

[illegible]

## This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the page.

# PHIẾU DUYỆT ĐỒ ÁN TỐT NGHIỆP

## I. Phần dành cho Sinh viên

1. *Họ và tên:* Nguyễn Thị Thanh Thủy

2. *Mã Sinh viên:* 102180273

3. *Lớp:* 18TCLC-Nhật

4. *Tên đề tài:* XÂY DỰNG WEBSITE NHÀ HÀNG HEALTHY FOOD KẾT HỢP KINH DOANH THỰC PHẨM ORGANIC

5. *Số điện thoại:* 0355343271

6. *E-mail:* 99thanhthuy99@gmail.com

7. *Họ và tên GVHD:* TS. Ninh Khánh Duy

## II. Phần dành cho Hội đồng

STT	Nội dung đánh giá	Kết luận
1.	<i>Trình bày báo cáo theo đúng mẫu qui định của Khoa</i>	
2.	<i>Không có sự sao chép nội dung báo cáo và chương trình đã có</i>	
3.	<i>Biên dịch mã nguồn và chạy được chương trình</i>	
4.	<i>Có kịch bản thực hiện với dữ liệu thử nghiệm</i>	
5.	<i>Kết quả thực hiện chương trình đúng theo báo cáo</i>	
6.	<i>Có sự đóng góp, phát triển của tác giả trong đồ án</i>	

**Ý kiến khác:** .....

.....

**Kết luận:**

☐ Đạt yêu cầu

☐ Phải sửa chữa lại

☐ Không đạt yêu cầu

*Đà Nẵng, ngày    tháng    năm 2022*

**Chủ tịch Hội đồng**

*(Ký và ghi họ tên)*

**Cán bộ duyệt kiểm tra**

*(Ký và ghi họ tên)*



## TÓM TẮT

Tên đề tài: XÂY DỰNG WEBSITE NHÀ HÀNG HEALTHY FOOD KẾT HỢP KINH DOANH THỰC PHẨM ORGANIC

Sinh viên thực hiện: Nguyễn Thị Thanh Thủy

Số thẻ SV: 102180273

Lớp: 18TCLC-Nhật

Trong thời đại công nghệ như hiện nay, để giải đáp cho thông tin đi đâu, ăn gì thì việc truy cập mạng internet mang tới những thông tin hữu ích, cần thiết cho mỗi người, mỗi nhà. Có thể tìm kiếm giữa nhiều địa chỉ, đưa ra cho mình quyết định đúng đắn và hợp lý dựa trên nhu cầu, mong muốn thực tế của chính mình. Bởi thế, các nhà hàng khi chú trọng tới việc tiếp cận khách hàng thông qua website là quan trọng, cần thiết và mang tới hiệu quả cao.

Đứng trước xu thế chung, khi mà thiết kế website nhà hàng trở nên phổ biến ở nhiều, thậm chí là ở nhiều lĩnh vực khác thì tìm hiểu để hoàn thiện chất lượng, hiệu quả là vấn đề quan trọng cần chú ý thực hiện. Thông qua đó, hoàn thiện website đưa vào sử dụng phát huy được tối đa giá trị, mang tới nhiều lợi ích lớn. Với lượng khách hàng tìm hiểu thông tin về nhà hàng thông qua website ngày càng nhiều thì sử dụng trang web dành riêng cho nhà hàng mình là điều cần được chú ý.

Khi việc mua sắm, sử dụng các dịch vụ theo phương thức trực tuyến ngày càng nhiều thì sở hữu trang web riêng cho nhà hàng là yêu cầu cần được đảm bảo. Một website chất lượng, phù hợp hỗ trợ cho quá trình nâng cao hiệu quả kinh doanh tới mức tối đa. Tiếp cận lượng lớn khách hàng thông qua mạng internet thì việc hoàn thiện website đưa vào sử dụng là yêu cầu cơ bản, quan trọng mà mỗi nhà hàng cần chú ý thực hiện.

## NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Họ tên sinh viên: NGUYỄN THỊ THANH THỦY

Số thẻ sinh viên: 102180273      Lớp: 18TCLC-Nhật.

Khoa: Công nghệ thông tin      Ngành: Công nghệ Phần Mềm

1. *Tên đề tài đồ án:* XÂY DỰNG WEBSITE NHÀ HÀNG HEALTHY FOOD KẾT HỢP  
KINH DOANH THỰC PHẨM ORGANIC

2. *Đề tài thuộc diện:* ☐ Có ký kết thỏa thuận sở hữu trí tuệ đối với kết quả thực hiện

3. *Các số liệu và dữ liệu ban đầu:*

Không có.

4. *Nội dung các phần thuyết minh và tính toán:*

Nội dung của thuyết minh gồm:

**Mở đầu:** Phần mở đầu của luận văn, giới thiệu về nhu cầu thực tế và lý do thực hiện đề tài, đồng thời giới thiệu sơ lược về đề tài và mục tiêu phải đạt được.

**Chương 1.** Cơ sở lý thuyết: trình bày những lý thuyết học được và đã áp dụng vào hệ thống.

**Chương 2.** Phân tích và thiết kế: trình bày các hồ sơ phân tích và hồ sơ thiết kế trong xây dựng hệ thống và luồng hoạt động của hệ thống.

**Chương 3.** Triển khai và đánh giá kết quả: mô tả cách cài đặt, vận hành hệ thống và đánh giá kết quả đạt được.

5. *Các bản vẽ, đồ thị ( ghi rõ các loại và kích thước bản vẽ ):*

Không có.

6. *Họ tên người hướng dẫn:* TS. Ninh Khánh Duy

7. *Ngày giao nhiệm vụ đồ án:*      /      / 2022

8. *Ngày hoàn thành đồ án:*      /      / 2022

Đà Nẵng, ngày      tháng      năm 2022

Trưởng Bộ môn .....

Người hướng dẫn



## LỜI NÓI ĐẦU

Em xin chân thành gửi lời cảm ơn đến các thầy cô trong Khoa Công nghệ Thông tin, Trường Đại học Bách Khoa – Đại học Đà Nẵng đã truyền đạt những kiến thức quý báu trong những năm học vừa qua để em có kiến thức thực hiện đề tài tốt nghiệp này.

Đặc biệt, em xin chân thành cảm ơn thầy giáo hướng dẫn TS. Ninh Khánh Duy đã trực tiếp hướng dẫn, tận tình góp ý, giúp đỡ và tạo điều kiện thuận lợi nhất cho em trong suốt quá trình thực hiện đề tài.

Quá trình thực hiện và nội dung trình bày ở đây chắc hẳn sẽ không tránh khỏi những thiếu sót. Em rất mong nhận được sự thông cảm, góp ý và tận tình chỉ bảo của quý Thầy Cô và các bạn để đề tài được hoàn thiện hơn.

Một lần nữa em xin chân thành cảm ơn!



## **CAM ĐOAN**

Em xin cam đoan:

1. Nội dung trong đồ án này là do em thực hiện dưới sự hướng dẫn trực tiếp của giảng viên TS. Ninh Khánh Duy
2. Các tham khảo dùng trong đồ án đều được trích dẫn rõ ràng tên tác giả, tên công trình, thời gian, địa điểm công bố.

