

**BỘ THÔNG TIN VÀ TRUYỀN THÔNG
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
KHOA CÔNG NGHỆ THÔNG TIN II**



BÁO CÁO MÔN HỌC

PHÂN TÍCH THIẾT KẾ VÀ ĐẢM BẢO CHẤT LƯỢNG PHẦN MỀM

Đề tài: “XÂY DỰNG WEBSITE KINH DOANH ĐIỆN THOẠI DI ĐỘNG”

GVHD: Nguyễn Anh Hào

Nguyễn Minh Thuận N19DCCN202

Giang Mạnh Tuấn N19DCCN176

Nguyễn Thành Thái N19DCCN189

Lớp: D19CQCNPM02

Khóa: 2019-2024

Hệ: Đại học chính quy

TPHCM, tháng 11 năm 2023

LỜI CẢM ƠN

Lời nói đầu tiên, chúng em xin được gửi lời cảm ơn chân thành nhất đến thầy *Nguyễn Anh Hào* đã giúp đỡ và hỗ trợ chúng em tận tình để hoàn thành đồ án môn học này.

Do trong khuôn khổ thời gian ngắn, trình độ chuyên môn, kinh nghiệm, kiến thức của bản thân còn hạn chế và là bước đầu làm quen với công việc nghiên cứu mang tính thực nghiệm thì chắc chắn kết quả đạt được của chúng em cũng không tránh khỏi những thiếu sót. Chúng em rất mong muốn được thầy và các bạn quan tâm và góp ý để đề tài nghiên cứu của chúng em ngày càng hoàn thiện hơn, được ứng dụng trong thực tế và cũng là kinh nghiệm cho chúng em về những đề tài sau.

Xin kính chúc thầy cùng tất cả những người đã hỗ trợ và đóng góp ý kiến cho chúng em lời chúc sức khỏe, hạnh phúc và thành công.

Chúng em xin chân thành cảm ơn.

Nhóm Sinh viên thực hiện.

MỤC LỤC

CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI	1
1.1. Mục đích	1
1.2. Mục tiêu	1
1.3. Phương pháp tiến hành	1
CHƯƠNG 2: CƠ SỞ KHOA HỌC	2
1.1. Bối cảnh và nhu cầu	2
1.2. Giới thiệu về NestJs framework – Backend	2
1.3. Giới thiệu về NextJS framework – Frontend	3
1.4. Giới thiệu về SQL Server	3
CHƯƠNG 3: PHÂN TÍCH HỆ THỐNG	4
3.1. Bối cảnh, hiện trạng của hệ thống	4
3.1.1. Định nghĩa vấn đề mà đề tài giải quyết	4
3.1.2. Hiện trạng trước khi sử dụng phần mềm	5
3.1.3. Đề xuất giải pháp của đề tài	6
3.2. Định nghĩa các tương tác cần thiết trên phần mềm	7
3.2.1. Usecase thanh toán	7
3.2.2. Usecase tìm kiếm sản phẩm theo chỉ nhất định	9
3.2.3. Usecase yêu cầu so sánh 2 sản phẩm bất kỳ từ khách hàng	11
3.2.4. Usecase xác nhận đơn hàng	13
3.2.5. Usecase giao hàng	15
3.3. Định nghĩa các yêu cầu và ràng buộc với phần mềm	17
3.3.1. Yêu cầu từ môi trường nghiệp vụ	17
3.3.2. Yêu cầu từ môi trường vận hành	18
3.3.3. Yêu cầu từ môi trường phát triển	19
CHƯƠNG 4: THIẾT KẾ PHẦN MỀM VÀ HỆ THỐNG CON	20
4.1. Lược đồ usecase cho thiết kế phần mềm	20
4.1.1. Usecase thanh toán	20
4.1.2. Usecase tìm kiếm sản phẩm theo chỉ nhất định	21
4.1.3. Usecase yêu cầu so sánh 2 sản phẩm bất kỳ từ khách hàng	22
4.1.4. Usecase xác nhận đơn hàng	23
4.1.5. Usecase giao hàng	24
4.2. Thiết kế phần mềm xử lý Usecase	25
4.2.1. Usecase ID/ Actor Name	25

4.2.2	Usecase ID/ Form ID	27
4.2.3	Usecase ID/ API ID	31
4.3	Thiết kế cơ sở dữ liệu cho phần mềm	37
4.3.1	Mức ý niệm (Lược đồ ERD)	37
4.3.2	Mức thiết kế (Lược đồ Bảng Chuẩn 3):	38
4.3.3	Ràng buộc toàn vẹn	48
4.4	Store Procedure:	48
4.4.2	U01/SP_GET_CART_BY_USER_ID	48
4.4.3	U01/SP_GET_PAYMENT	49
4.3.2	U01/SP_CREATE_ORDER_ONLINE	49
4.3.3	U01/SP_CREATE_ORDER_DETAIL	49
4.3.4	U01/SP_UPDATE_QUANTITY_PRODUCT	49
4.3.5	U02/SP_GET_PRODUCT_BY_NAME	49
4.3.6	U02/SP_GET_PRODUCT_BY_FILTERS	50
4.3.7	U02/SP_GET_PRODUCT_BY_RAM (các tiêu chí brand, memory, price (BETWEEN) tương tự).	50
4.3.8	U02/SP_GET_PRODUCT_BY_IDS	50
4.3.9	U03/SP_GET_PRODUCT_INFOR	50
4.3.10	U04/ SP_GET_ORDER	51
4.3.11	U05/ SP_CHECKLOGIN	51
4.3.12	U05/SP_GET_ORDERS_CONFIRMED	51
4.3.13	U05/ SP_UPDATE_STATUS_ORDER_SHIPPER	51
4.4	Bảng đối chiếu	51
CHƯƠNG 5: TỔNG KẾT		52
5.1	Kết quả đạt được	52
5.2	Mở rộng bổ sung cho Usecase So Sánh:	52
5.2.1	Ưu điểm	52
5.2.2	Nhược điểm	52
5.2.3	Hướng phát triển	52
5.3	Đánh giá ưu điểm khuyết điểm	52
5.2.1	Ưu điểm	52
5.2.2	Nhược điểm	52
5.2.3	Hướng phát triển trong tương lai	53

DANH MỤC CÁC HÌNH

Hình 1. Quy trình giao hàng tổng quát	2
Hình 2. Lược đồ cộng tác trước khi có phần mềm	5
Hình 3. Lược đồ cộng tác sau khi có phần mềm	6
Hình 4. Usecase thanh toán	7
Hình 5. Lược đồ tuần tự thanh toán	8
Hình 6. Usecase tìm kiếm sản phẩm theo chí nhất định	9
Hình 7. Lược đồ tuần tự tìm kiếm sản phẩm theo tiêu chí nhất định	10
Hình 8. Usecase yêu cầu so sánh 2 sản phẩm bất kỳ từ khách hàng	11
Hình 9. Lược đồ tuần tự yêu cầu so sánh 2 sản phẩm bất kỳ từ khách hàng	12
Hình 10. Usecas xác nhận đơn hàng	13
Hình 11. Lược đồ tuần tự xác nhận đơn hàng	14
Hình 12. Ussecase giao hàng	15
Hình 13. Lược đồ tuần tự giao hàng	16
Hình 14. Usecase thanh toán	20
Hình 15. Lược đồ tuần tự thanh toán	21
Hình 16. Usecase tìm kiếm sản phẩm theo tiêu chí nhất định	21
Hình 17. Lược đồ tuần tự tìm kiếm sản phẩm theo tiêu chí nhất định	22
Hình 18. Usecase yêu cầu so sánh 2 sản phẩm bất kỳ từ khách hàng	22
Hình 19. Lược đồ tuần tự yêu cầu so sánh 2 sản phẩm bất kỳ từ khách hàng	23
Hình 20. Usecase xác nhận đơn hàng	23
Hình 21. Lược đồ tuần tự xác nhận đơn hàng	24
Hình 22. Usecase giao hàng	24
Hình 23. Lược đồ tuần tự giao hàng	25
Hình 24. Lược đồ ERD	38
Hình 25: Lược đồ bảng đạt chuẩn 3	39
Hình 26. Mô hình diagram	40

DANH MỤC CÁC BẢNG

Bảng 1. Đặc tả Usecase thanh toán	9
Bảng 2. Đặc tả Usecase tìm kiếm sản phẩm theo chí nhất định	11
Bảng 3. Đặc tả yêu cầu so sánh 2 sản phẩm bất kỳ từ khách hàng	13
Bảng 4. Đặc tả Usecase xác nhận đơn hàng	15
Bảng 5. Đặc tả Usecase giao hàng	17
Bảng 6. Yêu cầu từ môi trường nghiệp vụ	18
Bảng 7. Yêu cầu từ môi trường vận hành	18
Bảng 8. Yêu cầu từ môi trường phát triển	19

KÍ HIỆU CÁC CỤM TỪ VIẾT TẮT

Từ viết tắt	Diễn giải
API	Application Programming Interface
CCCD	Căn cước công dân
CSDL	Cơ sở dữ liệu
DOM	Document Object Model
ERD	Entity Relationship Diagram
IDE	Integrated development environment
JS	Javascript
KM	Khuyến mãi
ORM	Object Relation Mapping
RDBMS	Relational Database Management System
REST	Representational State Transfer
SEO	Search Engine Optimization
SQL	Structured Query Language
TMDT	Thương mại điện tử
UI	User Interface

MỞ ĐẦU

Trong thời đại công nghệ số phát triển vượt bậc như hiện nay, việc sử dụng Internet để kinh doanh và tiếp cận thị trường đã trở thành xu hướng không thể phủ nhận. Với xu hướng tăng đáng kể của thương mại điện tử, dẫn đến thành lập một trang web kinh doanh đồng hồ trực tuyến đã trở thành một cơ hội hấp dẫn để khai thác thị trường và phát triển doanh nghiệp.

CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI

1.1. Mục đích

Mục đích của đề tài xây dựng web bán điện thoại di động là để tạo nên một nền tảng trực tuyến, giúp khách hàng tiện lợi hơn trong việc mua hàng, lựa chọn sản phẩm và thanh toán mà không cần phải di chuyển, cùng với đó là sự quản lý cửa hàng dễ dàng cho bên chủ doanh nghiệp.

1.2. Mục tiêu

Hiện nay với sự phát triển mạnh mẽ của thương mại số nên nhu cầu mua hàng online cho những khách hàng không có thời gian đi mua di chuyển đến tận cửa hàng nhiều, theo đó việc thanh toán và giao hàng tận nơi cho khách hàng. Phần mềm bán điện thoại di động sẽ giúp việc mua sắm của khách hàng trở nên tiện lợi hơn.

Với các khách hàng: phần mềm sẽ giúp các khách hàng không có thời gian để đến cửa hàng lựa chọn sản phẩm có thể mua sắm các sản phẩm phù hợp dựa vào các thông tin chủ doanh nghiệp đăng lên website, việc thanh toán cũng dễ dàng hơn thông qua việc thanh toán online hoặc thanh toán sau khi kiểm tra hàng và khách hàng có thể theo dõi đơn hàng của mình để sắp xếp thời gian nhận hàng hợp lý.

Với cửa hàng: phần mềm quản lý cửa hàng giúp cho việc quản lý các đơn hàng của khách hàng và sản phẩm trong cửa hàng dễ dàng hơn.

1.3. Phương pháp tiến hành

Phân tích yêu cầu: xác định và biểu diễn các yêu cầu chức năng và phi chức năng của trang web bán điện thoại di động từ người dùng và các bên liên quan

Thiết kế kiến trúc: Xác định các thành phần chức năng chính của hệ thống, tạo ra mô hình kiến trúc và xác định các quy tắc và tiêu chuẩn thiết kế

Thiết kế lớp: Xác định các lớp, thuộc tính, và phương thức cần thiết để triển khai các chức năng của hệ thống.

Thiết kế giao diện người dùng: Thiết kế các giao diện người dùng hấp dẫn và thân thiện với người dùng để tạo ra trải nghiệm tốt nhất cho người dùng khi sử dụng

Triển khai và kiểm thử: Triển khai các thành phần và chức năng của trang web, sau đó tiến hành kiểm thử và sửa lỗi để đảm bảo tính ổn định và chất lượng của hệ thống

Bảo trì và cải tiến: Duy trì và nâng cấp hệ thống theo thời gian để đáp ứng nhu cầu của người dùng và nâng cao trải nghiệm người dùng

CHƯƠNG 2: CƠ SỞ KHOA HỌC

1.1. Bối cảnh và nhu cầu

Hiện nay việc mua hàng trực tuyến ngày trở nên phổ biến, cùng với đó là sự phát triển vượt bậc của công nghệ số như các loại điện thoại di động làm cho nhu cầu mua sắm sở hữu các chiếc điện thoại di động thông minh hiện đại hơn cho người dùng ngày càng nhiều. Vậy nên cần tạo ra các phần mềm bán điện thoại di động rất cần thiết để giúp việc mua bán giữa khách hàng và chủ doanh nghiệp diễn ra một cách thuận tiện.

Quy trình tổng quát mua hàng tại cửa hàng:



Hình 1. Quy trình giao hàng tổng quát

Mô tả các bước thực hiện:

- 1: Đến (Khách hàng đến cửa hàng bán điện thoại mà mình muốn mua sản phẩm).
- 2: Chọn (Tại cửa hàng bán điện thoại di động, khách hàng lựa chọn sản phẩm phù hợp với yêu cầu của bản thân và trao đổi thông tin, giá sản phẩm với nhân viên bán hàng).
- 3: Thanh toán (Khách hàng sau khi lựa chọn sản phẩm phù hợp sẽ tiến hành toán toán trực tiếp hoặc online tại cửa hàng, cửa hàng kiểm tra sản phẩm, giao nhận tiền thanh toán của khách hàng và xuất hóa đơn, phiếu bảo hành sản phẩm cho khách hàng).
- 4: Giao (Sau khi khách hàng thanh toán, nhân viên chuẩn bị sản phẩm, cập nhật số lượng sản phẩm còn lại trong cửa hàng, ghi nhận hóa đơn của khách hàng).
5. Giao hàng (Nhân viên giao hàng trực tiếp cho khách hàng ngay tại cửa hàng hoặc theo yêu cầu giao hàng cho khách hàng tại địa chỉ mà khách hàng yêu cầu).

1.2. Giới thiệu về NestJs framework – Backend

NestJS là một framework Node.js được xây dựng trên TypeScript, cung cấp một kiến trúc và mô hình phát triển đơn giản và dễ dàng sử dụng.

❖ Điểm mạnh:

- Kiến trúc và Mô hình Phát triển: NestJS cung cấp một kiến trúc và mô hình phát triển dễ dàng sử dụng và rõ ràng. Nó giúp tạo cấu trúc ứng dụng dễ quản lý và bảo trì.
- Hỗ trợ TypeScript: Việc sử dụng TypeScript giúp kiểm tra lỗi tĩnh, cung cấp gợi ý mã và tạo ra mã dễ đọc hơn. TypeScript cũng giúp kiểm soát kiểu dữ liệu, giúp phát triển ứng dụng mạnh mẽ hơn.
- Hỗ trợ nhiều phong cách lập trình: NestJS cho phép bạn sử dụng nhiều phong cách lập trình như hướng đối tượng (OOP), lập trình hàm (FP), và lập trình chức năng phản ứng (FRP).

- Tích hợp các tính năng hữu ích: NestJS tích hợp nhiều tính năng như Dependency Injection, Middleware, Pipes, Guards, Exception Filters, GraphQL, WebSockets, ...

❖ **Điểm Yếu:**

- Tính phức tạp: Mặc dù NestJS cung cấp một kiến trúc rõ ràng, nó cũng có thể trở nên phức tạp đối với các dự án nhỏ. Điều này có thể làm cho việc học và triển khai mất thời gian đối với các nhà phát triển mới.

- Yêu cầu kiến thức TypeScript: NestJS chủ yếu được phát triển bằng TypeScript, vì vậy sử dụng nó đòi hỏi một sự hiểu biết về TypeScript. Điều này có thể tạo rào cản cho các nhà phát triển không quen thuộc với loại ngôn ngữ này.

1.3. Giới thiệu về NextJS framework – Frontend

Next.js là một framework mã nguồn mở dựa trên React, được bổ sung bởi các tính năng quan trọng như Server-Side Rendering (SSR) và Static Site Generation (SSG).

❖ **Điểm mạnh**

- Server-Side Rendering (SSR): SSR cho phép máy chủ thực hiện hầu hết các xử lý logic và xử lý JavaScript để tạo trang web. Giúp trang web tải nhanh hơn và giảm tải cho trình duyệt

- Search Engine Optimization (SEO): Next.js cũng cho phép chỉnh sửa thẻ <head> của trang web, điều quan trọng cho xếp hạng SEO, mà bạn không thể dễ dàng thực hiện trong React.

- Router được xây dựng sẵn: Next.js cung cấp định tuyến tự động thông qua thư mục "pages" trong dự án. Bạn chỉ cần tạo các tệp trong thư mục "pages," và Next.js sẽ tự động xử lý định tuyến cho bạn, giúp quản lý các đường dẫn trang web dễ dàng.

❖ **Điểm yếu**

- Khả năng tùy chỉnh gặp hạn chế

1.4. Giới thiệu về SQL Server

SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System - RDBMS) sử dụng câu lệnh SQL (Transact-SQL) để trao đổi dữ liệu giữa máy Client và máy cài SQL Server.

