

**TRƯỜNG ĐẠI HỌC MỎ - ĐỊA CHẤT
KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO
ĐỀ TÀI NCKH SINH VIÊN
ĐỀ TÀI:**

PHẦN MỀM BÁN HÀNG

Trưởng nhóm nghiên cứu: Nguyễn Quốc Vương, lớp CNPM E-K66

Thành viên tham gia thực hiện:

1. Nguyễn Việt Bách, lớp HTTT A-K67
2. Trần Quang Khải, lớp CNPM A-K66
3. Nguyễn Thành Đạt, lớp KHDL A-K67
4. Nguyễn Đức Quý, lớp CNPM B-K66
5. Nguyễn Quốc Vương, lớp CNPM-K66

Người hướng dẫn: GV.TS Nguyễn Tuấn Anh

HÀ NỘI, THÁNG 5/2025

LỜI CẢM ƠN

Trên thực tế, không có sự thành công nào mà không gắn liền với sự hỗ trợ, giúp đỡ dù ít hay nhiều, dù trực tiếp hay gián tiếp của mọi người. Qua hơn bảy tháng thực hiện đề tài nghiên cứu khoa học, nhóm chúng em đã nhận được nhiều sự quan tâm và giúp đỡ từ quý thầy cô, gia đình và bạn bè. Với lòng biết ơn sâu sắc nhất, nhóm chúng em xin gửi lời cảm ơn đến ban giám hiệu trường đại học mở - địa chất, các giảng viên khoa công nghệ thông tin cùng các phòng ban đã tạo điều kiện thuận lợi để nhóm có môi trường học tập và nghiên cứu tốt nhất nhằm hoàn thành đề tài “nghiên cứu xây dựng ứng dụng phần mềm quản lý bán hàng.”

Đặc biệt, nhóm chúng em xin gửi lời cảm ơn chân thành đến GV.Th.S Nguyễn Tuấn Anh, người đã tận tình hướng dẫn và luôn theo sát chỉ dẫn nhóm trong suốt quá trình thực hiện đề tài.

Đề tài “Nghiên cứu xây dựng ứng dụng phần mềm quản lý bán hàng” (đề tài NCKH cấp bộ kèm theo quyết định số 546/QĐ-MĐC) là một nghiên cứu có tính ứng dụng thực tiễn cao, giúp các cửa hàng và doanh nghiệp vừa và nhỏ dễ dàng quản lý hoạt động bán hàng, theo dõi hàng tồn kho và doanh thu một cách hiệu quả hơn. Hệ thống được xây dựng nhằm tối ưu hóa quy trình quản lý, giảm thiểu sai sót và nâng cao hiệu suất làm việc. Thông qua việc triển khai và hoàn thành đề tài, nhóm chúng em đã có cơ hội vận dụng các kiến thức chuyên ngành vào thực tế, rèn luyện khả năng làm việc nhóm, tư duy giải quyết vấn đề và phát huy tính sáng tạo. Việc tham gia nghiên cứu khoa học là cơ hội quý báu để chúng em nâng cao kiến thức và chuẩn bị hành trang vững chắc cho giai đoạn thực tập và làm việc sau này.

Một lần nữa, nhóm chúng em xin chân thành cảm ơn ban giám hiệu cùng quý thầy cô trong các phòng ban. Kính chúc sức khỏe quý thầy cô và chúc trường đại

học mở - địa chất ngày càng phát triển vững mạnh, đạt được nhiều thành công trong thời gian tới.

Hà Nội, Tháng 5 năm 2025

MỤC LỤC

LỜI CẢM ƠN	2
MỤC LỤC	3
1. Lí do chọn đề tài.....	6
2. Mục đích và yêu cầu của đề tài	7
2.1 Mục đích	7
2.2 Yêu cầu	7
2.3 Hiệu suất.....	7
3. Đối tượng, phạm vi nghiên cứu	8
3.1 Đối tượng nghiên cứu	8
3.2 Phạm vi nghiên cứu.....	8
4. Phương pháp nghiên cứu.....	8
A. Phương pháp nghiên cứu thực nghiệm:	8
B. Ngôn ngữ sử dụng:.....	8
5. Kết cấu đề tài.....	9
6. Dự kiến kết quả	9
7. Ý nghĩa khoa học và thực tiễn.....	9
CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI.....	9
1.1 Giới thiệu đề tài.....	9

1.2	<i>Tính cấp thiết của đề tài</i>	10
1.3	<i>Khảo sát hiện trạng</i>	11
1.3.1	<i>Hiện trạng quản lý bán hàng của doanh nghiệp/ cửa hàng nhỏ lẻ</i>	11
1.3.2	<i>Các phương pháp quản lý bán hàng hiện có</i>	11
1.4	<i>Yêu cầu đặt ra cho ứng dụng phần mềm quản lý bán hàng</i>	12
1.4.1	<i>Yêu cầu lưu trữ dữ liệu</i>	12
1.4.2	<i>Yêu cầu tra cứu và báo cáo</i>	12
1.4.3	<i>Sơ đồ tổng thể dự án</i>	13
1.4.4	<i>Yêu cầu về các chức năng chính</i>	14
1.4.5	<i>Yêu cầu phi chức năng</i>	15
CHƯƠNG 2.	<i>YÊU CẦU VỀ THIẾT KẾ CHUNG</i>	19
2.1	<i>Yêu cầu chung về giao diện</i>	19
2.2	<i>Yêu cầu về màn hình giao diện hệ thống</i>	20
2.2.1.	<i>Độ Phân Giải Màn Hình</i>	20
2.2.2	<i>Lợi Ích</i>	20
2.3	<i>Nền Ngoài (Background)</i>	21
2.4	<i>Font Chữ</i>	21
2.5	<i>Cỡ Chữ</i>	21
2.5.1	<i>Mục Tiêu</i>	21
2.5.2	<i>Yêu Cầu Giao Diện</i>	21
2.5.3	<i>Lợi Ích</i>	21
2.6	<i>Hiển Thị Banner</i>	22
2.6.1	<i>Mục Tiêu</i>	22
2.6.2	<i>Yêu Cầu Giao Diện</i>	22
2.6.3	<i>Lợi ích</i>	22
2.7	<i>Hiển Thị Footer</i>	22
2.7.1	<i>Mục Tiêu</i>	22

2.7.2 Yêu Cầu Giao Diện	23
2.7.3 Lợi Ích	23
3 Danh sách giao diện	23
CHƯƠNG 3. NGHIÊN CỨU XÂY DỰNG ỨNG DỤNG QUẢN LÝ	
BÁN HÀNG 26	
3.1 Ngôn ngữ lập trình.....	26
3.1.1 Ngôn ngữ C#	26
3.1.2 Ngôn ngữ HTML.....	27
3.1.3 Ngôn ngữ CSS	27
3.1.4 Ngôn ngữ JavaScript.....	28
3.1.5 Framework/Library	28
3.1.6 Công cụ định dạng tài liệu Roff	29
3.1.7 Công cụ quản lý source code	30
3.1.8 Công cụ triển khai container	32
3.2. Phân tích và thiết kế hệ thống.....	32
3.2.1 Mô tả hệ thống.....	32
3.2.2 Chức năng Import/Export Excel	38
3.2.3 Sơ đồ quy trình	39
3.3. Thiết kế dữ liệu	40
3.3.1 Biểu đồ chứng năng nghiệp vụ.....	40
3.3.2 Biểu đồ use case người dùng khi mua hàng.	42
3.3.3 Biểu đồ luồng dữ liệu.....	42
3.3.4. Thiết kế bảng.....	43
3.3.4 Giao diện người dùng mẫu.....	44
CHƯƠNG 4: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	47
4.1 Các kết quả đạt được:.....	48
4.2 Những hạn chế:	48
4.3 Hướng phát triển:	48
TÀI LIỆU THAM KHẢO	49

1. Lí do chọn đề tài

Trước đây, khi công nghệ thông tin chưa phát triển mạnh, các doanh nghiệp quản lý hoạt động bán hàng chủ yếu bằng phương pháp thủ công như ghi chép sổ sách, điều này dẫn đến nhiều khó khăn trong việc lưu trữ, tìm kiếm thông tin, theo dõi doanh thu và kiểm soát hàng hóa. Cùng với sự phát triển của công nghệ thông tin và internet, các phần mềm quản lý bán hàng ra đời đã mang lại nhiều lợi ích như tiết kiệm thời gian, giảm thiểu sai sót và nâng cao hiệu suất hoạt động kinh doanh.

Hiện nay, nhu cầu tin học hóa trong quản lý bán hàng ngày càng cao, đặc biệt là đối với các doanh nghiệp vừa và nhỏ. Việc ứng dụng phần mềm vào quá trình kinh doanh giúp tối ưu hóa quy trình, nâng cao khả năng kiểm soát, quản lý hàng tồn kho hiệu quả hơn và tạo điều kiện thuận lợi cho việc mở rộng hoạt động kinh doanh. Tuy nhiên, không phải doanh nghiệp nào cũng có điều kiện tiếp cận với một hệ thống quản lý phù hợp, nhiều đơn vị vẫn còn sử dụng phương pháp truyền thống gây ra những bất cập trong việc quản lý dữ liệu và báo cáo doanh thu.

Hiện nay, doanh nghiệp không chỉ cần lưu trữ dữ liệu mà còn cần hệ thống hỗ trợ phân tích hiệu suất để đưa ra quyết định chính xác. Phần mềm quản lý bán hàng giúp theo dõi doanh thu theo thời gian thực, kiểm tra tồn kho tự động và giảm sai sót thủ công.

Nhận thấy nhu cầu đó, nhóm đã chọn đề tài "Nghiên cứu xây dựng ứng dụng phần mềm quản lý bán hàng" nhằm xây dựng hệ thống hỗ trợ doanh nghiệp quản lý kinh doanh hiệu quả, đồng thời giúp nhóm áp dụng kiến thức vào thực tế và nâng cao kỹ năng chuyên môn.

Mặc dù có nhiều cố gắng nhưng do thời gian và trình độ còn hạn chế nên không thể tránh khỏi những thiếu sót. Rất mong nhận được sự góp ý của thầy cô, anh chị cùng các bạn để báo cáo được hoàn thiện hơn.

2. Mục đích và yêu cầu của đề tài

2.1 Mục đích

Xây dựng một ứng dụng phần mềm hỗ trợ quản lý bán hàng hiệu quả, dễ sử dụng và phù hợp với các cửa hàng, doanh nghiệp vừa và nhỏ. Tối ưu hóa quy trình quản lý hàng hóa và doanh thu. Ứng dụng các công nghệ hiện đại để đảm bảo tính bảo mật và khả năng mở rộng của hệ thống.

2.2 Yêu cầu

2.2.1 Yêu cầu chức năng của hệ thống

- Quản lý danh mục sản phẩm, đơn hàng, tồn kho, doanh thu và nhân viên.
- Hỗ trợ tìm kiếm và truy xuất dữ liệu nhanh chóng.
- Quản lý thông tin hóa đơn, thanh toán và xử lý giao dịch.
- Hệ thống báo cáo thống kê doanh thu, hàng tồn kho và lịch sử giao dịch.
- Tích hợp tính năng sao lưu và khôi phục dữ liệu nhằm đảm bảo an toàn.
- Quản lý quyền truy cập theo vai trò: chủ cửa hàng, nhân viên bán hàng, nhân viên kho.

2.2.2 Tính dễ sử dụng

- Giao diện trực quan, dễ sử dụng cho cả nhân viên và quản lý.
- Cho phép truy cập từ nhiều thiết bị khác nhau.
- Hỗ trợ hướng dẫn sử dụng và tài liệu tham khảo.

2.3 Hiệu suất

- Hệ thống hoạt động ổn định, có khả năng xử lý lượng dữ liệu lớn.

- Đảm bảo tốc độ truy xuất dữ liệu nhanh, hỗ trợ nhiều người dùng đồng thời.
- Hỗ trợ cập nhật và mở rộng hệ thống linh hoạt theo nhu cầu doanh nghiệp.

3. Đối tượng, phạm vi nghiên cứu

3.1 Đối tượng nghiên cứu

Hệ thống phần mềm quản lý bán hàng cho các mô hình quản lý kinh doanh tại các doanh nghiệp vừa và nhỏ.

3.2 Phạm vi nghiên cứu

Ứng dụng phần mềm quản lý bán hàng cho cửa hàng bán lẻ. Nghiên cứu và áp dụng công nghệ phù hợp để phát triển phần mềm.

4. Phương pháp nghiên cứu

A. Phương pháp nghiên cứu thực nghiệm:

- Tìm hiểu tài liệu về các hệ thống quản lý bán hàng hiện có.
- Khảo sát nhu cầu thực tế của doanh nghiệp về quản lý bán hàng.
- Áp dụng mô hình phát triển phần mềm phù hợp để thiết kế và triển khai hệ thống.

B. Ngôn ngữ sử dụng:

- .NET: kiến thức cần thiết: Framework ASP .NET
- Sql: SQL server
- HTML, CSS,
- JavaScript
- Framework/Library như: Bootstrap,..
- Sử dụng Git, Github
- Công cụ như Docker, ngôn ngữ định dạng Roff

5. Kết cấu đề tài

- Chương 1 : Tổng quan đề tài.
- Chương 2 : Yêu cầu về thiết kế chung
- Chương 3 : Nghiên cứu xây dựng ứng dụng quản lý bán hàng

6. Dự kiến kết quả

- Xây dựng thành công phần mềm quản lý bán hàng giúp doanh nghiệp tối ưu hóa quy trình quản lý.
- Hệ thống cho phép người dùng đăng nhập, phân quyền theo vai trò (quản trị viên, nhân viên) để thực hiện các tác vụ phù hợp.
- Hỗ trợ báo cáo doanh thu, kiểm soát hàng tồn kho và cải thiện độ chính xác trong quản lý dữ liệu.