Nếu có những sao chép không hợp lệ, vi phạm, em xin chịu hoàn toàn trách nhiệm.

Sinh viên thực hiện

Nguyễn Thị Thanh Thủy

# MỤC LỤC

TÓM TẮT	
NHIỆM VỤ ĐỒ ÁN	
LỜI NÓI ĐẦU VÀ CẢM ƠN	I
LỜI CAM ĐOAN LIÊM CHÍNH HỌC THUẬT	II
MỤC LỤC	III
DANH SÁCH CÁC BẢNG BIỂU, HÌNH VẼ VÀ SƠ ĐỒ	V
DANH SÁCH CÁC CỤM TỪ VIẾT TẮT	VII

## MỞ ĐẦU 9

1. TỔNG QUAN VỀ ĐỀ TÀI.....	9
2. Ý NGHĨA .....	9
3. PHẠM VI THỰC HIỆN.....	10
4. BỐ CỤC ĐỒ ÁN.....	11

## CHƯƠNG 1. CƠ SỞ LÝ THUYẾT VÀ CÔNG CỤ.....12

1.1. MERN STACK .....	12
1.1.1. Ngôn ngữ lập trình Nodejs .....	12
1.1.2. Reactjs framework .....	16
1.1.3. MongoDB và MongoDB Atlas .....	16
1.1.4. Express Framework.....	17
1.2. TỔNG QUAN VỀ API RESTFUL .....	17
1.2.1. API .....	17
1.2.2. RESTful API .....	17
1.3. QUẢN LÝ PHIÊN BẢN GIT .....	18
1.4. KẾT CHƯƠNG .....	19

## CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ .....20

2.1. KHẢO SÁT BÀI TOÁN THỰC TẾ .....	20
2.2. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG WEBSITE.....	22
2.2.1. Phân tích chức năng: .....	22
2.2.2. Biểu đồ ca sử dụng.....	24
2.2.3. Sơ đồ hoạt động .....	34
2.2.4. Sơ đồ tuần tự .....	35
2.2.5. Kiến trúc hệ thống.....	38
2.2.6. Thiết kế cơ sở dữ liệu.....	38

2.3.	KẾT CHƯƠNG .....	40
<b>CHƯƠNG 3. TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ .....</b>		<b>41</b>
3.1.	MÔI TRƯỜNG VÀ CÔNG CỤ PHÁT TRIỂN.....	41
3.2.	KẾT QUẢ TRIỂN KHAI .....	41
3.3.	ĐÁNH GIÁ .....	41
3.4.	KẾT CHƯƠNG .....	41
<b>KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN .....</b>		<b>42</b>
1.	KẾT QUẢ ĐẠT ĐƯỢC .....	42
2.	NHỮNG VẤN ĐỀ HẠN CHẾ .....	42
3.	HƯỚNG PHÁT TRIỂN .....	42
<b>TÀI LIỆU THAM KHẢO.....</b>		<b>43</b>

## DANH SÁCH CÁC BẢNG, HÌNH VẼ

Bảng 2. 1 Danh sách và mô tả các tác nhân hệ thống .....	24
Bảng 2. 2 Ca sử dụng đăng nhập .....	28
Bảng 2. 3 Ca sử dụng admin xem chi tiết người dùng .....	29
Bảng 2. 4 Ca sử dụng admin thêm sản phẩm .....	30
Bảng 2. 5 Ca sử dụng Admin chỉnh sửa đơn hàng .....	31
Bảng 2. 6 Ca sử dụng sử đổi thông tin cá nhân .....	32
Bảng 2. 7 Ca sử dụng quản lý giỏ hàng.....	33
Bảng 2. 8 Ca sử dụng đặt hàng.....	34
Bảng 2. 9 Thông tin chi tiết bảng Role.....	<b>Error! Bookmark not defined.</b>
Bảng 2. 10 Thông tin chi tiết bảng Users .....	<b>Error! Bookmark not defined.</b>
Bảng 2. 11 Thông tin chi tiết bảng Account.....	<b>Error! Bookmark not defined.</b>
Bảng 2. 12 Thông tin chi tiết bảng Address .....	<b>Error! Bookmark not defined.</b>
Bảng 2. 13 Thông tin chi tiết bảng User_address.....	<b>Error! Bookmark not defined.</b>
Bảng 2. 14 Thông tin chi tiết bảng Storage .....	<b>Error! Bookmark not defined.</b>
Bảng 2. 15 Thông tin chi tiết bảng Entity .....	<b>Error! Bookmark not defined.</b>
Bảng 2. 16 Thông tin chi tiết bảng Category .....	<b>Error! Bookmark not defined.</b>
Bảng 2. 17 Thông tin chi tiết bảng Product.....	<b>Error! Bookmark not defined.</b>
Bảng 2. 18 Thông tin chi tiết bảng Question.....	<b>Error! Bookmark not defined.</b>
Bảng 2. 19 Thông tin chi tiết bảng Review .....	<b>Error! Bookmark not defined.</b>
Bảng 2. 20 Thông tin chi tiết bảng Order.....	<b>Error! Bookmark not defined.</b>
Bảng 2. 21 Thông tin chi tiết bảng Order_product...	<b>Error! Bookmark not defined.</b>
Bảng 2. 22 Thông tin chi tiết bảng Promotion .....	<b>Error! Bookmark not defined.</b>
Bảng 2. 23 Thông tin chi tiết bảng Cart .....	<b>Error! Bookmark not defined.</b>
Bảng 2. 24 Thông tin chi tiết bảng Cart_product .....	<b>Error! Bookmark not defined.</b>
.....	
Hình 1. 1 Mô hình REST APIs.....	<b>Error! Bookmark not defined.</b>

Hình 1. 2 Mối tương quan giữa ES5, ES6 và Typescript **Error! Bookmark not defined.**

Hình 1. 3 Angular 9 ..... **Error! Bookmark not defined.**

Hình 1. 4 Kiến trúc mức cao của Angular 9 ..... **Error! Bookmark not defined.**

Hình 1. 5 Các thành phần cơ bản Angular 9 ..... **Error! Bookmark not defined.**

Hình 1. 6 Quản lý phiên bản Github..... 19

Hình 2. 1 Mô hình kinh doanh trực tuyến ..... 20

Hình 2. 2 Các thành phần làm nên bài toán kinh doanh online..... 21

Hình 2. 3 Sơ đồ use-case tổng quát ..... 25

Hình 2. 4 Sơ đồ use-case chức năng quản lý tài khoản người dùng của Admin..... 26

Hình 2. 5 Sơ đồ use-case chức năng quản lý sản phẩm của Admin **Error! Bookmark not defined.**