❖ **Điểm mạnh**

- Bảo mật mạnh mẽ: SQL Server cung cấp các tính năng bảo mật mạnh mẽ bao gồm kiểm soát truy cập, mã hóa dữ liệu, và quản lý danh sách người dùng và vai trò.

- Hiệu năng tối ưu: SQL Server cung cấp nhiều công cụ và tối ưu hóa hiệu năng để giúp tăng cường hiệu suất cơ sở dữ liệu. Các tính năng như truy cập song song, indexing và quản lý truy vấn giúp tối ưu hóa truy cập và xử lý dữ liệu.

❖ **Điểm yếu**

- SQL Server yêu cầu kiến thức và kỹ năng quản trị cao: Việc quản lý, tối ưu hóa và sao lưu cơ sở dữ liệu SQL Server có thể đòi hỏi thời gian và kiến thức chuyên sâu.

CHƯƠNG 3: PHÂN TÍCH HỆ THỐNG

3.1. Bối cảnh, hiện trạng của hệ thống

Trong hiện trạng có các đối tượng là người quản lý, khách hàng, nhân viên giao hàng và hỗ trợ thanh toán.

Trong bối cảnh hiện nay, Khách hàng thường rơi vào trạng thái băng khuâng không biết chọn như thế nào mới được sản phẩm điện thoại phù hợp với nhu cầu bản thân. Khách hàng luôn rơi vào 3 vấn đề tâm lý sau:

- Khách hàng không muốn cất công ra tận tận cửa hàng mà vẫn có thể xem được thông tin điện thoại.
- Một chiếc điện thoại nó có quá nhiều thông tin khiến Khách hàng cảm giác lười, ngao ngán khi tiếp thu một lúc quá nhiều thông tin khi đọc.
- Tâm lý khách hàng thường có sự tập trung vào việc so sánh đối chiếu giữa các tùy chọn khác nhau. Họ muốn đảm bảo chiếc điện thoại họ muốn mua phù hợp với nhu cầu mua sắm của họ.

3.1.1. Định nghĩa vấn đề mà đề tài giải quyết

Mục tiêu của đề tài là giải quyết các vấn đề

- Với vai trò người quản lý

- Chức năng quản lý đơn hàng: quản lý sẽ theo dõi, quản lý tình trạng các đơn hàng.
- Chức năng quản lý trạng thái đơn hàng: người quản lý sẽ quản lý các trạng thái đơn hàng (đang xử lý, đang giao, huỷ đơn...).
- Chức năng xác nhận đơn hàng: người quản lý sẽ tiến hành xác nhận hoặc huỷ các đơn giao hàng được khách hàng yêu cầu đến.

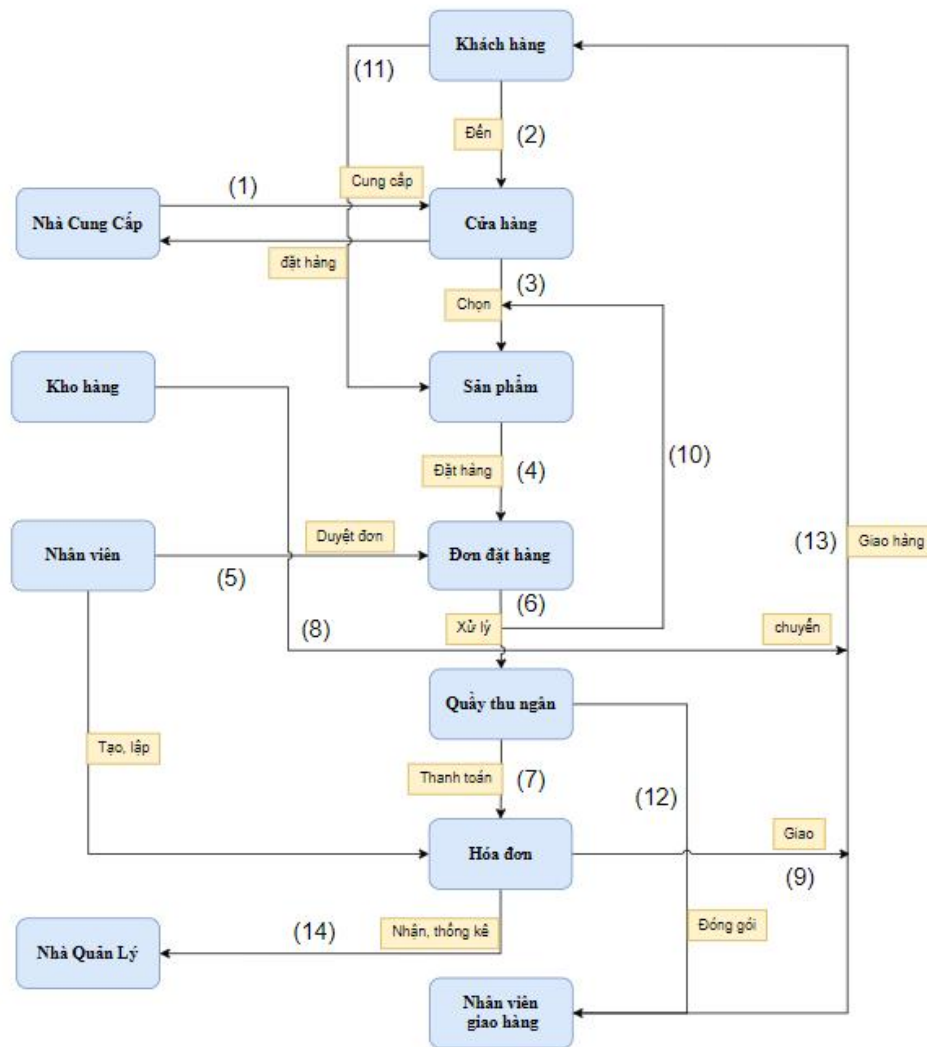
- Với vai trò khách hàng

- **Chức năng tạo đơn hàng (thanh toán):** khách hàng có thể tạo đơn hàng qua phần mềm bằng cách chọn sản phẩm, cung cấp thông tin cần thiết trong đơn hàng để bên cửa hàng dễ dàng xử lý.
- **Chức năng tìm kiếm sản phẩm theo tiêu chí:** khách hàng có thể tìm kiếm sản phẩm bằng từ khóa hoặc theo lựa chọn các danh mục có sẵn phù hợp với tiêu chí đặt ra của bản thân.
- **Chức năng so sánh sản phẩm:** khách hàng có thể so sánh thông tin chi tiết, giá, đánh giá, lượt bán giữa các sản phẩm với nhau để đưa ra lựa chọn đúng đắn nhất.

Mục tiêu của đề tài xây dựng một phần mềm website đáp ứng các chức năng trên, cho phép người quản lý nhận đơn hàng, quản lý các đơn hàng, quản lý trạng thái đơn hàng như (chưa xác nhận, xác nhận, đang giao, đã giao).

Khách hàng có thể biết được đơn hàng của mình có được xác nhận hay không, biết được trạng thái đơn hàng và được hoàn tiền nếu như đơn hàng bị hủy.

3.1.2 Hiện trạng trước khi sử dụng phần mềm



Hình 2. Lược đồ cộng tác trước khi có phần mềm

- Mô tả chi tiết:

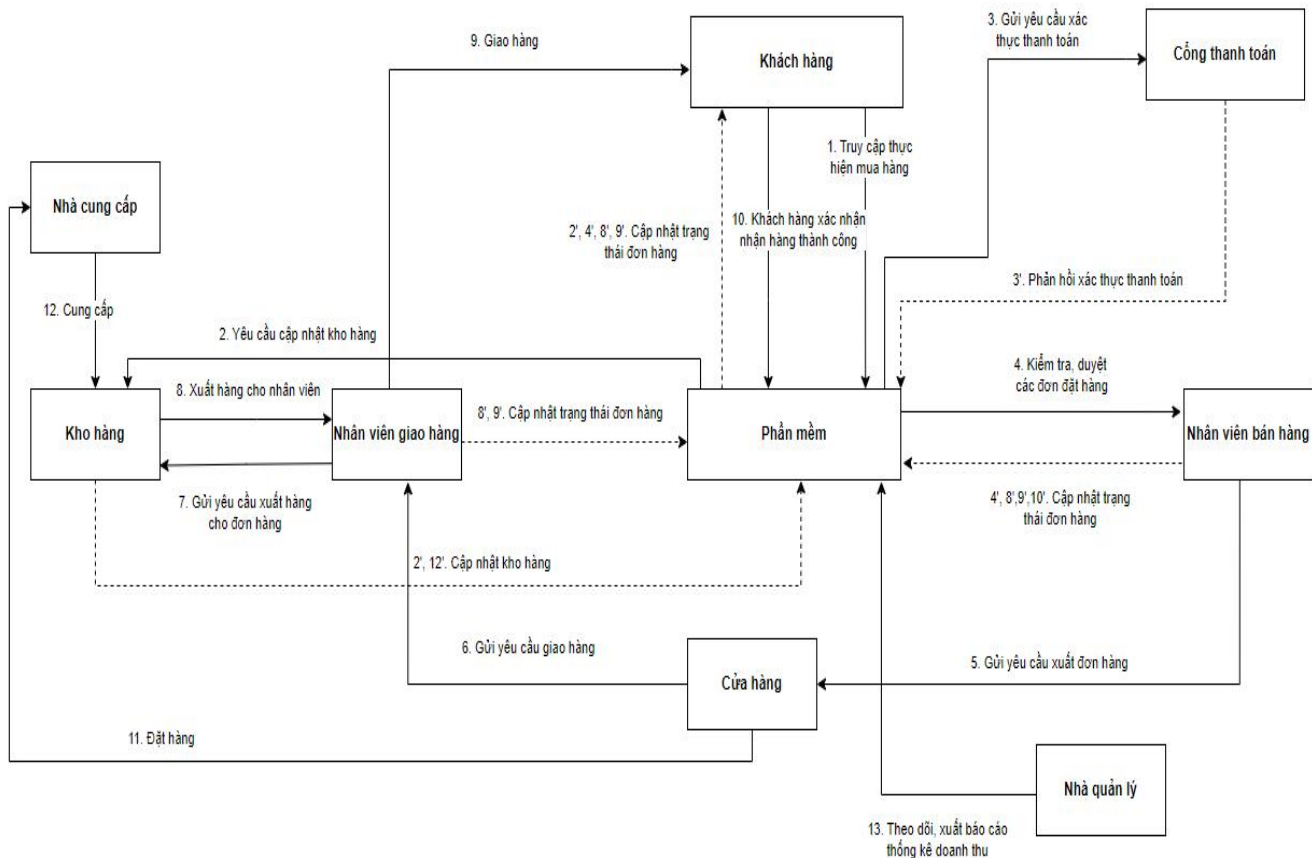
(1) Khi có yêu cầu đặt hàng từ cửa hàng, nhà cung cấp sẽ cung cấp vật phẩm cho cửa hàng theo phiếu đặt hàng.

(2) Khách hàng vào cửa hàng để xem các sản phẩm trong cửa hàng.

- (3) Sau khi vào cửa hàng, Khách hàng chọn các sản phẩm **định mua**.
- (4) Khi quyết định mua hàng, thông tin sản phẩm đó được ghi vào đơn đặt hàng.
- (5) Nhân viên nhận đơn đặt hàng của khách hàng.
- (6) Đơn đặt hàng được nhân viên xử lý, nếu đơn đặt hàng đó nằm trong trưng bày của cửa hàng thì chuyển đơn đặt hàng đó đến quầy thu ngân, còn không (hết hàng, sản phẩm kiểm tra có lỗi, ...) thì khách hàng có thể lại cửa hàng chọn mua sản phẩm khác (10).
- (7) Nhân viên ở quầy thu ngân tiến hành tạo lập hóa đơn, một bên giao hàng cho khách hàng (9), hóa đơn đó sẽ được chủ cửa hàng kiểm tra, thống kê cuối tháng (14).
- (8) Sản phẩm sẽ được nhân viên lấy từ kho hàng chuyển cho khách hàng.

3.1.3 Đề xuất giải pháp của đề tài

3.1.3.1 Lược đồ cộng tác sau khi có phần mềm



Hình 3. Lược đồ cộng tác sau khi có phần mềm

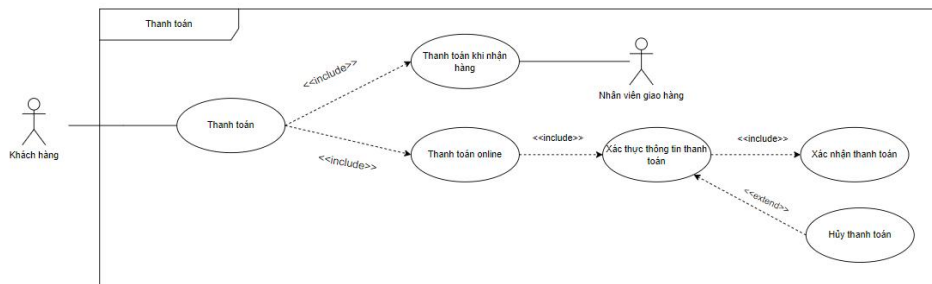
- Mô tả chi tiết:

- (1) Khách hàng truy cập vào phần mềm để thực hiện việc mua hàng.

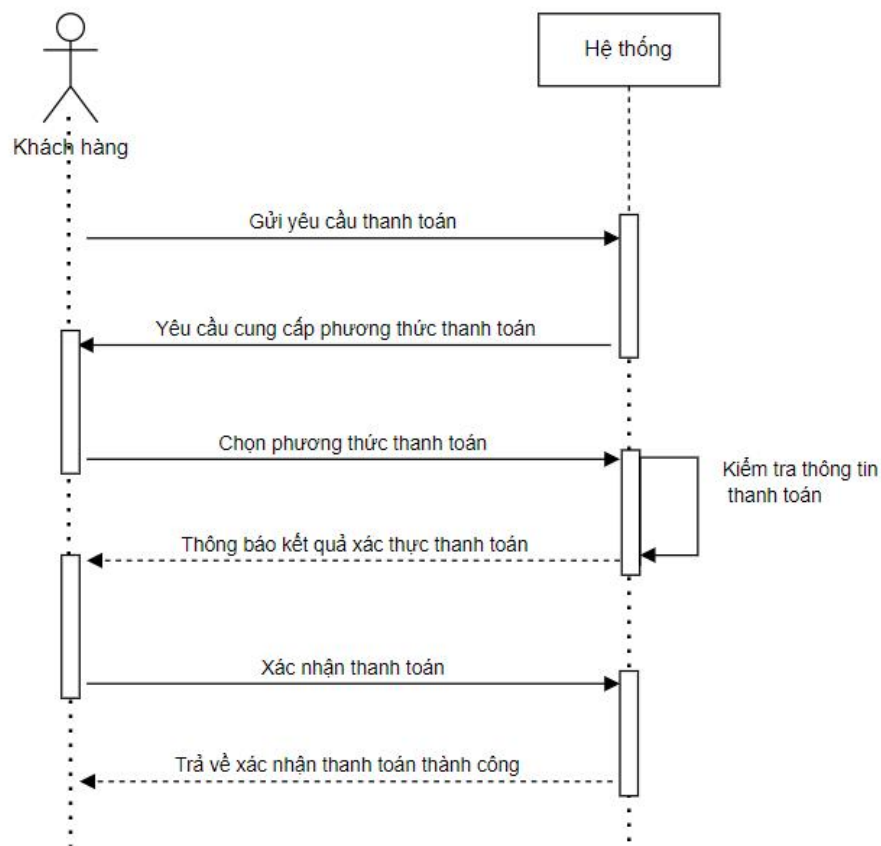
- (2) Phần mềm gửi yêu cầu cập nhật số lượng sản phẩm còn trong kho lên để khách hàng nắm được số lượng sản phẩm để mua hàng.
- (3) Sau khi lựa chọn sản phẩm theo yêu cầu của mình, khách hàng tiến hành thanh toán cho đơn hàng, gửi yêu cầu đến cổng thanh toán để thực hiện thanh toán tiền. Sau đó, cổng thanh toán xác thực thông tin thanh toán của khách hàng và gửi kết quả về phần mềm để thông báo xác thực với khách hàng.
- (4) Nhân viên bán hàng kiểm tra đơn hàng và xét duyệt các đơn hàng của khách hàng và cập nhật trạng thái đơn hàng lên phần mềm.
- (5) Nhân viên bán hàng gửi yêu cầu xử lý đơn hàng đến cửa hàng để phân công nhân viên giao hàng.
- (6) Cửa hàng tiếp nhận đơn hàng của nhân viên bán hàng, phân công nhân viên giao hàng xử lý đơn hàng và thu tiền nếu thanh toán khi nhận hàng.
- (7) Nhân viên giao hàng gửi yêu cầu xuất hàng cho hóa đơn của khách hàng.
- (8) Kho hàng xuất hàng theo đơn hàng cho nhân viên giao hàng và cập nhật trạng thái đơn hàng lên phần mềm.
- (9) Nhân viên giao hàng tiến hành giao hàng cho khách hàng và liên tục cập nhật trạng thái đơn hàng để khách hàng sắp xếp thời gian nhận hàng thuận lợi.
- (10) Sau khi khách hàng nhận hàng thành công phải tiến hành xác nhận nhận hàng thành công với phần mềm để kết thúc đơn hàng.
- (11) Cửa hàng gửi yêu cầu đặt hàng đến nhà cung cấp.
- (12) Nhà cung cấp cung cấp hàng cho kho hàng và kho hàng tiến hành cập nhật số lượng sản phẩm trong kho hàng lên phần mềm.
- (13) Nhà quản lý truy cập phần mềm để theo dõi và xuất báo cáo thống kê doanh thu.