7. Ý nghĩa khoa học và thực tiễn

Ứng dụng các kiến thức lập trình và công nghệ phần mềm vào thực tế. Giúp doanh nghiệp tiết kiệm thời gian, nâng cao hiệu suất làm việc và giảm thiểu rủi ro trong quản lý bán hàng.

Đề tài cung cấp một giải pháp hữu ích có thể mở rộng và ứng dụng rộng rãi trong thực tiễn.

Chương 1. TỔNG QUAN VỀ ĐỀ TÀI

1.1 Giới thiệu đề tài

Trong thời đại số hóa, quản lý bán hàng đóng vai trò then chốt đối với các doanh nghiệp, đặc biệt là cửa hàng vừa và nhỏ. Việc này không chỉ liên quan đến theo dõi doanh thu mà còn bao gồm quản lý kho, đơn hàng và các hoạt động tài

chính. Nhiều doanh nghiệp nhỏ vẫn dùng sổ sách hoặc Excel, gây khó khăn trong việc kiểm soát hàng hóa, thống kê doanh thu và phân tích hiệu quả kinh doanh.

Dự án "Nghiên cứu xây dựng ứng dụng phần mềm quản lý bán hàng" được triển khai nhằm giải quyết các hạn chế trên, cung cấp nền tảng hiện đại, dễ sử dụng và tích hợp đầy đủ chức năng: quản lý sản phẩm, đơn hàng, doanh thu, báo cáo và hỗ trợ ra quyết định dựa trên dữ liệu.

Hệ thống được thiết kế giao diện trực quan, dễ thao tác, tích hợp tự động hóa để giảm sai sót và tối ưu hóa quy trình làm việc. Qua đó, phần mềm sẽ giúp doanh nghiệp quản lý hiệu quả hơn và nâng cao năng lực cạnh tranh.

Bằng cách ứng dụng công nghệ phần mềm hiện đại, dự án này hứa hẹn mang đến một giải pháp toàn diện giúp doanh nghiệp quản lý bán hàng hiệu quả hơn, tối ưu hóa hoạt động kinh doanh và tăng cường khả năng cạnh tranh trong thị trường.

1.2 Tính cấp thiết của đề tài

Trong hoạt động kinh doanh của các doanh nghiệp nhỏ và cửa hàng bán lẻ tại Việt Nam nói chung, việc quản lý bán hàng vẫn chủ yếu dựa trên phương pháp thủ công, các công cụ đơn giản như sổ sách, Excel. Dẫn đến nhiều khó khăn trong việc theo dõi doanh thu, quản lý hàng tồn kho và xử lý đơn hàng.

Thực tế cho thấy, khi quy mô kinh doanh mở rộng, lượng dữ liệu giao dịch ngày càng lớn, việc kiểm soát thông tin bằng phương pháp thủ công không chỉ mất nhiều thời gian mà còn dễ xảy ra sai sót. Đặc biệt, quá trình tổng hợp doanh thu, phân tích hiệu suất bán hàng và dự báo tồn kho gặp nhiều hạn chế, ảnh hưởng đến việc ra quyết định kinh doanh kịp thời và chính xác.

Với mong muốn tối ưu hóa quy trình quản lý bán hàng, cũng như áp dụng các kiến thức chuyên ngành vào thực tiễn, nhóm chúng em đã lựa chọn đề tài "**Nghiên cứu xây dựng ứng dụng phần mềm quản lý bán hàng**". Đây sẽ là một hệ thống

phần mềm được thiết kế hiện đại, giúp doanh nghiệp quản lý bán hàng một cách tự động, chính xác và hiệu quả hơn.

1.3 Khảo sát hiện trạng

1.3.1 Hiện trạng quản lý bán hàng của doanh nghiệp/ cửa hàng nhỏ lẻ

Hiện nay, nhiều doanh nghiệp nhỏ và cửa hàng bán lẻ vẫn duy trì phương thức quản lý bán hàng thủ với các công cụ đơn giản như Excel. Quá trình bán hàng, theo dõi doanh thu và nhập kho phần lớn dựa vào sổ sách, dẫn đến hạn chế:

- **Khó khăn trong quản lý hàng tồn kho**
- **Mất thời gian trong việc tổng hợp doanh thu**
- **Thiếu công cụ phân tích kinh doanh:** Không có hệ thống báo cáo chi tiết về hiệu suất bán hàng, khó đưa ra quyết định kinh doanh dựa trên dữ liệu.

Những hạn chế trên cho thấy sự cần thiết của một hệ thống phần mềm quản lý bán hàng hiện đại, giúp tự động hóa và tối ưu hóa quy trình kinh doanh.

1.3.2 Các phương pháp quản lý bán hàng hiện có

Hiện nay, các doanh nghiệp và cửa hàng nhỏ lẻ thường sử dụng một trong ba phương pháp quản lý bán hàng sau:

- **Ghi chép thủ công:** đơn giản nhưng dễ sai sót, khó tra cứu và không phù hợp với quy mô lớn.
- **Dùng bảng tính (Excel, Google Sheets):** hỗ trợ tính toán tốt hơn nhưng vẫn phải nhập liệu thủ công, khó mở rộng.
- **Dùng phần mềm bán hàng:** như KiotViet, Sapo, Haravan..., tiện lợi và tự động hóa cao nhưng chi phí cao hoặc chưa phù hợp với nhu cầu từng doanh nghiệp nhỏ.

Dựa trên thực tế này, việc xây dựng một **phần mềm quản lý bán hàng** phù hợp với các doanh nghiệp nhỏ, dễ sử dụng, chi phí hợp lý và có khả năng mở rộng là rất cần thiết.

1.4 Yêu cầu đặt ra cho ứng dụng phần mềm quản lý bán hàng

1.4.1 *Yêu cầu lưu trữ dữ liệu*

- Hệ thống có khả năng lưu trữ thông tin sản phẩm, đơn hàng, nhân viên và các giao dịch bán hàng.
- Cơ sở dữ liệu đảm bảo tính bảo mật, toàn vẹn và có thể mở rộng khi quy mô doanh nghiệp phát triển.
- Hỗ trợ sao lưu dữ liệu định kỳ để tránh mất mát thông tin quan trọng.

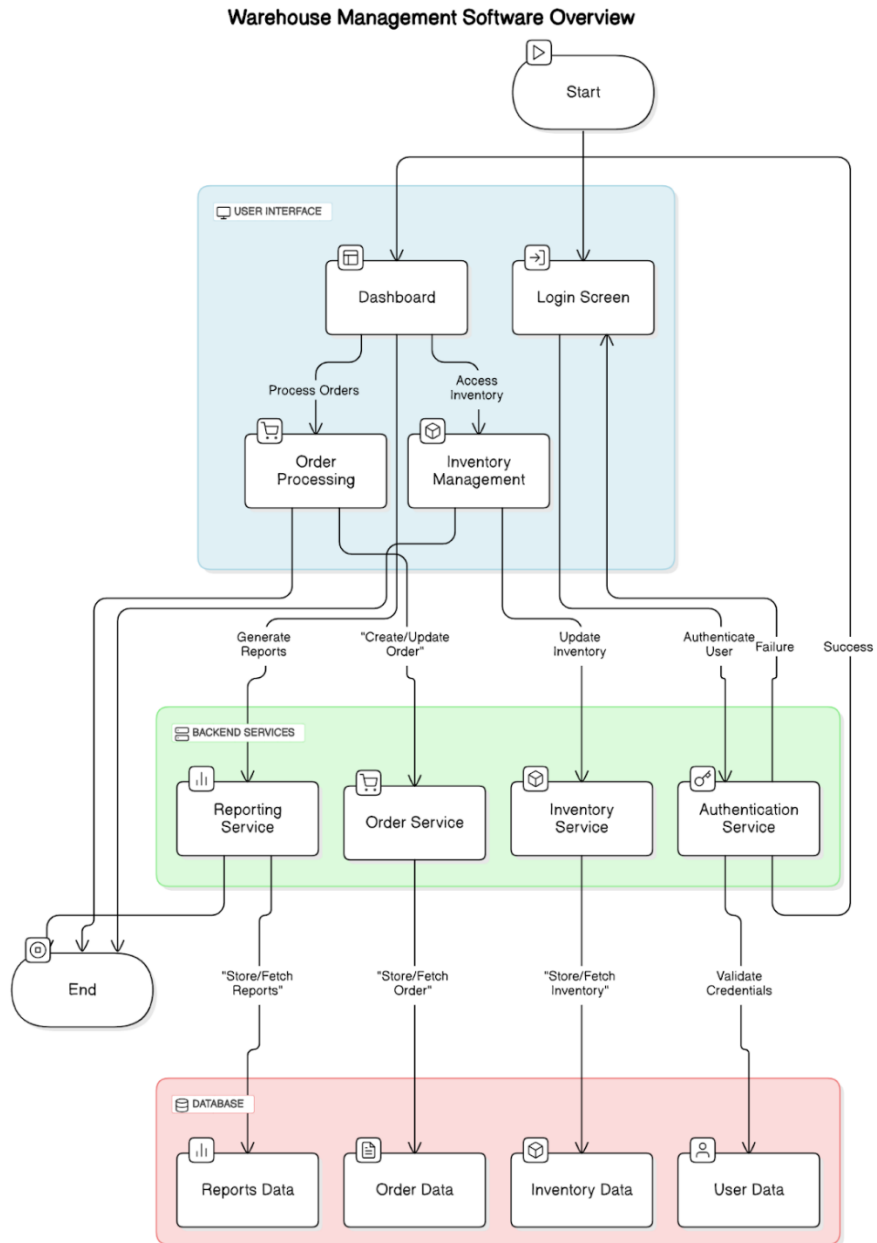
1.4.2 *Yêu cầu tra cứu và báo cáo*

Hệ thống cần hỗ trợ tìm kiếm nhanh thông tin sản phẩm, đơn hàng.

Cho phép lọc dữ liệu theo nhiều tiêu chí như thời gian, doanh thu, số lượng hàng bán chạy...

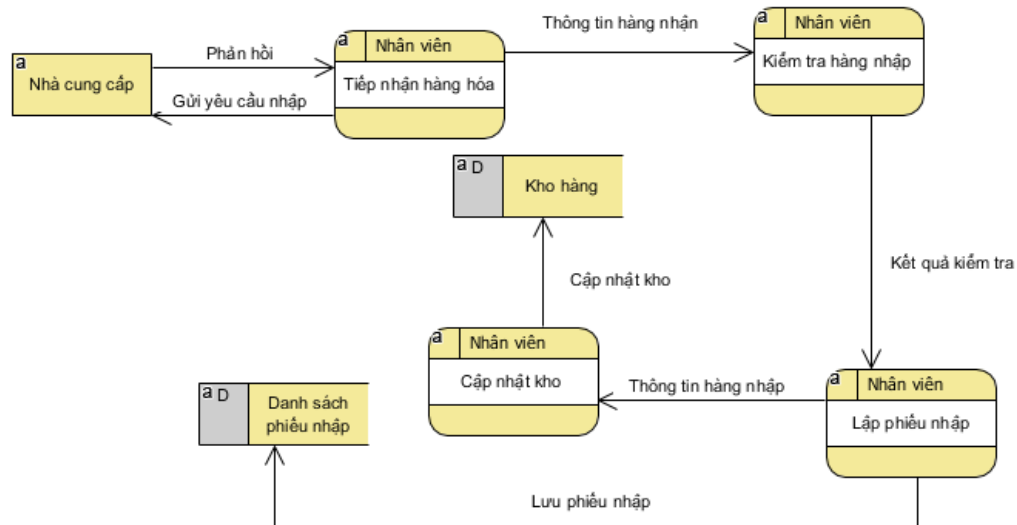
Hỗ trợ xuất báo cáo tổng hợp về doanh thu, hàng tồn kho, hiệu suất nhân viên dưới dạng biểu đồ, bảng biểu trực quan.

1.4.3 Sơ đồ tổng thể dự án



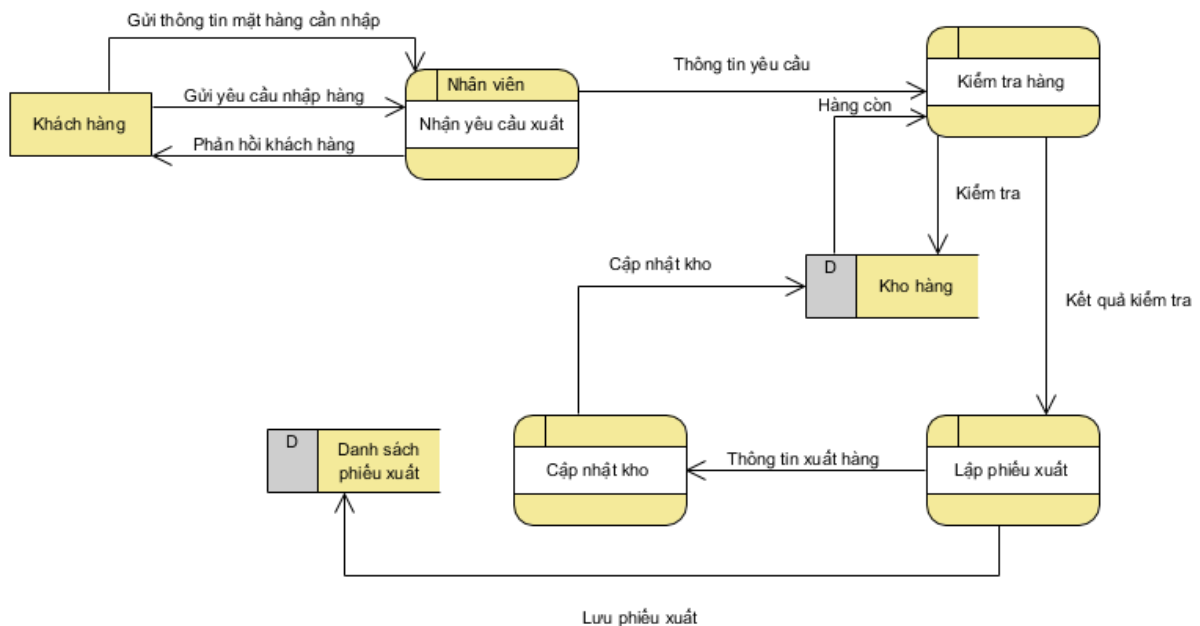
1.4.4 Yêu cầu về các chức năng chính

- **Quản lý sản phẩm:** Thêm, sửa, xóa sản phẩm, theo dõi số lượng hàng tồn kho.