Hình 2. 6 Sơ đồ use-case chức năng quản lý đơn hàng của Admin ..... 26

Hình 2. 7 Sơ đồ use-case chức năng quản lý tài khoản của người dùng..... 27

Hình 2. 8 Sơ đồ use-case chức năng quản lý giỏ hàng của người dùng..... 27

Hình 2. 9 Sơ đồ use-case chức năng đặt hàng của người dùng ..... 28

Hình 2. 10 Sơ đồ hoạt động việc đặt hàng và xác nhận đơn hàng ..... 35

Hình 2. 11 Sơ đồ tuần tự chức năng đặt hàng ..... 36

Hình 2. 12 Sơ đồ tuần tự chức năng xử lý đơn hàng..... 37

Hình 2. 13 Kiến trúc hệ thống ..... 38

Hình 2. 14 Biểu đồ lớp ..... **Error! Bookmark not defined.**

Hình 2. 15 Quan hệ cơ sở dữ liệu ..... 39

Hình 3. 1 Lưu trữ mã nguồn trên Github..... **Error! Bookmark not defined.**

Hình 3. 2 NodeJS..... **Error! Bookmark not defined.**

Hình 3. 3 Trình quản lý thư viện npm ..... **Error! Bookmark not defined.**

Hình 3. 4 Ngôn ngữ frontend sử dụng ..... **Error! Bookmark not defined.**

Hình 3. 5 Hệ quản trị cơ sở dữ liệu .....	<b>Error! Bookmark not defined.</b>
Hình 3. 6 Ngôn ngữ Backend sử dụng .....	<b>Error! Bookmark not defined.</b>
Hình 3. 7 Mô hình hệ thống.....	<b>Error! Bookmark not defined.</b>
Hình 3. 8 Màn hình Trang chủ - xem tất cả sản phẩm	<b>Error! Bookmark not defined.</b>
Hình 3. 9 Giao diện thông tin cá nhân.....	<b>Error! Bookmark not defined.</b>
Hình 3. 10 Giao diện trang giỏ hàng .....	<b>Error! Bookmark not defined.</b>
Hình 3. 11 Giao diện trang đặt hàng.....	<b>Error! Bookmark not defined.</b>
Hình 3. 12 Giao diện trang quản lý danh mục.....	<b>Error! Bookmark not defined.</b>
Hình 3. 13 Giao diện trang quản lý sản phẩm .....	<b>Error! Bookmark not defined.</b>
Hình 3. 14 Giao diện trang tạo hoặc chỉnh sửa thông tin sản phẩm	<b>Error! Bookmark not defined.</b>
Hình 3. 15 Giao diện chỉnh sửa thông tin thực thể sản phẩm trong kho .....	<b>Error! Bookmark not defined.</b>
Hình 3. 16 Giao diện danh sách đơn hàng.....	<b>Error! Bookmark not defined.</b>
Hình 3. 17 Giao diện trang quản lý đơn hàng .....	<b>Error! Bookmark not defined.</b>

## DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT

### CHỮ VIẾT TẮT:

STT	Viết Tắt	Ý nghĩa
1	CNTT	Công nghệ thông tin
2	OOP	Object-oriented programming
3	JDK	Java Development Kit
4	JVM	Java virtual machine
5	IO	Input/Output
6	MVC	Model-View-Controller
7	IoC	Inversion of Control
8	REST	REpresentational State Transfer
9	API	Application Programming Interface
10	JAR	Java Archive
11	IDE	Integrated Development Environment
12	CI	Continuous Integration
13	ES6	ECMAScript 2015
14	DI	Dependency Injecttion
15	DVCS	Distributed Version Control System
16	CSDL	Cơ sở dữ liệu
17	ERD	Entity Relationship Diagram
18	VCS	Version Control System

## MỞ ĐẦU

### 1. Tổng quan về đề tài

Trong xã hội hiện đại, công nghệ đang hiện diện mọi lúc mọi nơi trong toàn bộ khía cạnh của đời sống xã hội. Đặc biệt, CNTT đóng vai trò quan trọng không thể thiếu trong quá trình quản trị, điều hành các hoạt động sản xuất kinh doanh của mỗi doanh nghiệp. Sự phát triển và ứng dụng của Internet đã làm thay đổi mô hình và cách thức hoạt động kinh doanh của doanh nghiệp, việc chuyển dần các giao dịch truyền thống sang giao dịch điện tử đã ảnh hưởng đến vị trí, vai trò và cả nhu cầu của các bên hữu quan (khách hàng, nhà cung cấp, nhà đầu tư...) của doanh nghiệp.

Việc xây dựng website nhà hàng kết hợp kinh doanh thực phẩm là một trong những giải pháp giúp các nhà hàng, quán ăn thu hút được nhiều đối tượng khách hàng tạo ra nhiều nguồn thu nhập hơn mỗi ngày. Điều này sẽ giúp nhà hàng trở nên chuyên nghiệp hơn trong mắt các thực khách và cũng một phần nâng cao hơn giá trị thương hiệu trong lĩnh vực này.

### 2. Ý nghĩa

Thông qua website sử dụng dành riêng cho nhà hàng mang tới nhiều ưu điểm nổi bật, nhiều hỗ trợ hữu ích cho mỗi nhà hàng hoạt động kinh doanh ẩm thực trên thị trường. Trong đó, tiêu biểu phải kể tới như:

- Xây dựng kênh thông tin hữu ích, từ đó giúp việc tiếp cận khách hàng tiềm năng được thực hiện tốt. Khi nhu cầu sử dụng mạng internet ngày càng tăng cao thì lúc này tiếp cận của khách hàng được đảm bảo thực hiện tốt, đem tới hiệu quả cao, lý tưởng với lượng khách hàng tiềm năng đông đảo hơn.
- Khẳng định được sự chuyên nghiệp của chính nhà hàng mình là một lợi ích không thể bỏ qua. Cung cấp thông tin tới khách hàng trực quan, nhanh chóng và hiệu quả là điều được đảm bảo. Lúc đó việc nâng cao uy tín, sự chuyên nghiệp và chất lượng khi hoạt động trên thị trường đều dễ dàng đạt được.
- Mang tới cho khách hàng sự tin tưởng hơn, dễ dàng hiểu hơn về nhà hàng. Từ đó quyết định sử dụng dịch vụ, tới thưởng thức các món ăn được cung cấp dễ dàng đạt được. Cải thiện doanh thu đáng kể, mở rộng thị trường với lượng khách hàng đông đảo.
- Tạo thuận lợi cho việc cải thiện được sức cạnh tranh so với thị trường hiện nay.

Khi mà các nhà hàng hoạt động trên thị trường ngày càng đa dạng thì sức cạnh



tranh được đảm bảo giúp chúng ta có được vị trí lý tưởng, hoạt động kinh doanh hiệu quả hơn. Việc nâng cao nhận diện thương hiệu, khẳng định được sự uy tín và chuyên nghiệp được thực hiện tốt.

- Việc giao tiếp với khách hàng được hỗ trợ tốt và toàn diện. Lúc đó khách hàng tìm hiểu hiệu quả về những sản phẩm dịch vụ mà doanh nghiệp cung cấp, trong khi đó mỗi nhà hàng chủ động trong việc tiếp cận với khách hàng khi hoạt động trên thị trường.
- Từ những lợi ích thiết thực khi thiết kế website nhà hàng thì hoàn thiện trang web thích hợp là điều mà mỗi đơn vị cần chú ý thực hiện. Thông qua đó, sử dụng website hỗ trợ tốt hơn cho công việc kinh doanh là điều được đảm bảo.