3.2 Định nghĩa các tương tác cần thiết trên phần mềm

3.2.1 Usecase thanh toán



Hình 4. Usecase thanh toán



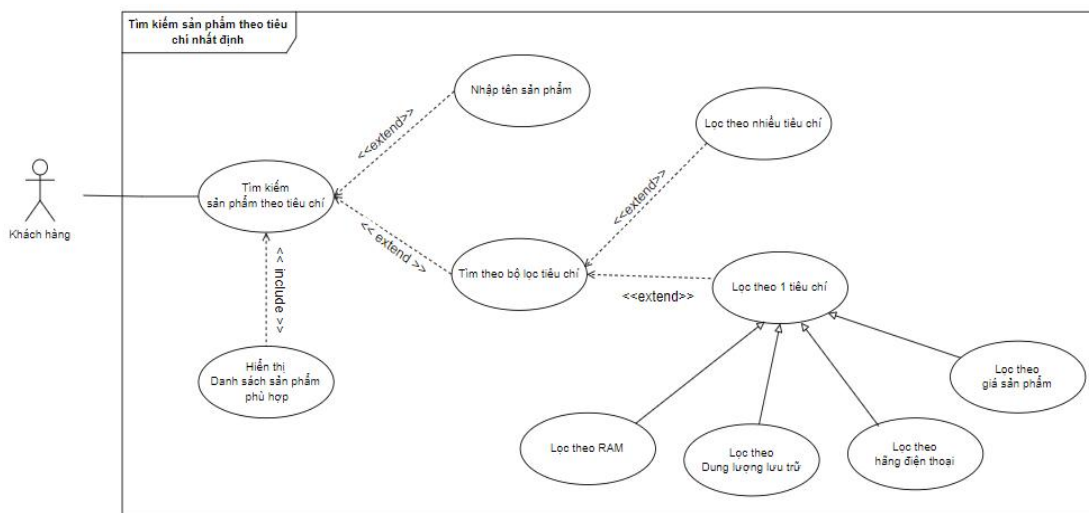
Hình 5. Lược đồ tuần tự thanh toán

USECASE NAME	Thanh toán
ID	U01
SCENARIO	Khách hàng muốn thanh toán đơn hàng.
TRIGGERING EVENT	Khách hàng gửi yêu thanh toán đơn hàng.
BRIEF DESCRIPTION	Khách hàng muốn hoàn thành đơn hàng của mình bằng cách thanh toán đơn hàng, hệ thống gửi yêu cầu khách hàng chọn phương thức thanh toán để xác thực, sau khi khách hàng xác nhận thanh toán, hệ thống thông báo xác nhận thanh toán thành công.
ACTORS	Khách hàng, Nhân viên giao hàng
PRECONDITION	Khách hàng phải chọn sản phẩm (điện thoại).
POSTCONDITION	Khách hàng nhận được thông báo thanh toán thành công

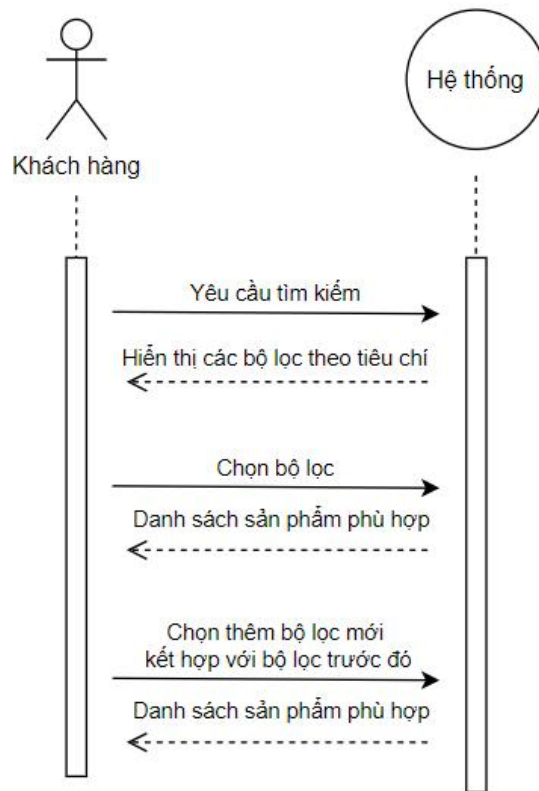
	Actor	System
FLOW OF EVENT	1. Khách hàng gửi yêu cầu thanh toán. 2. Khách hàng chọn phương thức thanh toán. 3. Khách hàng xác nhận thanh toán.	1. Hệ thống tiếp nhận yêu cầu của khách hàng và gửi yêu cầu chọn phương thức thanh toán cho khách hàng. 2. Hệ thống xác thực phương thức thanh toán của khách hàng và trả về thông báo. 3. Hệ thống tiếp nhận đơn thanh toán và trả về thông báo thanh toán thành công.
EXCEPTION CONDITION	Nếu thanh toán bị lỗi hay gặp vấn đề về kỹ thuật, thông báo “Thanh toán không thành công.”	

Bảng 1. Đặc tả Usecase thanh toán

3.2.2 Usecase tìm kiếm sản phẩm theo chỉ nhất định



Hình 6. Usecase tìm kiếm sản phẩm theo chỉ nhất định



Hình 7. Lược đồ tuần tự tìm kiếm sản phẩm theo tiêu chí nhất định

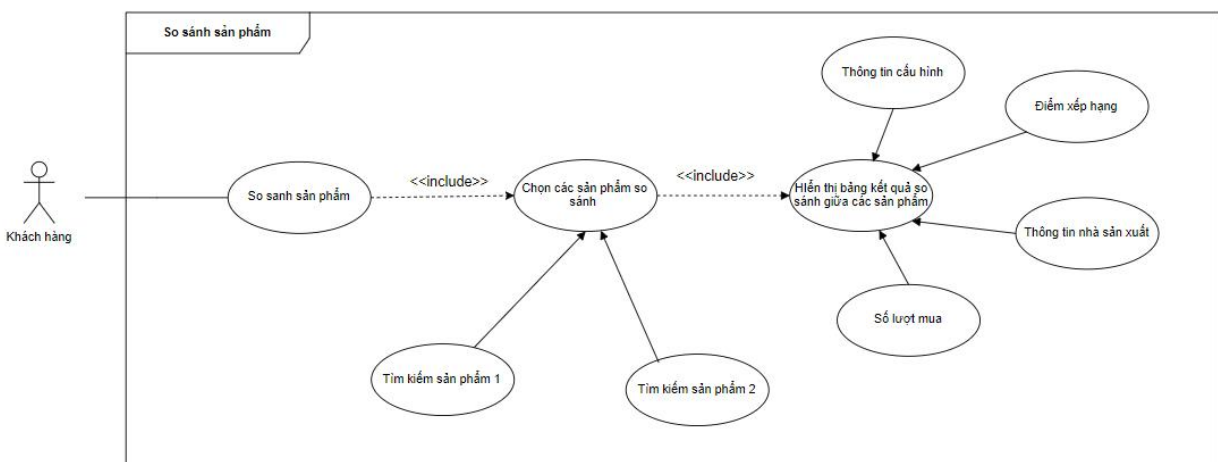
USECASE NAME	Tìm kiếm sản phẩm theo chí nhất định
ID	U02
SCENARIO	Khách hàng muốn tìm kiếm sản phẩm dựa vào tiêu chí.
TRIGGERING EVENT	Khách hàng gửi yêu cầu tìm kiếm sản phẩm
BRIEF DESCRIPTION	Danh sách điện thoại bao gồm rất nhiều thông tin cấu hình, khách hàng muốn kiếm một sản phẩm phù hợp theo mong muốn, hệ thống cung cấp các bộ lọc tiêu chí dựa trên thông tin của điện thoại.
ACTORS	Khách hàng
PRECONDITION	
POSTCONDITION	Khách hàng nhận được danh sách điện thoại ứng với tiêu chí đã chọn.

	Actor	System
FLOW OF EVENT	1. Khách hàng yêu cầu tìm kiếm sản phẩm. 2. Khách hàng chọn bộ lọc tiêu chí mong muốn. 3. Khách hàng chọn nhiều hơn 1 bộ lọc.	1. Hệ thống hiển thị các bộ lọc sẵn có. 2. Hệ thống hiển thị các sản phẩm tương ứng. 3. Hệ thống tiếp nhận theo bộ lọc mới, kết hợp với bộ lọc trước đó, hiển thị danh sách sản phẩm phù hợp.
EXCEPTION CONDITION	Nếu không có sản phẩm nào đáp ứng với bộ lọc mà khách hàng đã chọn, thông báo “Không tìm thấy.”	

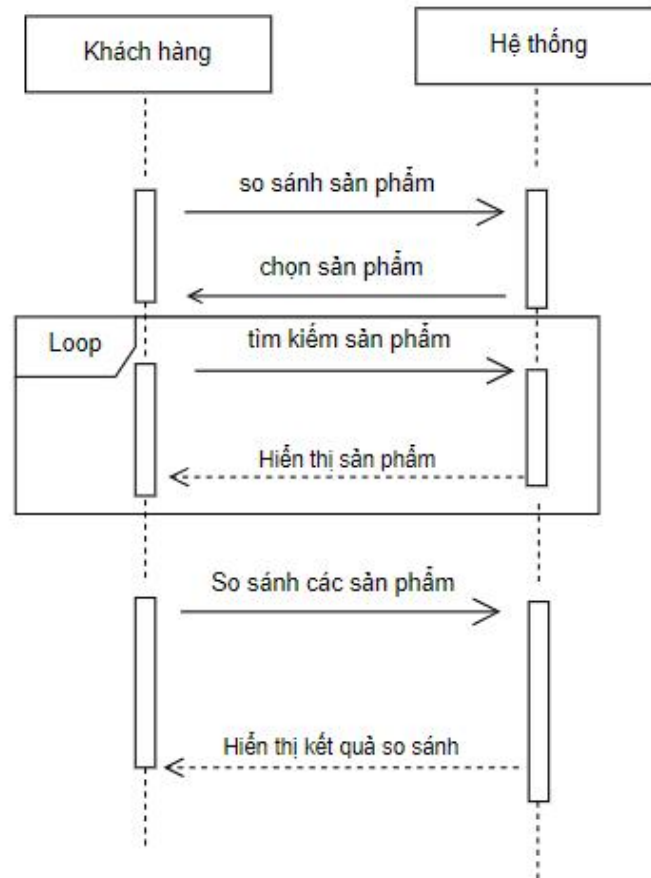
Bảng 2. Đặc tả Usecase tìm kiếm sản phẩm theo chỉ nhất định

3.2.3 Usecase yêu cầu so sánh 2 sản phẩm bất kỳ từ khách hàng

- ⇒ usecase So sánh 2 sản phẩm có giá trị sử dụng cao để phần mềm thực hiện, nó giúp người dùng có cái nhìn tổng quan nhất với nhu cầu của họ.
- ⇒ Vậy cần những thông tin gì khi so sánh:
- 1. thứ 1: **Thông tin nhà sản xuất** => vì thông tin nhà sản xuất có độ tin cậy cao.
 - 2. **Số lượt mua**: nếu mà ai cũng mua thì điện thoại này đang được chuộng. Nhưng lỡ mọi người mua theo trend, bị truyền thông lôi kéo mà mua nhầm. => cần phải thêm một yếu tố nữa.
 - 3. **Điểm xếp hạng** (lượt vote ...): Tức khách hàng họ đã sử dụng họ cảm giác hài lòng. Và nhiều người cùng chung cảm giác như vậy, thì điện thoại này mới khiến khách hàng mới có cảm giác đáng mua.
 - 4. điều cuối cùng không thể thiếu ở mỗi chiếc điện thoại đó là thông tin cấu hình



Hình 8. Usecase yêu cầu so sánh 2 sản phẩm bất kỳ từ khách hàng



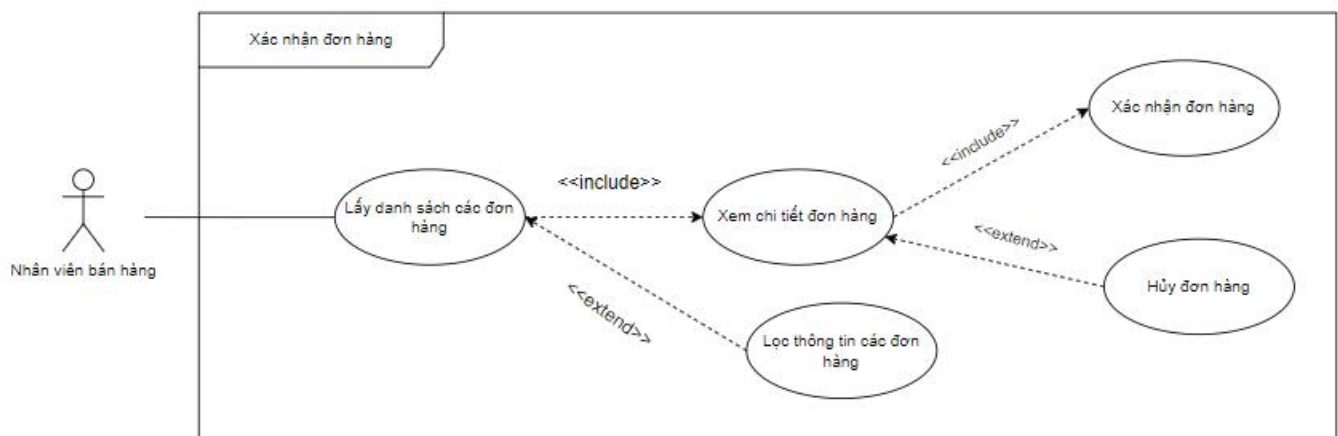
Hình 9. Lược đồ tuần tự yêu cầu so sánh 2 sản phẩm bất kỳ từ khách hàng

USECASE NAME	So sánh các sản phẩm điện thoại
ID	U03
SCENARIO	So sánh 2 sản phẩm điện thoại để hiểu rõ sự khác biệt giữa chúng.
TRIGGERING EVENT	Người dùng muốn so sánh chi tiết giữa hai sản phẩm điện thoại.
BRIEF DESCRIPTION	Use case này cho phép người dùng chọn hai sản phẩm điện thoại và xem các thông tin cấu hình, điểm xếp hạng, thông tin nhà sản xuất, số lượt mua

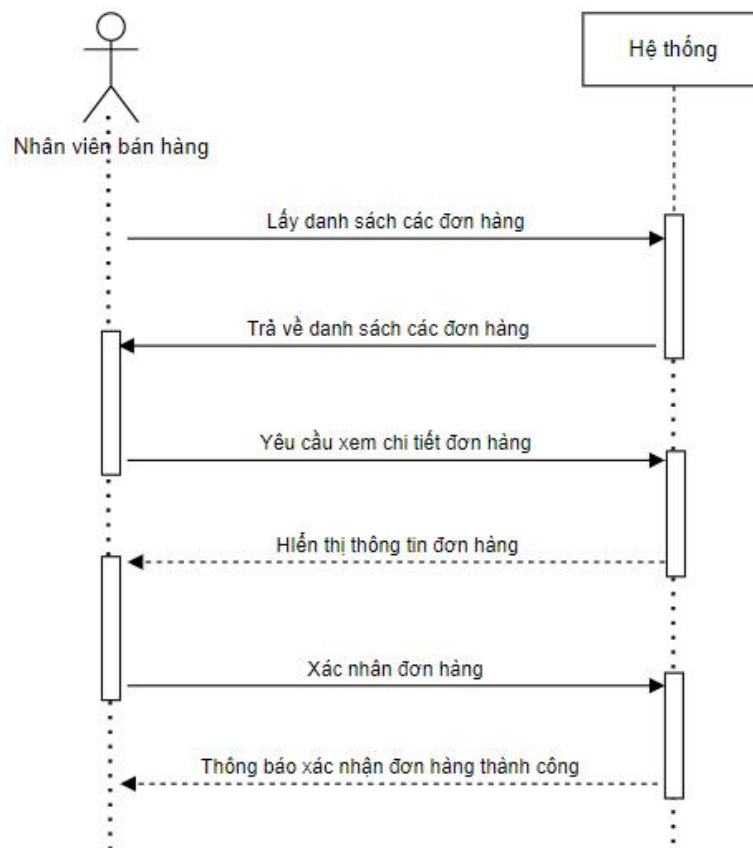
ACTORS	Khách hàng	
PRECONDITION	Hệ thống có sẵn các thông tin về các sản phẩm điện thoại trong cơ sở dữ liệu	
POSTCONDITION	Người dùng xem và so sánh 2 sản phẩm đã chọn	
FLOW OF EVENT	Actor	System
	1. Người dùng truy cập chức năng "So sánh " trên trang web. 2. Người dùng tìm kiếm và chọn các sản phẩm. 3. Người dùng xem và so sánh chi tiết của cả hai sản phẩm.	1. Hệ thống hiển thị giao diện cho phép người dùng tìm kiếm và chọn hai sản phẩm điện thoại để so sánh. 2. Hệ thống hiển thị chi tiết so sánh các sản phẩm bao gồm thông tin cấu hình, điểm xếp hạng, thông tin nhà sản xuất, số lượt mua.
EXCEPTION CONDITION	Nếu hệ thống không tìm thấy thông tin về sản phẩm hoặc sản phẩm không tồn tại, hệ thống sẽ thông báo cho người dùng. Nếu người dùng không chọn đủ hai sản phẩm để so sánh, hệ thống cũng sẽ thông báo và yêu cầu người dùng chọn lại.	