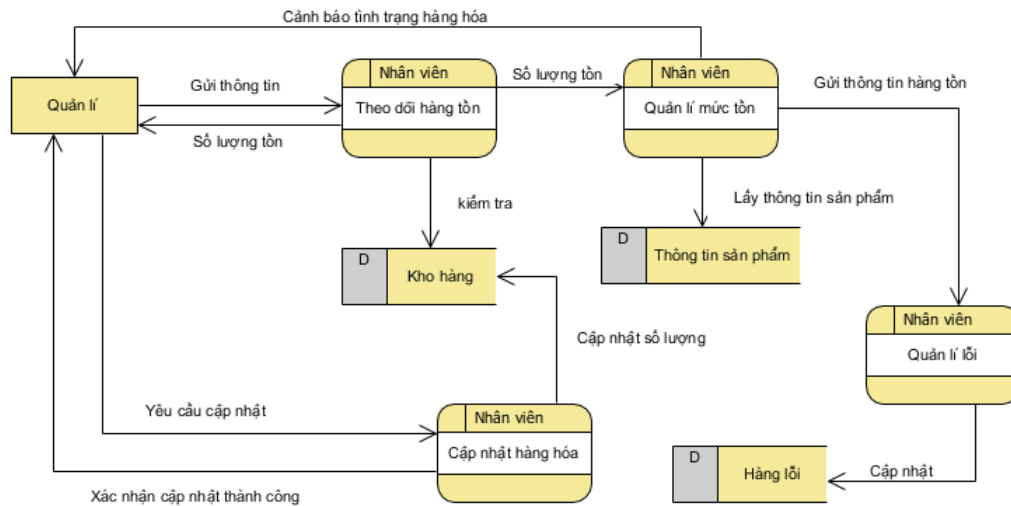
Hình 1.4.4.1: biểu đồ mô tả quản lý sản phẩm

- **Quản lý đơn hàng:** Tạo, cập nhật trạng thái đơn hàng, in hóa đơn.



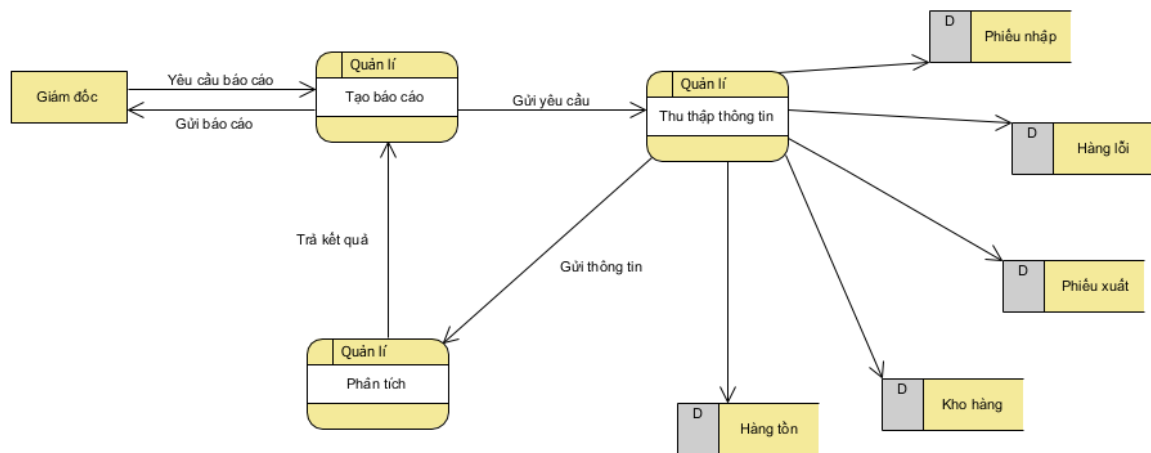
Hình 1.4.4.2: biểu đồ mô tả quản lý đơn hàng

- **Quản lý nhân viên:** Phân quyền sử dụng hệ thống cho nhân viên bán hàng.



Hình 1.4.4.3: biểu đồ mô tả quản lý nhân viên

- **Báo cáo và phân tích:** Cung cấp các báo cáo doanh thu, hàng bán chạy, quản lý công nợ.



Hình 1.4.4.4: biểu đồ mô tả quản lý báo cáo và phân tích

1.4.5 Yêu cầu phi chức năng

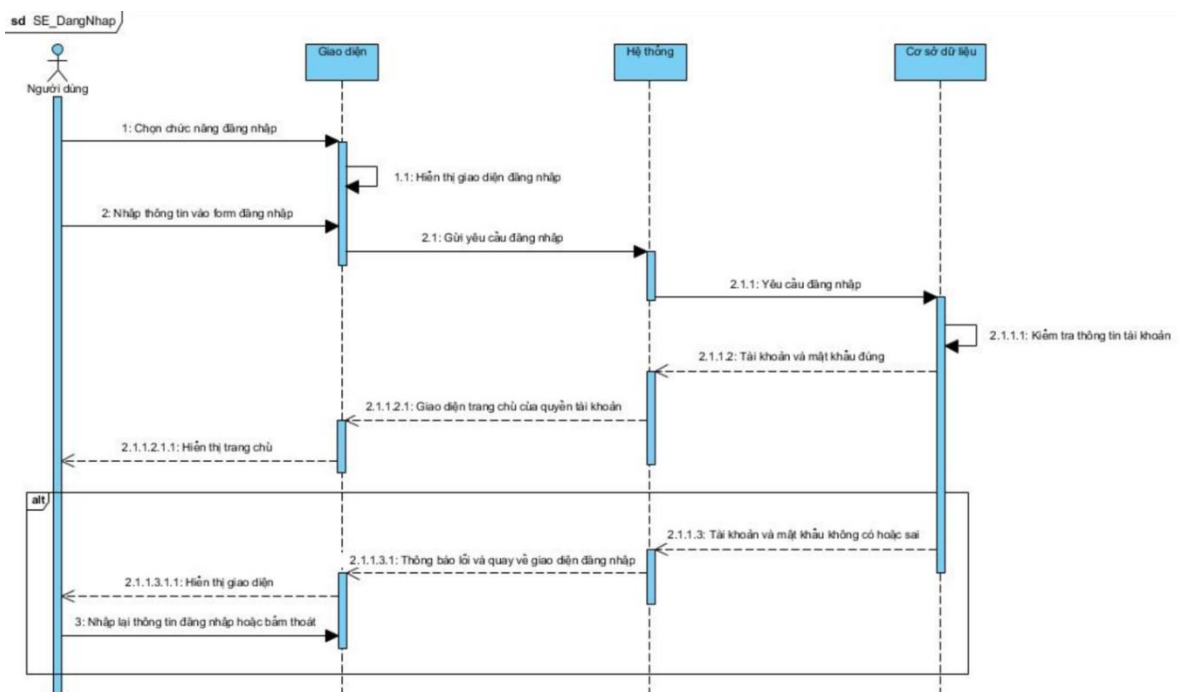
Các yêu cầu phi chức năng đảm bảo hệ thống hoạt động hiệu quả, dễ sử dụng và phù hợp triển khai thực tế. Với hệ thống quản lý bán hàng, yêu cầu chính gồm:

1. Giao Diện Thân Thiện, Dễ Sử Dụng

- Thiết kế trực quan (Intuitive UI), dễ hiểu, dễ thao tác
- Hướng đến người dùng không chuyên: biểu tượng, màu sắc rõ ràng
- Ngôn ngữ hiển thị rõ ràng
- Giao diện thống nhất, đồng bộ về bố cục, phong cách.

2. Tính Bảo Mật Cao

- Phân quyền người dùng:
 - Hệ thống phân chia người dùng theo vai trò (admin, nhân viên bán hàng, quản lý kho,...), mỗi vai trò có quyền truy cập khác nhau.



Hình 1.4.5.1: biểu đồ tuần tự mô tả quy trình người dùng truy cập, chỉnh sửa thông tin

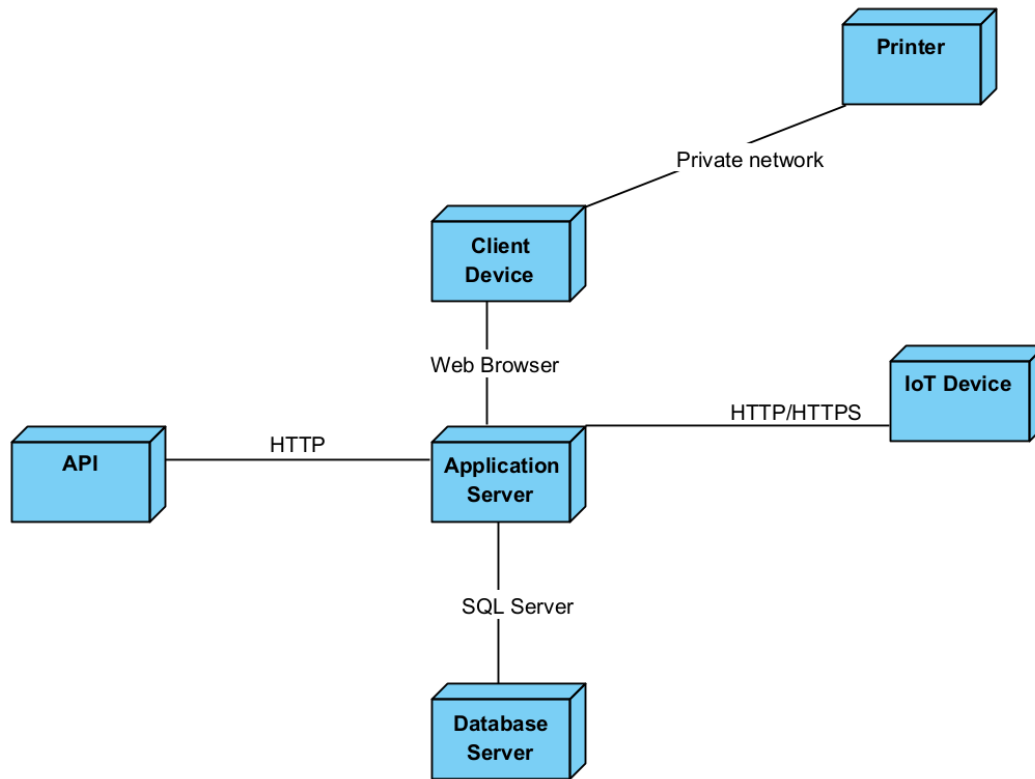
- Ví dụ:
 - **Admin:** Toàn quyền truy cập, thêm/sửa/xóa dữ liệu, tài khoản.
 - **Nhân viên:** Chỉ truy cập các chức năng bán hàng, xem tồn kho.

- **Xác thực và kiểm soát truy cập:** Đăng nhập bằng mật khẩu; hỗ trợ 2FA cho phiên bản nâng cao
- **Mã hóa dữ liệu:**
 - Thông tin quan trọng (tài khoản, giao dịch, mật khẩu) được mã hóa trước khi lưu trữ vào cơ sở dữ liệu.
 - Mật khẩu người dùng sử dụng thuật toán băm (hash) như SHA-256 hoặc bcrypt.
- **Lưu vết hoạt động (Audit trail):** Ghi lại các hành động quan trọng như đăng nhập, xóa đơn hàng, cập nhật giá... giúp theo dõi và truy vết sự cố.

3. Hiệu Suất Tốt

- **Tốc độ xử lý nhanh:** Phản hồi <1 giây với thao tác thường (tìm kiếm sản phẩm, tạo hóa đơn, in phiếu,...)
- **Xử lý khối lượng dữ liệu lớn:** Vận hành ổn định với hàng ngàn giao dịch/ngày.
- **Tối ưu hóa truy vấn cơ sở dữ liệu:** Dùng chỉ mục, tránh gọi dữ liệu thừa.
- **Tải trang nhẹ:** Thiết kế tối ưu ảnh, nén CSS/JS, chuyển trang nhanh.

Mô hình triển khai:



Hình 1.4.5.2: Mô hình kiến trúc hệ thống phần mềm với các thành phần chính và giao thức kết nối.

4. Tương Thích Đa Nền Tảng

- **Hỗ trợ đa thiết bị:**
 - Hệ thống có thể truy cập và sử dụng trên các thiết bị như:
 - Máy tính để bàn/laptop: Giao diện chi tiết, đầy đủ tính năng quản trị.
 - Máy tính bảng: Linh hoạt, thao tác nhanh trong kho/cửa hàng.
 - Điện thoại: Giao diện rút gọn, tập trung vào tạo hóa đơn, kiểm tra tồn kho.
- **Responsive Design:** Giao diện tự động điều chỉnh theo độ phân giải màn hình. Đảm bảo không bị lỗi hiển thị hoặc vỡ bố cục trên các thiết bị di động.

- **Tương thích trình duyệt:** Hoạt động ổn định trên Google Chrome, Firefox, Edge và Safari.
- **Khả năng mở rộng ứng dụng di động:** Có thể tích hợp app riêng hoặc phát triển native/hybrid trong tương lai.