### **3. Phạm vi thực hiện**

Đề tài thực hiện bao gồm hệ thống một website cho các đối tượng bao gồm khách (là người chỉ muốn xem sản phẩm, không thực hiện các hoạt động giao dịch, tương tác với nhà hàng), người dùng có đăng kí tài khoản, quản lý nhà hàng.

Ứng dụng web gồm có những chức năng chính sau đây:

- Đối với khách:
  - Xem sản phẩm;
  - Tìm kiếm, lọc sản phẩm;
  - Xem Blog.
- Đối với người dùng có đăng kí tài khoản:
  - Xem, đặt câu hỏi về sản phẩm;
  - Đặt bàn trực tuyến;
  - Quản lý tài khoản, địa điểm giao hàng, thanh toán;
  - Quản lý giỏ hàng, mua hàng, nhận xét, đánh giá sản phẩm.
- Đối với quản lý nhà hàng:
  - Quản lý danh mục sản phẩm;
  - Quản lý danh sách đặt bàn và Blog;
  - Quản lý thông tin khách hàng;
  - Quản lý các đơn hàng, dịch vụ giao hàng.

#### **4. Bố cục đề án**

Đề án được chia làm 3 chương chính:

Với chương thứ nhất đi vào tìm hiểu và nghiên cứu cơ sở lý thuyết để thực hiện áp dụng vào đề tài. Tìm hiểu công cụ và ngôn ngữ lập trình.

Trong chương thứ hai đi vào quá trình khảo sát hiện trạng thực tế, phân tích và thiết kế hệ thống thông tin, xác định các tác nhân chính của hệ thống, thiết kế và xây dựng cơ sở dữ liệu.

Tiếp nối chương thứ hai, chương thứ ba sẽ đi vào quá trình triển khai hệ thống, thực hiện xây dựng mã nguồn chương trình và chạy thử nghiệm.

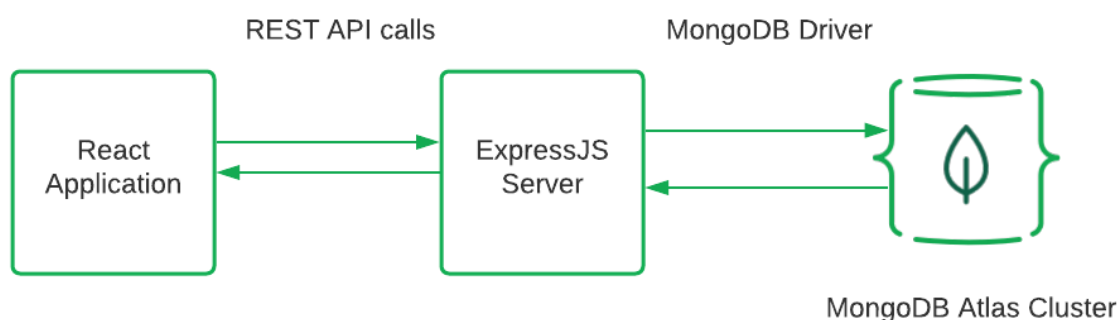
Đề tài được đánh giá lại qua phần kết luận, đưa ra các đánh giá kết quả về những vấn đề đã giải quyết, những vấn đề chưa giải quyết được và các giải pháp cho đề cải tiến, khắc phục những vấn đề đó.

## CHƯƠNG 1. CƠ SỞ LÝ THUYẾT VÀ CÔNG CỤ

### 1.1. MERN Stack

MERN là một thuật ngữ rút gọn của MongoDB, Express, React và Node; Stack MERN là một stack Javascript được thiết kế để giúp phát triển ứng dụng web toàn ngăn xếp dễ dàng hơn và nhanh hơn.

Tất cả bốn công nghệ này cung cấp một khuôn khổ hoàn chỉnh cho các nhà phát triển để tạo ra bất kỳ ứng dụng web nào. MERN đang tuân theo kiến trúc 3 tầng truyền thống, bao gồm tầng hiển thị front-end (React.js), tầng ứng dụng (Express.js và Node.js) và tầng cơ sở dữ liệu (MongoDB).

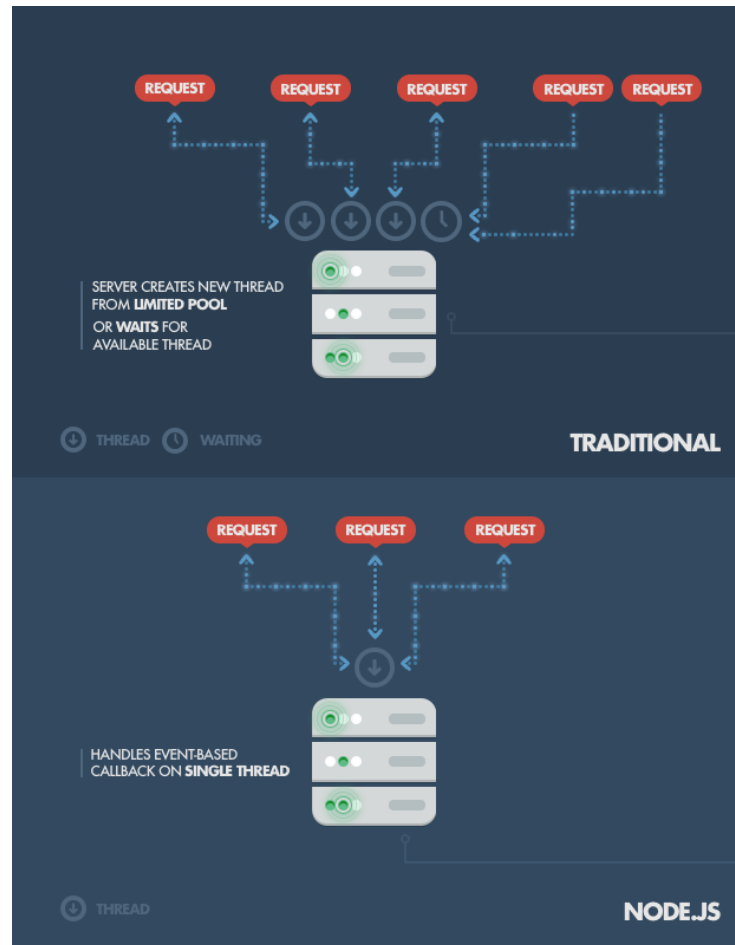


#### 1.1.1. Ngôn ngữ lập trình Nodejs

Node js được viết bằng ngôn ngữ javascript, nó là một trình biên dịch gói của Google's V8 JavaScript engine, libuv platform abstraction layer, và một thư viện lõi được viết bằng Javascript. Mục tiêu của Node js là làm cho web có khả năng push như trong một số ứng dụng gmail. Node js cung cấp công cụ giúp lập trình viên có thể làm việc trong non-blocking, mô hình I/O. Sau hơn 20 năm nghiên cứu, xây dựng và phát triển, nhóm kỹ sư đã cho ra đời sản phẩm ứng dụng web node js chạy thời gian thực và kết nối 2 chiều client và server, cho phép trao đổi dữ liệu một cách tự do.