Bảng 3. Đặc tả yêu cầu so sánh 2 sản phẩm bất kỳ từ khách hàng

3.2.4 Usecase xác nhận đơn hàng



Hình 10. Usecas xác nhận đơn hàng



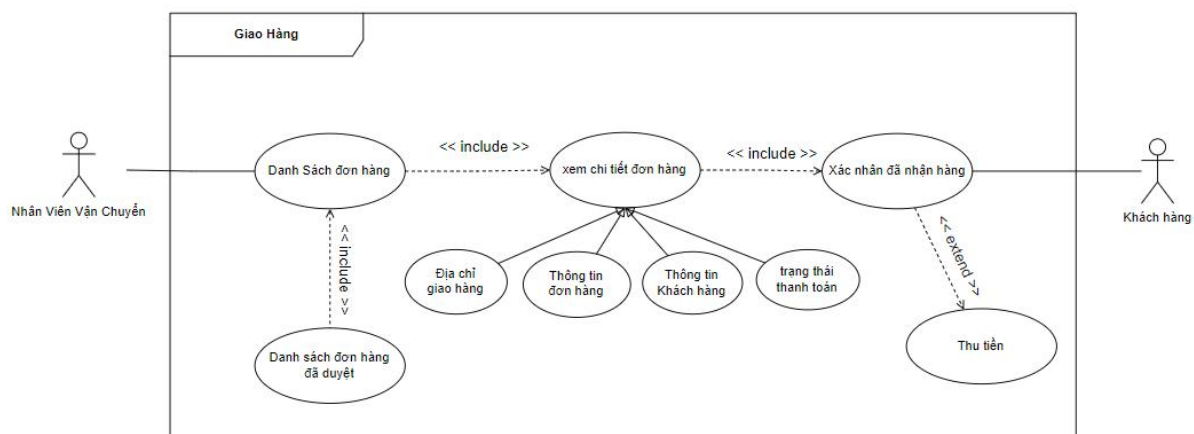
Hình 11. Lược đồ tuần tự xác nhận đơn hàng

USECASE NAME	Xác nhận đơn hàng	
ID	U04	
SCENARIO	Xác nhận đơn hàng cho khách hàng.	
TRIGGERING EVENT	Nhân viên bán hàng xử lý đơn hàng cho khách hàng.	
BRIEF DESCRIPTION	Nhân viên xác nhận đơn hàng cho khách hàng để khách hàng biết tình trạng đơn hàng của mình.	
ACTORS	Nhân viên bán hàng	
PRECONDITION	Danh sách các đơn hàng chưa được xử lý	
POSTCONDITION	Xác nhận đơn hàng thành công	
FLOW OF EVENT	Actor	System
	1. Nhân viên bán hàng yêu	1. Hệ thống hiển thị danh

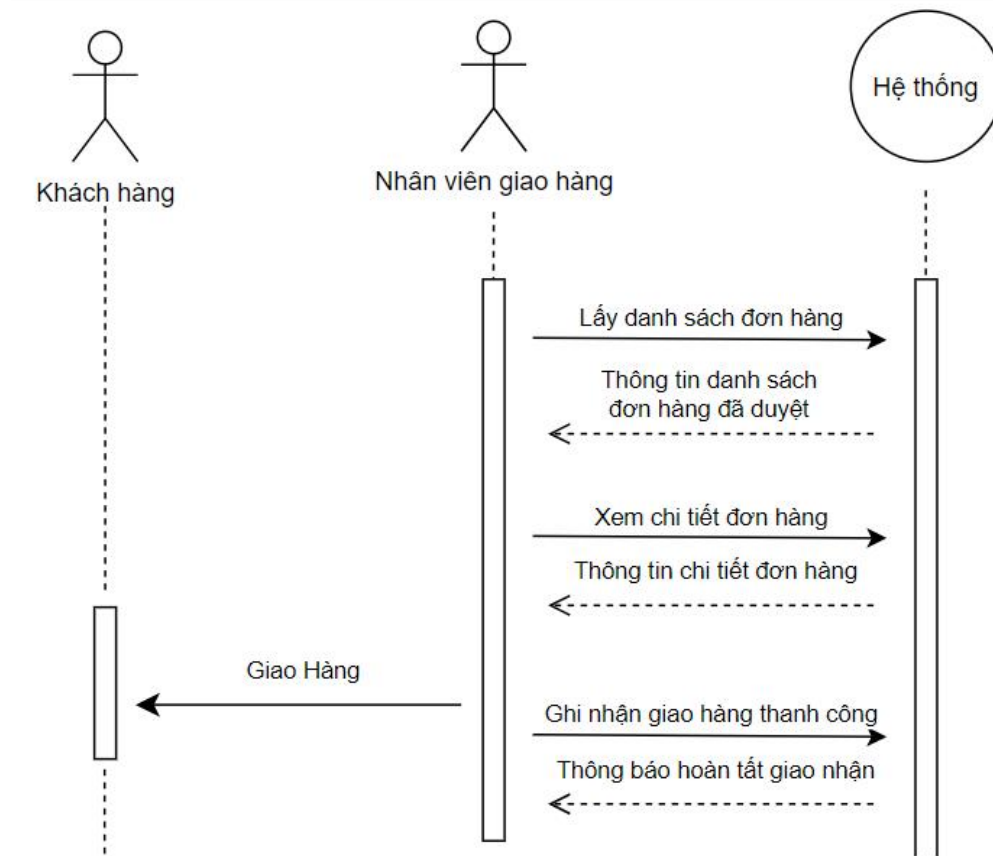
	cầu danh sách các đơn hàng chưa được xử lý. 2. Nhân viên bán hàng yêu cầu xem chi tiết đơn hàng. 3. Người dùng xác nhận đơn hàng được xử lý.	sách các đơn hàng chưa được xử lý. 2. Hệ thống hiển thị chi tiết thông tin đơn hàng. 3. Hệ thống trả về thông báo xác nhận đơn hàng thành công.
EXCEPTION CONDITION	Nếu đơn hàng gặp sự cố phát sinh về số lượng sản phẩm, nhân viên bán hàng sẽ hủy đơn hàng để khách hàng biết được tình trạng đơn hàng của mình.	

Bảng 4. Đặc tả Usecase xác nhận đơn hàng

3.2.5 Usecase giao hàng



Hình 12. Ussecase giao hàng



Hình 13. Lược đồ tuần tự giao hàng

USECASE NAME	Giao hàng
ID	U05
SCENARIO	Nhân viên giao hàng tiến hành giao hàng
TRIGGERING EVENT	Nhân viên giao hàng xem các đơn giao hàng được giao và tiến hành giao hàng
BRIEF DESCRIPTION	Nhân viên giao hàng vào đơn hàng được giao và tìm các đơn giao hàng
ACTORS	Nhân viên giao hàng
PRECONDITION	Khách hàng phải đặt hàng, và đơn hàng đã được duyệt bởi nhân viên tại cửa hàng.
POSTCONDITION	Nhân viên giao hàng vận chuyển đơn giao hàng

FLOW OF EVENT	Actor	System
	1. Nhân viên giao hàng bấm vào danh sách đơn hàng được giao	1. Hệ thống chuyển đến giao diện danh sách đơn hàng
	2. Chọn các đơn giao hàng	2. Hệ thống hiển thị các đơn giao hàng
EXCEPTION CONDITION	Nếu Khách hàng không nhận hàng, hoặc xảy ra tình huống khác khiến đơn hàng không thể giao, Ghi nhận “giao hàng thất bại”.	

Bảng 5. Đặc tả Usecase giao hàng

3.3 Định nghĩa các yêu cầu và ràng buộc với phần mềm

3.3.1 Yêu cầu từ môi trường nghiệp vụ

Usecase	Req-ID	Nội dung yêu cầu	Stack-Holder
U01	B00.1	Khách hàng có thể thanh toán online cho đơn hàng một cách nhanh chóng thông qua xác thực của cổng thanh toán.	Khách hàng
	B00.2	Hệ thống xác thực thông tin thanh toán với cổng thanh toán và gửi thông báo về cho khách hàng	Nhà quản lý
U02	B02.1	Hệ thống cần cung cấp thông tin cụ thể, phục vụ cho việc khách hàng tìm kiếm.	Khách hàng
U03	B00.3	Hệ thống cần lưu trữ và quản lý thông tin chi tiết của từng sản phẩm điện thoại, bao gồm các thông số kỹ thuật, mô tả, giá cả, đánh giá từ người dùng.	Nhà quản lý
	B00.4	đảm bảo rằng thông tin về sản phẩm và giá cả, đánh giá là chính xác, hệ thống cần được cập nhật có khả năng cung cấp thông tin thời gian thực.	Nhà quản lý
U04	B00.5	Hệ thống cần cập nhật các đơn hàng theo thời gian thực để xử lý đơn hàng của khách hàng không bị bỏ qua	Nhà quản lý

U05	B05.1	Nhân viên nhận hàng tiến hành giao hàng theo các đơn giao hàng được yêu cầu.	Nhân viên giao hàng
	B05.2	Nhân viên tiến hành thu tiền khách hàng	Nhân viên giao hàng

Bảng 6. Yêu cầu từ môi trường nghiệp vụ

3.3.2 Yêu cầu từ môi trường vận hành

Usecase	ID	Nội dung yêu cầu	Stack-Holder
U01	NF01.1	Hệ thống lưu đơn hàng mà khách hàng chưa kịp thanh toán	Khách hàng
U03	NF03.1	Hệ thống phải tích hợp với khả năng tìm kiếm cung cấp các tùy chọn lọc kết quả để người dùng có thể chọn sản phẩm để so sánh theo yêu cầu cụ thể	Khách hàng
U03	NF03.2	Hệ thống phải cung cấp hình ảnh sản phẩm để người dùng có thể xem trực quan.	Khách hàng
U04	NF04.1	Hệ thống tự động hủy đơn hàng của khách hàng nếu quá thời gian mà nhân viên không xác nhận.	Chủ doanh nghiệp
	NF04.2	Hệ thống cập nhật liên tục trạng thái đơn hàng của khách hàng nếu có thay đổi.	Chủ doanh nghiệp
U05	NF05.1	Hệ thống cần cập nhật thông tin các đơn hàng hoàn thành chính xác và nhanh nhất để quản lý có thể kiểm tra doanh thu.	Chủ doanh nghiệp
U06	NF06.1	Hệ thống không lưu thông tin thanh toán khách hàng	Điều 75 nghị định 52/2013/NĐ-CP

Bảng 7. Yêu cầu từ môi trường vận hành

3.3.3 Yêu cầu từ môi trường phát triển

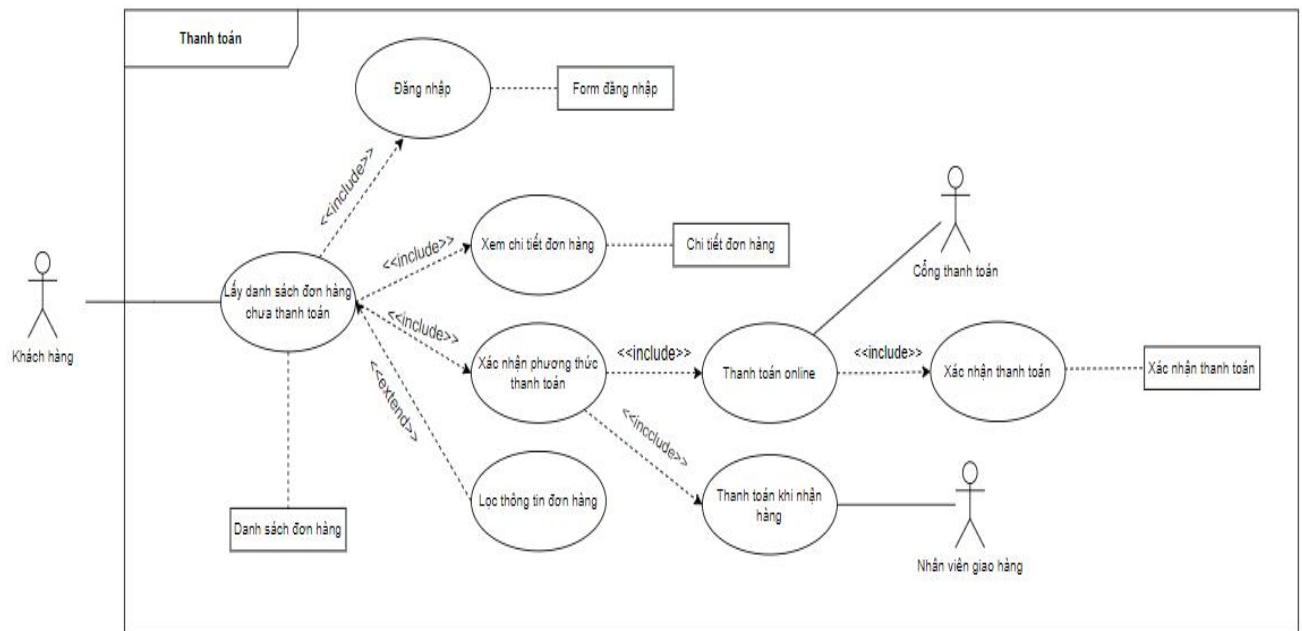
ID	Đối tượng	Nội dung yêu cầu	Stack-Holder
TA01	Hệ điều hành	Window 11	Khách hàng
TA02	Phần cứng	Ram 4GB, CPU intel core i3	Khách hàng
TD03	IDE	Viết code trên Visual Studio Code	Dev team
TD04	Ngôn ngữ lập trình	Typescripts 5.2.0	Dev team
TD05	Framework Frontend	Nextjs 13.3	Dev team
TD06	Payment Gateway	Stripe (Developer Version)	Dev team

Bảng 8. Yêu cầu từ môi trường phát triển

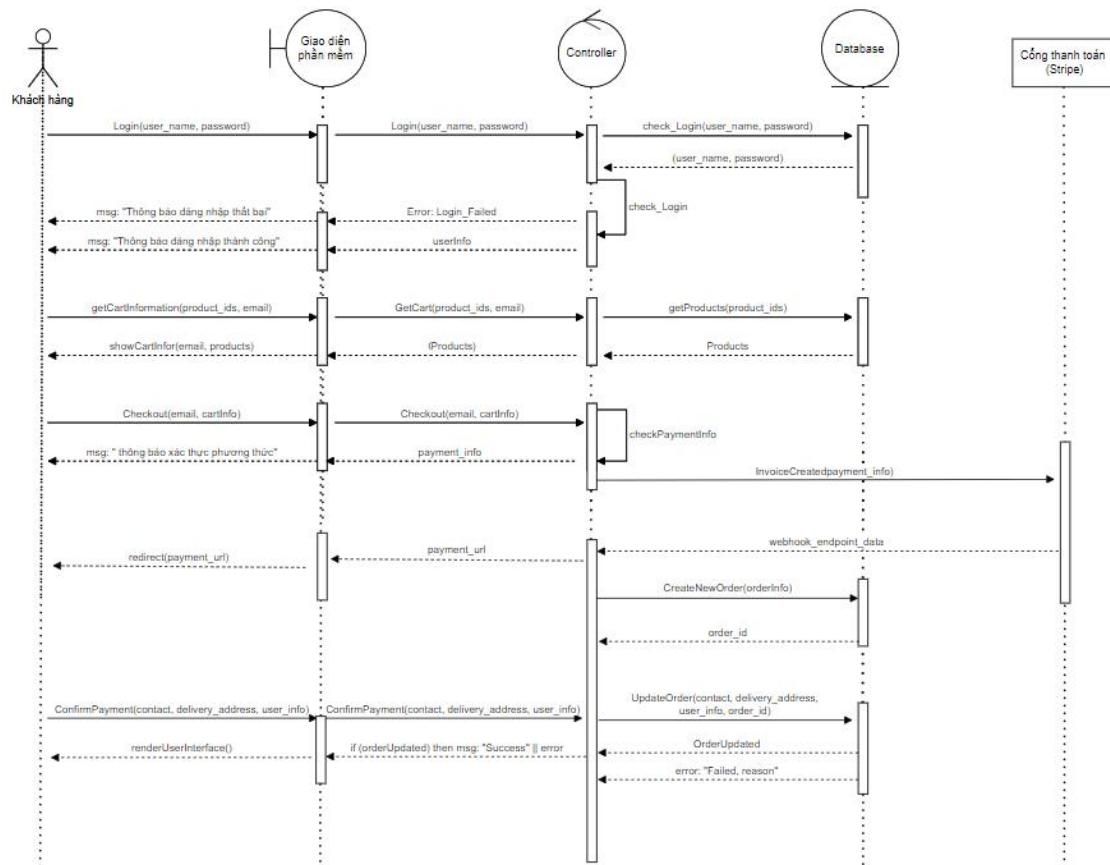
CHƯƠNG 4: THIẾT KẾ PHẦN MỀM VÀ HỆ THỐNG CON

4.1 Lược đồ usecase cho thiết kế phần mềm

4.1.1 Usecase thanh toán

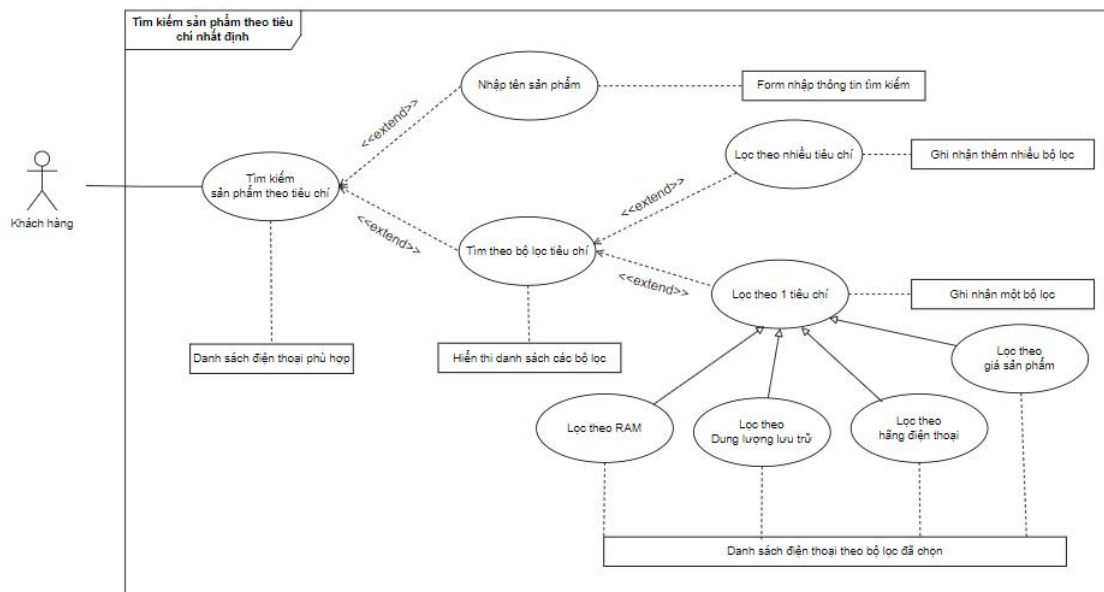


Hình 14. Usecase thanh toán

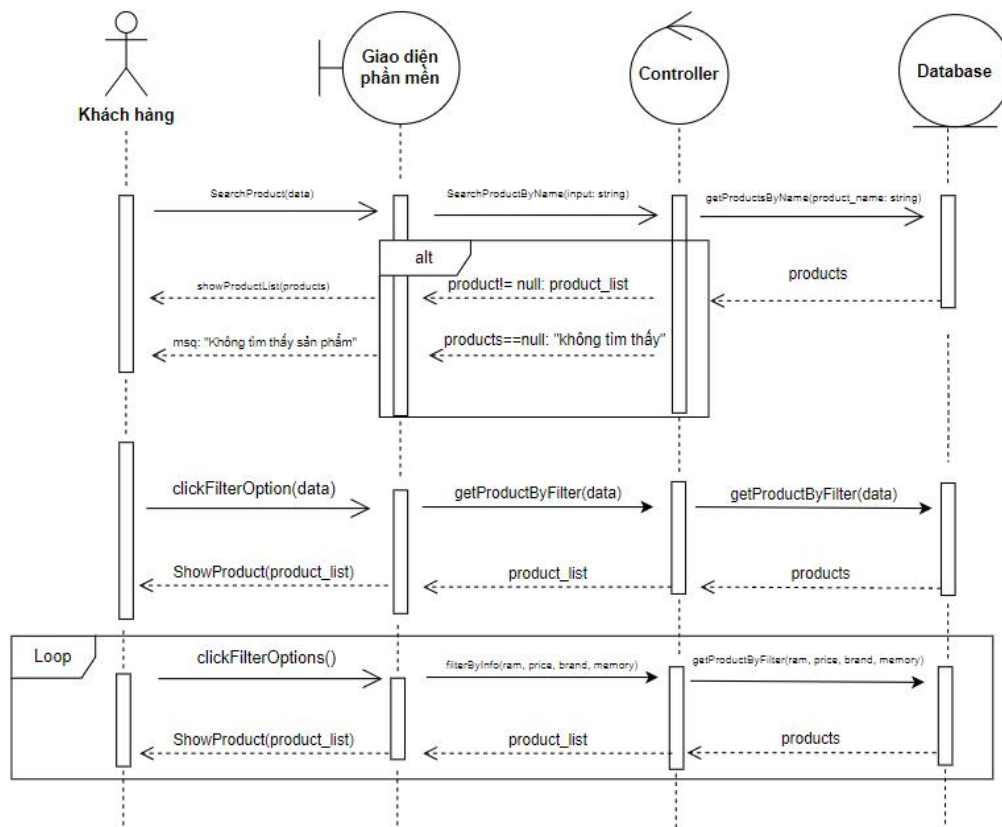


Hình 15. Lược đồ tuần tự thanh toán

4.1.2 Usecase tìm kiếm sản phẩm theo chỉ nhất định

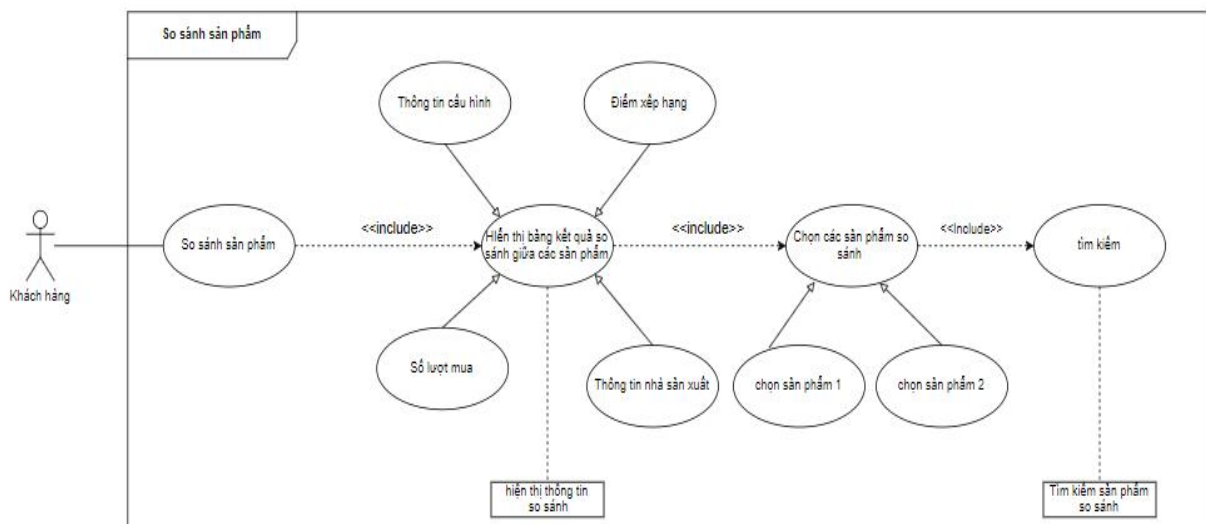


Hình 16. Usecase tìm kiếm sản phẩm theo tiêu chí nhất định

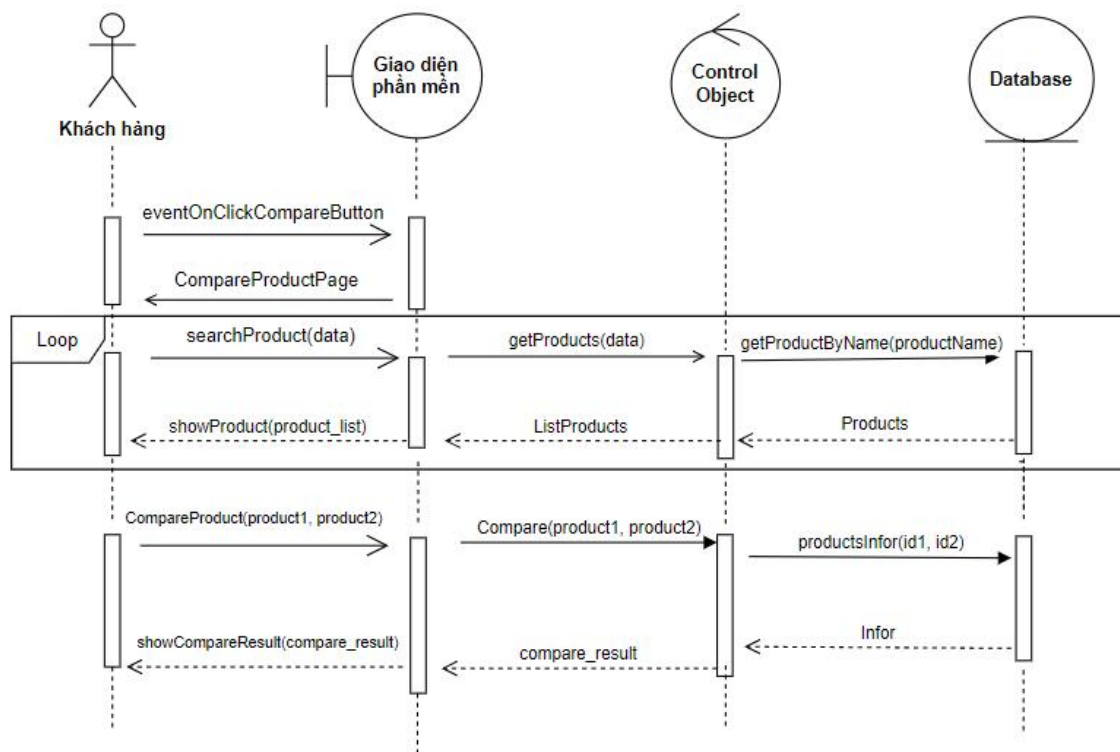


Hình 17. Lược đồ tuần tự tìm kiếm sản phẩm theo tiêu chí nhất định

4.1.3 Usecase yêu cầu so sánh 2 sản phẩm bất kỳ từ khách hàng

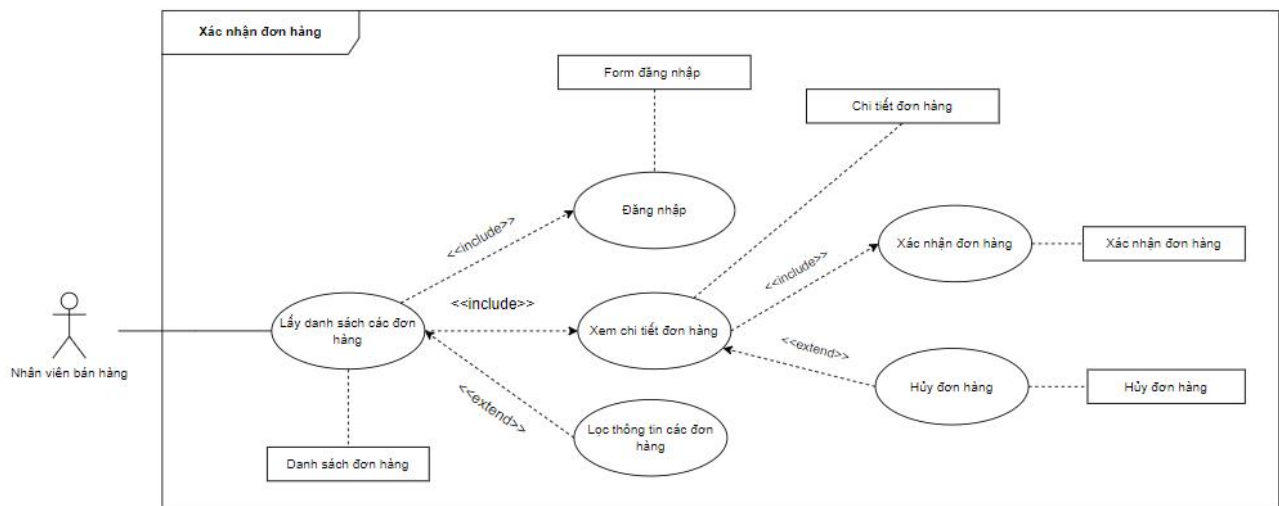


Hình 18. Usecase yêu cầu so sánh 2 sản phẩm bất kỳ từ khách hàng

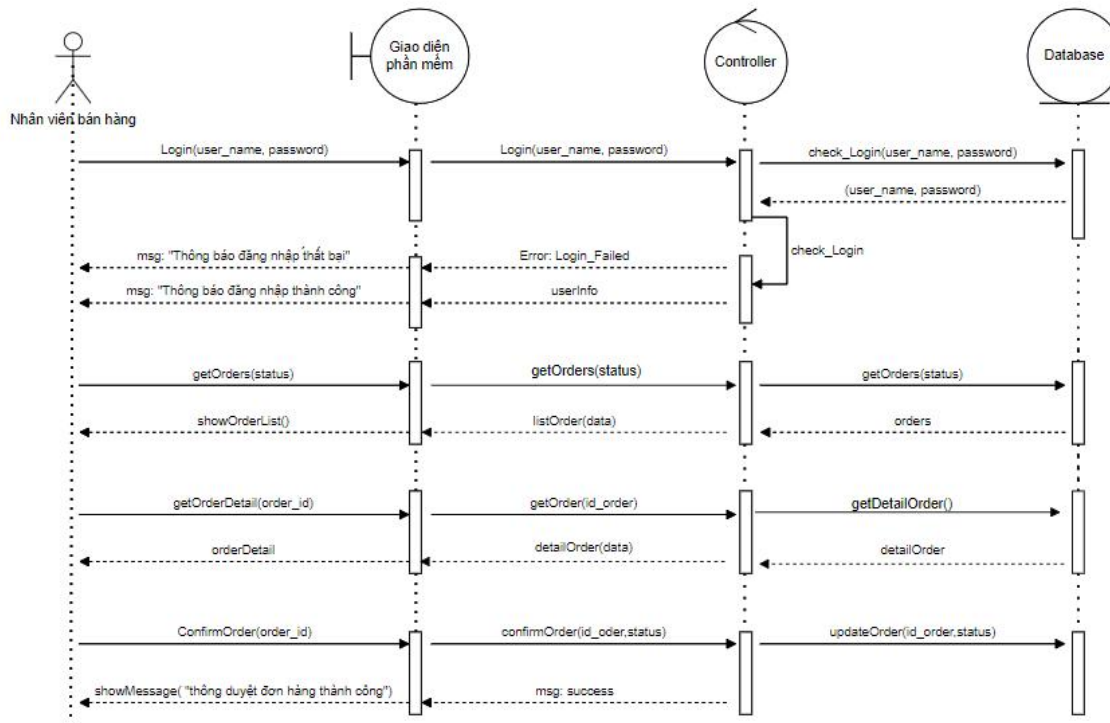


Hình 19. Lược đồ tuần tự yêu cầu so sánh 2 sản phẩm bất kỳ từ khách hàng

4.1.4 Usecase xác nhận đơn hàng

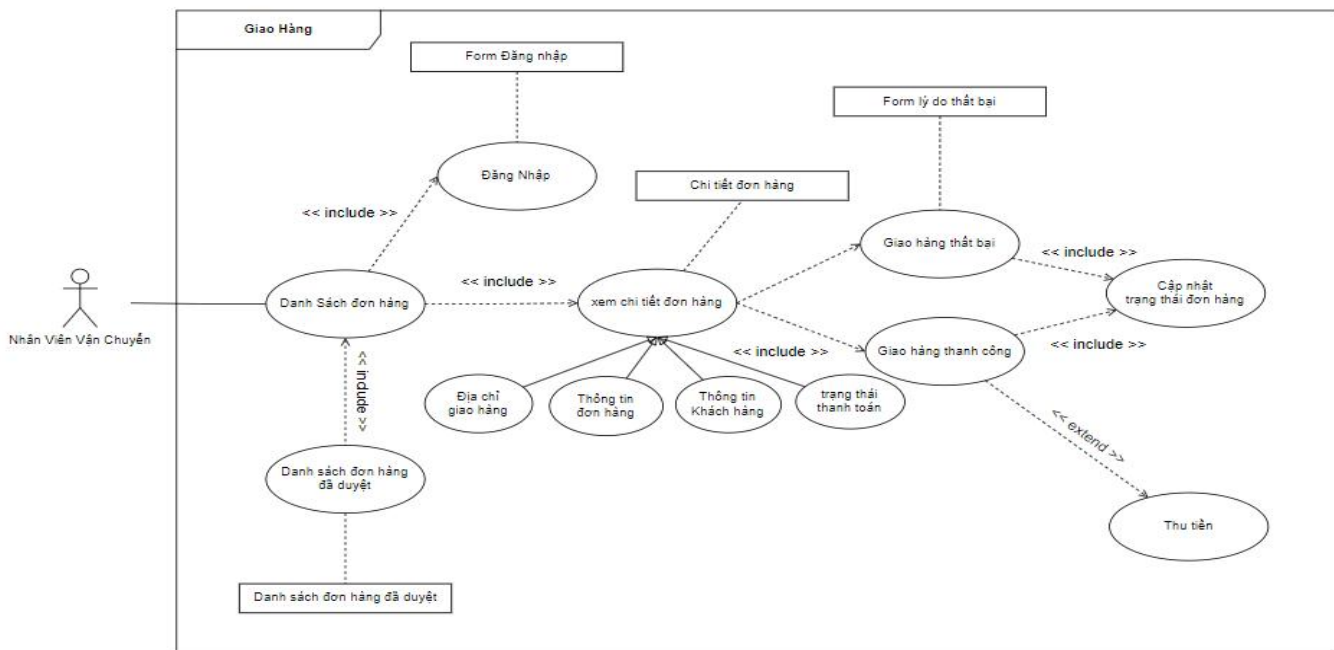


Hình 20. Usecase xác nhận đơn hàng

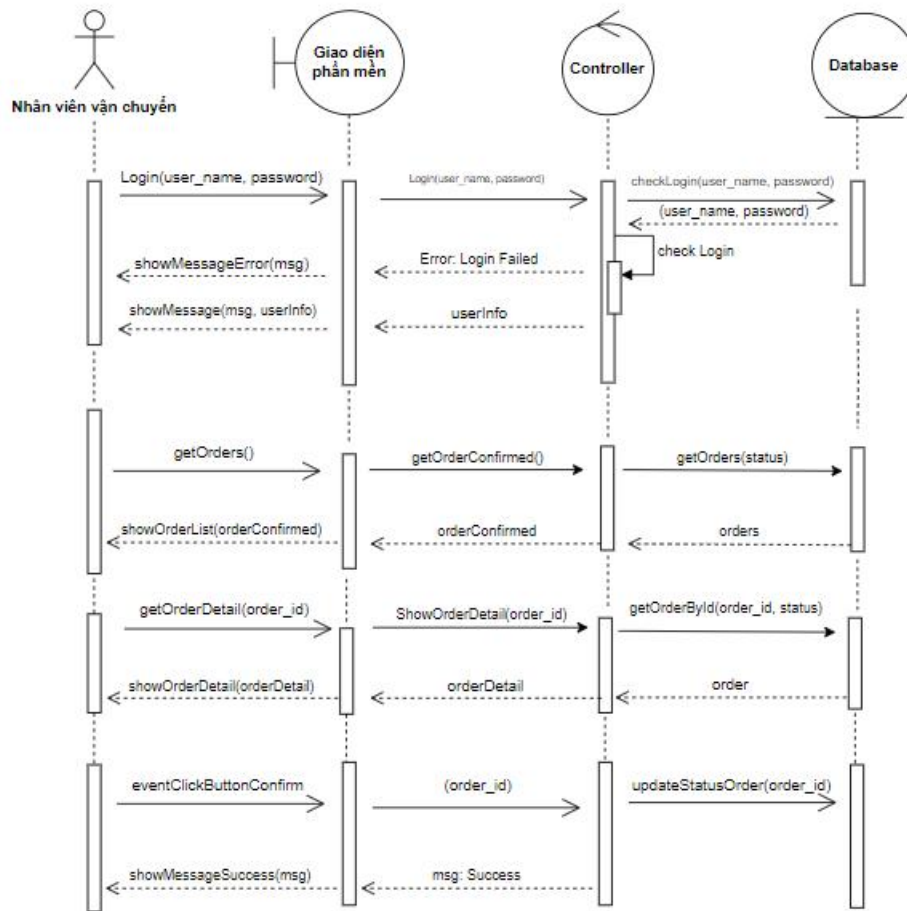


Hình 21. Lược đồ tuần tự xác nhận đơn hàng

4.1.5 Usecase giao hàng



Hình 22. Usecase giao hàng



Hình 23. Lược đồ tuần tự giao hàng

4.2 Thiết kế phần mềm xử lý Usecase

4.2.1 Usecase ID/ Actor Name

- U01/ Khách hàng

Vai trò	Tác nhân sử dụng usecase thanh toán đơn hàng
Tương tác	Form: <ul style="list-style-type: none"> - Form nhập thông tin đơn hàng chưa thanh toán (F01.1) - Form chọn phương thức thanh toán (F01.2) API:

	<ul style="list-style-type: none"> - API danh sách sản phẩm trong giỏ hàng (API 01.1) - API xác thực thanh toán (API 01.2). - API tạo hóa đơn với Stripe (API 01.3). - API lắng nghe phản hồi thông tin thanh toán của Stripe (API 01.4)
--	--

• **U02/ Khách hàng**

Vai trò	Tác nhân sử dụng usecase để tìm kiếm sản phẩm dựa trên các tiêu chí
Tương tác	<p>Form:</p> <ul style="list-style-type: none"> - Form nhập thông tin tìm kiếm (F 02.1) - Form chọn bộ lọc tiêu chí (F02.2) <p>API:</p> <ul style="list-style-type: none"> - API danh sách sản phẩm tìm theo tên (API 02.1) - API danh sách sản phẩm theo theo bộ lọc (API 02.2)

• **U03/Khách hàng**

Vai trò	Khách hàng sử dụng usecase để so sánh sự khác biệt giữa các sản phẩm
Tương tác	<p>Form:</p> <ul style="list-style-type: none"> - Form hiển thị thông tin so sánh (F 03.1) - Form tìm kiếm các sản phẩm so sánh (F03.2) <p>API:</p> <ul style="list-style-type: none"> - API danh sách sản phẩm tìm theo tên (API 02.1) - API so sánh các sản phẩm (API 03.1)

• **U04/ Nhân viên bán hàng**

Vai trò	Tác nhân sử dụng usecase duyệt đơn hàng
Tương tác	<p>Form:</p> <ul style="list-style-type: none"> - Form thông tin chi tiết đơn hàng (F 04.1)

	API: <ul style="list-style-type: none"> - Form chọn lọc các đơn hàng (F04.2) - API danh sách các đơn hàng chưa duyệt (API 04.1) - API danh sách các đơn hàng theo bộ lọc (API 04.2)
--	--

• **U05/Nhân viên vận chuyển**

Vai trò	Tác nhân sử dụng usecase để tìm kiếm sản phẩm dựa trên các tiêu chí
Tương tác	Form: <ul style="list-style-type: none"> - Form Chi tiết đơn hàng (F 05.1) - Form Lý do giao thất bại (F05.2) API: <ul style="list-style-type: none"> - API danh sách đơn hàng đã duyệt (API 05.1) - API Chi tiết đơn hàng (API 05.2) - API cập nhật trạng thái đơn hàng (API 05.3)

4.2.2 Usecase ID/ Form ID

• **U01/F01.1**

Tên Form	Form nhập thông tin đơn hàng
Giao tiếp với user	Khách hàng
Control	<ul style="list-style-type: none"> - Textbox nhập mã giảm giá. - Button “Chọn phương thức thanh toán”. - Button “Đặt hàng”.

Output: Trả về thông báo thanh toán thành công.

Xử lý: Dùng API 01.1 để đọc dữ liệu từ form và tìm kiếm.

• **U01/F01.3**

Tên Form	Form ghi nhận thông tin đơn hàng
Giao tiếp với user	Khách hàng
Control	<ul style="list-style-type: none"> - Textbox “Nhập địa chỉ giao hàng”.

	<ul style="list-style-type: none"> - Button “Chọn khu vực giao hàng”. - button “Chọn địa chỉ tỉnh”. - Button “Chọn huyện thuộc tỉnh”. - Button “Chọn xã thuộc huyện”. - Textbox “Nhập thông tin liên hệ”. - Textbox “Nhập mã thẻ thanh toán”. - Textbox “Nhập CVV”. - Button “Thanh toán”.
--	--

Output: Trả về đường dẫn “endpoint - url” thuộc Stripe cung cấp, giao diện trả về là trang giỏ hàng, kèm trạng thái thành công hoặc thất bại trên url. Đồng thời hiển thị thông báo trên mang hình.

Xử lý: Dùng API 01.3 để ghi nhận thông tin thanh toán vào cơ sở dữ liệu.

• **U02/F02.1**

Tên Form	Form nhập thông tin tìm kiếm.
Giao tiếp với user	Khách hàng
Control	<ul style="list-style-type: none"> - Textbox nhập tên tìm kiếm. - Button Tìm kiếm.

Output: Trả về danh sách sản phẩm phù hợp.

Xử lý: Dùng API 02.1 để đọc dữ liệu từ form và tìm kiếm.

• **U02/F02.2**

Tên Form	Form nhập thông tin tìm kiếm.
Giao tiếp với user	Khách hàng
Control	<ul style="list-style-type: none"> - Button “Lọc theo Ram”. - Button “Lọc theo Dung lượng lưu trữ”. - Buton “Lọc theo Giá thành.” - Button “Lọc Hãng điện thoại” - Button Lọc.

Output: Trả về danh sách sản phẩm phù hợp dựa trên bộ lọc tìm kiếm.

Xử lý: Dùng API 02.2 để đọc dữ liệu từ form và tìm kiếm.

• **U03/F03.1**

Tên Form	Form hiển thị thông tin so sánh
Giao tiếp với user	Khách hàng
Control	<ul style="list-style-type: none"> - TextView “tên phiên bản”. - TextView “giá”. - TextView “ram”. - TextView “bộ nhớ”. - TextView “chip” - TextView “pin”. - TextView “màn hình”. - TextView “camera trước”. - TextView “camera sau” - TextView “số lượt mua”.

Output: Trả về danh sách thông tin so sánh các sản phẩm.

Xử lý: Dùng API 02.1 để đọc dữ liệu vào form so sánh.

• **U03/F03.2**

Tên Form	tìm kiếm các sản phẩm so sánh
Giao tiếp với user	Khách hàng
Control	<ul style="list-style-type: none"> - TextBox “tìm kiếm sản phẩm”. - Button “thêm sản phẩm”. - Button “so sánh”.

Output: Trả về danh sách tên và hình ảnh các sản phẩm cần so sánh.

Xử lý: Dùng API 02.1 để đọc dữ liệu vào form tìm kiếm sản phẩm so sánh

Dùng API 03.1 để so sánh các sản phẩm.

• **U04/F04.1**

Tên Form	Form thông tin chi tiết đơn hàng
Giao tiếp với user	Nhân viên bán hàng
Control	<ul style="list-style-type: none"> - Textbox địa chỉ giao hàng (readonly). - Textbox Tên Khách hàng (readonly). - Textbox địa chỉ liên hệ khách hàng. (readonly).

	<ul style="list-style-type: none"> - Textbox Thông tin đơn hàng (readonly) - Textbox Thông tin trạng thái thanh toán (readonly). - Button “Xác nhận đơn hàng”.
--	---

Output: Trả về thông tin chi tiết đơn hàng.

Xử lý: Dùng API 06.1 để đọc dữ liệu từ form.

• **U04/F04.2**

Tên Form	Form chọn lọc các đơn hàng
Giao tiếp với user	Nhân viên bán hàng.
Control	<ul style="list-style-type: none"> - Button " Previous ". - Button " Next ".

Output: Trả về thông tin các đơn hàng.

Xử lý: Dùng API 04.2 để đọc dữ liệu từ form.

• **U05/F05.1**

Tên Form	Form chi tiết đơn hàng
Giao tiếp với user	Nhân viên vận chuyển.
Control	<ul style="list-style-type: none"> - Textbox địa chỉ giao hàng (readonly). - Textbox Tên Khách hàng (readonly). - Textbox địa chỉ liên hệ khách hàng. (readonly). - Textbox Thông tin đơn hàng (readonly) - Textbox Thông tin trạng thái thanh toán (readonly). - Button OK.

Output: Trả về thông tin chi tiết đơn hàng.

Xử lý: Dùng API 06.2 để đọc dữ liệu từ form.

• **U05/F05.2**

Tên Form	Form Lý do không nhận hàng.
Giao tiếp với user	Nhân viên vận chuyển.
Control	<ul style="list-style-type: none"> - Textbox nhập lý do hoàn đơn hàng - Textbox nhập thời gian khách không nhận hàng. - Button “Xác nhận đơn hàng giao thất bại.”

Output: Trả về thông báo gửi yêu cầu hoàn đơn.

Xử lý: Dùng API 05.3 ghi dữ liệu từ form và gửi yêu cầu về sever để thay đổi trạng thái đơn hàng và trả về thông báo hoàn đơn

4.2.3 Usecase ID/ API ID

- **U01/API 01.1 (API lấy danh sách các sản phẩm trong giỏ hàng thuộc khách hàng).**

Input	Form 01.1 param: email
Xử lý	Gọi SP_GET_CART_BY_USER_ID để lấy danh sách các sản phẩm thuộc khách hàng đang yêu cầu.
Output	[{ Products }]
Type	GET
Table	Order

- **U01/API 01.2 (API xác thực thông tin thanh toán)**

Input	Form 01.3 body ({address_delivery (VN + tỉnh + huyện + xã + số nhà), contact, mã thẻ, cvv}).
Xử lý	Gọi SP_CREATE_ORDER_ONLINE để tạo mới đơn hàng, một vài field chưa được ghi nhận thì để trống, ví dụ mới tạo thì chưa thể điền nhân viên nào mới tạo, trạng thái pending.
Output	[{ OrderCreated }]
Type	POST
Table	Order

- **U01/API 01.3 (API API tạo hóa đơn với Stripe).**

Input	Thông tin đơn hàng vừa tạo (order)
Xử lý	<ol style="list-style-type: none"> 1. Gọi api đến endpoint là cổng thanh toán stripe, lúc này stripe ghi nhận mã thẻ, thông tin thanh toán và sau đó đem các thông tin này đi xác thực, kiểm tra số dư, ... nếu hợp lệ, xác nhận thanh toán thành công, ngược lại thất bại. 2. Gọi SP_CREATE_ORDER_DETAIL để ghi nhận danh sách sản phẩm thuộc đơn hàng đã tạo, và ghi nhận số lượng mua.

	3. Gọi SP_UPDATE_QUANTITY_PRODUCT để trừ đi số lượng sản phẩm đã thanh toán thành công.
Output	[{ payment_intent.succeeded }]
Type	POST
Table	Order

• **U02/API 02.1 (API Tìm Danh sách sản phẩm theo tên)**

Input	Form 02.1, params: ./product_name: string
Xử lý	<ol style="list-style-type: none"> Sử dụng sp SP_GET_PRODUCT_BY_NAME (data) để tìm sản phẩm có tên phù hợp. Trả về danh sách các sản phẩm phù hợp theo tên.
Output	[{ "model_name", "vote", "price", "unit_price", "quantity", "status", "description", "operation_system", "hardware", "warranty_time", "category", "discount", "color", "battery", "screen", "memory", "front_camera", "behind_camera", "ram",

	<code>},</code> <code>]</code>
Type	GET
Table	Product

• **U02/API 02.2 (API Tìm Danh sách sản phẩm phù hợp theo bộ lọc)**

Input	Form 02.2, params: (ram, brand, price, memory)
Xử lý	<ol style="list-style-type: none"> Sử dụng sp SP_GET_PRODUCT_BY_FILTERS (ram, price, memory, brand) để tìm sản phẩm phù hợp. Trả về danh sách các sản phẩm phù hợp theo bộ lọc. Chuẩn bị sẵn các hàm bộ lọc theo tiêu chí (ví dụ: SP_GET_PRODUCT_BY_RAM – lọc theo ram, tương tự các bộ lọc “memory, price, brand”). SP_GET_PRODUCT_BY_RAM sẽ trả về một mảng các id của product phù hợp điều kiện lọc. Phần Body API có thể có đủ 4 tiêu chí hoặc không, tùy thuộc vào khách hàng chọn. Lấy các kết quả trả về của từng bộ lọc đem giao với nhau, được kết quả là mảng id của sản phẩm, từng mảng ấy đem đi tìm sản phẩm (SP_GET_PRODUCT_BY_IDS) và trả về một mảng object JSON.
Output	<pre>[{ "model_name", "vote", "price", "unit_price", "quantity", "status", "description", "operation_system", "hardware", "warranty_time", "category", "discount","color", "battery", "screen", "memory", "front_camera" },]</pre>
Type	GET
Table	Product

• **U03/API 03.1**

Input	Form 03.2, params (productID1, productID2)
Xử lý	<ol style="list-style-type: none"> Lấy thông tin sản phẩm thứ nhất qua hàm getProductById (productID1) thông qua SP_GET_PRODUCT_INFOR (productID1) Lấy thông tin sản phẩm thứ hai qua hàm getProductById (productID1) thông qua SP_GET_PRODUCT_INFOR

	(productID2) 3. Nếu tìm thấy thông tin 2 sản phẩm trả về OUTPUT 4. Nếu không tìm thấy thông tin sản phẩm trả về MESSAGE: “không tìm thấy sản phẩm”
Output	<pre>[{ "model_name", "category", "rating", "vote", "price", "unit_price", "quantity", "status", "description", "operation_system", "hardware", "warranty_time", "category", "discount", "color", "battery", "screen" "memory", "front_camera", "behind_camera", "ram", }, { "model_name", "category", "rating", "vote", "price", "unit_price", "quantity", "status",</pre>

	<pre> "description", "operation_system", "hardware", "warranty_time", "category", "discount", "color", "battery", "screen" "memory", "front_camera", "behind_camera", "ram", }] </pre>
Type	GET
Table	Products

• U04/API 04.1

Input	
Output	[{ Orders, }]
Type	GET
Table	Order

• U04/API 04.2

Input	
Output	[{ Orders, }]
Type	GET

Table	Order
-------	-------

• **U05/API 05.1**

Input	Status = “confirmed”
Xử lý	Gọi SP_GET_ORDERS_CONFIRMED truyền vào status để lấy ra danh sách các đơn hàng thỏa điều kiện đã confirmed.
Output	[{ Orders, }]
Type	GET
Table	Order

• **U05/API 05.2**

Input	Order_id
Xử lý	Gọi SP_GET_ORDER để lấy thông tin đơn hàng.
Output	{ Address_shipping, Order_info, User_info, Order_status, }
Type	GET
Table	Order

• **U05/API 05.3**

Input	Order_id
Xử lý	Gọi SP_UPDATE_STATUS_ORDER_SHIPPER để cập nhật trạng thái đơn hàng ghi nhận shipper đã giao.
Output	Message: (“huỷ đơn giao hàng, trả hàng về cửa hàng”) Message: (“giao hàng thành công”)
Type	PATCH

Table	Order
-------	-------

Mô tả: Khi nhân viên giao hàng không giao hàng được cho khách hàng trong thời gian dài thì nhân viên giao hàng tiến hành hoàn kho đơn hàng kèm theo lý do được ghi nhận

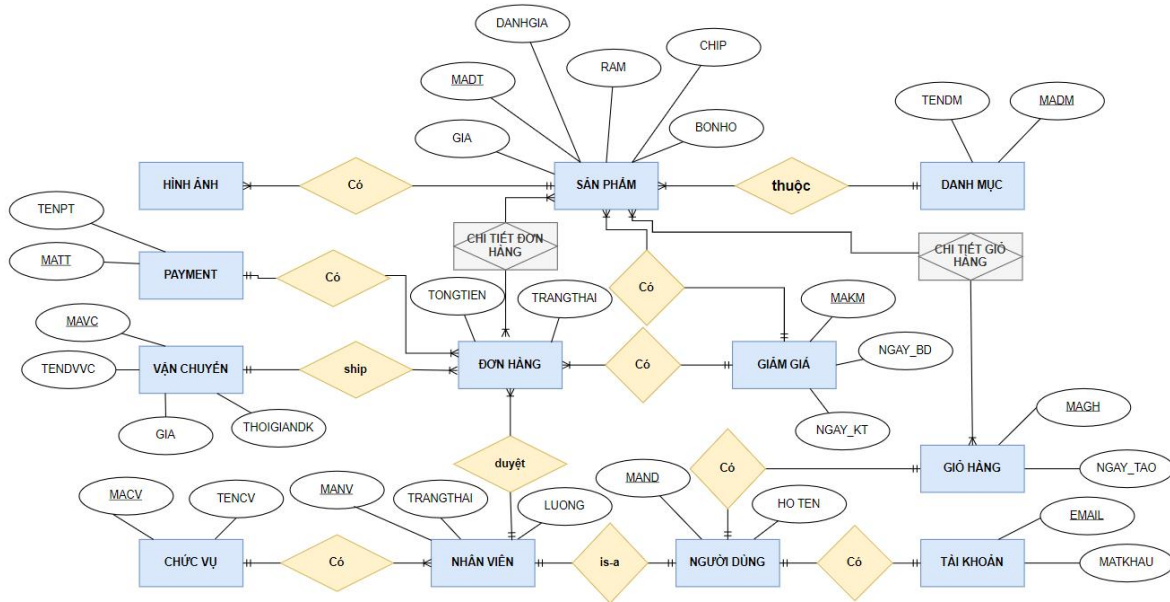
4.3 Thiết kế cơ sở dữ liệu cho phần mềm

4.3.1 Mức ý niệm (Lược đồ ERD)

4.3.1.1 Xác định thực thể

- SANPHAM (MADT, TEN_DT, GIA, SO_LUONG, MO_TA, TRANG_THAI,)
- HINH_ANH (MAHA, URL)
- DANH MUC (MADM, TENDANH MUC, MOTA)
- NHAN_VIEN (MANV, LUONG, TRANGTHAI, GIOI_TINH)
- KHUYEN_MAI (MAKM, MOTA, NGAY_BAT_DAU, NGAY_KET_THUC, PHANTRAM)
- GIOHANG (MAGH, NGAYTAO)
- CHUCVU (MACV, TENCV, CHITIET)
- VANCHUYEN (MAVC, TENDVVC, GIAVC, THOIGIANDK)
- DONHANG (MADONHANG, NGAYTAO, NGAYCAPNHAT, TONG TIEN, TRANGTHAI, DIACHIGIAO, LIENLAC)
- THANH TOAN (MATHANH TOAN, TENPT, MOTA)
- TAIKHOAN (EMAIL, MATKHAU, REFESHTOKEN)
- NGUOIDUNG (MAND, HO, TEN, PHONE, DIACHI, GIOITINH, AVATAR)

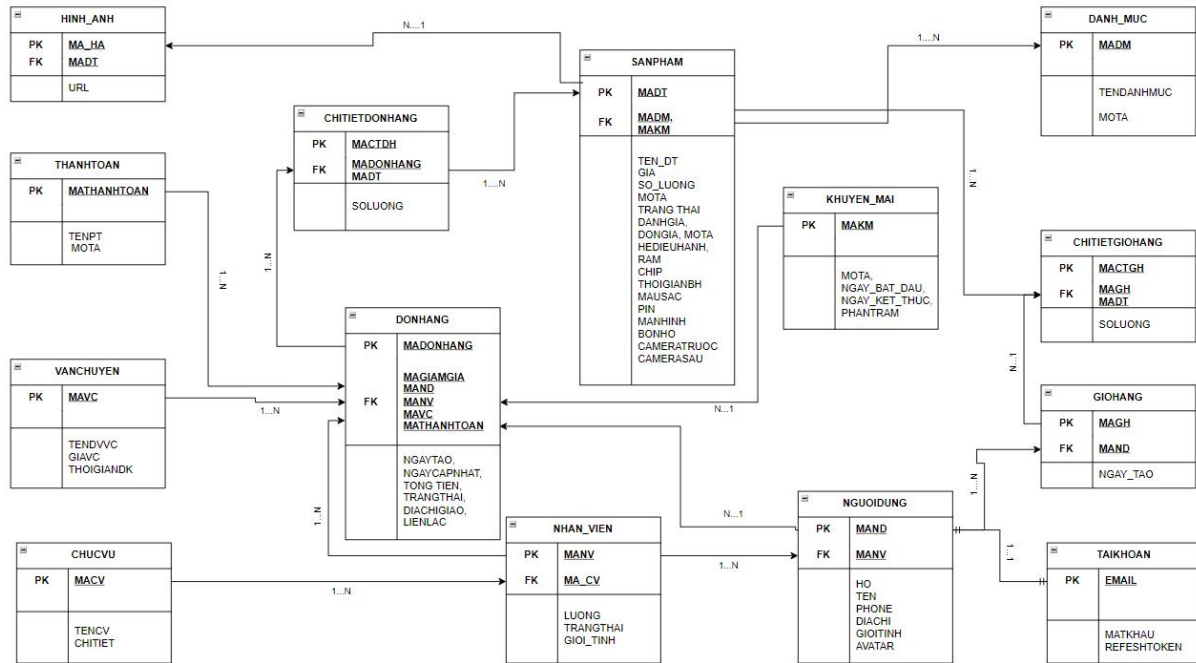
4.3.1.1 ERD



Hình 24. Lược đồ ERD

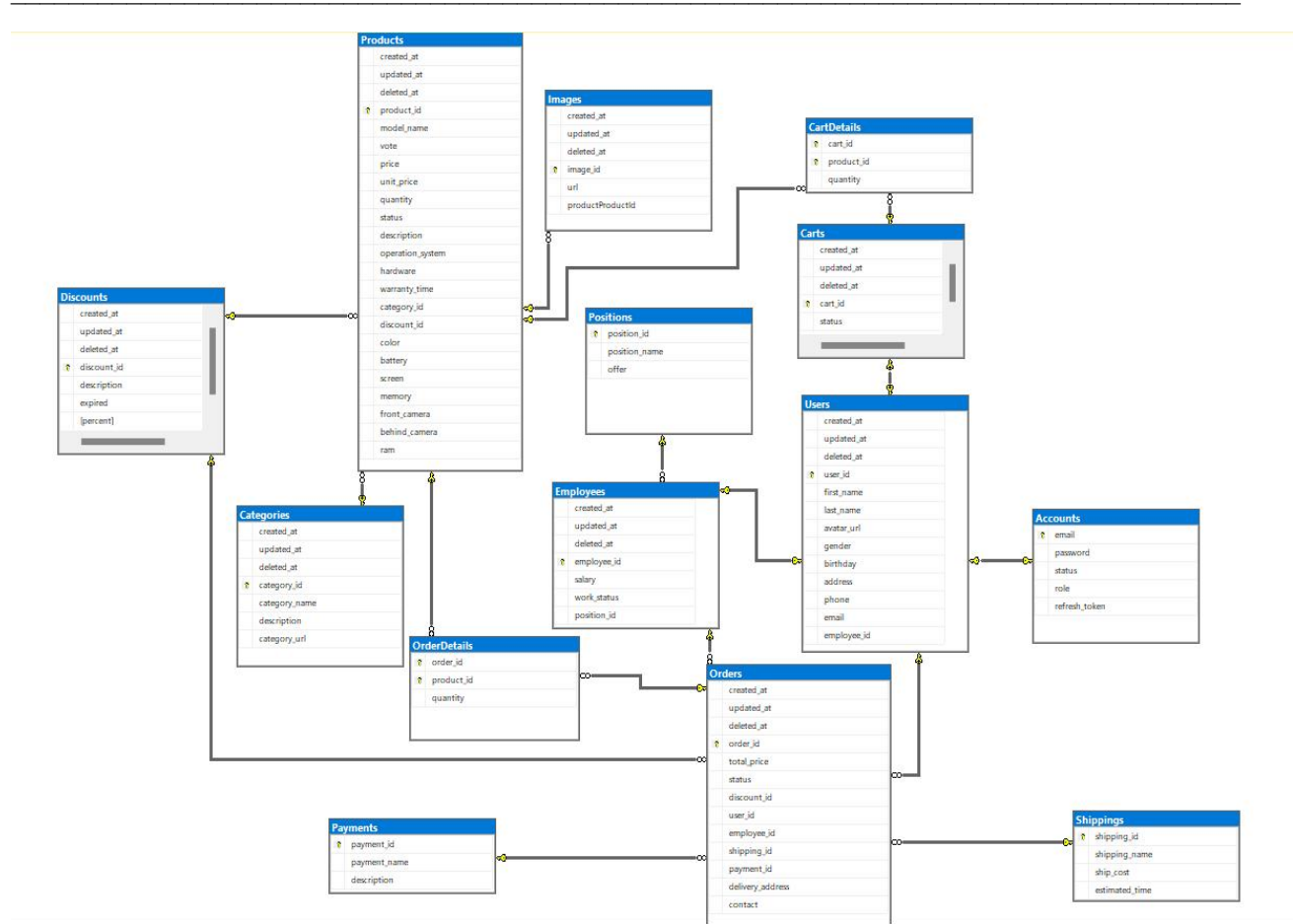
4.3.2 Mức thiết kế (Lược đồ Bảng Chuẩn 3):

4.3.2.1 Mô hình thực thể đạt chuẩn 3



Hình 25: Lược đồ bảng đạt chuẩn 3

4.3.2.1 Mô hình Diagram



Hình 26. Mô hình diagram

4.3.1.2 Từ điển dữ liệu

A. Nhóm Người Dùng

- ❖ **Accounts** (email, password, refresh_token, role, status)

Accounts Table			
FIELD	DATA TYPE	CONSTRAINT	MEANING
email	Nvarchar (50)	primary key	Tên tài khoản là email
password	Nvarchar (50)	Not null	Mật khẩu
refresh_token	Nvarchar (225)	Default: Null	Refresh token
role	Nvarchar (50)	Not Null, Default: user	Vai trò
status	bit	Default: 1	Trạng thái tài khoản

➤ Chi tiết phần quyền ở Accounts:

TABLE	PERMISSION	DATABASE ROLE				
		Nhà quản lý	Nhân viên bán hàng	Nhân viên vận chuyển	Khách hàng	Người dùng không đăng nhập

Accounts	Select	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Insert	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Update	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Delete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

❖ **User** (user_id, first_name, last_name, gender, birthday, address, phone, email, employee_id)

Users Table			
FIELD	DATA TYPE	CONSTRAINT	MEANING
user_id	int	Primary key, AUTO_INCREMENT	Mã Người dùng
email	Nvarchar (50)	FK, not null	Tài khoản
avatar_url	Nvarchar (50)	null	Ảnh đại diện
first_name	Nvarchar (50)	Not Null	Tên
last_name	Nvarchar (225)	Null	Họ
gender	Nvarchar (50)	null	Giới tính
birthday	Datetime	Not null	Sinh nhật
address	Nvarchar (225)	Not null	Địa chỉ
phone	Nvarchar (10)	Not null	Số điện thoại
employee_id	Nvarchar (50)	FK, Null	Mã nhân viên

➤ Chi tiết phân quyền ở bảng User:

TABLE	PERMISSION	DATABASE ROLE				
		Nhà quản lý	Nhân viên bán hàng	Nhân viên vận chuyển	Khách hàng	Người dùng không đăng nhập
Users	Select	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Insert	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Update	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Delete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

❖ **Positions** (position_id, position_name, offer)

Positions Table			
FIELD	DATA TYPE	CONSTRAINT	MEANING
position_id	int	Primary Key, AUTO_INCREMENT	Mã chức vụ
position_name	Nvarchar (50)	Not Null	Tên chức vụ
offer	int	Not null, default:1	Yêu cầu

➤ Chi tiết phân quyền ở bảng Position (Chức vụ):

TABLE	PERMISSION	DATABASE ROLE				
		Nhà	Nhân viên	Nhân viên	Khách	Người dùng

		quản lý	bán hàng	vận chuyển	hàng	không đăng nhập
Positions	Select	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Insert	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Update	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Delete	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

❖ **Employees** (employee_id, salary, work_status, position_id)

Employee Table			
FIELD	DATA TYPE	CONSTRAINT	MEANING
employee_id	Nvarchar (50)	Primary key	Mã Nhân viên
salary	int	Not null, default:0	Lương
work_status	bit	Not Null	Trạng thái làm
position_id	int	FK, not null	Mã chức vụ
timestamp	datetime	Not null	Dấu Thời gian

➤ Chi tiết phân quyền ở bảng Employee:

TABLE	PERMISSION	DATABASE ROLE				
		Nhà quản lý	Nhân viên bán hàng	Nhân viên vận chuyển	Khách hàng	Người dùng không đăng nhập
Employees	Select	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Insert	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Update	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Delete	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

B. Nhóm Sản phẩm❖ **Categories** (category_id, category_name, description)

Categories Table			
FIELD	DATA TYPE	CONSTRAINT	MEANING
category_id	int	Primary key, AUTO INCREMENT	Mã hãng sản xuất
category_name	Nvarchar (50)	Not null	Tên hãng sản xuất
description	Nvarchar (225)	Null	Mô tả
timestamp	datetime	Not null	Dấu Thời gian

➤ Chi tiết phân quyền ở bản Categories:

TABLE	PERMISSION	DATABASE ROLE				
		Nhà quản lý	Nhân viên bán hàng	Nhân viên vận chuyển	Khách hàng	Người dùng không đăng nhập
Categories	Select	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

	Insert	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Update	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Delete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

❖ **Discounts** (discount_id, expired, description, percent)

Discounts Table			
FIELD	DATA TYPE	CONSTRAINT	MEANING
discount_id	int	Primary key, AUTO_INCREMENT	Mã khuyến mãi
expired	datetime	Not null	Thời hạn
description	Nvarchar (50)	Null	Mô tả
percent	int	Not null	Phần trăm giảm
timestamp	datetime	Not null	Dấu thời gian

➤ Chi tiết phân quyền ở bảng Discounts:

TABLE	PERMISSION	DATABASE ROLE				
		Nhà quản lý	Nhân viên bán hàng	Nhân viên vận chuyển	Khách hàng	Người dùng không đăng nhập
Discounts	Select	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Insert	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Update	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Delete	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

❖ **Images** (image_id, url, product_id)

Images Table			
FIELD	DATA TYPE	CONSTRAINT	MEANING
image_id	uuid	Primary key	Mã Hình Ảnh
url	Nvarchar (50)	Not null	Đường liên kết
product_id	int	FK	Mã sản phẩm
timestamp	datetime	Not null	Dấu Thời gian

➤ Chi tiết phân quyền ở bảng Images:

TABLE	PERMISSION	DATABASE ROLE				
		Nhà quản lý	Nhân viên bán hàng	Nhân viên vận chuyển	Khách hàng	Người dùng không đăng nhập
Images	Select	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Insert	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Update	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Delete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

❖ **Products** (product_id, model_name, vote, price, unit_price, quantity, status, description, operation_system, hardware, warranty_time, color, front_camera, behind_camera, ram, memory)

Products Table			
FIELD	DATA TYPE	CONSTRAINT	MEANING
product_id	int	Primary key, AUTO_INCREMENT	Mã sản phẩm
model_name	Nvarchar (50)	Not null	Tên dòng sản phẩm
vote	int	Null	Lượt bình chọn
price	Decimal (5, 2)	Not null	Giá bán
unit_price	Decimal (5, 2)	Not null	Đơn giá (giá nhập)
quantity	int	Not null	Số lượng
status	bit	Not null	Trạng thái
description	Nvarchar (50)	null	Mô tả
operation_system	Nvarchar (50)	Not null	Hệ điều hành
hardware	Nvarchar (50)	Not null	Phần cứng
warranty_time	Nvarchar (50)	Null	Thời gian bảo hành
color	Nvarchar (50)	Not null	Màu sắc
battery	int	Not null	PIN
screen	Decimal (10, 2)	Not null	Kích thước màn hình
memory	int	Not null	Dung lượng lưu trữ
front_camera	Int	Not null	Camera trước
behind_camera	Int	Not null	Camera sau
ram	int	Not null	Bộ nhớ ram
category_id	Int	FK	Mã hãng sản xuất
discount_id	int	FK	Mã khuyến mãi
timestamp	datetime	Not null	Dấu thời gian

➤ Chi tiết phân quyền ở bảng Products:

TABLE	PERMISSION	DATABASE ROLE				
		Nhà quản lý	Nhân viên bán hàng	Nhân viên vận chuyển	Khách hàng	Người dùng không đăng nhập
Products	Select	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Insert	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Update	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Delete	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

C. Nhóm Đơn hàng

❖ **Carts** (cart_id, status, user_id)

Carts Table			
FIELD	DATA TYPE	CONSTRAINT	MEANING
cart_id	int	Primary key, AUTO INCREMENT	Mã Giỏ hàng
status	bit	Not null	Trạng thái Giỏ hàng
user_id	int	Not Null	Mã Người dùng
timestamp	datetime	Not null	Dấu Thời gian

➤ Chi tiết phân quyền ở bảng Carts:

TABLE	PERMISSION	DATABASE ROLE				
		Nhà quản lý	Nhân viên bán hàng	Nhân viên vận chuyển	Khách hàng	Người dùng không đăng nhập
Carts	Select	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Insert	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Update	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Delete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

❖ **CartDetails** (cart_id, product_id, quantity)

CartDetails Table			
FIELD	DATA TYPE	CONSTRAINT	MEANING
cart_id	int	foreign key (FK)	Mã Giỏ hàng
product_id	int	foreign key (FK)	Mã sản phẩm
quantity	int	Not Null	Mã Người dùng

➤ Chi tiết phân quyền ở bảng CartDetails:

TABLE	PERMISSION	DATABASE ROLE				
		Nhà quản lý	Nhân viên bán hàng	Nhân viên vận chuyển	Khách hàng	Người dùng không đăng nhập
Cart Details	Select	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Insert	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Update	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Delete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

❖ **Shippings** (shipping_id, shipping_name, ship_cost)

Shippings Table			
FIELD	DATA TYPE	CONSTRAINT	MEANING
shipping_id	int	Primary key, AUTO INCREMENT	Mã đơn vị vận chuyển

shipping_name	Nvarchar (50)	Not null	Tên đơn vị vận chuyển
ship_cost	int	Default: 0	Chi phí Vận chuyển
estimated_time	int	Not null	Thời gian dự kiến

➤ Chi tiết phân quyền bảng Shippings:

TABLE	PERMISSION	DATABASE ROLE				
		Nhà quản lý	Nhân viên bán hàng	Nhân viên vận chuyển	Khách hàng	Người dùng không đăng nhập
Shippings	Select	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Insert	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Update	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Delete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

❖ **Payments** (payment_id, payment_name, description)

Payemnts Table			
FIELD	DATA TYPE	CONSTRAINT	MEANING
payment_id	int	Primary key, AUTO INCREMENT	Mã phương thức thanh toán
payment_name	Nvarchar (100)	Not null	Tên phương thức thanh toán
description	Nvarchar (225)	Null	Mô tả

➤ Chi tiết bảng phân quyền bảng Payments:

TABLE	PERMISSION	DATABASE ROLE				
		Nhà quản lý	Nhân viên bán hàng	Nhân viên vận chuyển	Khách hàng	Người dùng không đăng nhập
Payments	Select	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Insert	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Update	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Delete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

❖ **Orders** (order_id, total_price, status, shipping_id, payment_id, employee_id, user_id, discount_id)

Orders Table			
FIELD	DATA TYPE	CONSTRAINT	MEANING
order_id	int	Primary key,	Mã đơn hàng

		AUTO INCREMENT	
total_price	Nvarchar (50)	Not null, default: 0	Tổng tiền
status	Nvarchar (50)	Not Null	Trạng thái đơn hàng
shipping_id	int	FK, null	Mã đơn vị vận chuyển
payment_id	int	FK, not null	Mã phương thức thanh toán
employee_id	Nvarchar (50)	FK, not null	Mã nhân viên
user_id	int	FK, not null	Mã Khách hàng
discount_id	int	FK, null	Mã khuyến mãi
delivery_address	Nvarchar (225)	Not null	Địa chỉ nhận hàng
contact	Nvarchar (25)	null	Liên hệ

➤ Chi tiết phân quyền bảng Orders:

TABLE	PERMISSION	DATABASE ROLE				
		Nhà quản lý	Nhân viên bán hàng	Nhân viên vận chuyển	Khách hàng	Người dùng không đăng nhập
Orders	Select	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Insert	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Update	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Delete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

❖ **OrderDetails** (order_id, product_id, quantity)

OrderDetails Table			
FIELD	DATA TYPE	CONSTRAINT	MEANING
order_id	int	foreign key (FK)	Mã đơn hàng
product_id	int	foreign key (FK)	Mã sản phẩm
quantity	int	Not Null	Mã Người dùng

➤ Chi tiết phân quyền bảng OrderDetails:

TABLE	PERMISSION	DATABASE ROLE				
		Nhà quản lý	Nhân viên bán hàng	Nhân viên vận chuyển	Khách hàng	Người dùng không đăng nhập
Order Details	Select	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Insert	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Update	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Delete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

4.3.3 Ràng buộc toàn vẹn

4.3.3.1 Ràng buộc toàn vẹn miền giá trị:

- RB: Trong lược đồ quan hệ NHANVIEN, ta có $Phai = \{ 'Nam', 'Nữ' \}$
NHAN_VIEN(NHAN_VIEN (MANV, LUONG, TRANGTHAI, GIOI_TINH, MACV)

Nội dung: $\forall NV \in NHANVIEN : NV.Phai = 'Nam' \text{ OR } NV.Phai = 'Nữ'$

Bảng tầm ảnh hưởng:

RB	Thêm	Xóa	Sửa
NHANVIEN	+	-	+(Phai)

- RB: Trong lược đồ quan hệ KHUYENMAI ta có PHANTRAM <100
KHUYEN_MAI (MAKM, MOTA, NGÀY_BAT_DAU, NGÀY_KET_THUC, PHANTRAM)
 Nội dung: $\forall KH \in KHUYENMAI : KH.PHANTRAM > 0 \text{ OR } KH.PHANTRAM < 100$

Bảng tầm ảnh hưởng:

RB	Thêm	Xóa	Sửa
KHUYENMAI	+	-	+(PHANTRAM)

4.3.3.2 Ràng buộc toàn vẹn liên thuộc tính:

- RB: Trong bảng DONHANG ngày tạo nhỏ hơn ngày cập nhật
DONHANG (MADONHANG, NGÀYTAO, NGÀYCAPNHAT, TONG TIEN, TRANGTHAI, DIACHIGIAO, LIENLAC, MAGIAMGIA, MAND, MANV, MAVC, MATHANHTOAN)

Nội dung: $\forall DH \in DONHANG : DH.NGÀYTAO \Rightarrow DH.NGÀYCAPNHAT$

Bảng tầm ảnh hưởng:

RB	Thêm	Xóa	Sửa
DONHANG	+	-	+(DH.NGÀYCAPNHAT)

4.4 Store Procedure:

4.4.2 U01/SP_GET_CART_BY_USER_ID

Name	SP_GET_CART
Input	User_id
Output	Carts (<u>cart_id</u> , status, user_id)
Table	Carts, Users

4.4.3 U01/SP_GET_PAYMENT

Name	SP_GET_PAYMENT
Input	PaymentID
Output	Payments (payment_id, payment_name, description)
Table	Payments

4.3.2 U01/SP_CREATE_ORDER_ONLINE

Name	SP_CREATE_ORDER_ONLINE
Input	Address_delivery, user_id, contact, status, payment_method
Output	Orders (order_id, total_price, status, shipping_id, payment_id, employee_id, user_id, discount_id, delivery_address, contact)
Table	ORDERS

4.3.3 U01/SP_CREATE_ORDER_DETAIL

Name	SP_CREATE_ORDER_DETAIL
Input	Product_ids, order_id
Output	OrderDetails (order_id, product_id, quantity)
Table	OrderDetails

4.3.4 U01/SP_UPDATE_QUANTITY_PRODUCT

Name	SP_UPDATE_QUANTITY_PRODUCT
Input	Product_ids, quatity
Output	
Table	Products

4.3.5 U02/SP_GET_PRODUCT_BY_NAME

Name	SP_GET_PRODUCT_BY_NAME
Input	model_name
Output	Products (model_name, vote, price, unit_price, quantity, status, description, operation_system, hardware, warranty_time, category, discount, color, battery, screen, memory, front_camera, behind_camera, ram)

Table	Products
-------	----------

4.3.6 U02/SP_GET_PRODUCT_BY_FILTERS

Name	SP_GET_PRODUCT_BY_FILTERS
Input	ram, price, memory, category
Output	Products (model_name, vote, price, unit_price, quantity, status, description, operation_system, hardware, warranty_time, category, discount, color, battery, screen, memory, front_camera, behind_camera, ram)
Table	Products

4.3.7 U02/SP_GET_PRODUCT_BY_RAM (các tiêu chí brand, memory, price (BETWEEN) tương tự).

Name	SP_GET_PRODUCT_BY_NAME
Input	ram
Output	Products (model_name, vote, price, unit_price, quantity, status, description, operation_system, hardware, warranty_time, category, discount, color, battery, screen, memory, front_camera, behind_camera, ram)
Table	Products

4.3.8 U02/SP_GET_PRODUCT_BY_IDS

Name	SP_GET_PRODUCT_BY_IDS
Input	[product_id, ...]
Output	[Products (model_name, vote, price, unit_price, quantity, status, description, operation_system, hardware, warranty_time, category, discount, color, battery, screen, memory, front_camera, behind_camera, ram), ...]
Table	Products

4.3.9 U03/SP_GET_PRODUCT_INFOR

Name	SP_GET_PRODUCT_INFOR
Input	ProductsID

Output	Products (model_name, vote, price, unit_price, quantity, status, description, operation_system, hardware, warranty_time, category, discount, color, battery, screen, memory, front_camera, behind_camera, ram)
Table	Products

4.3.10 U04/ SP_GET_ORDER

Name	SP_GET_ORDER
Input	OrderID
Output	Orders (order_id, total_price, status, shipping_id, payment_id, employee_id, user_id, discount_id, delivery_address, contact)
Table	Orders

4.3.11 U05/ SP_CHECKLOGIN

Name	SP_CHECKLOGIN
Input	User_name, password
Output	User (First_name, last_name), Account (email, role, status)
Table	Account, User

4.3.12 U05/SP_GET_ORDERS_CONFIRMED

Name	SP_GET_ORDERS_CONFIRMED
Input	STATUS = 'confirmed'
Output	Orders (order_id, total_price, status, shipping_id, payment_id, employee_id, user_id, discount_id, delivery_address, contact)
Table	Order

4.3.13 U05/ SP_UPDATE_STATUS_ORDER_SHIPPER

Name	SP_UPDATE_STATUS_ORDER_SHIPPER
Input	Order_id
Table	Order

4.4 Bảng đối chiếu

Usecase	Form	Api	Table
<U00, name>	<form name>	<api name>	<table name>,..

		<api name>	<table name>,..
<U01, Thanh toán>	<Form nhập thông tin đơn hàng chưa thanh toán >	<API danh sách sản phẩm trong giỏ hàng>	<Carts>, <CartDetails>
	<Form xác thực thanh toán >	<API Xác thực thanh toán>	<Orders>, <OrderDetails>, <Payments>
		<API xác thực thông tin thanh toán>	<Orders>, <OrderDetails>, <Product>
		<API API tạo hóa đơn với Stripe>	<Orders>, <OrderDetails>, <Product>
<U02, Tìm kiếm sản phẩm theo chỉ nhất định>	<Form Nhập tên>	<API tìm danh sách sản phẩm theo tên>	<Products>
	<Form Chọn tiêu chí>	<API tìm danh sách sản phẩm theo bộ lọc>	<Products>
<U03, So sánh 2 sản phẩm>	<Form hiển thị thông tin so sánh>	<API danh sách sản phẩm tìm theo tên>	<Products>
	<Form Tìm kiếm sản phẩm để so sánh >	<API so sánh các sản phẩm>	<Products>
<U04, Xác nhận đơn hàng >	<Form thông tin chi tiết đơn hàng>	<API danh sách các đơn hàng chưa duyệt>	<Orders>,
	<Form chọn lọc các đơn hàng>	<API danh sách các đơn hàng theo bộ lọc>	<Orders>,
<U05, Giao Hàng>	<Form Chi tiết đơn hàng>	<API danh sách đơn hàng đã duyệt>	<Orders>,
	<Form Chi tiết đơn hàng>	<API Chi tiết đơn hàng >	<Orders>, <OrderDetails>,
	<Form Lý do giao thất bại >	<API cập nhật trạng thái đơn hàng>	<Orders>,

CHƯƠNG 5: TỔNG KẾT

5.1 Kết quả đạt được

Phần mềm đạt được các tiêu chuẩn chất lượng phần mềm sau:

- Tính đúng đắn: Thực hiện đầy đủ yêu cầu người dùng
- Tính hiệu quả: Sử dụng tối ưu tài nguyên máy tính
- Tính tiện dụng: Giao diện đơn giản, dễ dùng, các chức năng dễ dàng xử lý.

5.2 Mở rộng bổ sung cho Usecase So Sánh:

5.2.1 Ưu điểm

- Tạo góc nhìn khách quan cho khách hàng.
- Nâng cao được sự hiển thị sản phẩm thông minh hơn.
- Tránh khách hàng rơi vào tình huống băng khuâng.

5.2.2 Nhược điểm

- Rơi vào tình trạng so sánh 2 sản phẩm có độ chênh lệch cao.
- Dễ Rơi vào trạng thái so sánh khắp khểnh.

5.2.3 Hướng phát triển

- Áp dụng khi chọn sản phẩm so sánh. sử dụng thuật toán dựa trên độ tương đồng để giảm tính khắp khểnh khi so sánh 2 sản phẩm.
- Đề xuất các thuật toán (1):
 - o Cosine Similarity:
 - Đo lường góc cosine giữa hai vector đặc trưng của sản phẩm.
 - Giá trị cosine similarity nằm trong khoảng $[-1, 1]$, với 1 đại diện cho sự giống nhau hoàn toàn và -1 đại diện cho sự khác biệt hoàn toàn.
 - o Euclidean Distance:
 - Đo lường khoảng cách giữa hai vector đặc trưng.
 - Giá trị càng nhỏ thì sự tương đồng càng cao.
- Đề xuất hiển thị thêm các lượt bình luận đánh giá (2).

5.3 Đánh giá ưu điểm khuyết điểm

5.2.1 Ưu điểm

- Đảm bảo tính nhất quán trong cơ sở dữ liệu.
- Xác thực thông tin trong cơ sở dữ liệu người dùng đăng nhập và tăng khả năng bảo mật cho thông tin khách hàng.
- Kiểm tra tính chính xác dữ liệu người dùng nhập vào. Hiển thị thông báo chi tiết khi cần thiết.
- Giao diện đơn giản dễ dùng.
- Đơn giản dễ sử dụng, quản lý mang lại cảm giác thân thiện cho người dùng.
- Chức năng tra cứu nhanh, tiện dụng lấy thông tin nhanh chóng.

5.2.2 Nhược điểm

- Các chức năng chính của hệ thống còn tương đối cơ bản, chưa đa dạng.

- Chưa đảm bảo hoạt động ổn định của website khi số lượng truy cập đồng thời quá lớn.
- Hệ thống chưa được đưa vào sử dụng thực tế nên tính thực dụng chưa cao.

5.2.3 Hướng phát triển trong tương lai

- Xem xét cải thiện giao diện và hiệu năng.
- Khắc phục các nhược điểm tồn tại trong hệ thống.
- Hoàn thành chương trình một cách tối ưu nhất để có thể đem vào sử dụng trong thực tế.

DANH MỤC TÀI LIỆU THAM KHẢO

Tiếng Việt:

- [1]. ThS. Nguyễn Anh Hào, Bài giảng Phân tích và thiết kế Hệ thống thông tin, lưu hành nội bộ, Học viện Công nghệ Bưu chính Viễn thông cơ sở tại TP.HCM
- [2]. ThS. Lưu Nguyễn Kỳ Thư, Bài giảng Cơ sở dữ liệu, lưu hành nội bộ, Học viện Công nghệ Bưu chính Viễn thông cơ sở tại TP.HCM.
- [3]. TS. Võ Xuân Thê, Bài giảng Kiến trúc và thiết kế phần mềm, lưu hành nội bộ, Học viện Công nghệ Bưu chính Viễn thông cơ sở tại TP.HCM.
- [4]. TS. Trương Đức Nga, Bài giảng Thương mại điện tử, lưu hành nội bộ, Học viện Công nghệ Bưu chính Viễn thông cơ sở tại TP.HCM.

CHÚNG EM CẢM ƠN THẦY ĐÃ XEM VÀ ĐỌC BÀI BÁO CÁO CỦA CHÚNG EM