Chương 2. YÊU CẦU VỀ THIẾT KẾ CHUNG

2.1 Yêu cầu chung về giao diện

Hạng mục xây dựng hệ thống kỹ thuật:

Hệ thống kỹ thuật của phần mềm quản lý bán hàng cần được xây dựng với các yêu cầu và tiêu chí sau:

- **Hỗ trợ toàn diện quy trình bán hàng:** Quản lý sản phẩm, giỏ hàng, đơn hàng, khách hàng, thanh toán và xuất hóa đơn; thao tác thêm/sửa/xóa nhanh chóng, chính xác.
- **Giao diện người dùng trực quan và nhất quán:** Thiết kế rõ ràng, dễ sử dụng, bố cục logic, tối ưu thao tác cho nhân viên và quản lý.
- **Tích hợp với các hệ thống khác:** Hỗ trợ kết nối với phần mềm kế toán, thanh toán điện tử, vận chuyển, CRM để đồng bộ dữ liệu.
- **Tùy chỉnh và mở rộng:** Hỗ trợ thêm tính năng như khuyến mãi, công nợ, phân tích hành vi khách hàng, triển khai đa chi nhánh.
- **Bảo mật và phân quyền rõ ràng:** Phân quyền người dùng theo vai trò; bảo vệ dữ liệu hiệu quả.
- **Đảm bảo hiệu năng và ổn định:** Xử lý nhanh với dữ liệu lớn, đảm bảo hoạt động mượt trong giờ cao điểm.
- **Tuân thủ các tiêu chuẩn công nghệ hiện đại:** Phát triển trên nền tảng web (MVC, REST API, SPA), tương thích đa thiết bị, sẵn sàng triển khai trên nền tảng đám mây.

Điều khiển giao diện:

Trong hệ thống quản lý kho, các điều khiển giao diện giúp người dùng thao tác nhanh, trực quan và hiệu quả. Thành phần UI được chọn theo chuẩn HTML/CSS, dễ mở rộng và tích hợp trên nền web hiện đại.

- **Button (Nút nhấn):** Thêm, sửa, xóa, tìm kiếm, xuất báo cáo
- **Textbox (Ô nhập liệu):**
- **Dropdownlist (Danh sách chọn):**
- **Radio Button / Checkbox:**
 - **Radio Button:** Dùng trong các lựa chọn duy nhất, ví dụ: *Chọn trạng thái hàng hóa* (Mới / Đã qua sử dụng).
 - **Checkbox:** Dùng khi có thể chọn nhiều, như *Chọn nhiều mục hàng để xuất kho*.
- **Listbox / Table (Danh sách):** Hiển thị và lọc danh sách hàng hóa
- **Modal Dialog / Popup:** Xác nhận các thao tác quan trọng
- **Thanh tìm kiếm và bộ lọc: Tìm và lọc nhanh theo tiêu chí**
 - Tất cả các điều khiển được thiết kế đồng nhất về màu sắc, font chữ và kích thước.

2.2 Yêu cầu về màn hình giao diện hệ thống

2.2.1. Độ Phân Giải Màn Hình

Chọn độ phân giải tối thiểu 1024x768 pixels để đảm bảo hiển thị rõ ràng văn bản và hình ảnh, tương thích tốt với thiết bị văn phòng. Giao diện cần tự điều chỉnh để hiển thị tốt trên các màn hình có độ phân giải cao hơn.

2.2.2 Lợi Ích

- Hiển thị rõ nét, giảm mỏi mắt.
- Phù hợp với hầu hết thiết bị văn phòng.

2.3 Nền Ngoài (Background)

Nền ngoài của giao diện: môi trường làm việc thoải mái, không gây phân tâm.

2.4 Font Chữ

Dùng "Lucida Grande", Arial, sans-serif – phổ biến, dễ đọc, tương thích tốt với các trình duyệt và độ phân giải khác nhau.

2.5 Cỡ Chữ

2.5.1 Mục Tiêu

Cỡ chữ đóng vai trò quan trọng trong việc xác định mức độ dễ đọc của giao diện. Cỡ chữ cần phải vừa đủ để người dùng có thể đọc được thông tin mà không bị mỏi mắt.

2.5.2 Yêu Cầu Giao Diện

Cỡ chữ bình thường 12px: Đây là cỡ chữ lý tưởng, không quá lớn để chiếm diện tích màn hình nhưng vẫn đảm bảo độ dễ đọc cho người sử dụng. Cỡ chữ này giúp tạo sự đồng đều và dễ nhìn trong toàn bộ giao diện.

Cỡ chữ có thể thay đổi: Trong các trường hợp đặc biệt (ví dụ: phần tiêu đề, hoặc báo cáo quan trọng), cỡ chữ có thể được điều chỉnh lớn hơn để làm nổi bật thông tin quan trọng.

2.5.3 Lợi Ích

Giảm mỏi mắt: Cỡ chữ 12px dễ đọc, không chiếm diện tích.

Tính đồng nhất: Sử dụng cỡ chữ đồng nhất giúp giao diện dễ nhìn và dễ dàng cho làm quen, tạo trải nghiệm liền mạch.

2.6 Hiện Thị Banner

2.6.1 Mục Tiêu

Banner là phần hình ảnh quan trọng, nhận diện thương hiệu khi họ truy cập vào phần mềm. Banner còn giúp thu hút sự chú ý của người dùng, tạo ấn tượng. Việc hiển thị banner giúp tạo sự liên kết và nhất quán trải nghiệm người dùng.

2.6.2 Yêu Cầu Giao Diện

- Đặt tại đầu trang, dưới thanh điều hướng.
- Kích thước vừa phải, nội dung gồm logo, slogan, hình ảnh rõ nét.
- Có thể là slideshow nếu cần, nhưng không gây phiền nhiễu.
- Responsive trên mọi thiết bị.

Tính tương thích:

Banner cần được thiết kế sao cho tự động điều chỉnh (responsive) khi hiển thị trên các thiết bị có độ phân giải khác nhau, bao gồm máy tính để bàn, laptop, và điện thoại di động.

2.6.3 Lợi Ích

- Người dùng dễ tìm thông tin hỗ trợ.
- Duy trì sự thống nhất trong giao diện.
- Tăng tính tiện dụng khi điều hướng.

2.7 Hiện Thị Footer

2.7.1 Mục Tiêu

Footer là phần quan trọng của giao diện, cung cấp các thông tin hỗ trợ người dùng như các liên kết liên hệ, các điều khoản sử dụng, chính sách bảo mật, và các thông tin khác mà người dùng có thể cần. Footer sẽ xuất hiện trên toàn bộ các trang

trong hệ thống để người dùng dễ dàng tìm thấy thông tin này mà không phải tìm kiếm hoặc chuyển trang.

2.7.2 Yêu Cầu Giao Diện

Vị trí footer: Hiện thị ở cuối mỗi trang, kéo dài theo chiều rộng màn hình, dễ tiếp cận sau khi xem xong nội dung chính.

Nội dung footer:

- **Liên hệ:** Gồm email, số điện thoại và các phương thức liên hệ khác.
- **Điều khoản & chính sách:** Liên kết đến các trang về chính sách bảo mật, điều khoản sử dụng, quyền lợi người dùng.
- **Liên kết khác:** Bao gồm FAQ, hỗ trợ trực tuyến, giới thiệu công ty hoặc dịch vụ khách hàng.
- **Định dạng:** Font chữ nhỏ hơn nội dung chính, nhưng liên kết cần dễ nhận diện và thao tác.
- **Tương thích:** Footer hiện thị đúng trên các độ phân giải khác nhau, bố trí liên kết hợp lý, dễ truy cập.

2.7.3 Lợi Ích

- Tập hợp thông tin quan trọng, giúp người dùng dễ dàng truy cập.
- Đảm bảo tính nhất quán giao diện trên toàn bộ phần mềm.
- Tăng tính tiện lợi khi cung cấp liên kết hỗ trợ ngay cuối trang.

3 Danh sách giao diện

STT	Mã Giao Diện	Tên giao diện	Mô Tả Chi Tiết
1	GD001	Đăng Nhập	- Logo và tên hệ thống: Hiện thị phía trên cùng để dễ nhận diện.

			- Trường nhập thông tin: Gồm ô tên đăng nhập và mật khẩu, có tùy chọn ẩn/hiện mật khẩu. - Nút đăng nhập: Gửi yêu cầu xác thực. - Quên mật khẩu: Liên kết khôi phục. - Thông báo lỗi: Khi sai thông tin, hệ thống hiện lỗi rõ ràng.
2	GD002	Đăng Xuất	- Xác nhận đăng xuất: Hộp thoại xác nhận. - Nút đăng xuất: Kết thúc phiên làm việc. - Thông báo đăng xuất: Xác nhận thành công. - Chuyển hướng: Quay về giao diện đăng nhập hoặc trang chủ.
3	GD003	Kiểm kê hàng hoá	- Giao diện cho phép kiểm kê hàng thực tế trong kho. - Hiển thị danh sách hàng tồn, số lượng hiện có, số lượng kiểm thực tế. - Cho phép cập nhật số lượng và ghi chú nếu có sai lệch.
4	GD004	Tìm kiếm hàng hoá	- Thanh tìm kiếm theo mã sản phẩm, tên, danh mục. - Hiển thị kết quả theo dạng bảng với các cột thông tin: mã, tên, giá, tồn kho, nhà cung cấp. - Tương tác trực tiếp để chọn hoặc chỉnh sửa nhanh.
5	GD005	Danh sách nhà cung cấp	- Bảng danh sách: Thể hiện thông tin các nhà cung cấp. - Tìm kiếm: Theo tên hoặc tiêu chí khác.

			- Chức năng thêm/sửa/xóa: Qua nút thao tác và các form chi tiết. - Xác nhận xóa: Tránh thao tác nhầm. - Thông báo kết quả: Thành công/thất bại.
6	GD006	Danh sách khách hàng	- Bảng hiển thị: Tên, địa chỉ, điện thoại, email của khách hàng. - Tìm kiếm: Theo tên hoặc các tiêu chí. - Chức năng thêm/sửa/xóa: Qua các form nhập dữ liệu. - Xác nhận thao tác: Đảm bảo chắc chắn khi xóa. - Thông báo: Hiện rõ ràng trạng thái thao tác.
7	GD007	Quản lý tài khoản khách hàng	- Danh sách tài khoản: Hiện thị tên đăng nhập, vai trò, trạng thái. - Thêm/sửa tài khoản: Qua form chi tiết, có phần thiết lập quyền. - Phân quyền: Chọn vai trò (Admin, Nhân viên,...). - Xác nhận và thông báo: Hiện thị khi thao tác thành công/thất bại.
8	GD008	Báo Cáo	- Thống kê doanh thu: Theo ngày, tháng, năm hoặc theo sản phẩm. - Báo cáo tồn kho: Hiện thị hàng tồn quá mức hoặc sắp hết. - Biểu đồ: Dạng cột, tròn giúp dễ hình dung. - Tải xuống: Có nút xuất file (PDF, Excel). - Sửa/xóa báo cáo tùy chọn (nếu cho phép).

Chương 3. NGHIÊN CỨU XÂY DỰNG ỨNG DỤNG QUẢN LÝ BÁN HÀNG

3.1 Ngôn ngữ lập trình

3.1.1 *Ngôn ngữ C#*

C# là ngôn ngữ lập trình hướng đối tượng hiện đại do Microsoft phát triển, ra mắt năm 2000 cùng nền tảng .NET. Nó kết hợp sức mạnh của C++ với sự an toàn và đơn giản của Java, tích hợp với hệ sinh thái Microsoft, phù hợp cho phát triển ứng dụng desktop, web, mobile (Xamarin), và đa nền tảng (.NET Core, .NET 5+).

C# có cú pháp quen thuộc với lập trình viên C-style, hỗ trợ quản lý bộ nhớ tự động, hệ thống kiểu mạnh, lập trình bất đồng bộ, biểu thức lambda, LINQ, và nhiều tính năng hiện đại. Ngôn ngữ này cho phép xây dựng ứng dụng từ hệ thống nhỏ đến quy mô doanh nghiệp một cách rõ ràng, hiệu quả và dễ bảo trì.

Đặc điểm chính của C#:

- **Hướng đối tượng hoàn toàn (pure OOP)**
- **Hệ thống kiểu mạnh và an toàn:** tránh lỗi thời gian chạy
- **Quản lý bộ nhớ tự động**
- **Hỗ trợ lập trình bất đồng bộ:** với async/await, dễ dàng xử lý I/O, thao tác mạng, giao diện người dùng mà không block luồng chính.
- **Tích hợp LINQ:** cho phép viết truy vấn dữ liệu trong ngôn ngữ
- **Tính mở rộng:** hỗ trợ generic, delegate, event, extension method pattern matching.
- **Cập nhật thường xuyên**

Ứng dụng thực tế:

- **Ứng dụng desktop (Windows Forms, WPF)**
- **Ứng dụng web ASP.NET, Web API**

- **Ứng dụng mobile: Xamarin, MAUI**
- **Game: Unity Engine**
- **Cloud và microservice :Azure, .NET Core:**
- **AI, IoT, và khoa học dữ liệu**

C# được doanh nghiệp ưa chuộng nhờ tính ổn định, dễ bảo trì, và tích hợp tốt với hệ thống Microsoft như SQL Server, Active Directory, SharePoint.

3.1.2 Ngôn ngữ HTML

HTML (HyperText Markup Language) là một ngôn ngữ đánh dấu được sử dụng để tạo và cấu trúc nội dung trên các trang web. Nó là một ngôn ngữ giao diện cho trình duyệt web, giúp hiển thị thông tin và tạo cấu trúc cho các phần tử trên trang.

HTML sử dụng các thẻ (tag) để xác định các phần tử trên trang web. Mỗi thẻ bắt đầu bằng dấu nháy mở `<` và kết thúc bằng dấu nháy đóng `>`. Ví dụ, thẻ `<h1>` và `</h1>` được sử dụng để định dạng tiêu đề cấp 1 trong HTML. Các thẻ có thể chứa văn bản, hình ảnh, liên kết và nhiều phần tử khác.