Ý tưởng chính của Node js là sử dụng non-blocking, hướng sự vào ra dữ liệu thông qua các tác vụ thời gian thực một cách nhanh chóng. Bởi vì, Node js có khả năng mở rộng nhanh chóng, khả năng xử lý một số lượng lớn các kết nối đồng thời bằng thông lượng cao. Nếu như các ứng dụng web truyền thống, các request tạo ra một luồng xử lý yêu cầu mới và chiếm RAM của hệ thống thì việc tài nguyên của hệ thống sẽ được sử dụng không hiệu quả. Chính vì lẽ đó giải pháp mà Node js đưa ra là sử dụng luồng đơn

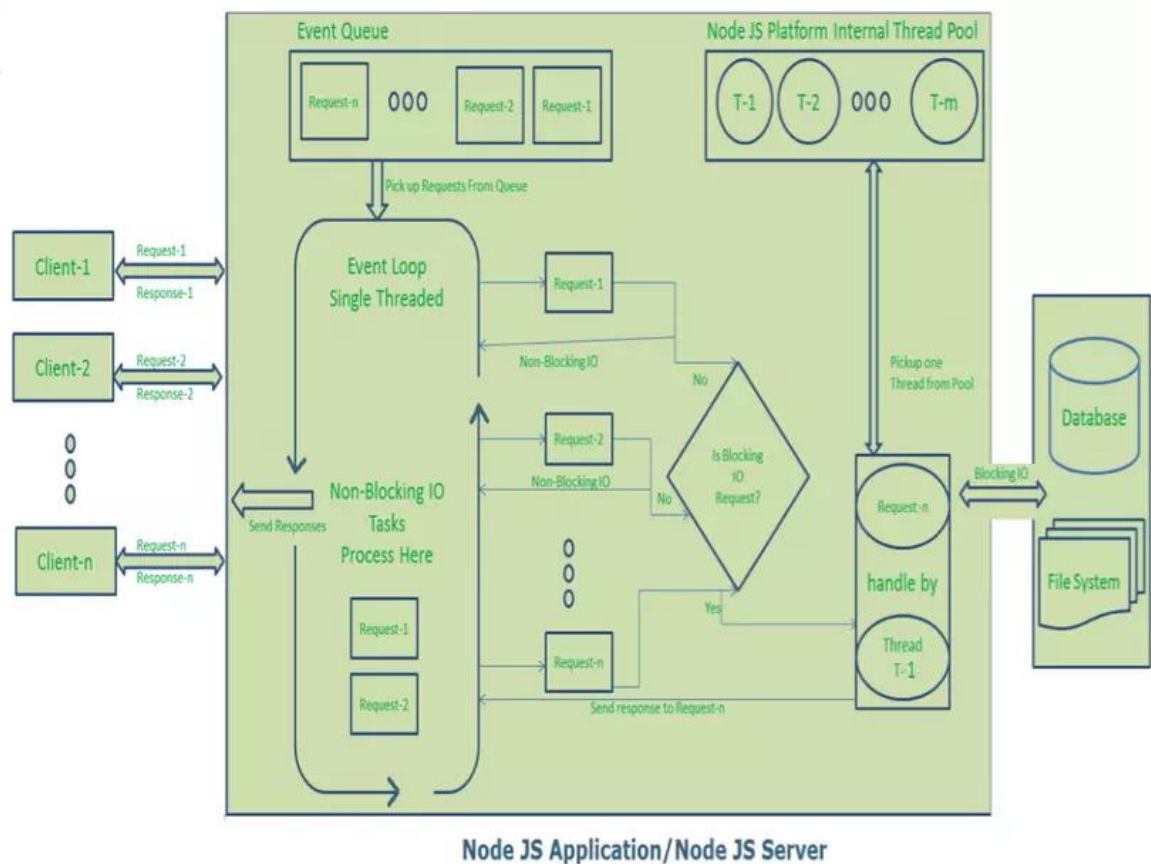
(Single-Threaded), kết hợp với non-blocking I/O để thực thi các request, cho phép hỗ trợ hàng chục ngàn kết nối đồng thời.



Hình 1: So sánh xử lý request truyền thống và request trên NodeJs

- Tính toán nhanh như sau: giả sử mỗi một luồng đính kèm 2MB dữ liệu được gửi lên server chạy trong hệ thống với 8GB RAM hệ thống, thì có tối đa khoảng 4000 kết nối đồng thời. Tuy nhiên với Node js thì nó có khả năng mở rộng ra cả hàng triệu kết nối cùng lúc. Thật vậy, cơ chế của Node js Application xử lý Model như sau:
- Client gửi request đến Web Server
- Node js Web Service duy trì trong nội bộ một luồng giới hạn để cung cấp dịch vụ cho Client Request.
- Node js Web Service nhận tất cả các request và đặt chúng vào một trong Queue. Nó được gọi là một Event Queue.
- Node js Web Service nội bộ có một thành phần được gọi là "Event Loop".
- Event Loop chỉ sử dụng một luồng đơn để xử lý Model.

- Event Loop kiểm tra tất cả các Request đặt trong Event Queue. Nếu không có request nào thì chờ request đến vô thời hạn
  - Nếu có request thì sẽ lấy một request từ Event Queue:
  - Khởi động quá trình xử lý tiến trình từ client request
  - Nếu Client Request không chứa nhiều Blocking I/O thì xử lý tất cả mọi thứ và chuẩn bị cho quá trình gửi lại phản hồi cho phía client.
  - Nếu Client Request chứa nhiều Blocking I/O như việc tương tác với cơ sở dữ liệu, tập tin hệ thống, dịch vụ mở rộng, thì nó sẽ thực hiện theo các phương pháp tiếp cận khác nhau.
1. Kiểm tra các luồng sẵn có từ nội bộ bên trong của request gửi lên
  2. Chọn một luồng và chỉ định cho client request tương ứng với luồng đó
  3. Luồng đó phải có trách nhiệm với request đó, xử lý nó, thực thi các hoạt động Blocking I/O, chuẩn bị các phản hồi và gửi lại cho Event Loop.
  4. Event Loop gửi lại phản hồi tương ứng cho client.



Hình 2: Sơ đồ xử lý request từ client của NodeJs

### **NPM: The Node Package Manager**

- Khi thảo luận về Node.js thì một điều chắc chắn không nên bỏ qua là xây dựng package quản lý sử dụng các công cụ NPM mà mặc định với mọi cài đặt Node.js. Ý tưởng của mô-đun NPM là khá tương tự như Ruby-Gems: một tập hợp các hàm có sẵn có thể sử dụng được, thành phần tái sử dụng, tập hợp các cài đặt dễ dàng thông qua kho lưu trữ trực tuyến với các phiên bản quản lý khác nhau.
- Danh sách các mô-đun có thể tìm trên web NPM package hoặc có thể truy cập bằng cách sử dụng công cụ NPM CLI sẽ tự động cài đặt với Node.js.

#### **Một số các module NPM phổ biến nhất hiện nay là:**

- [expressjs.com/](https://expressjs.com/) - Express.js, một Sinatra-inspired web framework khá phát triển của Node.js, chứa rất nhiều các ứng dụng chuẩn của Node.js ngày nay.
- connect - Connect là một mở rộng của HTTP server framework cho Node.js, cung cấp một bộ sưu tập của hiệu suất cao "plugins" được biết đến như là trung gian; phục vụ như một nền tảng cơ sở cho Express
- socket.io and sockjs - Hai thành phần Server-side websockets components nổi tiếng nhất hiện nay.
- Jade - Một trong những engines mẫu, lấy cảm hứng từ HAML, một phần mặc định trong Express.js.
- mongo and mongojs - MongoDB hàm bao để cung cấp các API cho cơ sở dữ liệu đối tượng trong MongoDB Node.js
- redis - thư viện Redis client.
- coffee-script - CoffeeScript trình biên dịch cho phép developers viết các chương trình Node.js của họ dùng Coffee.
- underscore (lodash, lazy) - Thư viện tiện ích phổ biến nhất trong JavaScript, package được sử dụng với Node.js, cũng như hai đối tác của mình, hứa hẹn hiệu suất tốt hơn bằng cách lấy một cách tiếp cận thực hiện hơi khác nhau.
- forever - Có lẽ là tiện ích phổ biến nhất để đảm bảo rằng một kịch bản nút cho chạy liên tục. Giữ quá trình Node.js của bạn lên trong sản xuất đối mặt với bất kỳ thất bại không ngờ tới.

### 1.1.2. *Reactjs framework*

React là thư viện JavaScript phổ biến nhất để xây dựng giao diện người dùng (UI). Nó cho tốc độ phản hồi tuyệt vời khi user nhập liệu bằng cách sử dụng phương pháp mới để render trang web.

Components của công cụ này được phát triển bởi Facebook. Nó được ra mắt như một công cụ JavaScript mã nguồn mở vào năm 2013. Hiện tại, nó đã đi trước các đối thủ chính như Angular và Bootstrap, hai thư viện JavaScript bán chạy nhất thời bấy giờ.

➤ Ưu điểm:

- Reactjs giúp việc viết các đoạn code JS dễ dàng hơn: Đây là đặc tính thú vị của Reactjs. Nó sẽ chuyển đổi các đoạn HTML thành các hàm khởi tạo đối tượng HTML bằng bộ biến đổi JSX.
- Nó có nhiều công cụ phát triển: Sau khi bạn cài đặt ứng dụng này, bạn sẽ có cái nhìn trực tiếp vào virtual DOM như thể bạn đang xem cây DOM thông thường.
- Render tầng server: Reactjs là một thư viện component, nó có thể vừa render ở ngoài trình duyệt sử dụng DOM và cũng có thể render bằng các chuỗi HTML mà server trả về.
- Làm việc với vấn đề test giao diện: Nó cực kì dễ để viết các test case giao diện vì virtual DOM được cài đặt hoàn toàn bằng JS.
- Hiệu năng cao đối với các ứng dụng có dữ liệu thay đổi liên tục, dễ dàng cho bảo trì và sửa lỗi.

### 1.1.3. *MongoDB và MongoDB Atlas*

#### ***MongoDB***

MongoDB là một database hướng tài liệu (document), một dạng NoSQL database. Vì thế, MongoDB sẽ tránh cấu trúc table-based của relational database để thích ứng với các tài liệu như JSON có một schema rất linh hoạt gọi là BSON. MongoDB sử dụng lưu trữ dữ liệu dưới dạng Document JSON nên mỗi một collection sẽ các các kích cỡ và các document khác nhau. Các dữ liệu được lưu trữ trong document kiểu JSON nên truy vấn sẽ rất nhanh. [5]



Hình 1.1 MongoDB

### ***MongoDB Atlas***

MongoDB Atlas là cloud database của MongoDB được ra mắt vào năm 2016 chạy trên AWS, Microsoft Azure và Google Cloud Platform. [6]

#### ***1.1.4. Express Framework***

Express là một framework giành cho nodejs. Nó cung cấp cho chúng ta rất nhiều tính năng mạnh mẽ trên nền tảng web cũng như trên các ứng dụng di động. Express hỗ trợ các phương thức HTTP và middleware tạo ra một API vô cùng mạnh mẽ và dễ sử dụng. Có thể tổng hợp một số chức năng chính của express như sau:

- Thiết lập các lớp trung gian để trả về các HTTP request
- Định nghĩa router cho phép sử dụng với các hành động khác nhau dựa trên phương thức HTTP và URL
- Cho phép trả về các trang HTML dựa vào các tham số.

## **1.2. Tổng quan về API RESTful**

### ***1.2.1. API***

API (application programming interface) là giao diện dùng để tương tác giữa ứng dụng này với ứng dụng khác. API có thể trả về dữ liệu mà bạn cần cho ứng dụng của mình ở những kiểu dữ liệu phổ biến như JSON hay XML. Facebook, google là hai ứng dụng khá quen thuộc. Nó có những API riêng để cho chúng ta sử dụng để nó có thể lấy được thông tin của người dùng.

### ***1.2.2. RESTful API***

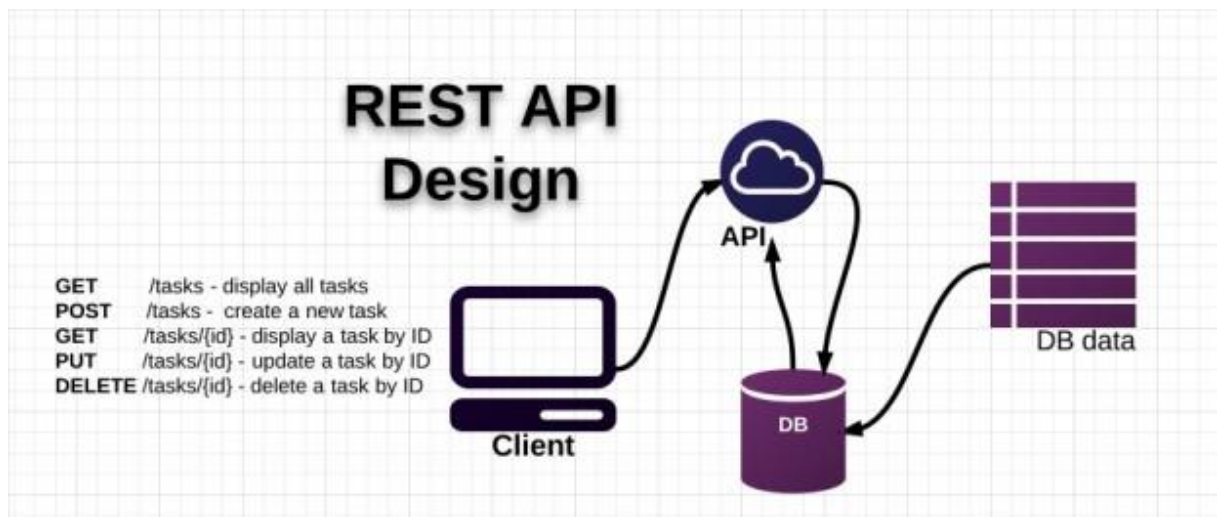
RESTful API là một tiêu chuẩn dùng trong việc thiết kế các API cho các ứng dụng web để quản lý các resource(tài nguyên). RESTful là một trong những kiểu thiết kế API được sử dụng phổ biến nhất ngày nay. Trọng tâm của REST quy định cách sử dụng các HTTP method (như GET, POST, PUT, DELETE...) và cách định dạng các URL cho ứng dụng web để quản lý các resource. RESTful không quy định logic code ứng dụng và không giới hạn bởi ngôn ngữ lập trình ứng dụng.



## RESTful hoạt động như thế nào?

Có 4 hoạt động chủ yếu ta có thể kể đến khi làm việc với server: lấy dữ liệu ở một định dạng nào đó (JSON), tạo dữ liệu mới, cập nhật dữ liệu, xóa dữ liệu. REST hoạt động chủ yếu dựa vào giao thức HTTP. Các hoạt động cơ bản nêu trên sẽ sử dụng những phương thức HTTP riêng.

- GET: lấy dữ liệu
- POST: tạo mới
- PUT: cập nhật (thay đổi)
- DELETE: Xóa dữ liệu

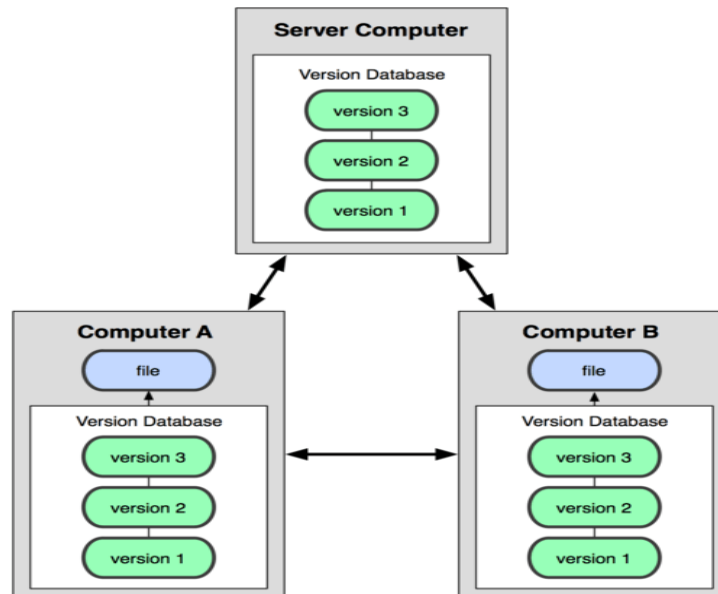


Những phương thức hay hoạt động này thường được gọi là CRUD tương ứng với Create, Read, Update, Delete – Tạo, Đọc, Sửa, Xóa. Mỗi phương thức trên phải được API gọi thông qua để gửi chỉ thị cho server phải làm gì.

### 1.3. Quản lý phiên bản Git

Git là tên gọi của một Hệ thống quản lý phiên bản phân tán (Distributed Version Control System – DVCS) là một trong những hệ thống quản lý phiên bản phân tán phổ biến nhất hiện nay. DVCS nghĩa là hệ thống giúp mỗi máy tính có thể lưu trữ nhiều phiên bản khác nhau của một mã nguồn được nhân bản (clone) từ một kho chứa mã nguồn (repository), mỗi thay đổi vào mã nguồn trên máy tính sẽ có thể ủy thác (commit) rồi đưa lên máy chủ nơi đặt kho chứa chính. Và một máy tính khác (nếu họ có quyền truy cập) cũng có thể clone lại mã nguồn từ kho chứa hoặc clone lại một tập hợp các

thay đổi mới nhất trên máy tính kia. Trong Git, thư mục làm việc trên máy tính gọi là Working Tree.



Hình 1. 2 Quản lý phiên bản Github

Ngoài ra, có một cách hiểu khác về Git đơn giản hơn đó là nó sẽ giúp bạn lưu lại các phiên bản của những lần thay đổi vào mã nguồn và có thể dễ dàng khôi phục lại dễ dàng mà không cần copy lại mã nguồn rồi cất vào đâu đó. Và một người khác có thể xem các thay đổi của bạn ở từng phiên bản, họ cũng có thể đối chiếu các thay đổi của bạn rồi gộp phiên bản của bạn vào phiên bản của họ. Cuối cùng là tất cả có thể đưa các thay đổi vào mã nguồn của mình lên một kho chứa mã nguồn. Cơ chế lưu trữ phiên bản của Git là nó sẽ tạo ra một “ảnh chụp” (snapshot) trên mỗi tập tin và thư mục sau khi commit, từ đó nó có thể cho phép bạn tái sử dụng lại một ảnh chụp nào đó mà bạn có thể hiểu đó là một phiên bản. Đây cũng chính là lợi thế của Git so với các DVCS khác khi nó không “lưu cứng” dữ liệu mà sẽ lưu với dạng snapshot [7].

Github là một dịch vụ máy chủ repository công cộng, mỗi người có thể tạo tài khoản trên đó để tạo ra các kho chứa của riêng mình để có thể làm việc.

#### 1.4. Kết chương

Trong chương này đã trình bày cơ sở lý thuyết xây dựng nên hệ thống: giới thiệu về ngôn ngữ lập trình NodeJs, một số ưu điểm mà NodeJs mang lại; giới thiệu về React framework, MongoDB, Express; trình bày kiến trúc REST APIs được sử dụng. Đồng thời cũng trình bày về công cụ quản lý source code của dự án.

## CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ

Chương này trình bày các phân tích về chức năng hệ thống, các sơ đồ mô tả các chức năng chính gồm sơ đồ use-case, sơ đồ tuần tự, sơ đồ hoạt động và các thiết kế tổng quát hệ thống, sơ đồ lớp, cơ sở dữ liệu.

### 2.1. KHẢO SÁT BÀI TOÁN THỰC TẾ

#### 2.1.1. Tổng quan bài toán

Ngày nay, việc áp dụng CNTT vào trong việc điều hành và quản lý doanh nghiệp, kinh doanh buôn bán đang trở nên phổ biến rộng rãi hơn. CNTT giúp cho việc tiếp cận thông tin, việc quản lý trở nên dễ dàng, thông minh, chính xác, tập trung và dễ mở rộng hơn nhiều. Trong quá trình tìm hiểu em thấy việc kinh doanh nhà hàng truyền thống gặp nhiều trở ngại mà internet có thể giải quyết vấn đề này như tiếp cận khách hàng từ xa từ nhiều phía, phát triển về mặt hình ảnh tốt hơn, dễ dàng tìm kiếm sản phẩm cần mua và có được thời gian giao hàng nhanh chóng.

#### 2.1.2. Điều kiện áp dụng bài toán

Nhà hàng thường có hai bên cụ thể là người mua và người bán. Người mua tập trung vào tính năng xem sản phẩm, thắc mắc về sản phẩm và mua hàng. Phía người bán là nhân viên hoặc quản lý nhà hàng phải có quyền quản lý người dùng, danh mục sản phẩm, sản phẩm, doanh thu,... Từ đó em đã xây dựng nên hệ thống giúp nhà hàng có thể sử dụng được.

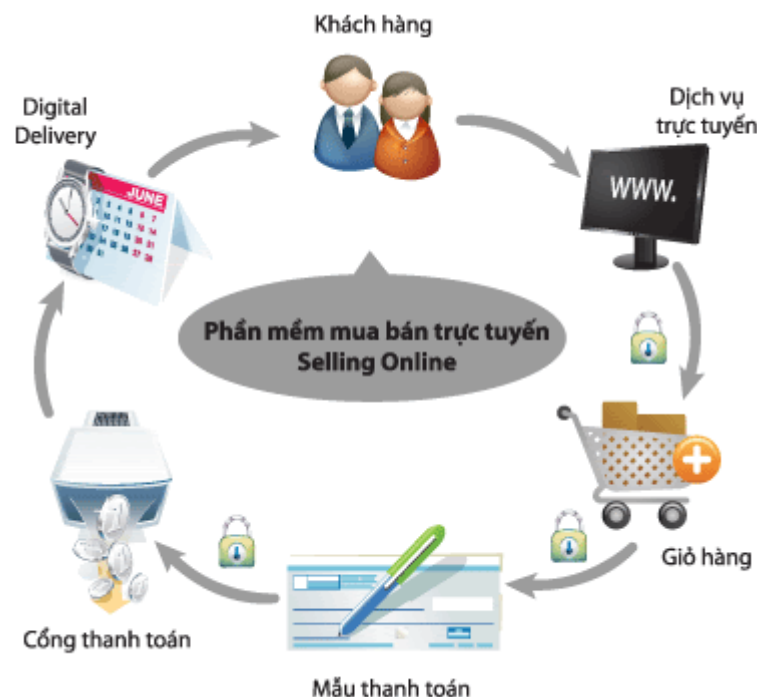


Hình 2. 1 Mô hình kinh doanh trực tuyến

#### 2.1.3. Phân tích yêu cầu

Để làm nên bài toán kinh doanh online thì sẽ có các thành phần chính dưới đây:

- Đặt bàn trực tiếp;
- Công cụ hỗ trợ tìm kiếm tối ưu, lọc sản phẩm theo các tiêu chí;
- Giỏ hàng;
- Blog văn hóa ẩm thực của nhà hàng;
- Tính năng thanh toán;
- Mẫu và phương thức thanh toán;
- Giao nhận hàng;
- Hỗ trợ đa ngôn ngữ.



Hình 2. 2 Các thành phần làm nên bài toán kinh doanh online

❖ **Dịch vụ trực tuyến:**

Dịch vụ trực tuyến là một website có thiết kế nhận diện riêng của người bán hàng, cung cấp thông tin, giới thiệu sản phẩm của người bán. Người dùng có thể tìm hiểu về sản phẩm, nhà cung cấp (hoặc nhà phát hành), so sánh giá của nhà trang bán hàng khác nhau để đưa ra quyết định mua hàng hay không.

❖ **Giỏ hàng:**

Giỏ hàng có thể hiểu như xe đẩy khi đi siêu thị, như tiện dụng hơn nhiều. Khách hàng chỉ cần chọn sản phẩm muốn mua và thêm vào giỏ hàng, tất cả sản phẩm được chọn trong giỏ hàng sẽ được thanh toán với vài cú nhấp chuột. Ngoài ra giỏ hàng trực tuyến luôn tồn tại với khách hàng đăng kí tài khoản, họ hoàn toàn có thể lưu lại sản phẩm yêu thích đến khi có khả năng thanh toán cho chúng mà không mất thời gian để tìm kiếm lại từng sản phẩm.

❖ **Mẫu thanh toán:**

Mẫu thanh toán là cam kết giữa người mua và người bán, hay còn gọi hóa đơn thanh toán bao gồm thông tin sản phẩm, số lượng mua, thành tiền trên từng loại sản phẩm và tổng giá trị đơn hàng; thời gian đặt hàng và thời gian giao hàng dự kiến và địa điểm giao nhận hàng; phương thức thanh toán cho đơn hàng; thông tin người mua và người bán.

❖ **Phương thức thanh toán:**

Các phương thức hiện đang phổ biến với hình thức kinh doanh online hiện nay gồm thanh toán trực tiếp khi nhận hàng, thanh toán điện tử bằng thẻ, thanh toán qua ví điện tử, thanh toán bằng điện thoại thông minh, sử dụng cổng thanh toán điện tử,.. Với các hình thức thanh toán điện tử khách hàng có được sự tiện lợi tuy nhiên tâm lý lo ngại với độ tin cậy về dịch vụ của bên thứ ba có thể khiến ít người muốn bắt đầu với hình thức thanh toán này. Hình thức thanh toán khi nhận hàng đảm bảo việc mặt hàng đến tận tay khách hàng và được kiểm tra sản phẩm trước khi quyết định thanh toán. Điều này phần nào giúp khách hàng yên tâm hơn và dễ dàng áp dụng với nhà bán hàng mới tạo dựng thương hiệu hay nhà bán lẻ quy mô nhỏ.

❖ **Giao hàng:**

Hình thức giao hàng tận nơi hiện đang rất phổ biến vì sự tiện lợi, và thời gian giao hàng ngày càng được rút ngắn. Với những nhà bán hàng cung cấp dịch vụ giao hàng tận nơi chi phí giao hàng đã được tốt ưu với mức giá hợp lý khách hàng có thể bỏ ra để tận hưởng dịch vụ.

## **2.2. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG WEBSITE**

### **2.2.1. Phân tích chức năng:**

Hệ thống phục vụ cho 3 loại người dùng: Quản trị hệ thống, Người dùng và Khách.

➤ **Quản trị hệ thống thường có chức năng sau:**

- ***Quản lý danh mục:***

Chức năng cho phép admin quản lý các danh mục sản phẩm của nhà hàng.

- ***Quản lý người dùng:***

Chức năng cho phép admin quản lý danh sách các người dùng: xem thông tin, thêm mới, sửa, xóa, khóa.

- ***Quản lý sản phẩm:***

Chức năng cho phép admin quản lý danh sách các sản phẩm: xem thông tin, thêm mới, sửa, xóa.

- ***Quản lý đơn hàng:***

Chức năng cho phép admin quản lý danh sách các đơn hàng: xem, sửa, xác nhận hoặc hủy bỏ.

- ***Quản lý khuyến mãi:***

Chức năng cho phép admin quản lý các mã khuyến mãi: xem, sửa, hoặc hủy bỏ.

- ***Quản lý câu hỏi:***

Chức năng cho phép admin quản lý các câu hỏi về sản phẩm: xem, trả lời, hoặc xóa bỏ.

➤ **Người dùng có chức năng sau:**

- ***Quản lý thông tin cá nhân:***

Chức năng cho phép người dùng quản lý thông tin cá nhân: thông tin cơ bản, ảnh đại diện, danh sách địa chỉ, đổi mật khẩu.

- ***Quản lý giỏ hàng:***

Chức năng giúp cho người dùng quản lý các sản phