỚP: ĐHCNTT21B

**GNHD: ThS. Phạm Huệ Minh**

**Đồng tháp, ngày tháng năm 2024**

**LỜI CẢM ƠN**

*Kính gửi cô Phạm Huệ Minh,*

Em xin gửi lời cảm ơn và lòng biết ơn chân thành đến cô với những hướng dẫn và sự hỗ trợ trân thành trong môn Đồ án môn học 2 với đề tài “Xây dụng webite quản lý cửa hàng Tạp hóa” của em. Đây là một lời tri ân chân thành và biểu hiện lòng biết ơn sâu sắc của em về sự chỉ dẫn và hướng dẫn mà cô đã dành cho em.

Trong suốt quá trình làm việc với cô, em đã học được rất nhiều từ kiến thức chuyên môn đến kỹ năng thực tế. Sự am hiểu sâu sắc của cô về lĩnh vực này đã giúp tôi hiểu rõ hơn về yêu cầu và phạm vi của đề tài, cung cấp cho em cái nhìn toàn diện và các khía cạnh quan trọng cần được xem xét.

Không chỉ là một người hướng dẫn, cô còn là người thầy và nguồn cảm hứng lớn đối với em. Sự tận tâm và sự chia sẻ kiến thức từ cô đã truyền cảm hứng cho em để không ngừng nỗ lực và vượt qua khó khăn trong quá trình nghiên cứu.

Em cũng muốn bày tỏ sự biết ơn đối với cô vì thời gian và cống hiến mà cô đã dành cho em. Sự tận tâm và sự hỗ trợ không ngừng nghỉ từ cô đã giúp em vượt qua những thách thức và hoàn thành đề tài thành công.

Hơn cả những gì em có thể diễn đạt, em hy vọng rằng cô hiểu rằng sự hướng dẫn của cô đã có tác động sâu sắc đến sự phát triển cá nhân và sự thành công của em. Em tự hào và biết ơn vì đã có cô là người thầy của mình.

Một lần nữa, xin chân thành cảm ơn cô vì sự tận tâm, kiên nhẫn và hỗ trợ trong suốt quá trình thực hiện đề tài của em. Em sẽ luôn trân trọng những kiến thức và kinh nghiệm mà cô đã truyền đạt, và em hứa sẽ áp dụng chúng vào công việc và sự nghiệp của mình trong tương lai.

*Đồng Tháp, ngày tháng 12 năm 2024*

**Người thực hiện**

Nguyễn Thị cẩm Trân

**LỜI DẪN**

Đề tài "Xây dựng website quản lý cửa hàng Tạp hóa Trân Trân" nhằm phát triển một hệ thống quản lý hiện đại, giúp tối ưu hóa các quy trình bán hàng, quản lý hàng hóa, và theo dõi doanh thu của cửa hàng tạp hóa. Trong bối cảnh nhu cầu tiêu dùng ngày càng tăng cao và đòi hỏi sự tiện lợi trong quản lý, việc xây dựng một website quản lý không chỉ giúp cửa hàng Tạp hóa tiết kiệm thời gian mà còn nâng cao độ chính xác và hiệu quả trong các hoạt động kinh doanh.

Website này sẽ cung cấp các chức năng như quản lý sản phẩm,loại sản phẩm, quản lý hóa đơn, quản lý nhân viên, quản lý phiếu nhập, quản lý nhà cung cấp và tài khoản cũng như theo dõi doanh thu theo ngày, tháng, và năm. Giao diện người dùng sẽ được thiết kế trực quan, dễ sử dụng và thân thiện với cả nhân viên lẫn chủ cửa hàng. Đồng thời, việc tích hợp các công nghệ hiện đại như ASP .NET MVC, Boostrap, HTML5 giúp hệ thống đạt hiệu suất cao và dễ dàng mở rộng trong tương lai.

Qua đề tài này, người thực hiện sẽ được trải nghiệm quá trình xây dựng một hệ thống quản lý toàn diện, từ thiết kế cơ sở dữ liệu đến triển khai các tính năng phức tạp trên website, góp phần cải thiện quy trình quản lý cho cửa hàng Tạp hóa.

**MỤC LỤC**

[PHẦN MỘT 1](#_Toc184846249)

[1. Đặt vấn đề 1](#_Toc184846250)

[a. Tính cấp thiết 1](#_Toc184846251)

[b. Lý do chọn đề tài 1](#_Toc184846252)

[2. Ý nghĩa lý thuyết, thực tiễn của đề tài 1](#_Toc184846253)

[a. Ý nghĩa lý thuyết 1](#_Toc184846254)

[b. Ý nghĩa thực tiễn 2](#_Toc184846255)

[3. Mục tiêu của đề tài 2](#_Toc184846256)

[4. Phạm vi của đề tài 2](#_Toc184846257)

[5. Các phương pháp nghiên cứu 2](#_Toc184846258)

[a. Phương pháp nghiên cứu lý thuyết 2](#_Toc184846259)

[b. Phương pháp nghiên cứu thực nghiệm 2](#_Toc184846260)

[6. Kế hoạch thực hiện đề tài 3](#_Toc184846261)

[PHẦN HAI: NỘI DUNG 4](#_Toc184846262)

[CHƯƠNG MỘT: CƠ SỞ LÝ THUYẾT 4](#_Toc184846263)

[1.1. Giới thiệu về C# 4](#_Toc184846264)

[1.1.1. C# là gì? 4](#_Toc184846265)

[1.1.2. Đặc điểm ngôn ngữ C# 4](#_Toc184846266)

[1.2. Giới thiệu về Bootstrap 5](#_Toc184846267)

[1.2.1. Bootstrap là gì? 5](#_Toc184846268)

[1.2.2. Tại sao chúng ta nên sử dụng Bootstrap? 5](#_Toc184846269)

[1.2.3. Ưu và nhược điểm của Bootstrap 6](#_Toc184846270)

[1.3. Giới thiệu về CSS3 7](#_Toc184846271)

[1.3.1. CSS3 là gì? 7](#_Toc184846272)

[1.3.2. Các tính năng của CSS3 8](#_Toc184846273)

[1.3.3. Ưu và nhược điểm 8](#_Toc184846274)

[1.4. Giới thiệu về HTML5 9](#_Toc184846275)

[1.4.1. HTML5 là gì? 9](#_Toc184846276)

[1.4.2. Ưu và nhược điểm 10](#_Toc184846277)

[1.5. Giới thiệu vài nét về ASP .NET MVC 10](#_Toc184846278)

[1.5.1. Kiến trúc ASP .NET MVC 11](#_Toc184846279)

[1.5.2. Một số đặc điểm 12](#_Toc184846280)

[1.5.3. Ưu và nhược điểm của ASP .NET MVC 14](#_Toc184846281)

[CHƯƠNG HAI: XÂY DỰNG CHƯƠNG TRÌNH 15](#_Toc184846282)

[2.1. Đặc tả hệ thống 15](#_Toc184846283)

[2.2. Phân tích hệ thống 16](#_Toc184846284)

[2.2.1. Sơ đồ hệ thống 16](#_Toc184846285)

[2.2.2. Sơ đồ Usecase 17](#_Toc184846286)

[2.2.2.1. Sơ đồ Usecase tổng quát 17](#_Toc184846287)

[2.2.2.2. Sơ đồ Usecase thành phần 17](#_Toc184846288)

[a. Sơ đồ Usecase đăng nhập 17](#_Toc184846289)

[b. Sơ đồ Usecase quản lý sản phẩm 18](#_Toc184846290)

[c. Sơ đồ Usecase quản lý hóa đơn 19](#_Toc184846291)

[d. Sơ đồ Usecase quản lý tài khoản 19](#_Toc184846292)

[e. Sơ đồ Usecase quản lý loại sản phẩm 20](#_Toc184846293)

[f. Sơ đồ Usecase quản lý Nhân viên 20](#_Toc184846294)

[g. Sơ đồ Usecase quản lý phiếu nhập 21](#_Toc184846295)

[2.2.3. Sơ đồ hoạt động 21](#_Toc184846296)

[2.2.3.1. Sơ đồ hoạt động đăng nhập 21](#_Toc184846297)

[2.2.3.2. Sơ đồ hoạt động quản lý sản phẩm 22](#_Toc184846298)

[2.2.3.3. Sơ đồ hoạt động quản lý Nhân viên 23](#_Toc184846299)

[2.2.4. Sơ đồ thành phần 24](#_Toc184846300)

[2.3. Thuyết kế CSDL 25](#_Toc184846301)

[2.3.1. Thuyết kế dữ liệu vật lý 25](#_Toc184846302)

[2.3.1.1. Bảng Tài khoản 25](#_Toc184846303)

[2.3.1.2. Bảng Sản phẩm 25](#_Toc184846304)

[2.3.1.3. Bảng Phiếu nhập hàng 26](#_Toc184846305)

[2.3.1.4. Bảng Nhân viên 26](#_Toc184846306)

[2.3.1.5. Bảng Nhà cung cấp 26](#_Toc184846307)

[2.3.1.6. Bảng Loại sản phẩm 27](#_Toc184846308)

[2.3.1.7. Bảng Hóa đơn 27](#_Toc184846309)

[2.3.1.8. Bảng Chi tiết phiếu nhập hàng 28](#_Toc184846310)

[2.3.1.9. Bảng Chi tiết hóa đơn 28](#_Toc184846311)

[2.3.1.10. Sơ đồ lớp hệ thống 29](#_Toc184846312)

[2.4. Một số giải thuật tiêu biểu 29](#_Toc184846313)

[2.4.1. Giải thuật tìm kiếm theo tên sản phẩm 29](#_Toc184846314)

[2.4.1. Giải thuật thanh toán hóa đơn 29](#_Toc184846315)

[2.5. Phát triển hệ thống 30](#_Toc184846316)

[2.5.1. Cộng cụ và môi trường phát triển 30](#_Toc184846317)

[2.5.2. Một số giao diện chính của chương trình 31](#_Toc184846318)

[2.5.2.1. Giao diện trang chủ 31](#_Toc184846319)

[2.5.2.2. Giao diện quản lý sản phẩm 32](#_Toc184846320)

[2.5.2.3. Giao diện quản lý hóa đơn 32](#_Toc184846321)

[PHẦN BA: KẾT LUẬN 34](#_Toc184846322)

[1. Kết quả đạt được 34](#_Toc184846323)

[2. Hạn chế 34](#_Toc184846324)

[3. Hướng phát triển 34](#_Toc184846325)

**DANH MỤC HÌNH**

[Hình 2.1. Sơ đồ hệ thống 16](#_Toc184846326)

[Hình 2.2. Hình Usecase tổng quát 17](#_Toc184846327)

[Hình 2.3. Hình sơ đồ Usecase đăng nhập 18](#_Toc184846328)

[Hình 2.4. Hình sơ đồ Usecase quản lý sản phẩm 18](#_Toc184846329)

[Hình 2.5. Hình sơ đồ Usecase quản lý đơn hàng 19](#_Toc184846330)

[Hình 2.6. Hình sơ đồ Usecase quản lý tài khoản 19](#_Toc184846331)

[Hình 2.7. Hình sơ đồ Usecase quản lý loại sản phẩm 20](#_Toc184846332)

[Hình 2.8. Hình sơ đồ Usecase quản lý nhân viên 20](#_Toc184846333)

[Hình 2.9. Hình sơ đồ usecase quản lý phiếu nhập 21](#_Toc184846334)

[Hình 2.10. Hình sơ đồ hoạt động đăng nhập 22](#_Toc184846335)

[Hình 2.11. Hình sơ đồ hoạt động quản lý sản phẩm 23](#_Toc184846336)

[Hình 2.12. Hình sơ đồ hoạt động quản lý nhân viên 24](#_Toc184846337)

[Hình 2.13. Hình sơ đồ thành phần 24](#_Toc184846338)

[Hình 2.14. Sơ đồ lớp hệ thống 29](#_Toc184846339)

[Hình 2.15. Giao diện trang chủ 31](#_Toc184846340)

[Hình 2.16. Hình giao diện trang quản lý sản phẩm 32](#_Toc184846341)

[Hình 2.17. Giao diện quản lý hóa đơn 33](#_Toc184846342)

**DANH MỤC BẢNG**

[Bảng 1. Bảng kế hoạch thực hiện đề tài 3](#_Toc184846343)

[Bảng 2.1. Bảng tài khoản 25](#_Toc184846344)

[Bảng 2.2. Bảng sản phẩm 25](#_Toc184846345)

[Bảng 2.3. Bảng phiếu nhập hàng 26](#_Toc184846346)

[Bảng 2.4. Bảng nhân viên 26](#_Toc184846347)

[Bảng 2.5. Bảng nhà cung cấp 27](#_Toc184846348)

[Bảng 2.6. Bảng loại sản phẩm 27](#_Toc184846349)

[Bảng 2.7. Bảng hóa đơn 27](#_Toc184846350)

[Bảng 2.8. Bảng chi tiết phiếu nhập 28](#_Toc184846351)

[Bảng 2.9. Bảng chi tiết hóa đơn 28](#_Toc184846352)

PHẦN MỘT

--- oOo ---

1. Đặt vấn đề

a. Tính cấp thiết

Trong thời đại công nghệ số phát triển nhanh chóng, nhu cầu tự động hóa và quản lý hiệu quả tại các cửa hàng tạp hóa ngày càng trở nên quan trọng. Đặc biệt, với các cửa hàng có số lượng lớn hàng hóa và giao dịch hằng ngày, việc quản lý thủ công không chỉ mất nhiều thời gian mà còn tiềm ẩn nhiều sai sót. Cửa hàng cần phải tối ưu hóa quy trình vận hành để phục vụ khách hàng tốt hơn, quản lý hàng hóa dễ dàng hơn và theo dõi doanh thu chính xác hơn. Chính vì vậy, việc xây dựng một hệ thống website quản lý hiệu quả sẽ giúp cửa hàng nâng cao năng lực cạnh tranh, tiết kiệm thời gian và chi phí vận hành.

b. Lý do chọn đề tài

Đề tài "Xây dựng website quản lý cửa hàng tạp hóa Trân Trân" được chọn với mong muốn tạo ra một công cụ hỗ trợ hữu ích cho việc quản lý, từ theo dõi sản phẩm, quản lý hóa đơn, nhân viên, quản lý nhà cung cấp, quản lý phiếu nhập, quản lý tài khoản, đến giám sát doanh thu hàng ngày. Hệ thống này không chỉ đáp ứng nhu cầu thực tế của cửa hàng tạp hóa mà còn giúp người thực hiện rèn luyện các kỹ năng phát triển phần mềm quản lý, từ khâu phân tích yêu cầu đến triển khai thực tế. Qua đề tài này, người thực hiện mong muốn đóng góp vào việc hiện đại hóa quy trình quản lý cho các cửa hàng tạp hóa, đồng thời tích lũy kinh nghiệm thực tế và chuyên sâu hơn trong lĩnh vực phát triển phần mềm quản lý.

2. Ý nghĩa lý thuyết, thực tiễn của đề tài

a. Ý nghĩa lý thuyết

Đề tài "Xây dựng website quản lý cửa hàng tạp hóa Trân Trân" đóng góp vào việc nghiên cứu và ứng dụng các mô hình quản lý thông tin trong kinh doanh bán lẻ, đặc biệt là trong lĩnh vực tạp hóa. Đề tài này giúp hệ thống hóa kiến thức về thiết kế cơ sở dữ liệu, tổ chức và xử lý dữ liệu liên quan đến quản lý sản phẩm, khách hàng, hóa đơn, và nhân viên. Bên cạnh đó, quá trình xây dựng hệ thống còn củng cố hiểu biết về các phương pháp xây dựng và triển khai phần mềm quản lý, cùng các phương thức tối ưu hóa hiệu quả của hệ thống thông tin cho doanh nghiệp bán lẻ.

b. Ý nghĩa thực tiễn

Hệ thống quản lý này mang lại nhiều lợi ích thiết thực cho cửa hàng tạp hóa Trân Trân, giúp tối ưu hóa quy trình quản lý hàng hóa, tiết kiệm thời gian và nguồn lực, đồng thời tăng cường độ chính xác trong việc theo dõi doanh thu và quản lý kho hàng. Hệ thống này còn hỗ trợ việc phân quyền và quản lý nhân viên hiệu quả hơn, giúp chủ cửa hàng dễ dàng kiểm soát các hoạt động kinh doanh. Ngoài ra, ứng dụng này có thể mở rộng để áp dụng cho nhiều cửa hàng tạp hóa khác, góp phần nâng cao năng lực cạnh tranh cho các doanh nghiệp bán lẻ trong bối cảnh kinh doanh hiện đại.

3. Mục tiêu của đề tài

Mục tiêu của đề tài "Xây dựng website quản lý cửa hàng tạp hóa Trân Trân" là xây dựng thành công một hệ thống quản lý toàn diện, thân thiện với người dùng nhằm hỗ trợ cửa hàng trong việc quản lý hàng hóa, nhân viên và theo dõi doanh thu một cách hiệu quả và chính xác.

4. Phạm vi của đề tài

Đề tài "Xây dựng website quản lý cửa hàng tạp hóa Trân Trân" tập trung vào việc thiết kế và phát triển các chức năng cơ bản và cần thiết để quản lý hoạt động của một cửa hàng tạp hóa Trân Trân và có khả năng mở rộng trong tương lai.

5. Các phương pháp nghiên cứu

a. Phương pháp nghiên cứu lý thuyết

Phương pháp nghiên cứu lý thuyết tập trung vào việc thu thập và phân tích tài liệu để xây dựng nền tảng kiến thức cho đề tài. Cụ thể, người thực hiện sẽ tìm hiểu các tài liệu về quản lý cửa hàng, quản lý dữ liệu bán lẻ, và thiết kế hệ thống thông tin để áp dụng vào việc xây dựng website quản lý cửa hàng tạp hóa. Bên cạnh đó, phương pháp này còn bao gồm việc nghiên cứu về các kiến trúc phần mềm, mô hình MVC, hệ quản trị cơ sơ dữ liệu SQL Server và các phương pháp tối ưu hóa hiệu suất hệ thống nhằm đảm bảo website hoạt động hiệu quả và thân thiện với người dùng.

b. Phương pháp nghiên cứu thực nghiệm

Phương pháp nghiên cứu thực nghiệm sẽ được thực hiện thông qua quá trình xây dựng và thử nghiệm hệ thống. Thực hiện triển khai các chức năng quản lý sản phẩm, hóa đơn, nhân viên, và doanh thu dựa trên yêu cầu thực tế của cửa hàng tạp hóa. Sau đó, hệ thống sẽ được kiểm thử để đảm bảo các chức năng hoạt động chính xác và đáp ứng yêu cầu đề ra. Quá trình thử nghiệm cũng bao gồm việc thu thập phản hồi từ người dùng, như nhân viên và chủ cửa hàng, để cải tiến giao diện và tính năng, giúp hệ thống dễ sử dụng và phù hợp với nhu cầu thực tế.

6. Kế hoạch thực hiện đề tài

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Thời gian thực hiện** | **Nội dung thực hiện** | **Người thực hiện** |
| 1 | 7/10/2024 – 13/10/2024 | Tìm hiểu, nghiên cứu tài liệu để nắm vững cơ sở lý thuyết | Nguyễn Thị Cẩm Trân |
| 2 | 14/10/2024 – 28/10/2024 | Nghiên cứu nghiệp vụ, thiết kế và phân tích đề tài.  Xây dựng cơ sở dữ liệu |
| 3 | 29/10/2024 – 27/11/2024 | Thiết kế và phát triển hệ thống |
| 4 | 28/11/2024 – 7/12/2024 | Bước đầu hoàn chỉnh hệ thống |
| 5 | 8/12/2024 - 12/12/2024 | Kiểm thử và gỡ lỗi hệ thống |
| 6 | 13/12/2024 - 15/12/2024 | Đánh giá hiệu suất |
| 7 | 16/12/2024 – 19/12/2024 | Tối ưu hóa cải thiện  Viết báo cáo |
| 8 | 20/12/2024 – 22/12/2024 | Hoàn thành báo cáo và sản phẩm |

Bảng 1. Bảng kế hoạch thực hiện đề tài

PHẦN HAI: NỘI DUNG

--- oOo ---

CHƯƠNG MỘT: CƠ SỞ LÝ THUYẾT

1.1. Giới thiệu về C#

1.1.1. C# là gì?

C# (hay C sharp) là một ngôn ngữ lập trình đơn giản, được phát triển bởi đội ngũ kỹ sư của Microsoft vào năm 2000. C# là ngôn ngữ lập trình hiện đại, hướng đối tượng và được xây dựng trên nền tảng của hai ngôn ngữ mạnh nhất là C++ và Java.

Trong các ứng dụng Windows truyền thống, mã nguồn chương trình được biên dịch trực tiếp thành mã thực thi của hệ điều hành.

Trong các ứng dụng sử dụng .NET Framework, mã nguồn chương trình (C#, VB.NET) được biên dịch thành mã ngôn ngữ trung gian MSIL (Microsoft intermediate language).

Sau đó mã này được biên dịch bởi Common Language Runtime (CLR) để trở thành mã thực thi của hệ điều hành. Hình bên dưới thể hiện quá trình chuyển đổi MSIL code thành native code.

1.1.2. Đặc điểm ngôn ngữ C#

**Lịch sử C#**

* C# được phát triển bởi nhóm của Anders Hejlsberg và Scott Wiltamuth tại Microsoft.
* C# là ngôn ngữ lập trình hướng đối tượng và được cung cấp các thư viện xây dựng sẵn.

+ Cho phép lập trình viên phát triển ứng dụng nhanh chóng.

* C# và Visual Basic chia sẻ thư viện của Framework Class Library (FCL).
* C# thích hợp cho các yêu cầu phát triển ứng dụng quản lý.

+ đặc biệt là các ứng dụng trên windows và Web.

**Đặc điểm**

* C# là ngôn ngữ hướng sự kiện và hỗ trợ lập trình trực quan.

+ Chương trình được xây dựng sử dụng môi trường lập trình tích hợp (Integrated Development Environment-IDE).

* C# được chuẩn hóa bởi ECMA (European Computer Manufacturers Association) International.

**Lý do sử dụng C#**

* C# đơn giản

+ Loại bỏ các vấn đề của C++, Java (như macro, template…)

* C# là ngôn ngữ hiện đại
* C# là ngôn ngữ lập tình hướng đối tượng

+ Hỗ trợ tính năng bao đóng, kế thừa và tính đa hình

* C# mạnh mẽ và linh hoạt

+ Sử dụng phát triển nhiều loại ứng dụng (xử lý văn bản, hình ảnh, bảng tính phục vụ cho quản lý…)

**Ứng dụng**

* Ứng dụng Console
* Ứng dụng Windows Form
* Dịch vụ Web XML
* Ứng dụng dạng Client-Server
* Ứng dụng CSDL
* C# còn cung cấp trình soạn thảo mã nguồn thuận tiện cho thiết kế giao diện người dùng (interface designer), trình kiếm lỗi (debugger), và nhiều công cụ khác giúp dễ dàng hơn cho việc phát triển ứng dụng.

1.2. Giới thiệu về Bootstrap

1.2.1. Bootstrap là gì?

Bootstrap là 1 framework HTML, CSS, và JavaScript cho phép người dùng dễ dàng thiết kế website theo 1 chuẩn nhất định, tạo các website thân thiện với các thiết bị cầm tay như mobile, ipad, tablet,...

**Tại sao lại như vậy?**

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.

1.2.2. Tại sao chúng ta nên sử dụng Bootstrap?

Bootstrap là một trong những framework được sử dụng nhiều nhất trên thế giới để xây dựng nên một website. Bootstrap đã xây dựng nên 1 chuẩn riêng và rất được người dùng ưa chuộng. Chính vì thế, chúng ta hay nghe tới một cụm từ rất thông dụng "Thiết kế theo chuẩn Bootstrap".

Từ cái "chuẩn mực" này, chúng ta có thể thấy rõ được những điểm thuận lợi khi sử dụng Bootstrap.

Rất dễ để sử dụng: Nó đơn giản vì nó được base trên HTML, CSS và Javascript chỉ cẩn có kiến thức cơ bản về 3 cái đó là có thể sử dụng bootstrap tốt.

Responsive: Bootstrap xây dựng sẵn reponsive css trên các thiết bị Iphones, tablets, và desktops. Tính năng này khiến cho người dùng tiết kiệm được rất nhiều thời gian trong việc tạo ra một website thân thiện với các thiết bị điện tử, thiết bị cầm tay.

Tương thích với trình duyệt: Nó tương thích với tất cả các trình duyệt (Chrome, Firefox, Internet Explorer, Safari, and Opera). Tuy nhiên, với IE browser, Bootstrap chỉ hỗ trợ từ IE9 trở lên. Điều này vô cùng dễ hiểu vì IE8 không support HTML5 và CSS3.

1.2.3. Ưu và nhược điểm của Bootstrap

**Ưu điểm:**

* **Phát triển giao diện nhanh chóng:** Bạn dễ dàng phát triển giao diện website một cách rất nhanh, nếu một trang bình thường thì bạn có thể cắt xong trong một ngày hoặc chưa tới một ngày. Chưa kể đến tính tương thích với các trình duyệt và thiết bị di động.
* **Dễ học, dễ sử dụng:** Cộng đồng đông đúc và tài liệu tham khảo rõ ràng chính là sức mạnh của Bootstrap.
* **Nền tảng tối ưu:** Trong bootstrap đã tạo sẵn một thư viện để lưu trữ mà các nhà thiết kế có thể sử dụng và tuỳ ý chỉnh sửa theo mục đích cá nhân. Hơn nữa, bootstrap sự tương thích với trình duyệt và thiết bị đã được kiểm tra nhiều lần nên bạn hoàn toàn có thể yên tâm với kết quả mình làm ra, thậm chí bạn còn có thể bỏ qua cả bước kiểm tra lại, và bạn sẽ tiết kiệm được thời gian, tiền bạc cho website của mình.
* **Tương tác tốt với Smartphone:** Nếu như trước đây khi truy cập website bằng điện thoại di động bạn thường nhận được result từ trang tìm kiếm như mobile.trangweb.com, tức là trang web này được lập trình cho cả 2 phiên bản, nhưng với bootstrap có sử dụng grid system nên bootstrap mặc định hỗ trợ responsive và viết theo xu hướng mobile first ưu tiên giao diện mobile trước. Điều này cải thiện đáng kể hiệu suất trang web khi có người dùng truy cập bằng mobile. Khách hàng thiết kế web của bạn không còn nỗi lo trang web của mình có thể chạy trên nền tảng di động hay không.
* **Giao diện đầy đủ sang trọng:** Giao diện của bootstrap có màu xám bạc rất sang trọng và hỗ trợ gần như đầy đủ các thành phần mà một website hiện đại cần có. Cầu trúc HTML rõ ràng giúp bạn nhanh chóng nắm bắt được cách sử dụng và phát triển. Không những vậy, bootstrap còn giúp website hiển thị tốt khi chúng ta co dãn màn hình windows.
* **Dễ dàng tùy biến:** Để phù hợp cho nhiều loại website, bootstrap cũng hỗ trợ thêm tính năng customizer, bạn có thể thay đổi gần như tất cả những thuộc tính của nó để phù hợp với chương trình của bạn. Nếu những tuỳ chình này vẫn không đáp ứng được yêu cầu của bạn, bạn hoàn toàn có thể chỉnh sửa trực tiếp trên mã nguồn của bootstrap. Boostrap tương thích rất tốt với HTML5.
* **Hỗ trợ SEO tốt**: Đây là lý do quan trọng nhất bởi vì hiện nay Google đã cập nhật thuật toán tìm kiếm và Responsive là một yếu tố rất quan trọng để đưa từ khóa lên top.

**Nhược điểm:**

* **Tính kém phổ biến:** Bootstrap không phải là ứng dụng web phổ biến nên để tìm được một tổ chức, cá nhân thành thạo bootstrap để có thể sử dụng với nền tảng lập trình web không nhiều.
* **Sản phẩm nặng, tốc độ tối ưu chưa cao:** nên nếu dự án của bạn đòi hỏi sản phẩm nhẹ thì việc sử dụng bootstrap sẽ là cả một gánh nặng cho web.
* ​**Chưa hoàn thiện**: Bootstrap chưa đầy đủ các thư viện cần thiết. Các phát triển chưa thể tạo ra một framework riêng hoàn hảo, do đó một số trang web vẫn phải dùng phiên bản dành riêng cho mobile
* **Nhiều code thừa:** Không thể phủ nhận rằng Bootstrap có rất nhiều ưu điểm khi nó cũng cấp gần như đầy đủ những tính năng cơ bản của một trang web responsive hiện đại. Tuy nhiên, mặt trái của việc này là website của bạn sẽ phải tải thêm rất nhiều dòng code không cần thiết khi mà bạn chỉ cần chưa đến 10% những gì Bootstrap cung cấp.
* **Bootstrap không khuyến khích sáng tạo:** Chỉ cần nhét Bootstrap vào themes sẵn có, gọi ra cái .class từ stylesheet và thế là bạn đã có một trang web responsive trông cũng ổn ổn. Sự tiện dụng và dễ dàng của Bootstrap nhiều khi sẽ khuyến khích tính lười sáng tạo, vốn luôn thường trực trong mỗi chúng ta. Kết quả là, chúng ta thướng thoả hiệp những gì mình thực sự muốn cho website để đổi lấy sự tiện dụng và tiết kiệm thời gian mà Bootstrap mang lại.

1.3. Giới thiệu về CSS3

1.3.1. CSS3 là gì?

CSS3 (Cascading Style Sheets) là phiên bản mới nhất của ngôn ngữ định kiểu được sử dụng rộng rãi trong phát triển web. CSS3 cung cấp cho các nhà phát triển web một bộ công cụ mạnh mẽ để tạo kiểu cho các trang web, cho phép kiểm soát chính xác giao diện và trải nghiệm người dùng.

Đi sâu hơn, CSS3 còn giới thiệu các tính năng mới như flexbox, grid layout, chuyển đổi mượt mà, và biến đổi, giúp xây dựng các bố cục phức tạp và hiệu ứng động một cách dễ dàng. Các tính năng này mở ra nhiều khả năng sáng tạo, nâng cao tính thẩm mỹ và tương tác của website.

Ngoài ra, CSS3 còn được tích hợp tốt với các công cụ phát triển web khác, cho phép tạo ra các website đáp ứng và tương thích với nhiều thiết bị khác nhau.

1.3.2. Các tính năng của CSS3

* **Selectors:** Selectors cho phép nhà thiết kế chọn ở các cấp chính xác hơn của trang web. Chúng là các lớp cấu trúc giả thực hiện đối sánh từng phần để giúp khớp lệnh các giá trị thuộc tính và thuộc tính nhắm mục tiêu đến một lớp giả để tạo kiểu cho các phần tử trong URL. Các bộ chọn cũng bao gồm một lớp giả đã kiểm tra để tạo kiểu cho các phần tử như hộp kiểm và nút radio.
* **Hiệu ứng văn bản và bố cục:** Với CSS3, chúng ta có thể thay đổi cách căn lề của văn bản, điều chỉnh khoảng trắng của tài liệu và tạo kiểu cho dấu gạch ngang của các từ.
* **First - Letter and First – Line Pseudo – Classes:** CSS3 bao gồm các thuộc tính hỗ trợ kerning (điều chỉnh khoảng cách giữa các ký tự để đạt được hiệu ứng trực quan đẹp mắt) và định vị các dấu gạch xuống (chữ hoa trang trí lớn ở đầu đoạn văn).
* **Paged Media và Generated Content:** CSS3 có các lựa chọn bổ sung trong Paged Media, chẳng hạn như số trang, đầu trang và footers đang chạy. Có các thuộc tính bổ sung để in Generated Content, như thuộc tính cho các tham chiếu chéo và chú thích cuối trang.
* **Bố cục nhiều cột:** Tính năng này bao gồm các thuộc tính cho phép các nhà thiết kế trình bày nội dung của họ trong nhiều cột với các tùy chọn như số cột, khoảng cách cột và chiều rộng cột.

1.3.3. Ưu và nhược điểm

**Ưu điểm:**

* **Trình bày nội dung linh hoạt:** CSS3 cung cấp nhiều tính năng mới như sử dụng gradient, box-shadow, text-shadow, border-radius, etc. giúp tăng tính thẩm mỹ và trực quan của trang web.
* **Tối ưu hoá hiệu năng:** Một số tính năng của CSS3 như Transform, Transition, Animation giúp tăng hiệu năng, cải thiện trải nghiệm người dùng mà không cần dùng JavaScript.
* **Hỗ trợ đa nền tảng:** CSS3 được thiết kế để hỗ trợ đa nền tảng, có thể được sử dụng trên desktop, mobile, tablet với các trình duyệt phổ biến.
* **Giảm code HTML:** CSS3 cho phép làm nhiều hơn với ít code HTML, giúp trang web gọn nhẹ và dễ bảo trì.
* **Tăng trải nghiệm người dùng:** Các tính năng mới của CSS3 như Animation, Transform,...giúp tăng tương tác, trải nghiệm người dùng.

**Nhược điểm:**

* **Tính tương thích trình duyệt:** Các tính năng mới của CSS3 không được hỗ trợ đầy đủ trên tất cả các trình duyệt, đòi hỏi phải sử dụng các tiền tố nhà cung cấp.
* **Độ phức tạp tăng:** Khi sử dụng nhiều tính năng mới của CSS3, code CSS có thể trở nên phức tạp và khó bảo trì.
* **Khả năng lạm dụng:** Nếu sử dụng quá nhiều hiệu ứng CSS3 có thể khiến trang web trở nên rối loạn và ảnh hưởng đến trải nghiệm người dùng.
* **Hiệu năng có thể bị ảnh hưởng:** Một số hiệu ứng CSS3 như Animation, Transform có thể làm chậm trang web, đặc biệt là trên các thiết bị cũ hoặc yếu.
* **Hạn chế về trình duyệt cũ:** Người dùng trình duyệt cũ có thể không được trải nghiệm đầy đủ các tính năng mới của CSS3.

1.4. Giới thiệu về HTML5

1.4.1. HTML5 là gì?

HTML5 là phiên bản cập nhật lần thứ 5 của ngôn ngữ HTML. Đây là ngôn ngữ cấu trúc và trình bày nội dung được giới thiệu bởi World Wide Web Consortium (W3C). HTML5 hiện là ngôn ngữ chuẩn mới nhất thay thế cho cả HTML4, XHTML, HTML DOM Level 2.

**Lợi ích của HTML5:**

* HTML5 cung cấp một cơ sở bộ đệm ẩn, các ứng dụng ngoại tuyến sẽ tải trang mà người dùng đã truy cập ngay cả khi người dùng tạm thời ngoại tuyến. Tính năng này sẽ giúp các tập tin tải nhanh hơn và giảm tải trên máy chủ.
* HTML5 hỗ trợ các phần tử đa phương tiện như: Khả năng phát âm thanh và video.

1.4.2. Ưu và nhược điểm

**Ưu điêm:**

* **Cấu trúc trang web rõ ràng**: HTML5 cung cấp các phần tử mới như <header>, <nav>, <article>, <section>, <aside>, <footer> giúp cấu trúc trang web trở nên rõ ràng và rành mạch hơn.
* **Hỗ trợ đa phương tiện:** HTML5 bổ sung các phần tử mới như <video>, <audio> cho phép nhúng trực tiếp các nội dung đa phương tiện mà không cần phụ thuộc vào plugin.
* **Cải thiện khả năng tiếp cận:** Các phần tử mới trong HTML5 như <nav>, <article>, <section> cải thiện khả năng truy cập và sử dụng cho người khuyết tật.
* **Tăng hiệu suất**: HTML5 cung cấp các API mới như Offline, Geolocation, Local Storage giúp cải thiện hiệu suất và trải nghiệm người dùng.
* **Giảm code JavaScript:** Nhiều tính năng mới của HTML5 như form validation, date/time pickers, etc. có thể được thực hiện mà không cần sử dụng JavaScript.

**Nhược điểm:**

* **Tính tương thích trình duyệt:** Các tính năng mới của HTML5 chưa được hỗ trợ đầy đủ trên các trình duyệt cũ, đòi hỏi phải sử dụng các kỹ thuật fallback.
* **Độ phức tạp tăng:** Với nhiều tính năng mới, HTML5 trở nên phức tạp hơn so với các phiên bản trước, đòi hỏi thời gian học tập và hiểu biết sâu rộng hơn.
* **Thiếu các tiêu chuẩn rõ ràng:** Một số tính năng mới của HTML5 chưa có tiêu chuẩn rõ ràng, khiến các trình duyệt thực hiện không thống nhất.
* **Ảnh hưởng đến trải nghiệm người dùng:** Nếu sử dụng quá nhiều tính năng mới của HTML5 trên các trình duyệt cũ có thể ảnh hưởng đến trải nghiệm người dùng.
* **Yêu cầu cập nhật trình duyệt:** Các tính năng mới của HTML5 yêu cầu người dùng phải cập nhật trình duyệt mới để trải nghiệm tối ưu.

1.5. Giới thiệu vài nét về ASP .NET MVC

* ASP.NET MVC là một framework hỗ trợ đầy đủ cho việc xây dựng ứng dụng web theo mô hình MVC.
* Giống ASP.NET Web Forms, ASP.NET MVC được xây dựng dựa trên ASP.NET Framework. Điều này có nghĩa là có thể sử dụng các API trong ứng dụng ASP.NET Web
* Forms truyền thống vào trong ứng dụng ASP.NET MVC.
* Trong thế giới ASP.NET MVC, nhiều cải tiến về ASP.NET đã được đưa vào framework.
* Mục đích chính của mẫu thiết kế này là cô lập xử lý nghiệp vụ từ giao diện người dùng để tập trung vào khả năng bảo trì, cải tiến testing tốt hơn và làm cho một cấu trúc ứng
* dụng gọn hơn.
* MVC framework được định nghĩa trong namespace System.Web.MVC.
* Lịch sử phát triển của ASP.NET MVC

1.5.1. Kiến trúc ASP .NET MVC

* **Model:** là các thành phần có nhiệm vụ

+ Truy xuất dữ liệu từ database và cung cấp dữ liệu cho database.

+ Lưu trữ thông tin, trạng thái của các đối tượng.

+ Tất cả các nghiệp vụ logic đều được thực thi ở Model: Dữ liệu vào từ người dùng sẽ thông qua View để kiểm tran Model trước khi lưu váo database. Việc truy xuất, xác nhận và lưu dữ liệu là phần việc của Model.

* **View:** Là các thành phần chịu trach nhiệm hiển thị thông tin cho người dùng thông qua giao diện User Interface (UI).Thông thường, các thông tin cần hiển thị trên View được tạo ra từ các thành phần Models trong mô hình dữ kiệu (model data).
* **Controllers:** Là các thành phần xử lý tương tác với người dùng, làm việc với model, chọn một View thích hợp để hiển thị thông tin lên giao diện người dùng. Trong một số ứng dụng MVC, View chỉ hiển thị thông tin, còn điều khiển dòng nhập xuất của người dùng vẫn do Controller đảm nhiệm.

**Quá trình xử lý yêu cầu:**

* Một trong những khái niệm quan trọng nhất để hiểu về ứng dụng MVC là không có mối quan hệ tồn tại giữa một request và một tập tin vật lý bên trong web server.

+ Trong một Web Forms truyền thống, mỗi request được dịch thành một lời gọi đến một tập tin vật lý trong web server.

+ Web server sẽ thông dịch yêu cầu này bằng cách tìm kiếm trong thư mục gốc của website một tập tin tương ứng. Sau đó xử lý tập tin này và trả về kết quả dạng HTML.

+ Đối với MVC, khi thực hiện một yêu cầu (ví dụ:http://myapp/product/list), một thành phần được gọi là “Routing engine” sẽ so sánh yêu cầu với route cụ thể.

* Route sẽ xác định các yêu cầu bằng cách sử dụng một chuỗi nền, thiết lập Controller và phương thức trong Controller Class cần xử lý yêu cầu.
* Khi route được nhận diện, công cụ định tuyến tạo ra bộ giám sát yêu cầu, bộ này lần lượt sẽ tạo ra các đối tượng Controller để xử lý yêu cầu (ví dụ trên là controller “product”). Đối tượng Controller sẽ gọi phương thức trong Controller Class (ví dụ trên có tên là “list”).

+ Những phương thức trong Controller Class xử lý các yêu cầu đgl “action methods”.

* Khi xử lý các yêu cầu kết thúc, action method tạo ra một kết quả để gửi lại cho người sử dụng. Thông thường kết quả là dạng HTML (được render bởi View) mà người dùng sẽ thấy trong trình duyệt.

1.5.2. Một số đặc điểm

* Tiếp tục hỗ trợ các tính năng trong ASP.NET

+ Hỗ trợ sử dụng các tập tin: .ASPX, .ASCX, .Master như là thành phần View

+ Hỗ trợ đầy đủ các tính năng bảo mật của ASP.Net: Form/ Windows authenticate, URL authorization, membership/roles, output và data catching, section/ profile state, configuration system, provider architecture, …

+ Tách rõ ràng các mối liên quan, mở ra khả năng test TDD (Test Driven Developer).

* Có thể test unit trong ứng dụng mà không cần phải chạy Controllers cùng với tiến trình của ASP.Net và có thể dùng bất kỳ một unit testing framework nào như Nunit, MBUnit, MS test, …
* Có khả năng mở rộng, mọi thứ trong MVC được thiết kế để dễ thay thế, dễ dàng tùy biến.
* Ánh xạ URL mạnh mẽ, cho phép xây dựng ứng dụng với những URL gọn gàng.
* Không sử dụng mô hình post-back từ giao diện gửi đến server nữa. Thay vào đó, có thể chủ động đưa những post-back từ View đến thẳng lớp Controller.
* Không còn ViewState hay Page Lifecycle tồn tại trong mô hình MVC.
* Hỗ trợ nhiều công cụ tạo View (Support for Multiple View Engines)

+ Cho phép chọn công cụ tạo View

+ Hộp thoại New Project cho phép xác định View Engine mặc định cho một project

+ Các loại View Engine: Web Forms (ASPX), Razor. Hay một View Engine nguồn mở như Spark, Nhaml, Ndjango.

* Hỗ trợ định tuyến

+ ASP .NET MVC Framework có một bộ máy ánh xạ URL thật sự mạnh mẽ.

+ Bộ máy cung cấp phương pháp linh hoạt trong việc ánh xạ URLs sang các Controller Calsses.

+ Dễ dàng định dạng các quy luật, cài đặt đường đi, ASP .NET dựa vào các quy luật đường đi đó để xác định Controller và action cần thực thi.

+ ASP .NET còn có khả năng phân tích URL, chuyển các thông số trong URL thành các tham số trong lời gọi hàm của Controller.

* Model Binding

+ Model Binding là tính năng thế mạnh của ASP .NET MVC.

+ Hỗ trợ viết phương thức nhận một đối tượng tùy biến như một tham số.

+ Với sự hỗ trợ của Model Binding, chỉ cần tập trung vào việc cài đạt các nghiệp vụ logic, không cần phải suy nghĩ làm cách nào để ánh xạ dữ liệu từ người sử dụng sang đối tượng .NET.

* Tính năng Binding được áp dụng mặc định

+ Không cần gọi các scipt hoặc file css riêng biệt, vì tất cả các script hoặc file css được tham chiếu sẽ được đóng gói và nén lại.

* Filters

+ Đây là tính năng mạnh trong ASP .NET MVC. Hỗ trợ cho việc kiểm tra tính hợp lệ trước khi một action method được gọi hoặc sau khi một action method thi hành.

* Razor View

+ Từ ASP .NET MVC 3 trở đi kèm với một công cụ View mới có tên là Razor có những lợi ích như:

* Cú pháp Razor gọng gàn và xúc tích , đòi hỏi một số lượng tối thiểu các tập hợp phím.
* Việc tìm hiểu Razor tương đối dễ dàng vì nó dựa trên ngôn ngữ C# và VB.
* Visual Studio bao gồm InteliSense và mã cú pháp Razor được mã hóa.
* Với Razor View có thể kiểm tra từng đơn vị mà không đòi hỏi phải chạy các ứng dụng hoặc phải chạy Web Server.

**Một số đặc điểm của MVC 5**

* MVC: Tách bạch các phần việc trong xử lý yêu cầu.
* Sát với giao thức web: Không hướng sự kiện như ASP.NET.
* Không duy trì thông tin trạng thái như ASP.NET.
* Tự động nhận diện thiết bị: Tự lựa chọn view phù hợp.
* Razor: sinh giao diện.
* Kiểu dữ liệu động: ViewBag/DataView
* Cải thiện Ajax: Jquery + Helper Ajax.
* Kiểm lỗi: lập trình 1 lần áp dụng cho cả 2 client và server.
* Web API: thư viện web dùng cho nhiều loại thiết bị
* Action Filter: kiểm soát các Action.
* Dễ test: dễ dàng test các Action của các Controller
* NuGet: quản lý các gói mở rộng.

1.5.3. Ưu và nhược điểm của ASP .NET MVC

**Ưu điểm:**

* Có tính mở rộng do thay thế từng thành phần một cách dễ dàng.
* Không sử dụng Viewstate để điều khiển ứng dụng.
* Bổ sung mới hệ thống định tuyến (routers)
* Hỗ trợ tối đa cho việc kiểm thử bằng việc tạo và cài đặt các unitests tự động.
* Hỗ trợ kết hợp rất tốt giữa người lập trình và thiết kế giao diện.

**Nhược điểm:**

* Thay đổi cách lập trình của webform nên gây nhiều khó khăn cho những người chuyên sử dụng webform chuyển sang mô hình này (không còn hướng đến các event của các control).
* Người lập trình phải biết nhiều về HTTP, HTML, CSS và JavaScript.
* Áp dụng cho dự án nhỏ thì quá cồng kềnh và phức tạp.

CHƯƠNG HAI: XÂY DỰNG CHƯƠNG TRÌNH

2.1. Đặc tả hệ thống

Hệ thống quản lý cửa hàng tạp hóa Trân Trân được thiết kế để hỗ trợ chủ cửa hàng và nhân viên quản lý các hoạt động hàng ngày một cách dễ dàng, hiệu quả. Hệ thống bao gồm các thành phần và chức năng chính sau đây:

* **Quản lý sản phẩm**

+ Thêm sản phẩm mới: Cho phép thêm sản phẩm vào hệ thống, bao gồm các thông tin cơ bản như tên sản phẩm, mã sản phẩm, loại sản phẩm, giá cả, số lượng, ngày nhập, mô tả và hình ảnh sản phẩm.

+ Cập nhật thông tin sản phẩm: Hỗ trợ chỉnh sửa thông tin của các sản phẩm đã có trong hệ thống.

+ Xóa sản phẩm: Cho phép xóa sản phẩm không còn bán khỏi hệ thống.

+ Xuất file Excel: Cho phép xuất toàn bộ thông tin của từng loại sản phẩm thành file.

+ Tìm kiếm**:** Cung cấp tính năng tìm kiếm sản phẩm theo tên, loại sản phẩm, giúp người dùng dễ dàng tìm thấy thông tin cần thiết.

* **Quản lý loại sản phẩm**

**+** Thêm, sửa, xóa, tìm kiếm, xuất file excel thông tin loại sản phẩm: Cung cấp chức năng quản lý danh sách loại sản phẩm, bao gồm thông tin loại sản phẩm, tài khoản.

* **Quản lý hóa đơn**

**+** Tạo hóa đơn bán hàng: Hệ thống hỗ trợ tạo hóa đơn khi bán hàng, bao gồm việc chọn sản phẩm, số lượng, tự động tính tổng tiền, và lưu trữ chi tiết hóa đơn.

**+** Lưu trữ lịch sử hóa đơn**:** Các hóa đơn được lưu lại để dễ dàng tra cứu và theo dõi lịch sử bán hàng.

* **Quản lý nhân viên**

+ Thêm, sửa, xóa, tìm kiếm, xuất file excel thông tin nhân viên: Cung cấp chức năng quản lý danh sách nhân viên, bao gồm thông tin cá nhân, vai trò, tài khoản.

+ Phân quyền truy cập: Phân quyền người dùng theo vai trò nhân viên hoặc chủ cửa hàng để đảm bảo an toàn dữ liệu.

* **Quản lý phiếu nhập**

**+** Thêm, sửa, xóa, tìm kiếm, xuất file excel thông tin phiếp nhập: Cung cấp chức năng quản lý danh sách phiếu nhập, bao gồm thông tin sản phẩm, ngày nhập,tổng tiền, tài khoản.

* **Quản lý tài khoản**

+ Về Nhân viên thì có thể đăng nhập đăng xuất

+ Về chủ cửa hàng thì có thể thực hiện quyền của nhân viên ngoài ra còn có thể đăng ký tài khoản, thêm sửa, xóa, phân quyền, xuất file excel thông tin tài khoản. Về việc phân quyền tài khoản giúp hệ thống đảm bảo tính bảo mật.

* **Báo cáo doanh thu (chỉ có chủ cửa hàng được quyền xem báo cáo)**

+ Thống kê doanh thu: Cung cấp tính năng thống kê doanh thu theo các khoảng thời gian cụ thể (tháng, năm) để chủ cửa hàng có cái nhìn tổng quan về hiệu quả kinh doanh.

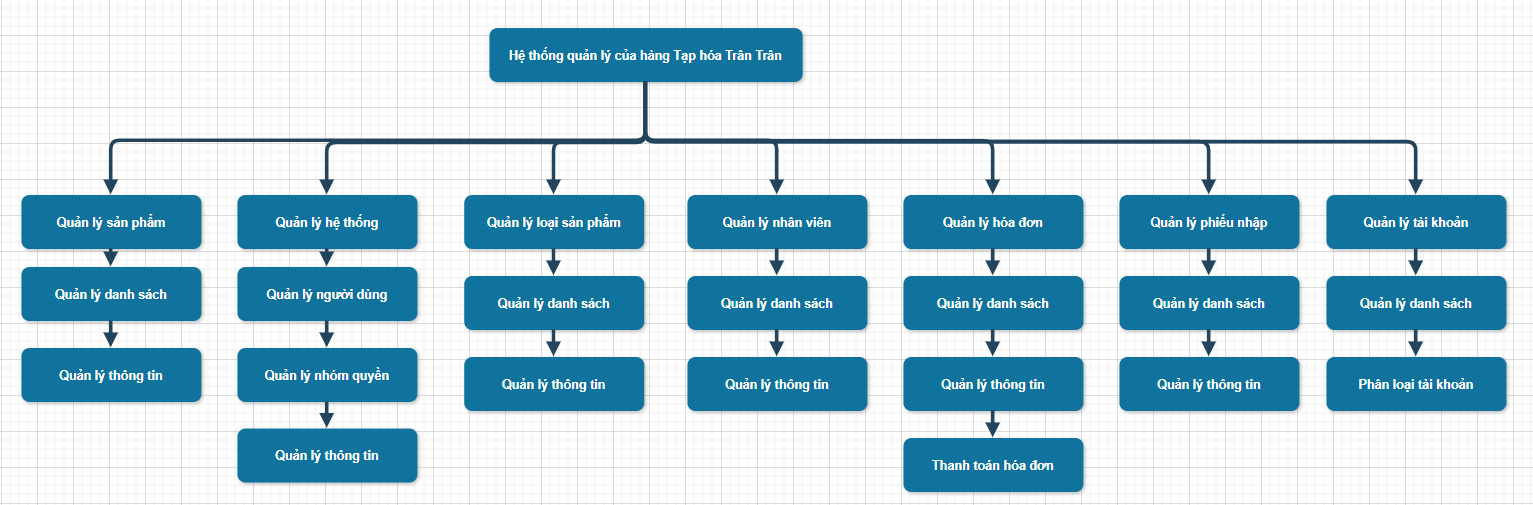
* **Giao diện người dùng**

Thiết kế thân thiện và trực quan: Giao diện được thiết kế để dễ sử dụng, trực quan, giúp người dùng không cần nhiều kiến thức công nghệ vẫn có thể thao tác trên hệ thống.

2.2. Phân tích hệ thống

2.2.1. Sơ đồ hệ thống

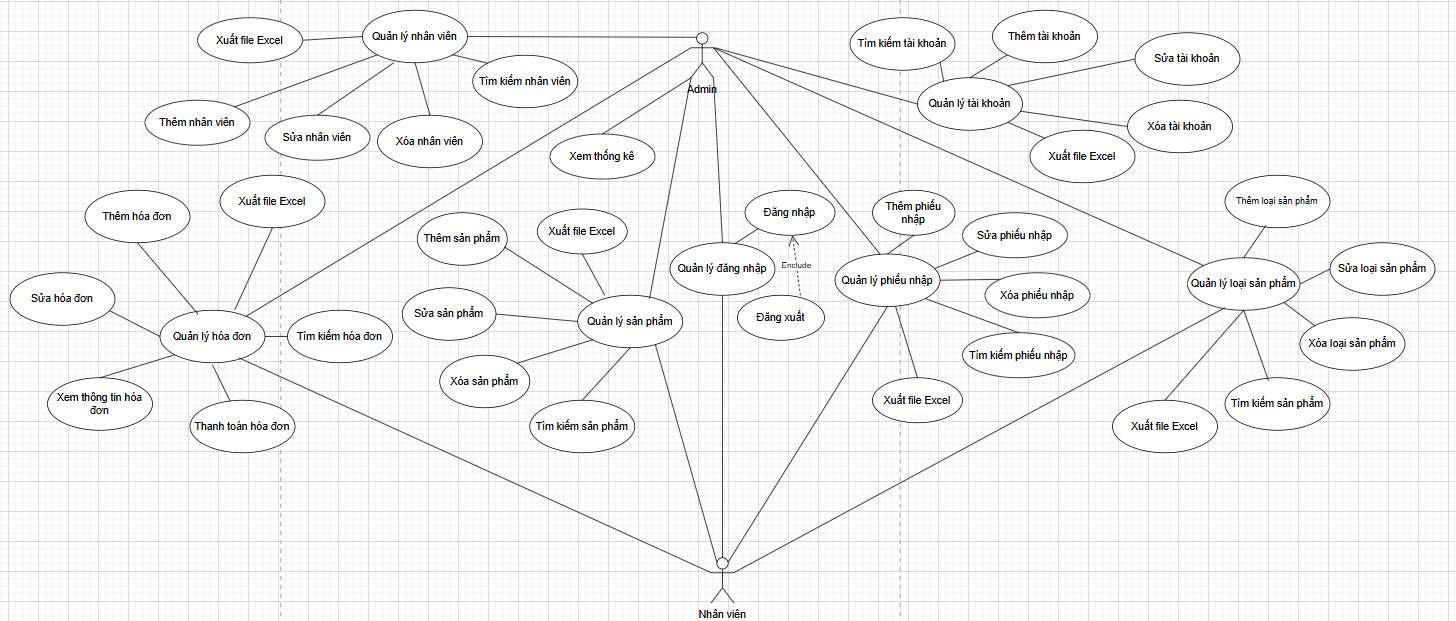
Hình dưới đây sẽ mô tả các chức mà 2 loại người dùng có thể thực hiện trên hệ thống như là các chức năng về quản lý sản phẩm, quản lý nhân viên, quản lý phiếu nhập, quản lý hóa đơn, quản lý tài khoản, quản lý hệ thống,…



Hình 2.1. Sơ đồ hệ thống

2.2.2. Sơ đồ Usecase

2.2.2.1. Sơ đồ Usecase tổng quát

****

Hình 2.2. Hình Usecase tổng quát

Mô tả các actor:

* Nhân viên: có quyền quản lý thông tin của hệ thống như quản lý hóa đơn, sản phẩm, phiếu nhập, loại sản phẩm và có quyền thêm sửa xóa các thông tin trên hệ thống trong phạm vi quyền của nhân viên.
* Quản trị viên: là người quản trị cao nhất của hệ thống, quản trị viên có quyền quản lý các thông tin của hệ thống như: quản lý đơn hàng, sản phẩm, tài khoản, loại sản phẩm,…và có quyền thêm sửa xóa các thông tin trên hệ thống.

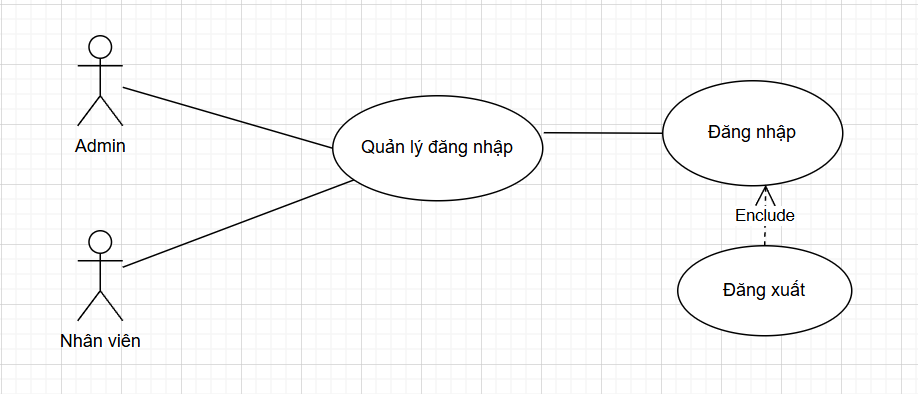
2.2.2.2. Sơ đồ Usecase thành phần

a. Sơ đồ Usecase đăng nhập

Tác nhân: Quản trị viên, nhân viên

Điều kiện: có tài khoản, mật khẩu

Mô tả: Quản trị viên/nhân viên đăng nhập hệ thống bằng tài khoản và mật khẩu cá nhân. Tùy vào phân quyền của tài khoản, quản trị viên/nhân viên có thể truy cập các chức năng tương ứng của hệ thống.



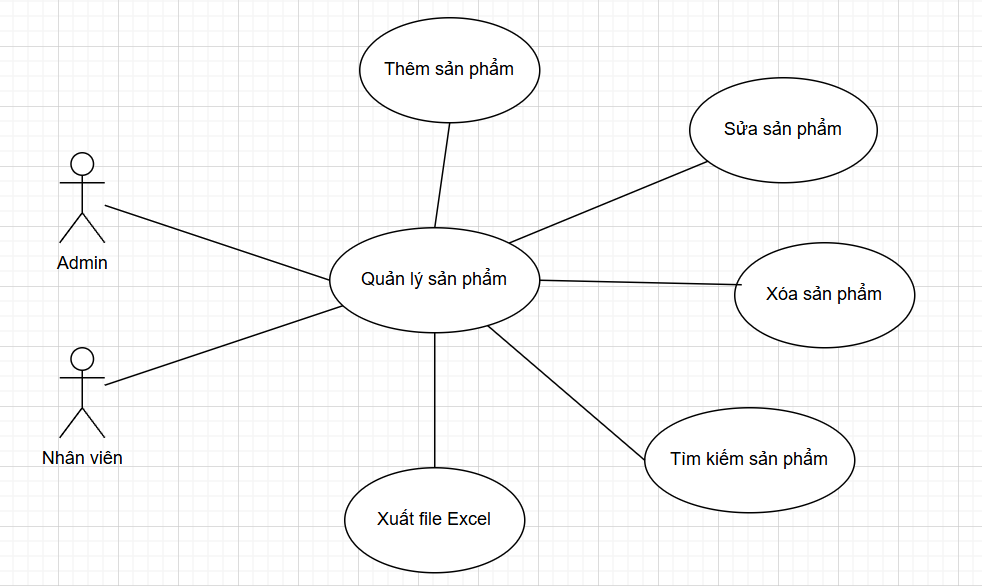
Hình 2.3. Hình sơ đồ Usecase đăng nhập

b. Sơ đồ Usecase quản lý sản phẩm

Tác nhân: Quản trị viên, nhân viên

Điều kiện: có tài khoản, mật khẩu

Mô tả: Trong quản lý sản phẩm, người dùng là quản trị viên/nhân viên có quyền thêm sửa xóa, bên cạnh đó còn có thể tìm kiếm sản phẩm, xem chi tiết sản phẩm, xuất file Excel.



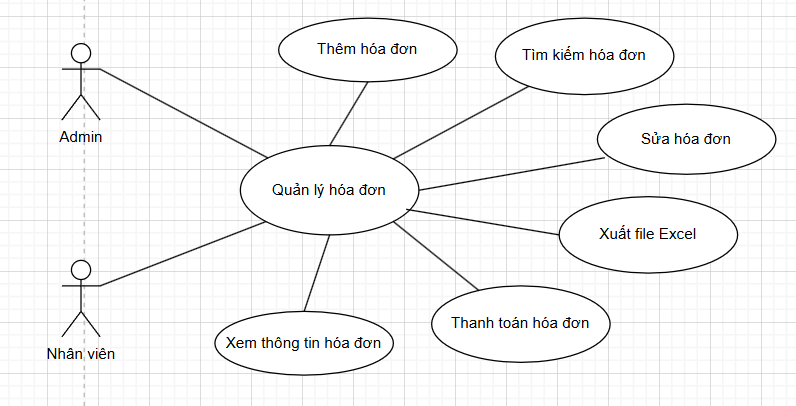
Hình 2.4. Hình sơ đồ Usecase quản lý sản phẩm

c. Sơ đồ Usecase quản lý hóa đơn

Tác nhân: Quản trị viên, nhân viên

Điều kiện: có tài khoản, mật khẩu

Mô tả: Trong quản lý hóa đơn, người dùng là quản trị viên/nhân viên có quyền thanh toán hóa đơn, thêm, sửa và xem chi tiết hóa đơn, bên cạnh đó còn có thể tìm kiếm đơn hàng và xuất file Excel.



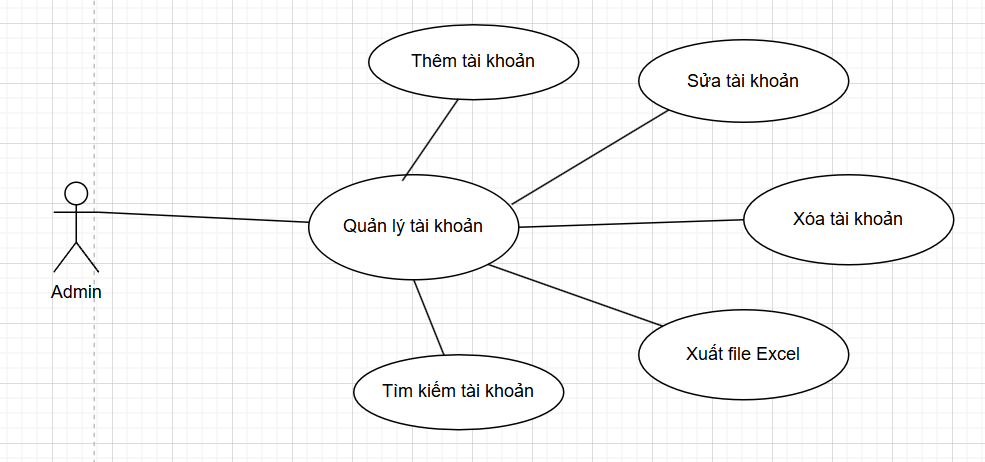
Hình 2.5. Hình sơ đồ Usecase quản lý đơn hàng

d. Sơ đồ Usecase quản lý tài khoản

Tác nhân: Quản trị viên

Điều kiện: có tài khoản, mật khẩu

Mô tả: Trong quản lý tài khoản, người dùng là quản trị viên có quyền thêm, sửa, phân quyền và xem chi tiết tài khoản, bên cạnh đó còn có thể tìm kiếm tài khoản và xuất file Excel.



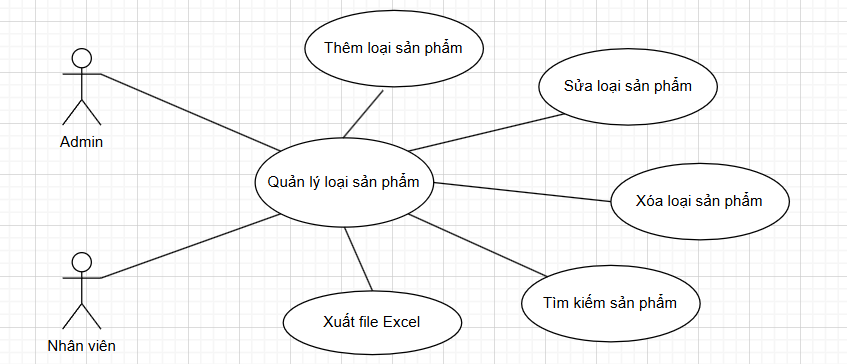
Hình 2.6. Hình sơ đồ Usecase quản lý tài khoản

e. Sơ đồ Usecase quản lý loại sản phẩm

Tác nhân: Quản trị viên, nhân viên

Điều kiện: có tài khoản, mật khẩu

Mô tả: Trong quản lý loại sản phẩm, người dùng là quản trị viên/nhân viên có quyền thêm, sửa, xóa và xem chi tiết loại sản phẩm, bên cạnh đó còn có thể tìm kiếm loại sản phẩm và xuất file Excel.



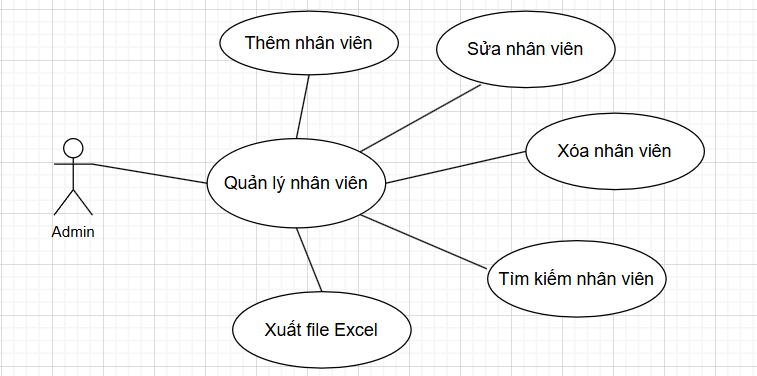
Hình 2.7. Hình sơ đồ Usecase quản lý loại sản phẩm

f. Sơ đồ Usecase quản lý Nhân viên

Tác nhân: Quản trị viên

Điều kiện: có tài khoản, mật khẩu

Mô tả: Trong quản lý nhân viên, người dùng là quản trị viên có quyền thêm, sửa, xóa và xem chi tiết thông tin của nhân viên, bên cạnh đó còn có thể tìm kiếm nhân viên và xuất file Excel.



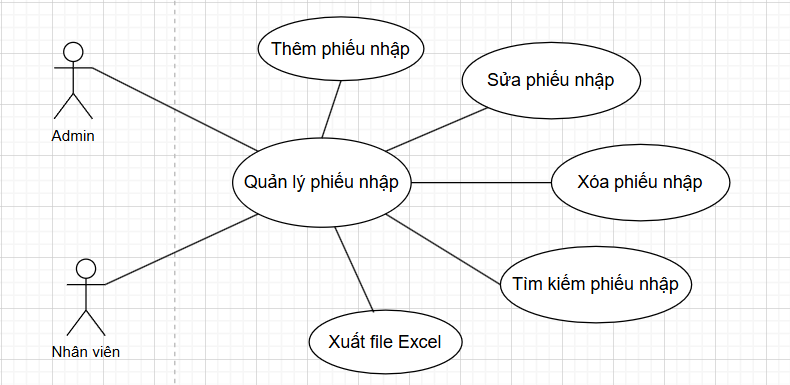
Hình 2.8. Hình sơ đồ Usecase quản lý nhân viên

g. Sơ đồ Usecase quản lý phiếu nhập

Tác nhân: Quản trị viên, nhân viên

Điều kiện: có tài khoản, mật khẩu

Mô tả: Trong quản lý phiếu nhập, người dùng là quản trị viên/ nhân viên có quyền thêm, sửa và xem chi tiết thông tin của phiêu nhập, bên cạnh đó còn có thể tìm kiếm phiếu nhập và xuất file Excel.

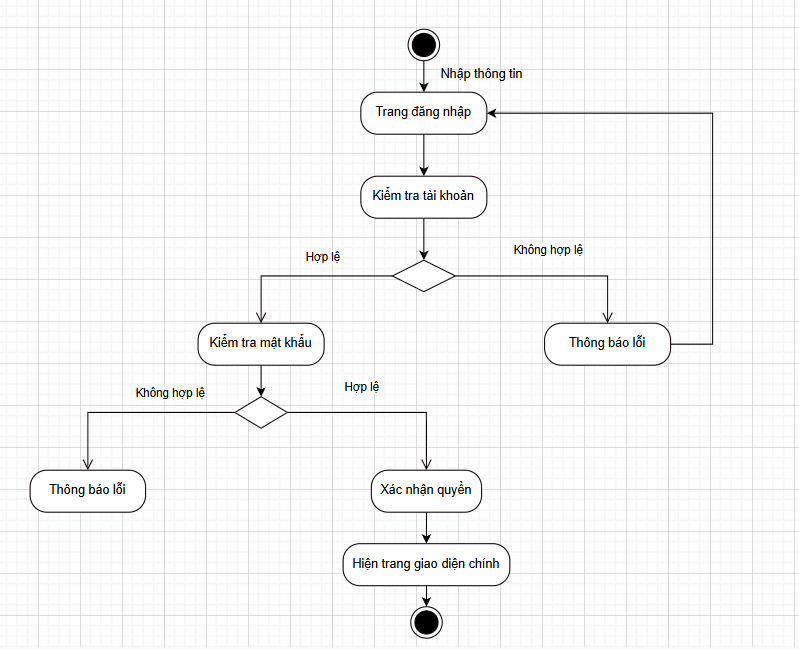


Hình 2.9. Hình sơ đồ usecase quản lý phiếu nhập

2.2.3. Sơ đồ hoạt động

2.2.3.1. Sơ đồ hoạt động đăng nhập

Khi người dùng đăng nhập, người dùng cần nhập đầy đủ thông tin mà hệ thống yêu cầu như tài khoản và mật khẩu tại trang đăng nhập. Sau đó, hệ thống sẽ kiểm tra tài khoản, nếu tài khoản hợp lệ thì “đăng nhập thành công”, xác nhận quyền của tài khoản và hiện trang chủ hệ thống và kết thúc. Ngược lại thông báo tài khoản hoặc mật khẩu không đúng và kết thúc.



Hình 2.10. Hình sơ đồ hoạt động đăng nhập

2.2.3.2. Sơ đồ hoạt động quản lý sản phẩm

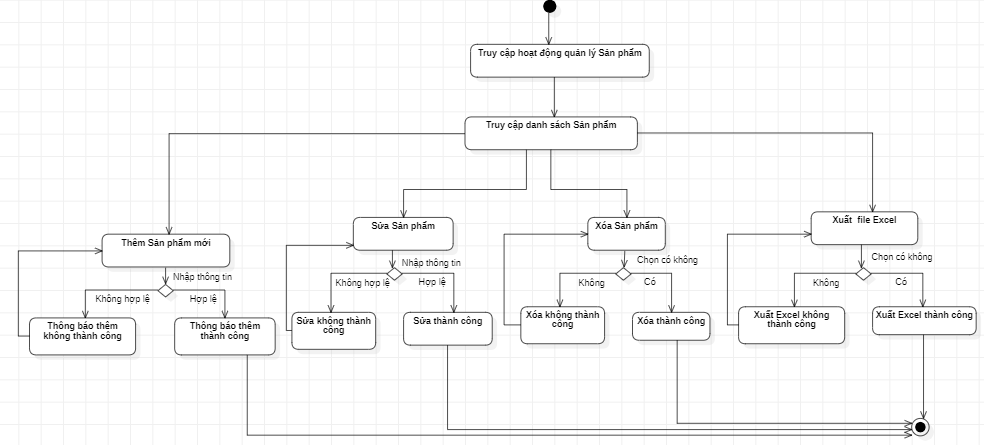
Trong quản lý sản phẩm, người dùng là quản trị viên/nhân viên có thể dùng các chức năng như thêm sản phẩm, sửa sản phẩm, xóa sản phẩm, tìm kiếm sản phẩm, xem chi tiết sản phẩm và xuất file Excel.

Trong hoạt động thêm sản phẩm: người dùng cần nhập các thông tin mà hệ thống yêu cầu để có thể thêm sản phẩm, sau khi nhập đầy đủ và thêm thì hệ thống sẽ hiển thị thông báo thêm thành công và kết thúc hoạt động thêm, ngược lại thì thông báo thêm không thành công và trở về hoạt động thêm.

Trong hoạt động sửa sản phẩm: người dùng cần chọn sản phẩm cần sửa và chỉnh sửa lại thông tin sản phẩm, nếu hợp lệ và đầy đủ thông tin thì hệ thống sẽ thông báo cập nhật thành công và kết thúc hoạt động sửa sản phẩm, ngược lại thì thông báo cập nhật sản phẩm không thành công.

Trong hoạt động xóa sản phẩm: người dùng cần chọn sản phẩm muốn xóa, hệ thống sẽ hiển thị thông báo lựa chọn, nếu người dùng chọn có thì hệ thống sẽ thông báo xóa sản phẩm thành công và kết thúc hoạt động này, ngược lại thì sẽ thông báo xóa không thành công và quay lại ban đầu.

Trong hoạt động xuất excel sản phẩm: người dùng cần bấm vào nút xuất file Excel hệ thống sẽ hiển thị lựa chọn, nếu người dùng chọn có hệ thống sẽ thông báo xuất Excel thành công và kết thúc hoạt động xuất excel, ngược lại hệ thống sẽ thông báo xuất excel không thành công và quay lại hoạt ban đầu.



Hình 2.11. Hình sơ đồ hoạt động quản lý sản phẩm

2.2.3.3. Sơ đồ hoạt động quản lý Nhân viên

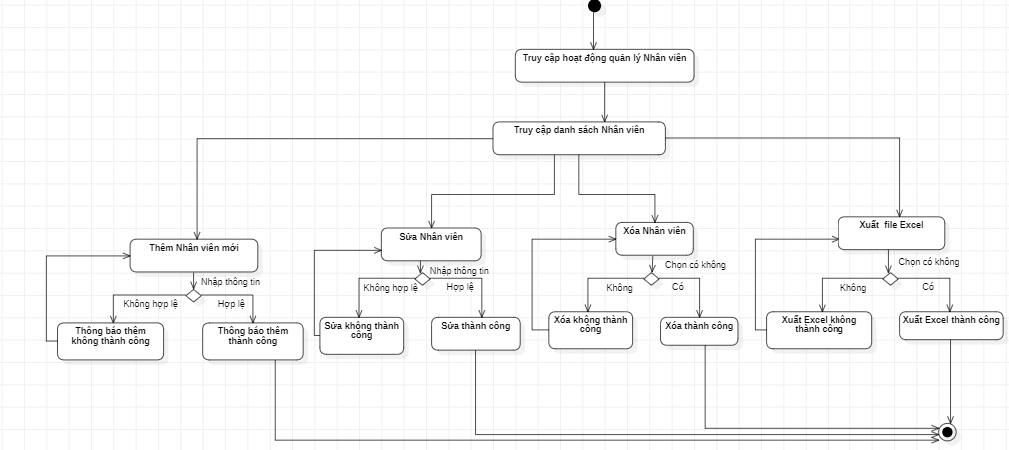
Trong quản lý nhân viên, người dùng là quản trị viên có thể dùng các chức năng như thêm nhân viên, sửa nhân viên, xóa nhân viên, xuất file Excel.

Trong hoạt động thêm nhân viên: người dùng cần nhập các thông tin mà hệ thống yêu cầu để có thể thêm nhân viên, sau khi nhập đầy đủ và thêm thì hệ thống sẽ hiển thị thông báo thêm thành công và kết thúc hoạt động thêm, ngược lại thì thông báo thêm không thành công và trở về hoạt động thêm.

Trong hoạt động sửa nhân viên: người dùng cần chọn nhân viên cần sửa và chỉnh sửa lại thông tin nhân viên, nếu hợp lệ và đầy đủ thông tin thì hệ thống sẽ thông báo cập nhật thành công và kết thúc hoạt động sửa nhân viên, ngược lại thì thông báo cập nhật nhân viên không thành công.

Trong hoạt động xóa nhân viên: người dùng cần chọn nhân viên muốn xóa, hệ thống sẽ hiển thị thông báo lựa chọn, nếu người dùng chọn có thì hệ thống sẽ thông báo xóa nhân viên thành công và kết thúc hoạt động này, ngược lại thì sẽ thông báo xóa không thành công và quay lại ban đầu.

Trong hoạt động xuất excel nhân viên: người dùng cần bấm vào nút xuất file Excel hệ thống sẽ hiển thị lựa chọn, nếu người dùng chọn có hệ thống sẽ thông báo xuất Excel thành công và kết thúc hoạt động xuất excel, ngược lại hệ thống sẽ thông báo xuất excel không thành công và quay lại hoạt ban đầu.

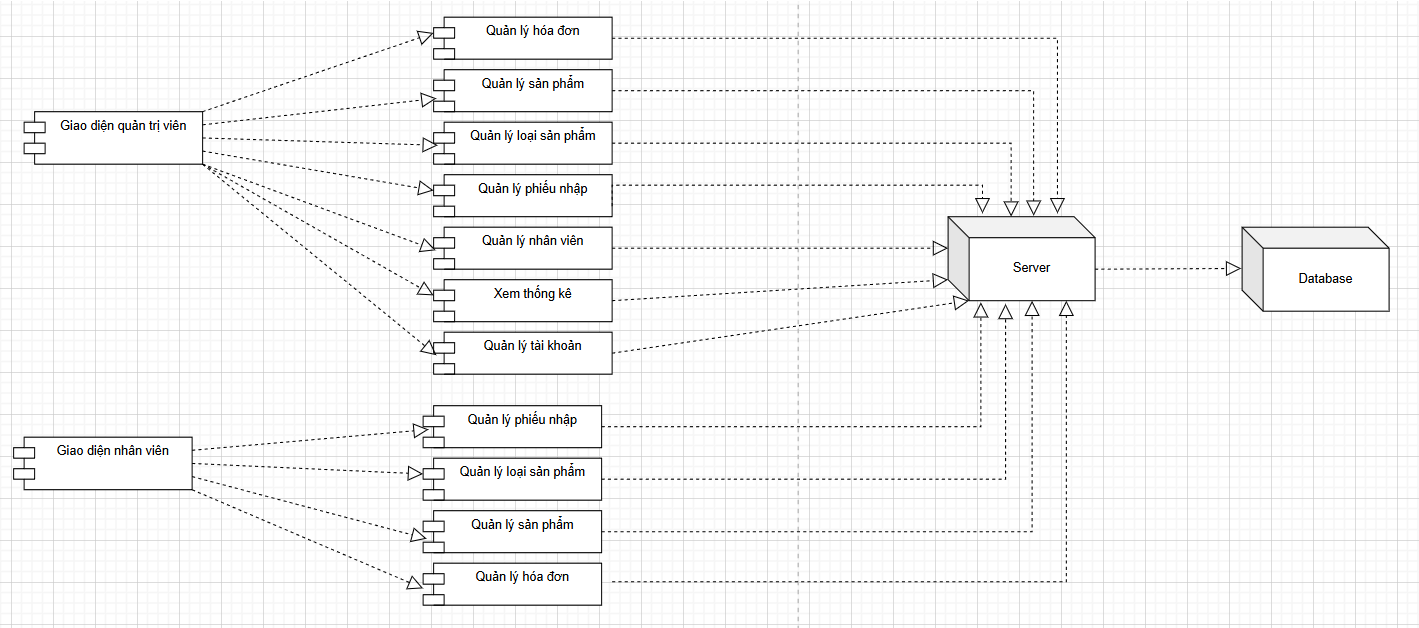


Hình 2.12. Hình sơ đồ hoạt động quản lý nhân viên

2.2.4. Sơ đồ thành phần

Sơ đồ thành phần sẽ khái quát giao diện về chức năng chính của hệ thống như là:

* Giao diện người quản trị: quản lý hóa đơn, quản lý tài khoản, quản lý phiếu nhập, quản lý nhân viên, quản lý sản phẩm, quản lý loại sản phẩm, xem thống kê.
* Giao diện nhân viên: quản lý hóa đơn, quản lý phiếu nhập, quản lý sản phẩm, quản lý loại sản phẩm.



Hình 2.13. Hình sơ đồ thành phần

2.3. Thuyết kế CSDL

2.3.1. Thuyết kế dữ liệu vật lý

2.3.1.1. Bảng Tài khoản

Bảng Tài khoản dùng để lưu trữ thông tin các loại tài khoản của hệ thống như tên tài khoản, mật khẩu, email, số điện thoại…

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | maTaiKhoan | int | Mã tài khoản, khóa chính |
| 2 | tenDangNhap | varchar(50) | Tên đăng nhập |
| 3 | matKhau | nvarchar(255) | Mật khẩu |
| 4 | email | varchar(100) | Email |
| 5 | soDienThoai | varchar(20) | Số điện thoại |
| 6 | ngayTao | date | Ngày tạo |
| 7 | loaiTaiKhoan | varchar(50) | Loại tìa khoản |

Bảng 2.1. Bảng tài khoản

2.3.1.2. Bảng Sản phẩm

Bảng sản phẩm dùng để lưu trữ thông tin các loại sản phẩm của hệ thống như tên sản phẩm, số lượng, đơn giá…

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | maSanPham | int | Mã sản phẩm, khóa chính |
| 2 | tenSanPham | nvarchar(255) | Tên sản phẩm |
| 3 | giaBan | decimal(10,2) | Giá bán |
| 4 | soLuong | int | Số lượng |
| 5 | ngayNhap | nvarchar(250) | Ngày nhập |
| 6 | moTa | nvarchar(4000) | Mô tả |
| 7 | hinhAnh | bit | Hình ảnh |
| 8 | maLoaiSanPham | bit | Mã loại sản phẩm |

Bảng 2.2. Bảng sản phẩm

2.3.1.3. Bảng Phiếu nhập hàng

Bảng phiếu nhập hàng dùng để lưu trữ thông tin các phiếu nhập của hệ thống như mã phiếu nhập, mã nhà cung cấp, mã nhà cung cấp,…

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | maPhieuNhap | int | Mã phiếu nhập, khóa chính |
| 2 | maNhaCungCap | int | Mã nhà cung cấp |
| 3 | ngayNhap | date | Ngày nhập |
| 4 | tongTien | decimal(10,2) | Tổng tiền |
| 5 | maTaiKhoan | int | Mã tài khoản |

Bảng 2.3. Bảng phiếu nhập hàng

2.3.1.4. Bảng Nhân viên

Bảng Nhân viên dùng để lưu trữ thông tin nhân viên của hệ thống như tên nhân viên, số diện thoại, địa chỉ, email, chức vụ…

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | maNhanVien | int | Mã nhân viên, khóa chính |
| 2 | tenNhanVien | nvarchar(255) | Tên nhân viên |
| 3 | soDienThoai | varchar(20) | Số điện thoại |
| 4 | diaChi | nvarchar(255) | Địa chỉ |
| 5 | email | nvarchar(255) | Email |
| 6 | chucVu | nvarchar(255) | Chức vụ |
| 7 | luong | decimal(10, 2) | Lương |
| 7 | maTaiKhoan | int | Mã tài khoản |

Bảng 2.4. Bảng nhân viên

2.3.1.5. Bảng Nhà cung cấp

Bảng Nhà cung cấp dùng để lưu trữ thông tin các nhà cung cấp của hệ thống như tên nhà cung cấp, số diện thoại, địa chỉ,…

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | maNhaCungCap | int | Mã nhà cung cấp, khóa chính |
| 2 | tenNhaCungCap | nvarchar(255) | Tên nhà cung cấp |
| 3 | soDienThoai | varchar(20) | Số điện thoại |
| 4 | diaChi | nvarchar(255) | Địa chỉ |
| 5 | email | varchar(100) | Email |

Bảng 2.5. Bảng nhà cung cấp

2.3.1.6. Bảng Loại sản phẩm

Bảng Loại sản phẩm dùng để lưu trữ thông tin các loại sản phẩm của hệ thống như mã loại sản phẩm, tên sản phẩm.

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | maLoaiSanPham | int | Mã loại sản phẩm, khóa chính |
| 2 | tenLoaiSanPham | nvarchar(255) | Tên loại sản phẩm |

Bảng 2.6. Bảng loại sản phẩm

2.3.1.7. Bảng Hóa đơn

Bảng Hóa đơn dùng để lưu trữ thông tin các hóa đơn của hệ thống như ngày lập, tổng tiền, trạng thái…

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | maHoaDon | int | Mã hóa đơn, khóa chính |
| 2 | ngayLap | date | Ngày lập |
| 3 | tongTien | decimal(10, 20) | Tổng tiền |
| 4 | maTaiKhoan | int | Mã tài khoản |
| 5 | TrangThai | varchar(50) | Trạng thái |

Bảng 2.7. Bảng hóa đơn

2.3.1.8. Bảng Chi tiết phiếu nhập hàng

Bảng Chi tiết phiếu nhập dùng để lưu trữ thông tin các phiếu nhập của hệ thống như mã sản phẩm, số lượng, giá nhập,…

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | maPhieuNhap | int | Mã phiếu nhập, khóa chính |
| 2 | maSanPham | int | Mã sản phẩm, khóa chính |
| 3 | soLuong | int | Số lượng |
| 4 | giaNhap | decimal(10, 2) | Giá nhập |
| 5 | thanhTien |  | Thành tiển |

Bảng 2.8. Bảng chi tiết phiếu nhập

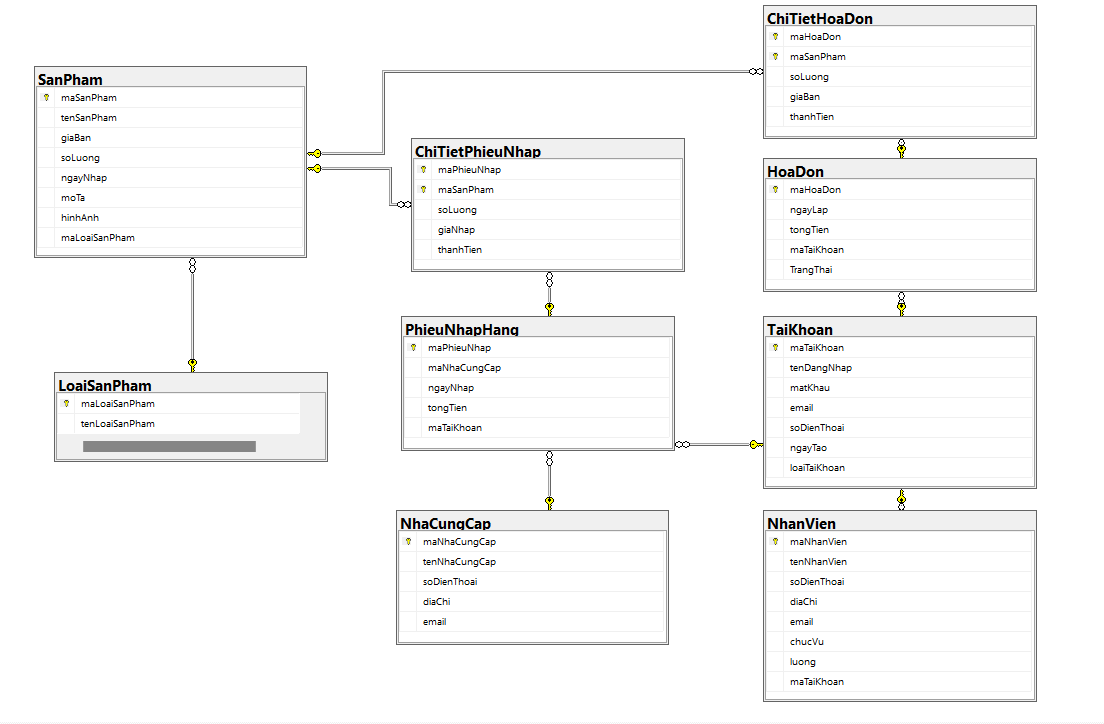
2.3.1.9. Bảng Chi tiết hóa đơn

Bảng Chi tiết hóa đơn dùng để lưu trữ thông tin chi tiết các hóa đơn của hệ thống như mã sản phẩm, số lượng, giá bán,…

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | maHoaDon | int | Mã hóa đơn, khóa chính |
| 2 | maSanPham | int | Mã sản phẩm |
| 3 | soLuong | int | Số lượng |
| 4 | giaNhap | decimal(10, 2) | Giá nhập |
| 5 | thanhTien |  | Thành tiển |

Bảng 2.9. Bảng chi tiết hóa đơn

2.3.1.10. Sơ đồ lớp hệ thống

****

Hình 2.14. Sơ đồ lớp hệ thống

2.4. Một số giải thuật tiêu biểu

2.4.1. Giải thuật tìm kiếm theo tên sản phẩm

**Mục tiêu:** Tìm kiếm đúng và gần đúng tên sản phẩm với nội dung người dùng nhập.

**Thuật toán sử dụng:** Giải thuật tìm kiếm chuỗi thường sử dụng so sánh ký tự từng vị trí của từ khóa với dữ liệu trong tập dữ liệu.

**Quá trình tìm kiếm:**

* Nhập dữ liệu: chuỗi từ khóa được nhập từ người dùng.
* Xử lý dữ liệu: dữ liệu trong tập dữ liệu được duyệt qua để tìm kiếm chuỗi từ khóa.
* So sánh và đánh giá: so sánh từng ký tự của từ khóa với dữ liệu để xác định kết quả tìm kiếm.

**Kết quả:** trả về tên chính xác thông tin của sản phẩm hoặc một danh sách thông tin của các sản phẩm gần đúng.

Sử dụng giải thuật này giúp tối ưu hóa quá trình tìm kiếm của người dùng, thuận tiện cho người dùng tìm kiếm sản phẩm.

2.4.1. Giải thuật thanh toán hóa đơn

**Mục tiêu:** Cập nhật trạng thái hóa đơn thành "Đã thanh toán" khi người dùng chọn thanh toán một hóa đơn và tạo phiên bản có thể in được của hóa đơn đó.

**Thuật toán sử dụng**: Giải thuật này thực hiện cập nhật trạng thái của hóa đơn và tạo hóa đơn có thể in được dựa trên ID của hóa đơn.

**Quá trình thanh toán:**

**Chọn hóa đơn:** Người dùng sẽ chọn hóa đơn cần thanh toán bằng cách nhấp vào dòng dữ liệu chứa thông tin của hóa đơn trong bảng. Mã hóa đơn (ID) của hóa đơn này sẽ được lấy từ thao tác chọn của người dùng.

**Xử lý dữ liệu:** Sau khi lấy được ID của hóa đơn từ bước trước, hệ thống sẽ duyệt qua danh sách hóa đơn trong cơ sở dữ liệu để tìm kiếm hóa đơn có mã tương ứng với ID người dùng đã chọn. Mỗi hóa đơn sẽ được kiểm tra và so sánh với ID được chọn để xác định đúng hóa đơn cần thanh toán.

**So sánh và đánh giá:** Sau khi tìm được hóa đơn có mã tương ứng, hệ thống sẽ kiểm tra trạng thái hiện tại của hóa đơn. Nếu trạng thái của hóa đơn là chưa thanh toán, hệ thống sẽ thực hiện cập nhật trạng thái của hóa đơn thành "Đã thanh toán". Nếu hóa đơn đã được thanh toán trước đó, có thể hệ thống sẽ thông báo cho người dùng rằng hóa đơn đã thanh toán hoặc không thực hiện thay đổi.

**Tạo hóa đơn**: Sau khi cập nhật trạng thái của hóa đơn thành "Đã thanh toán", hệ thống sẽ tạo một bản hóa. Bản hóa đơn này sẽ bao gồm các thông tin chi tiết như:

+ Tên sản phẩm

+ Số lượng sản phẩm

+ Giá bán của từng sản phẩm

+ Tổng tiền của hóa đơn

+ Trạng thái của hóa đơn ("Đã thanh toán")

**Kết quả:** Sau khi hoàn tất, hệ thống sẽ trả về một bản in chi tiết của hóa đơn đã thanh toán hoặc thông báo lỗi nếu không tìm thấy hóa đơn. Điều này giúp tối ưu hóa quy trình thanh toán và cung cấp một bản sao của hóa đơn cho mục đích lưu trữ.

Sử dụng giải thuật này giúp tối ưu hóa quy trình thanh toán, thuận tiện cho người dùng khi thanh toán hóa đơn và cung cấp bản in cho mục đích lưu trữ.

2.5. Phát triển hệ thống

2.5.1. Cộng cụ và môi trường phát triển

**Môi trường phát triển**

* Ứng dụng xây dựng trên nền tảng ASP.NET MVC Framework.
* Ứng dụng được xây dựng theo mô hình client-server, mang lại lợi ích đáng kể trong việc đơn giản hóa quá trình phát triển và tối ưu hóa hiệu suất. Việc triển khai mô hình client-server giúp tách biệt logic và chức năng giữa phía client và phía server. Phía client chịu trách nhiệm về giao diện người dùng và tương tác người dùng, trong khi phía server xử lý logic, lưu trữ dữ liệu và thực hiện các tác vụ chức năng. Lợi ích của mô hình client-server bao gồm sự linh hoạt trong việc phát triển và bảo trì, vì có thể thay đổi và cập nhật client mà không ảnh hưởng đến server và ngược lại.

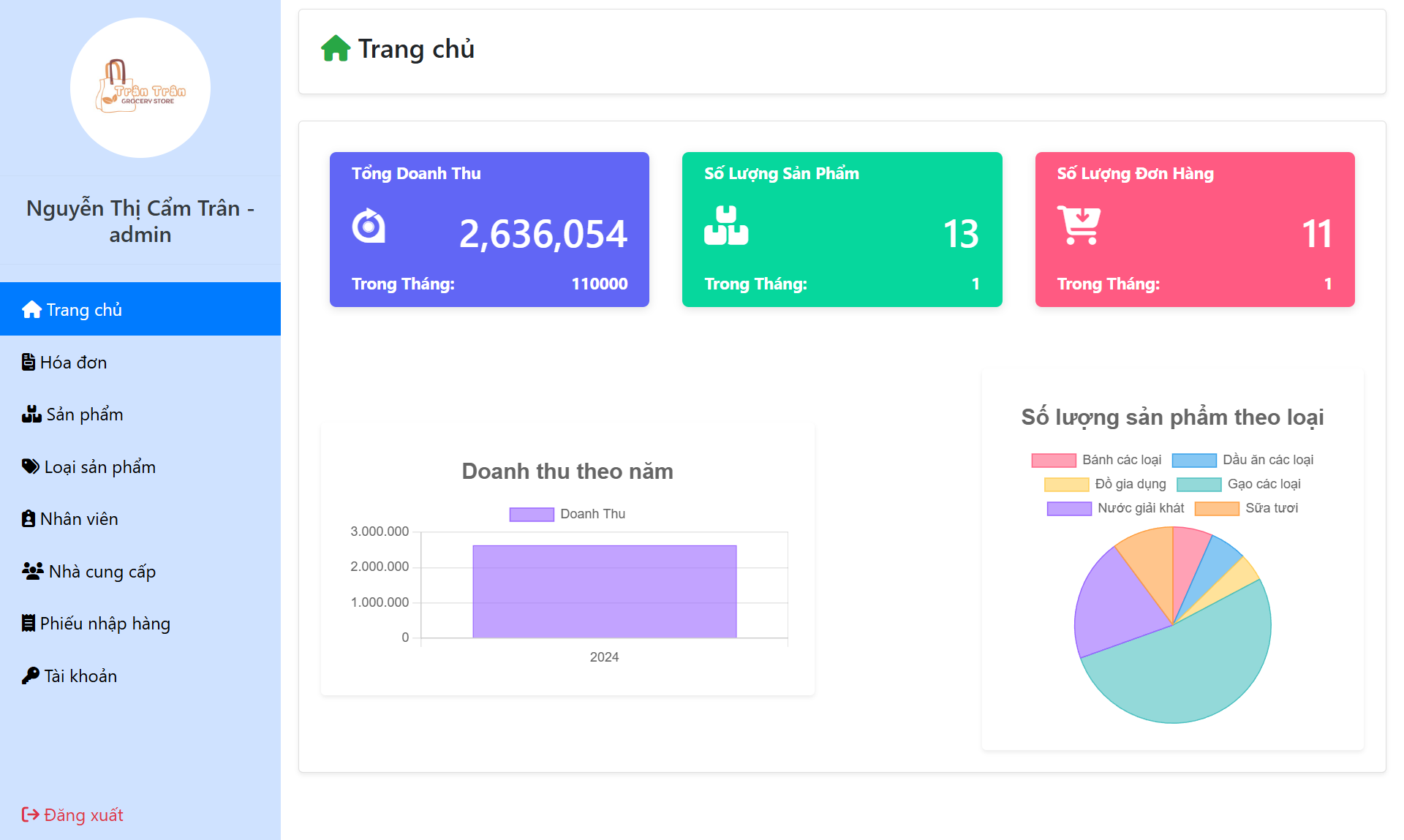
**Công cụ phát triển hệ thống:**

* Visual Studio 2022 công cụ phát triển chính của hệ thống.
* Hệ quản trị cơ sở dữ liệu của ứng dụng SQL server.
* Đồng thời xây dựng hệ thống qua các sơ đồ usecase, sơ đồ hoạt động, sơ đồ tuần tự dựa trên công cụ StarUML.

2.5.2. Một số giao diện chính của chương trình

2.5.2.1. Giao diện trang chủ

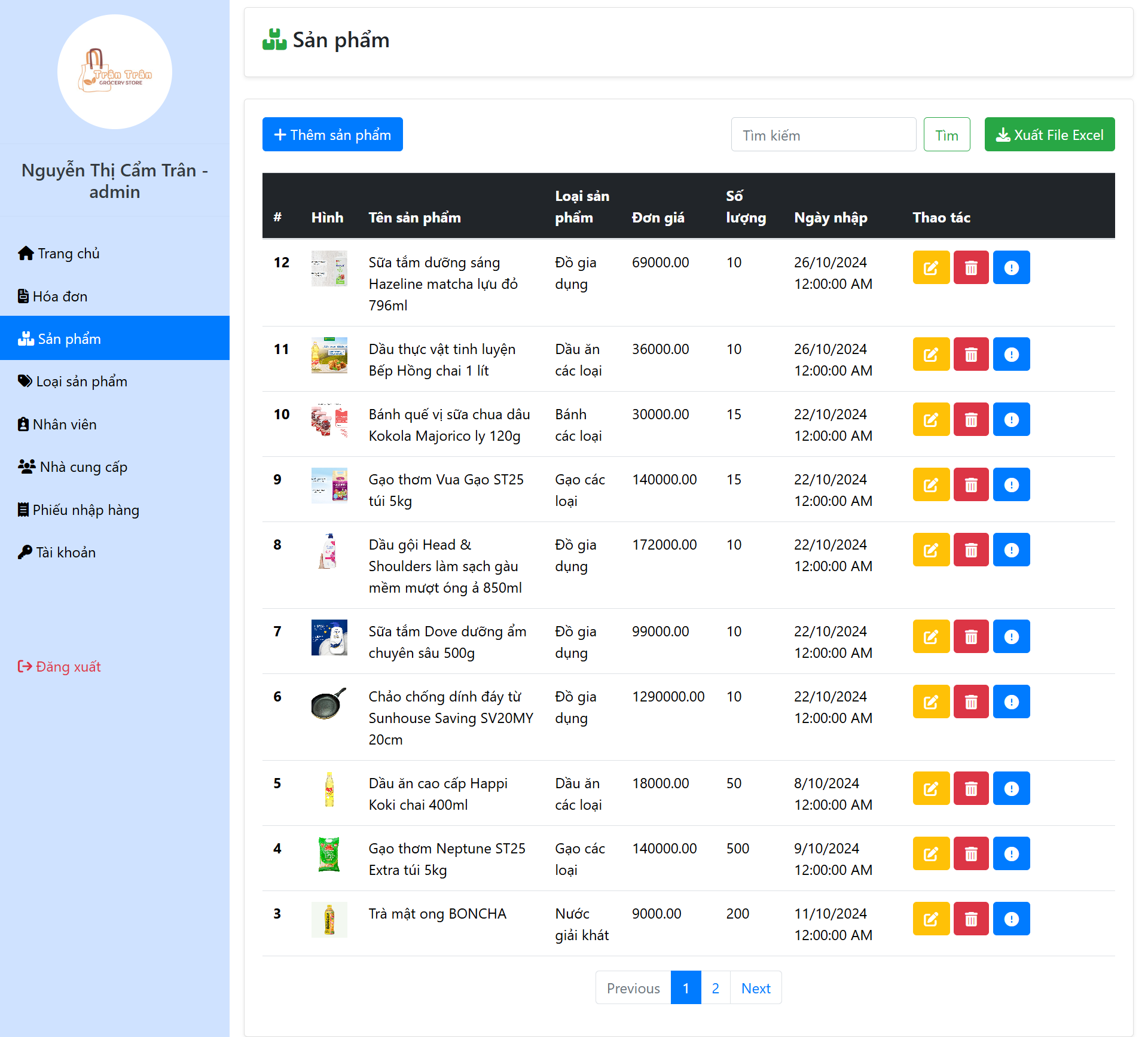
Tại đây người dùng có thể xem thông kê về tổng doanh thu theo tháng, năm hiện tại, tổng số lượng sản phẩm trong tháng, năm, tổng số lượng đơn hàng trong tháng, năm.



Hình 2.15. Giao diện trang chủ

2.5.2.2. Giao diện quản lý sản phẩm

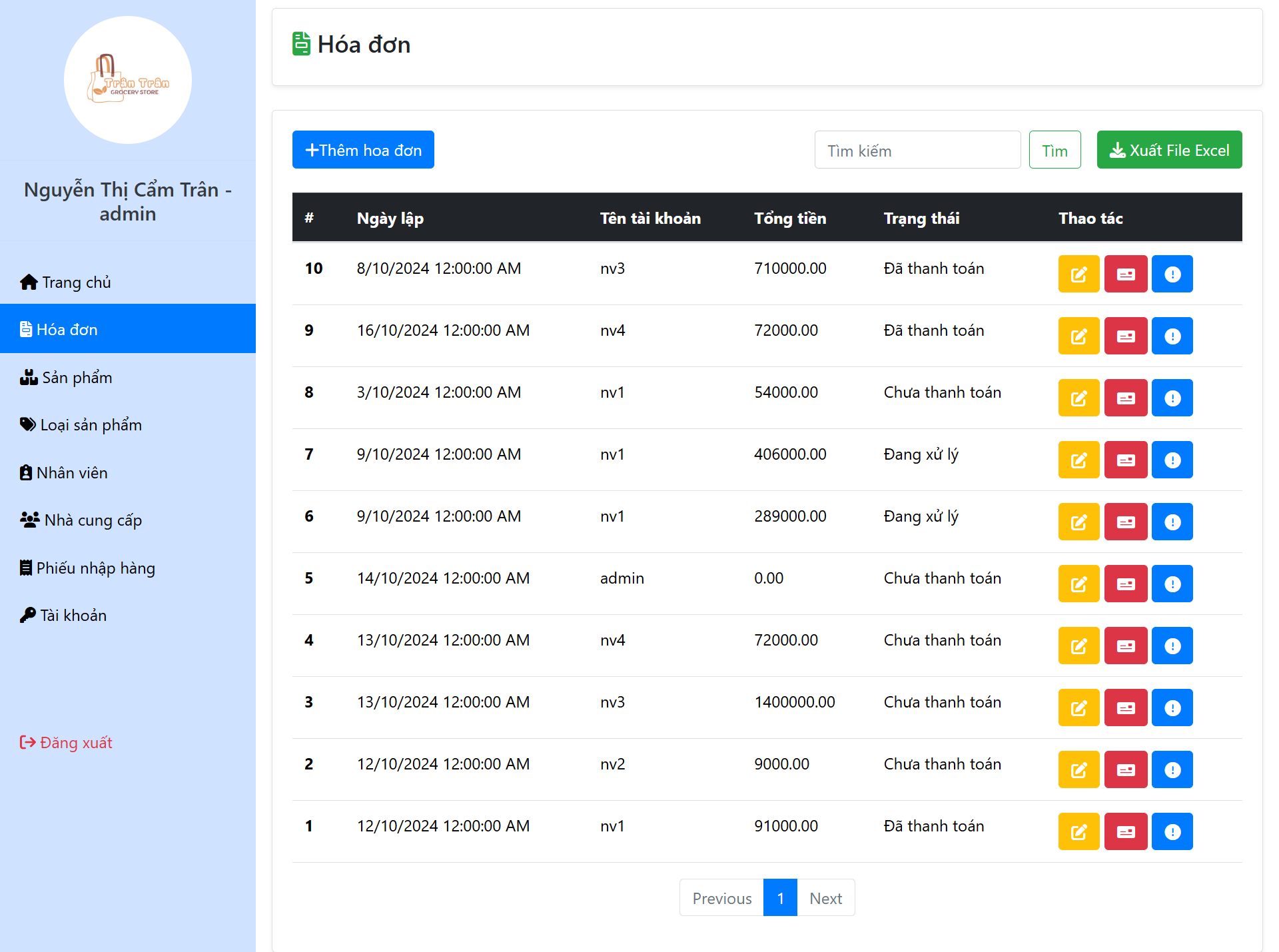
Khi người dùng chuyển đến trang quản lý sản phẩm, người dùng sẽ xem được thông tin các loại sản phẩm như tên sản phẩm, đơn giá, số lượng,…Bên cạnh đó người dùng có thể thêm sản phẩm, sửa sản phẩm, xóa sản phẩm, xem chi tiết sản phẩm và có thể xuất ra file Excel.



Hình 2.16. Hình giao diện trang quản lý sản phẩm

2.5.2.3. Giao diện quản lý hóa đơn

Khi người dùng chuyển đến trang quản lý hóa đơn, người dùng sẽ xem được thông tin các hóa đơn, bên cạnh đó người dùng có thể thêm hóa đơn, sửa hóa đơn, thanh toán hóa đơn, xem chi tiết hóa đơn và có thể xuất ra file Excel.



Hình 2.17. Giao diện quản lý hóa đơn

PHẦN BA: KẾT LUẬN

1. Kết quả đạt được

* Qua thời gian nghiên cứu tìm hiểu công nghệ và viết ứng dụng, em đã đạt được một số kết quả nhất định. Hoàn thành được mục tiêu xây dựng website quản lý cửa hàng Tạp hóa. Các chức năng đã đáp ứng được yêu cầu đặt ra ban đầu, em đã nghiên cứu và áp dụng các công nghệ và thư viện để hoàn thành ứng dụng này.
* Những chức năng cụ thể mà em đã thực hiện được như sau:

+ Người dùng là Nhân viên: có thể quản lý sản phẩm, quản lý hóa đơn, quản lý phiếu nhập, quản lý loại sản phẩm như thêm sửa xóa, xem chi tiết, xuất file excel…

+ Người dùng là quản trị viên hệ thống: quản trị hệ thống, quản lý người dùng hệ thống (đăng ký, chỉnh sửa, xóa người dùng), quản lý vai trò gồm: Quản trị viên (phân quyền, cấp và xóa tài khoản), quản lý Sản phẩm, quản lý hóa đơn, quản lý Nhân viên, quản lý loại sản phẩm.

* Bên cạnh đó, hiểu được nghiệp vụ, phân tích đề tài cụ thể và thiết kế cơ sở dữ liệu.
* Đồng thời xây dựng hệ thống thông qua các sơ đồ usecase, sơ đồ hoạt động, sơ đồ tuần tự dựa trên công cụ StarUML.

2. Hạn chế

* Phần mềm chưa tích hợp với đa dạng thiết bị như điện thoại, máy tính bản.
* Giao diện chưa chuyên nghiệp, bố cục và màu sắc chưa hài hòa.
* Phần mềm chưa nhiều chức năng như một ứng dụng

3. Hướng phát triển

* Hoàn thiện và bổ sung thêm nhiều chức năng mới để đáp ứng yêu cầu nghiệp vụ của đề tài.
* Xây dựng giao diện tích hợp với nhiều giao diện.
* Thiết kế giao diện hài hòa, đẹp mắt thích hợp với người dùng để tăng tính trải nghiệm người dùng hệ thống.

**TÀI LIỆU THAM KHẢO**

**1. Giáo trình, bài giảng**

[1] Trần Kim Hương & et al. (2021), *Bài giảng Công Nghệ Web,* Trường Đại học Đồng Tháp khoa Công nghệ và Kỹ thuật.

[2] Jon Galloway, Brad Wilson, K. Scott Allen, David Matson, (2014), *Professional ASP.NET MVC 5,* Trung tâm học liệu Lê Vũ Hùng Trường Đại học Đồng Tháp.

[3] José Rolando Guay Paz, (2013), *Beginning ASP.NET MVC 4,* Trung tâm học liệu Lê Vũ Hùng Trường Đại học Đồng Tháp.

**2. Các website tham khảo**

[1] Vietnix, Cao Lê Viết Tiến, (10-09-2024), *Mô hình MVC là gì? Ví dụ và cách ứng dụng MVC vào lập trình,* [*https://vietnix.vn/tim-hieu-mo-hinh-mvc-la-gi/*](https://vietnix.vn/tim-hieu-mo-hinh-mvc-la-gi/)*,* truy cập ngày 14, 11, 2024.

[2] Codegym, Codegym, (24-07-2024), *C# là gì? Tìm hiểu về ngôn ngữ lập trình C#*, <https://codegym.vn/blog/c-la-gi-tim-hieu-ve-ngon-ngu-lap-trinh-c/>, truy cập ngày 14, 11, 2024.

[3] Wikipedia, Wikipedia, (14-10-2024), *C Sharp (ngôn ngữ lập trình),* <https://vi.wikipedia.org/wiki/C_Sharp_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)>, truy cập ngày 14, 11, 2024.