3.1.3 Ngôn ngữ CSS

CSS (Cascading Style Sheets) là ngôn ngữ dùng để định dạng các phần tử do HTML tạo ra, chịu trách nhiệm tạo phong cách cho trang web. CSS xác định thuộc tính như màu sắc, kích thước, vị trí, đường viền, từ đó xây dựng giao diện và trải nghiệm người dùng.

HTML và CSS gắn kết chặt chẽ: HTML là cấu trúc nội dung, CSS định hình giao diện. CSS có thể được viết trực tiếp trong HTML, liên kết từ file riêng, hoặc sử dụng framework như Bootstrap.

CSS cho phép kiểm soát toàn diện giao diện, từ văn bản đến bố cục và hiệu ứng, với tính linh hoạt và khả năng tái sử dụng cao, giúp tạo giao diện đẹp và đồng nhất trên nhiều trang và thiết bị.

Tóm lại, CSS là công cụ thiết yếu để tùy chỉnh giao diện, nâng cao trải nghiệm và tính tương tác của trang web.

3.1.4 Ngôn ngữ JavaScript

JavaScript là ngôn ngữ lập trình thông dịch, cấp cao, đa mô hình, phát triển ban đầu để tạo tương tác cho trình duyệt web. Từ nền tảng đơn giản, JavaScript đã trở thành ngôn ngữ quan trọng và phổ biến trong ngành phần mềm hiện đại.

JavaScript cùng với HTML và CSS tạo nên bộ ba cốt lõi của frontend: HTML xây dựng cấu trúc, CSS định dạng giao diện, JavaScript điều khiển hành vi và logic tương tác. Với Node.js, JavaScript mở rộng sang backend và server-side, trở thành ngôn ngữ toàn diện dùng cho cả frontend và backend trong hệ thống full-stack.

3.1.5 Framework/Library

3.1.1 ASP.NET Core

ASP.NET Core là nền tảng phát triển ứng dụng web mã nguồn mở, đa nền tảng do Microsoft xây dựng, thay thế ASP.NET truyền thống. Nó hỗ trợ xây dựng ứng dụng web hiện đại, API, realtime (SignalR) và backend di động với hiệu suất cao, bảo mật tích hợp.

Hoạt động trên Windows, Linux, macOS, tích hợp với .NET Runtime, cho phép xây dựng theo MVC, Razor Pages hoặc RESTful API. Không phụ thuộc IIS, có thể chạy độc lập hoặc trong Docker, Nginx.

Đặc điểm:

- Đa nền tảng, hiệu năng cao, kiến trúc modular qua NuGet.
- Dependency Injection tích hợp sẵn, middleware pipeline linh hoạt.

- Hỗ trợ Razor, Blazor, Web API, SignalR.
- Routing linh hoạt (attribute và convention).
- Tích hợp bảo mật: authentication, authorization, OAuth2, OpenID Connect.
- Hosting qua Kestrel hoặc reverse proxy.

Cấu trúc ứng dụng:

- Program.cs: cấu hình khởi tạo WebApplication (.NET 6+).
- Startup.cs: cấu hình dịch vụ và middleware (.NET 5 trở về trước).
- Controllers/: chứa controller MVC hoặc API.
- Models/: dữ liệu và logic nghiệp vụ.
- Views/: giao diện Razor (nếu dùng MVC).

Ứng dụng phù hợp:

- Backend Web API cho di động, web client, SPA.
- Ứng dụng MVC, Razor Pages truyền thống.
- Ứng dụng realtime với SignalR.
- Microservices, hệ thống phân tán, container hóa (Docker/Kubernetes).
- Ứng dụng doanh nghiệp như ERP, CRM, CMS.
- RESTful services tích hợp frontend JS (React, Vue).

3.1.6 Công cụ định dạng tài liệu Roff

Roff (Runoff) là ngôn ngữ định dạng văn bản (markup language) phát triển từ những năm 1960-1970, ban đầu dùng trong các hệ thống máy tính lớn và sau trở thành phần quan trọng của hệ sinh thái Unix. Roff không phải ngôn ngữ lập trình mà là ngôn ngữ điều khiển định dạng văn bản qua các chỉ thị bắt đầu bằng dấu chấm (.) hoặc dấu nháy đơn ('). Mục đích là định dạng và trình bày văn bản để in hoặc hiển thị với tính nhất quán và kiểm soát chi tiết.

Trong Unix, Roff thường dùng để soạn thảo trang manual (man page), tài liệu hướng dẫn phần mềm hay lệnh hệ thống. Dù là văn bản thuần túy, các tệp này dùng chỉ thị Roff để hiển thị tiêu đề, đoạn văn, chữ in đậm, in nghiêng, danh sách và khối mã rõ ràng.

Roff là một ngôn ngữ định dạng dòng đơn (line-based formatting language), trong đó mỗi chỉ thị được bắt đầu bằng dấu chấm (.) hoặc dấu nháy đơn ('). Mỗi dòng trong tệp Roff có thể là một dòng văn bản thông thường hoặc một chỉ thị định dạng.

Cấu trúc cơ bản của một chỉ thị Roff:

`.<macro> <arguments>`

Đặc điểm chính:

- Định dạng dựa trên dòng, xử lý độc lập từng dòng.
- Không cần trình biên dịch, được diễn giải bởi nroff hoặc troff.
- Hỗ trợ các chỉ thị điều khiển định dạng như .SH (tiêu đề), .B (in đậm), .I (in nghiêng).
- Hỗ trợ macro để tái sử dụng định dạng hoặc khối văn bản phức tạp.
- Khả năng mở rộng qua các gói macro như man, ms, me, phục vụ các mục đích viết tài liệu, bài báo, tài liệu kỹ thuật.

3.1.7 Công cụ quản lý source code

3.1.7.1 Git

Git là hệ thống quản lý phiên bản phân tán (DVCS) phổ biến, cho phép mỗi lập trình viên có kho lưu trữ riêng chứa toàn bộ lịch sử thay đổi. Git giúp nhóm phát triển làm việc đồng thời trên cùng dự án mà không ghi đè lên nhau, bằng cách tạo các nhánh riêng và hợp nhất chúng vào nhánh chính (master/main).

Các lợi ích của Git:

- Theo dõi thay đổi mã nguồn theo thời gian, dễ dàng quay lại phiên bản trước.

- Hỗ trợ làm việc nhóm, tạo và hợp nhất nhánh cho tính năng hoặc sửa lỗi.
- Quản lý xung đột khi nhiều người sửa cùng một đoạn code.
- Lưu chi tiết lịch sử thay đổi, bao gồm tác giả và lý do sửa.
- Hỗ trợ code review qua các nền tảng như GitHub, GitLab.
- Cho phép rollback khi lỗi phát sinh, bảo vệ sự ổn định.
- Mỗi thành viên có bản sao đầy đủ, tránh mất dữ liệu.
- Hoạt động offline, thuận tiện cho nhóm làm việc phân tán.
- Mã nguồn mở, hỗ trợ hợp tác trên các dự án công khai.
- Linh hoạt, có thể áp dụng cho dự án nhỏ đến doanh nghiệp lớn.
- Cho phép tùy chỉnh quy trình làm việc phù hợp nhu cầu.

3.1.7.2 Github

GitHub là dịch vụ lưu trữ mã nguồn dựa trên Git, cho phép người dùng tạo kho riêng, quản lý nhánh, gửi pull request và làm việc nhóm hiệu quả. GitHub hỗ trợ xem, sao chép và đóng góp vào các dự án mã nguồn mở. Git là hệ thống quản lý phiên bản phân tán, còn GitHub là nền tảng chia sẻ và hợp tác phát triển dựa trên Git, giúp quản lý mã nguồn và phối hợp nhóm dễ dàng hơn.

3.1.7.3 Gitflow

Gitflow là quy trình phân nhánh được phát triển bởi Vincent Driessen, thiết lập cấu trúc rõ ràng quanh các bản release dự án. Nó giúp quản lý các môi trường development, staging và production tách biệt, hỗ trợ QA, feedback và xử lý lỗi hiệu quả. Gitflow giúp nhóm làm việc song song mà không gây xung đột, thuận tiện theo dõi, review và xử lý các vấn đề phát sinh từ tính năng hay bản sửa lỗi, đảm bảo quá trình phát triển và deploy được kiểm soát chặt chẽ.

3.1.8 Công cụ triển khai container

3.1. Docker

Docker là nền tảng phát triển và chạy ứng dụng trong container, tạo môi trường độc lập, dễ deploy trên mọi server.

Docker hoạt động như hệ điều hành chuyên biệt cho container, thay thế ảo hóa phần cứng bằng ảo hóa hệ điều hành, cung cấp lệnh dựng, khởi động, dừng container đơn giản.

Lợi ích khi sử dụng Docker:

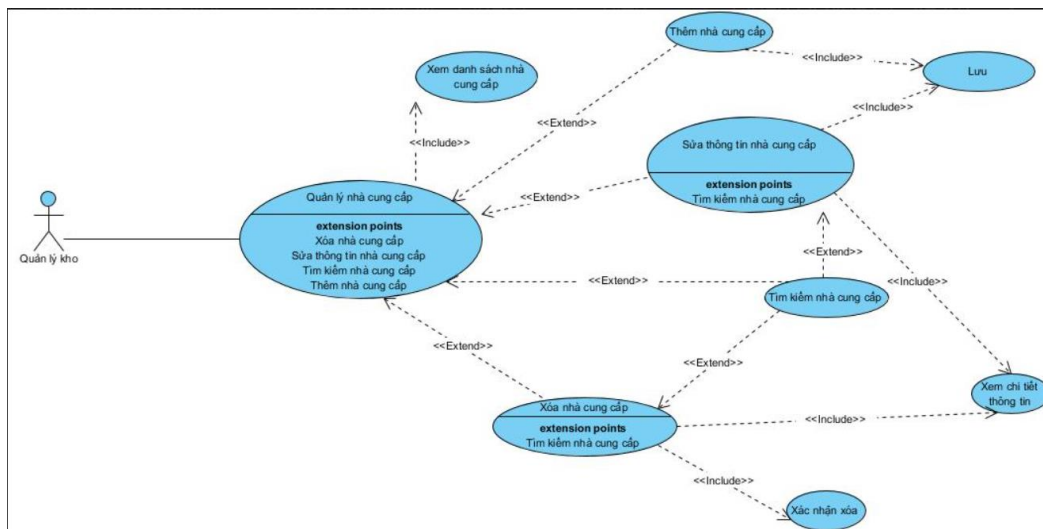
- Khởi động, dừng nhanh hơn máy ảo.
- Xây dựng và loại bỏ container nhanh.
- Thiết lập môi trường một lần, tái sử dụng dễ dàng.
- Giữ workspace sạch, tránh ảnh hưởng đến các phần khác khi xóa container.

3.2. Phân tích và thiết kế hệ thống

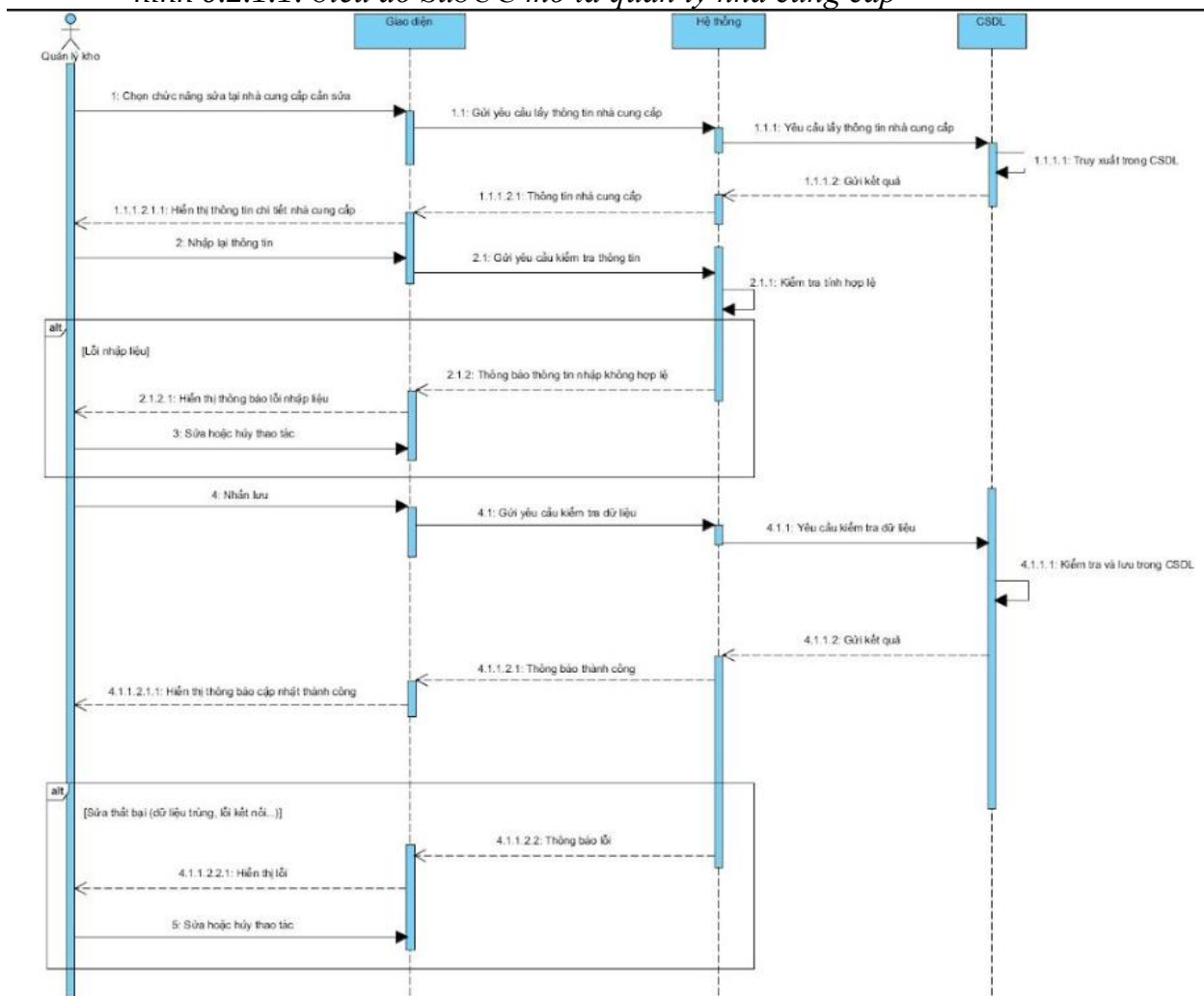
3.2.1 Mô tả hệ thống

Hệ thống quản lý kho hàng là một giải pháp phần mềm được xây dựng nhằm hỗ trợ các tổ chức, doanh nghiệp trong việc quản lý hoạt động nhập kho, xuất kho, theo dõi tồn kho và kiểm soát thông tin hàng hóa. Hệ thống này cung cấp các chức năng cơ bản như:

- **Quản lý hàng hóa:** Cho phép thêm mới, chỉnh sửa, xóa thông tin hàng hóa như mã hàng, tên hàng, đơn vị tính, số lượng, vị trí lưu trữ.
- **Quản lý danh mục:** Gồm các danh mục như loại hàng, nhà cung cấp, đơn vị tính.

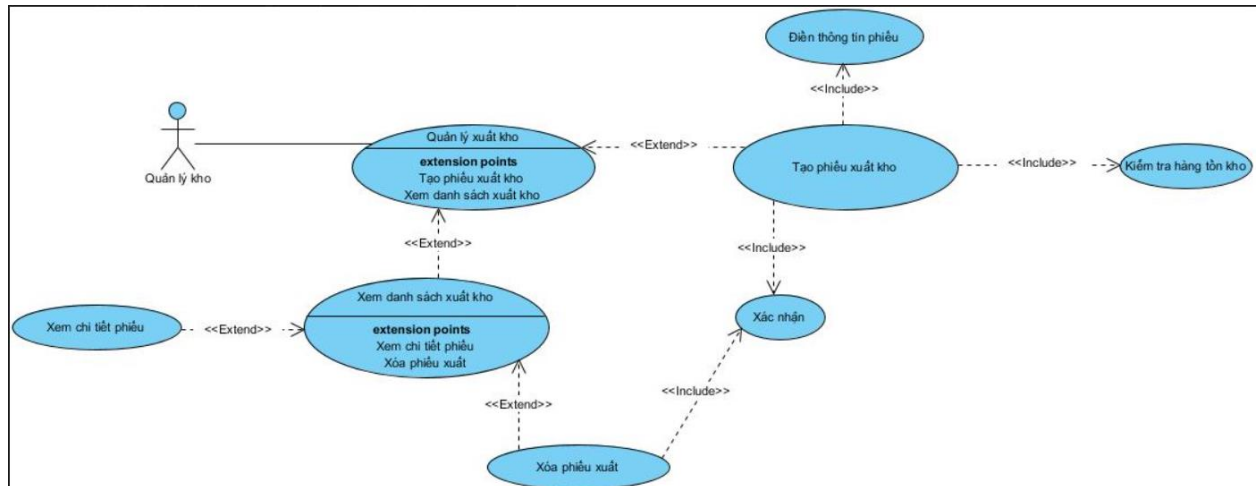


hình 0.2.1.1: biểu đồ SubUC mô tả quản lý nhà cung cấp

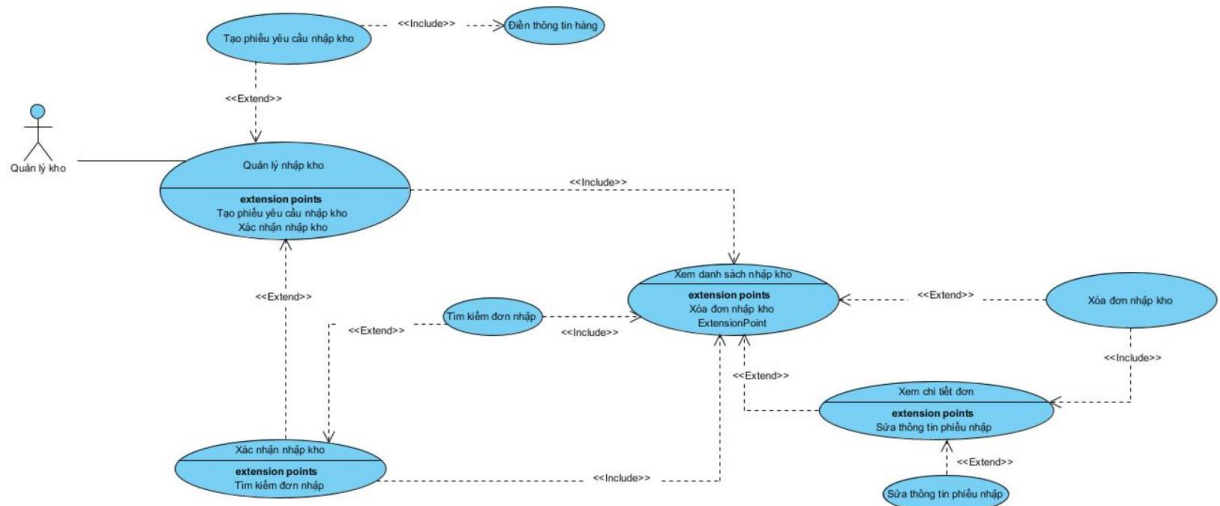


Bảng 0.2.1.2: biểu đồ tuần tự mô tả quy trình chỉnh sửa thông tin nhà cung cấp

- **Quản lý phiếu nhập, phiếu xuất:** Cập nhật và lưu trữ các giao dịch nhập – xuất hàng hóa.

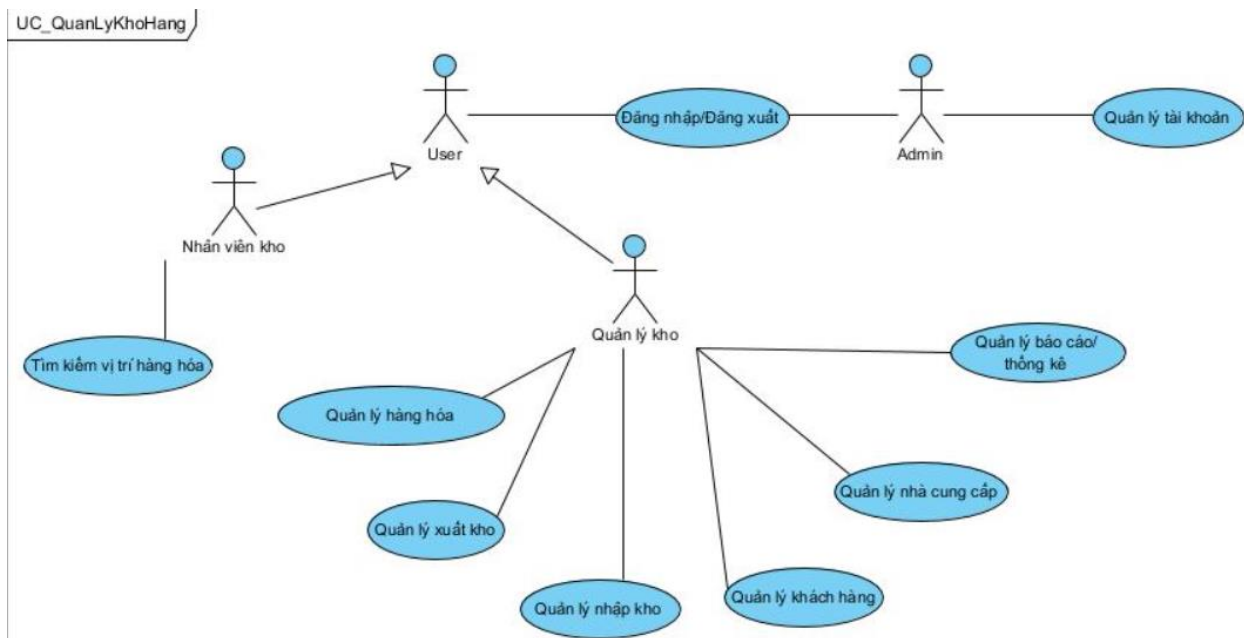


Hình 0.2.1.3: biểu đồ use case mô tả quản lý phiếu xuất kho

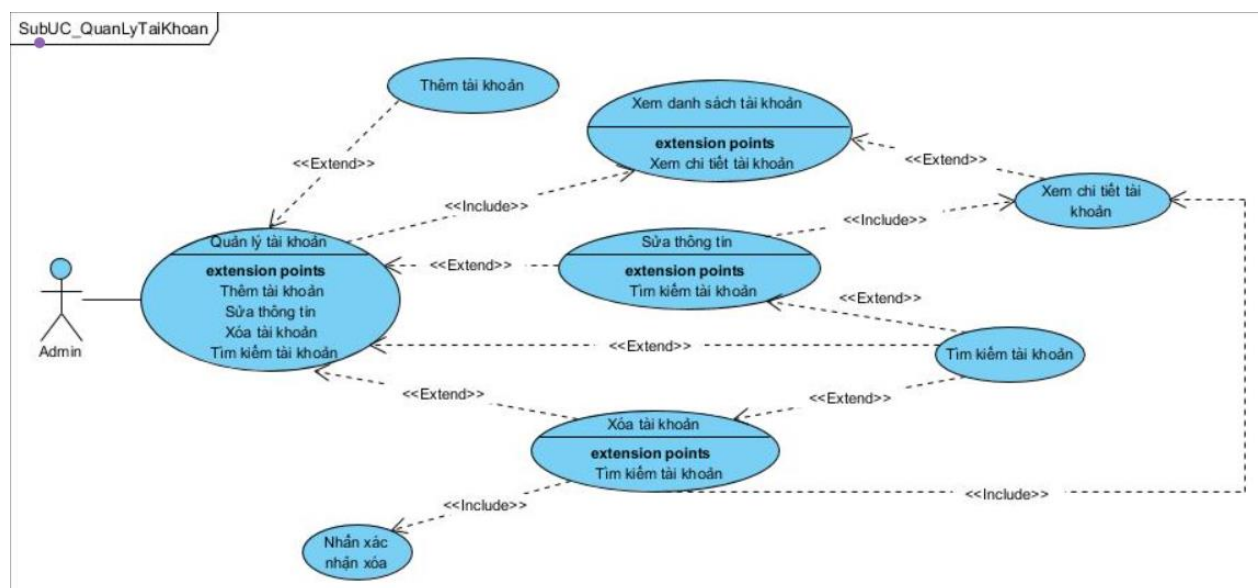


Hình 0.2.1.4: biểu đồ use case mô tả quản lý phiếu nhập kho

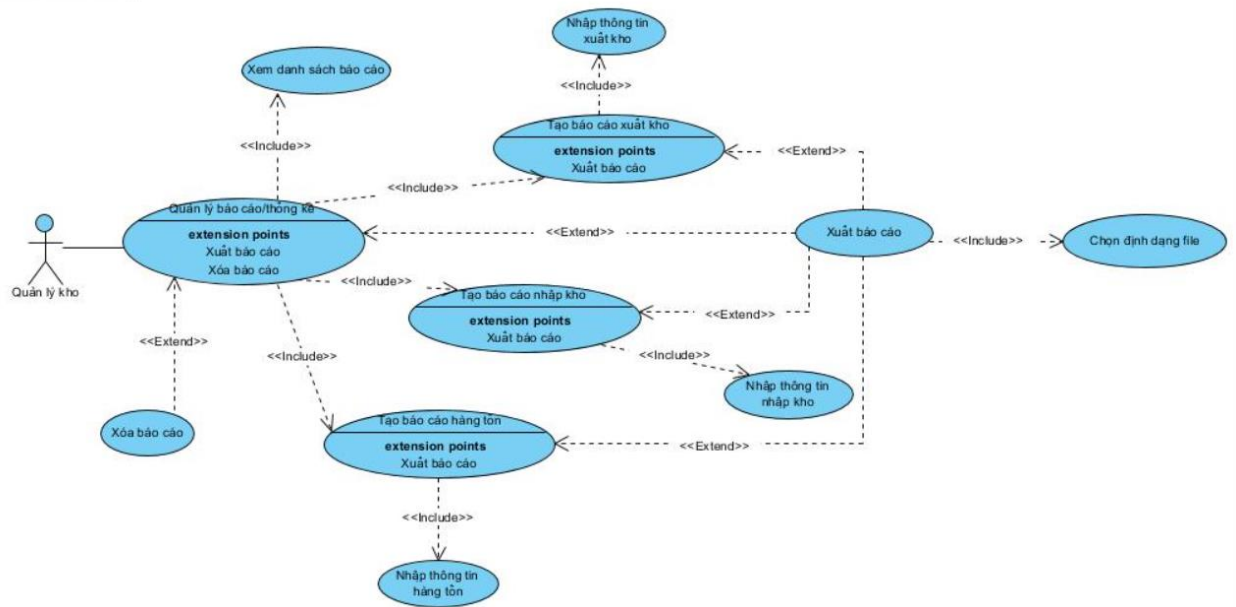
Quản lý người dùng: Phân quyền theo vai trò (quản trị viên, nhân viên) và đảm bảo bảo mật truy cập.



Hình 0.2.1.5: biểu đồ use case mô tả quan hệ, vai trò và quyền truy cập của users

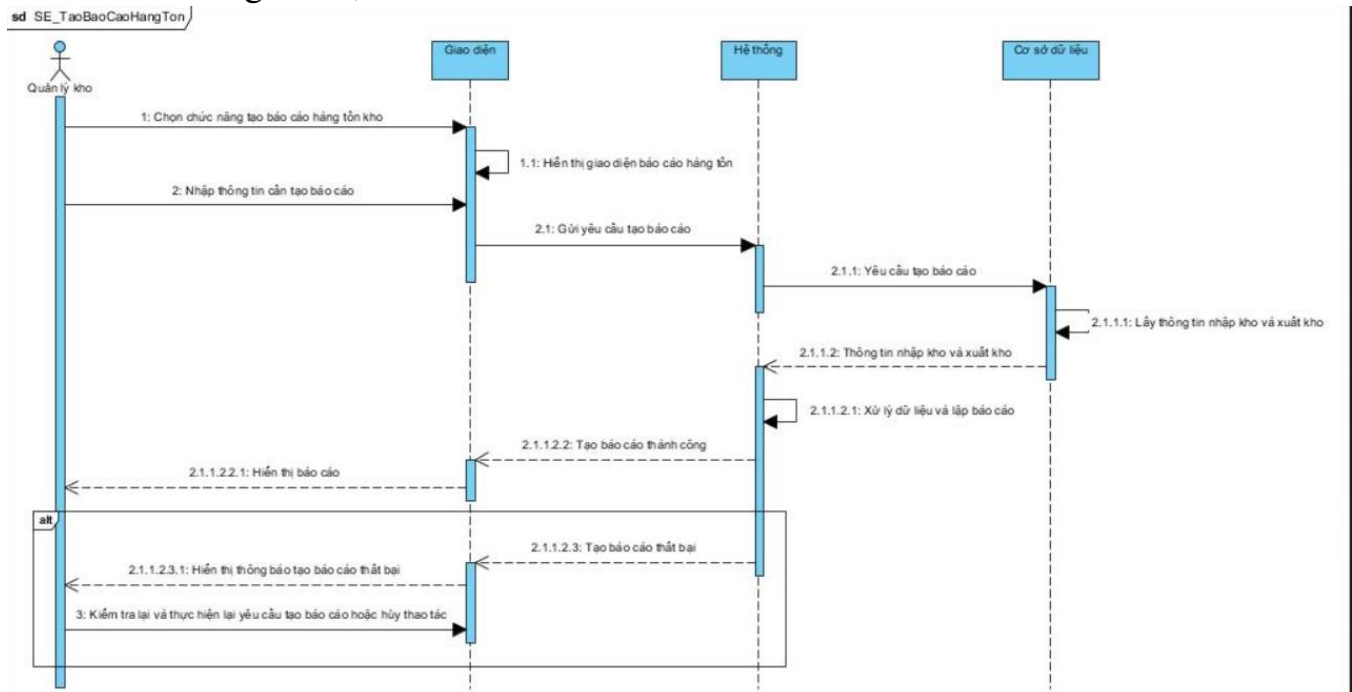


Hình 0.2.1.6: biểu đồ sub use case mô tả vai trò và quyền truy cập của Admin



Hình 0.2.1.7: biểu đồ sub use case mô tả vai trò và quyền truy cập của quản lý kho

- **Tra cứu và thống kê:** Cho phép tra cứu lịch sử giao dịch, thống kê tồn kho theo thời gian thực.



Hình 0.2.1.8: biểu đồ tuần mô tả quy trình báo cáo hàng tồn kho

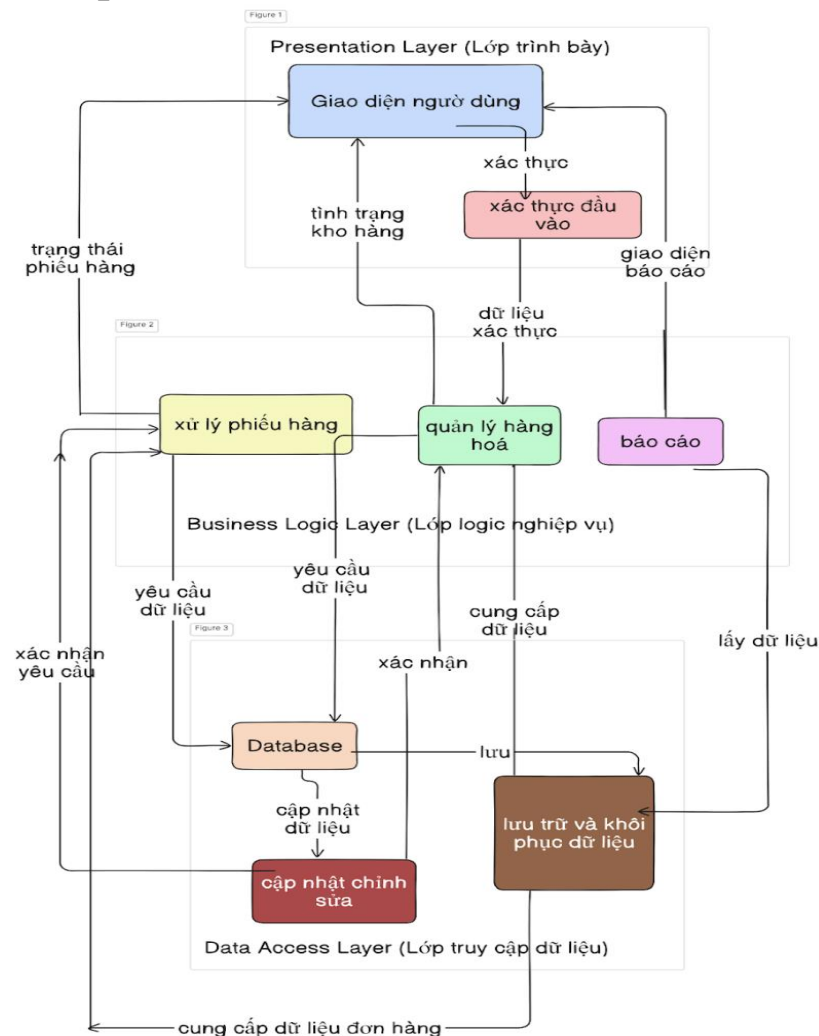
Đối tượng sử dụng:

- **Quản trị viên (Admin):** Toàn quyền hệ thống (tạo người dùng, xem thống kê và giám sát hoạt động).
- **Nhân viên kho:** Thực hiện nhập, xuất kho, theo dõi tồn kho và cập nhật.

Công nghệ sử dụng:

- Backend: ASP.NET Core (.NET 5+), C#
- ORM: Entity Framework Core
- Triển khai: Docker
- Cơ sở dữ liệu: SQL Server

Mô hình phân lớp của dự án:



Hình 0.2.1.9: Biểu đồ phân lớp hệ thống

3.2.2 Chức năng Import/Export Excel

Cho phép quản trị nhập xuất dữ liệu sản phẩm nhanh qua file Excel, giảm thao tác thủ công, hỗ trợ nhập kho hàng loạt và sao lưu dữ liệu.

3.2.2.1 Yêu cầu chức năng:

- Tải lên file Excel (.xlsx) để thêm hoặc cập nhật sản phẩm hàng loạt.
- Xuất danh sách sản phẩm ra file Excel để lưu hoặc chỉnh sửa.
- Kiểm tra định dạng, dữ liệu lỗi trước khi nhập.
- Cung cấp file mẫu chuẩn để import.
- Ghi nhật ký thao tác import/export.

3.2.2.2 Quy trình hoạt động:

Import: chọn file → kiểm tra dữ liệu → nhập hệ thống hoặc báo lỗi.

Export: nhấn xuất → hệ thống tạo file chuẩn → tải về máy.

3.2.2.3 Giao diện người dùng:

- Nút “Nhập Excel” nằm tại trang quản lý sản phẩm, mở dialog chọn file.
- Nút “Xuất Excel” nằm cạnh đó, thực hiện tải ngay file khi nhấn.
- Thông báo lỗi/hoàn tất hiển thị rõ ràng.
- Có thể có tab “Nhật ký nhập/xuất” để tra cứu.

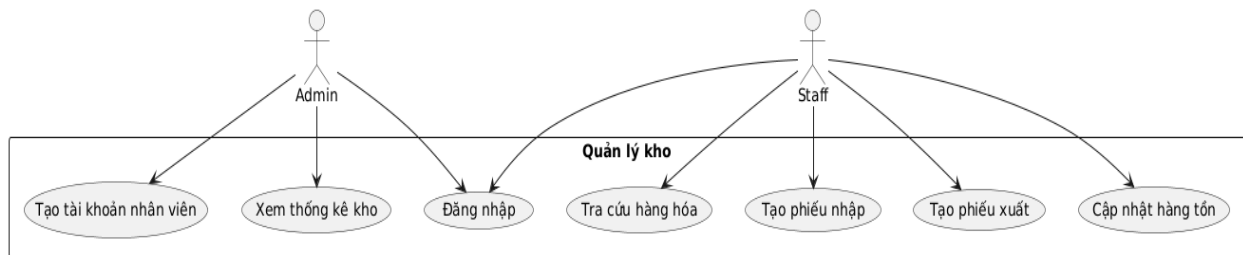
3.2.2.4 Lợi ích:

- Tăng tốc nhập dữ liệu, tiết kiệm thời gian.
- Giảm sai sót nhập liệu thủ công.
- Hỗ trợ sao lưu và chia sẻ dữ liệu.
- Giảm thời gian bảo trì hệ thống.

3.2.3 Sơ đồ quy trình

3.2.2.1 Biểu đồ Use Case

Biểu đồ Use Case thể hiện mối quan hệ giữa các tác nhân (actors) và các chức năng chính của hệ thống. Nó mô tả rõ phạm vi hệ thống, các vai trò và những gì hệ thống cung cấp cho người dùng.



Hình 0.2.2.1: biểu đồ tổng quan hệ thống: có thể thực hiện, như đăng nhập, quản lý tài khoản, tạo và xử lý phiếu nhập/xuất kho, truy vấn thông tin hàng hóa.

3.2.3.1 Sơ đồ hoạt động

Biểu đồ hoạt động mô tả luồng công việc (workflow) của một quy trình cụ thể trong hệ thống, ví dụ như quy trình tạo phiếu nhập kho. Đây là công cụ trực quan giúp hiểu rõ các bước xử lý và điều kiện kiểm tra.



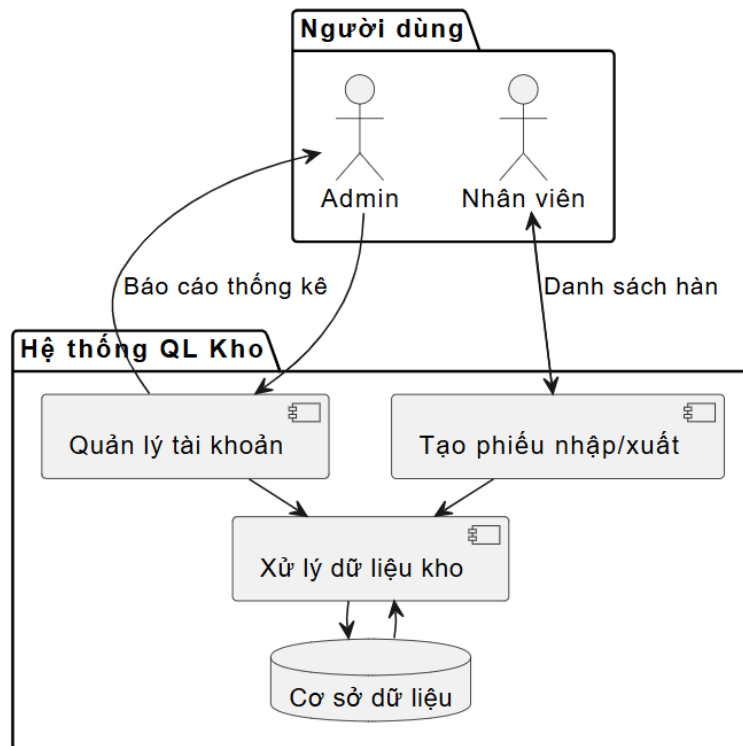
Hình 0.2.2.2: Biểu đồ thể hiện luồng hoạt động khi thực hiện nghiệp vụ nhập kho

3.3. Thiết kế dữ liệu

3.3.1 Biểu đồ chứng năng nghiệp vụ

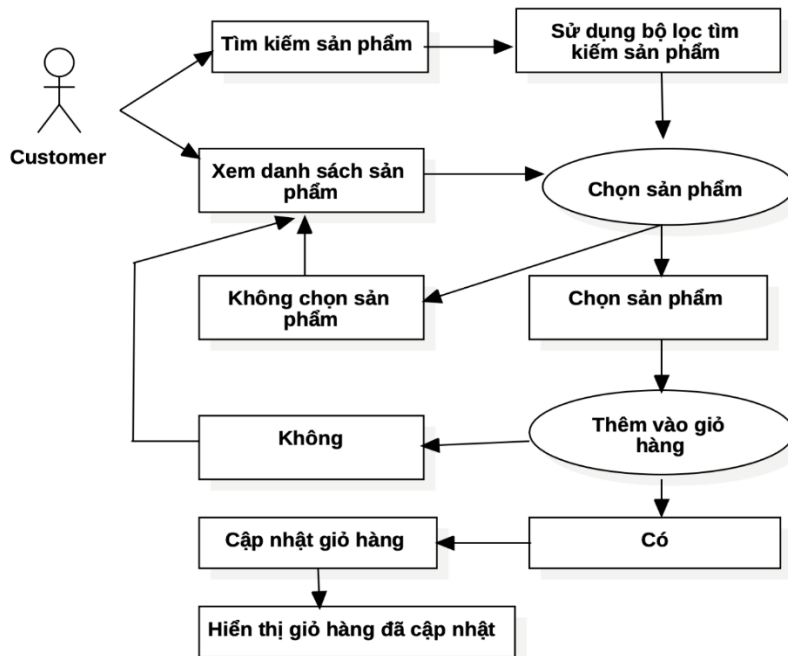
Trong biểu đồ này bao gồm các chức năng của từng người dùng cụ thể như sau:

- **Admin:** Quản lý toàn bộ hệ thống, bao gồm tạo các vai trò, cấp quyền cho tài khoản, thêm/xóa tài khoản nhân viên, xem và phân tích báo cáo thống kê kho.
- **Nhân viên:** Tạo và cập nhật phiếu nhập kho, xuất kho; kiểm tra và tra cứu hàng hóa tồn kho; chỉnh sửa thông tin hàng hóa.
- **Hệ thống:** Lưu trữ và truy xuất dữ liệu trong cơ sở dữ liệu (CSDL) bao gồm thông tin hàng hóa, danh mục, phiếu nhập/xuất và người dùng.



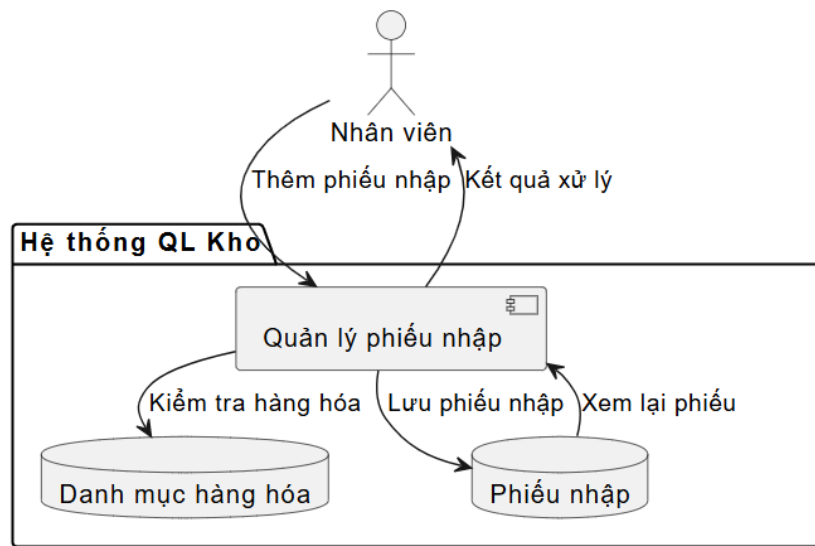
Hình 0.3.1.1: Biểu đồ khái quát mối quan hệ, chức năng chính của hệ thống quản lý kho của người dùng

3.3.2 Biểu đồ use case người dùng khi mua hàng.



3.3.3 Biểu đồ luồng dữ liệu

mô tả tổng quan các dòng dữ liệu chính giữa các đối tượng bên ngoài (external entities), các quá trình xử lý (processes), và các kho dữ liệu (data stores).



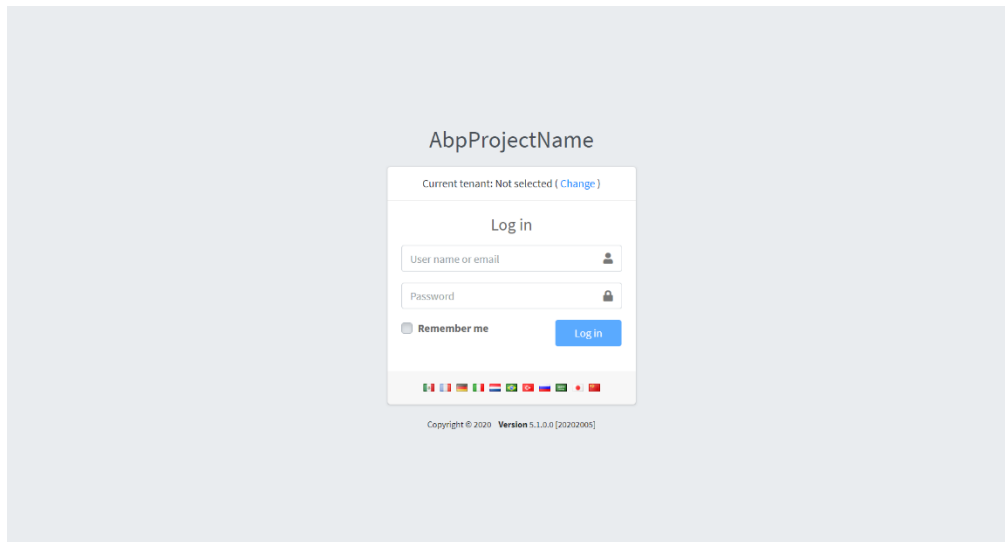
Hình 0.3.2.2: biểu đồ use case mô tả chức năng quản lý phiếu nhập kho của nhân viên

3.3.4. Thiết kế bảng

Tên cột	Kiểu dữ liệu	Mô tả
ID	uniqueidentifier	Khóa chính, định danh duy nhất cho mỗi giao dịch
ProductId	uniqueidentifier	Khóa ngoại tới bảng (Product)
Quantity	int	Số lượng sản phẩm được nhập hoặc xuất
TransactionType	int hoặc varchar	Loại giao dịch: nhập hoặc xuất
TransactionDate	datetime	Ngày thực hiện giao dịch
Notes	Nvarchar (MAX)	Ghi chú tùy ý
CreatedByUserId	uniqueidentifier	Người thực hiện giao dịch
WarehouseId	uniqueidentifier	Kho liên quan đến giao dịch

3.3.4 Giao diện người dùng mẫu

1. Trang đăng nhập



Đường dẫn: /Account/Login

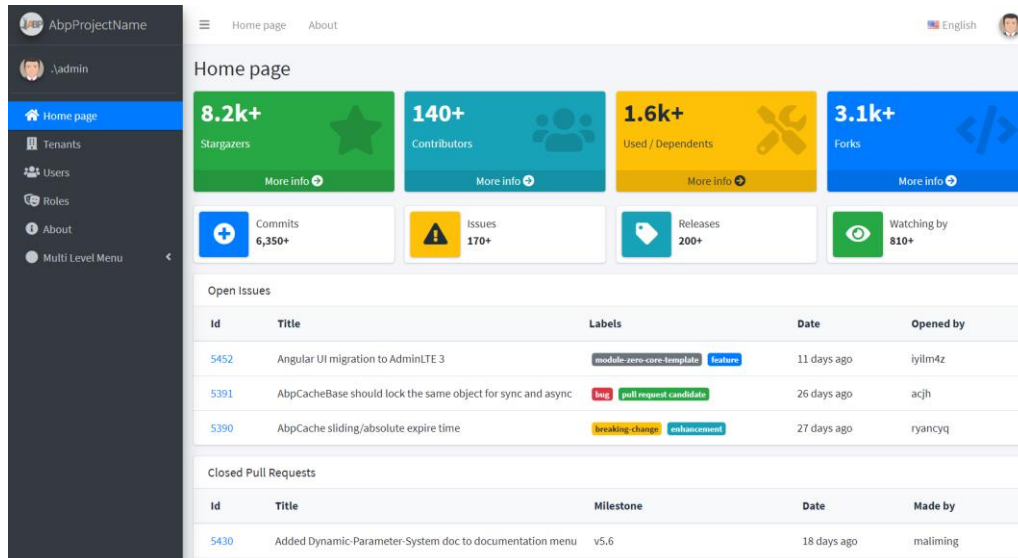
Tính năng:

- Giao diện đơn giản, gọn gàng (theo mặc định của ABP)
- Hỗ trợ xác thực qua Identity / JWT

Thành phần:

- Ô (Username, Password)
- Nút Login
- Thông báo lỗi khi đăng nhập sai

2. Dashboard (Trang chính)



- **Đường dẫn:** /
- **Tính năng:**
 - Hiển thị thông tin tổng quan (có thể tùy chỉnh bằng widgets)
 - Giao diện sử dụng layout của LeptonX/Metronic
- **Ví dụ nội dung hiển thị:**
 - Tổng số đơn hàng
 - Tổng sản phẩm tồn kho
 - Biểu đồ xuất - nhập kho

3. Quản lý Sản phẩm (Product Management)

- **URL:** /products
- **Thành phần chính:**
 - Danh sách sản phẩm (ABP DataTable)
 - Bộ lọc: Tìm kiếm theo tên, loại, giá

- Nút Create, Edit, Delete từng sản phẩm
- Popup modal để tạo / sửa sản phẩm

4. Quản lý Phiếu nhập kho / xuất kho

- **URL:** /stock-transactions
- **Giao diện dạng bảng:**
 - Mã phiếu, Loại (Nhập/Xuất), Ngày tạo, Người lập, Trạng thái
 - Nút tạo mới / chỉnh sửa / xem chi tiết
- **Form nhập liệu:**
 - Combo chọn sản phẩm
 - Số lượng, đơn giá, ghi chú
 - Xác nhận lưu (Submit)

5. Quản lý Người dùng & Phân quyền

- **Dựa theo module Identity** có sẵn trong ABP
- **URL:** /identity/users và /identity/roles
- **Tính năng:**
 - Quản lý user, phân quyền theo role
 - Giao diện ABP chuẩn: form modal, data table
 - Các nút chức năng: Create User, Edit Role, Assign Permissions

6. Thống kê và Báo cáo

- **Tùy chỉnh Dashboard hoặc tạo riêng module** /reports
- **Hiển thị:**
 - Biểu đồ bar/pie (dùng Chart.js hoặc ApexCharts)
 - Bảng dữ liệu báo cáo (có lọc theo thời gian, loại giao dịch)

7. Thanh điều hướng & Layout

- **Menu Sidebar** tự động render theo cấu hình permission
- Giao diện theo theme:
 - **LeptonX**: Gọn nhẹ, hiện đại
 - **Metronic**: Đầy đủ widget, biểu đồ đẹp, nhiều layout

**** Một số chức năng nổi bật:**

- **Phân quyền động**: Áp dụng Permission Management của ABP
- **Tự động mapping & validation** với AutoMapper và Data Annotations
- **Giao diện hiện đại**: theme của ABP

CHƯƠNG 4: Kết luận và hướng phát triển

Trong thời đại công nghệ số hiện nay, việc áp dụng các hệ thống phần mềm trong quản lý hoạt động kinh doanh không chỉ giúp doanh nghiệp tối ưu hóa quy trình vận hành mà còn nâng cao hiệu quả và khả năng cạnh tranh trên thị trường. Đề tài “Hệ thống quản lý bán hàng” được xây dựng với mục tiêu cụ thể là tạo ra một giải pháp quản lý sản phẩm, kho hàng, đơn hàng và người dùng một cách toàn diện, linh hoạt và dễ dàng mở rộng.

Trong quá trình thực hiện, nhóm đã lựa chọn ABP Framework – một nền tảng phát triển ứng dụng hiện đại trên nền .NET – để xây dựng hệ thống. ABP mang lại nhiều lợi thế như kiến trúc phân tầng rõ ràng, hỗ trợ mạnh mẽ cho việc quản lý quyền hạn người dùng, hệ thống module hóa dễ bảo trì, đồng thời tích hợp sẵn nhiều tính năng như quản lý đa ngôn ngữ, xác thực, phân quyền, nhật ký hệ thống,...

4.1 Các kết quả đạt được:

- Quản lý hiệu quả sản phẩm, đơn hàng, kho và người dùng với đầy đủ CRUD và giao diện trực quan.
- Áp dụng kiến trúc Layered giúp hệ thống dễ bảo trì và mở rộng.
- Sử dụng các công nghệ như EF Core, AutoMapper, DI, Unit of Work để đảm bảo hiệu năng và khả năng phát triển.
- Frontend kết nối backend qua REST API, hỗ trợ cập nhật dữ liệu mượt mà, phù hợp ứng dụng hiện đại.
- Đảm bảo bảo mật qua xác thực và phân quyền theo vai trò.

4.2 Những hạn chế:

- Do sử dụng ABP Framework với nhiều lớp trừu tượng, cấu hình ban đầu có phần phức tạp, đòi hỏi người lập trình phải hiểu rõ kiến trúc framework.
- Giao diện chưa được tối ưu cho thiết bị di động trong phiên bản hiện tại.
- Một số tính năng nâng cao như thanh toán trực tuyến, thống kê nâng cao, xuất hóa đơn... chưa được triển khai do giới hạn về thời gian và phạm vi đề tài.

4.3 Hướng phát triển:

- Bổ sung các module chức năng mới như **quản lý khuyến mãi, tích hợp thanh toán điện tử, quản lý nhà cung cấp**.
- Nâng cấp giao diện theo hướng **responsive design** để hoạt động tốt trên các thiết bị di động.
- Tích hợp hệ thống báo cáo thống kê nâng cao (BI), kết nối với Power BI hoặc các dashboard visualization để hỗ trợ quản lý ra quyết định nhanh chóng hơn.

- Triển khai hệ thống lên các **nền tảng đám mây** (Cloud) để hỗ trợ truy cập mọi lúc, mọi nơi và nâng cao khả năng mở rộng theo nhu cầu thực tế.
- ➔ Với thời gian và phần tìm hiểu có hạn, bài viết còn vẫn còn một vài thiếu sót rất mong nhận được những góp ý và nhận xét từ các Thầy (Cô), các anh (chị) và các bạn.

Nhóm chúng em xin chân thành cảm ơn!

TÀI LIỆU THAM KHẢO

STT	DANH MỤC
[1]	Bootstrap: https://getbootstrap.com/docs/5.3/getting-started/introduction/
[2]	ASP.Net: Microsoft. “ASP.NET Documentation.” https://learn.microsoft.com/en-us/aspnet/core
[3]	PostgreSQL Global Development Group. “PostgreSQL 15 Documentation.” https://www.postgresql.org/docs
[4]	Phạm Huy Hoàng, <i>Lập trình Web với ASP.NET Core và Angular</i> , NXB Khoa học và Kỹ thuật, 2020.
[5]	Nguyễn Minh Đức, <i>Thiết kế giao diện người dùng trong phát triển phần mềm</i> , NXB Thống kê, 2018.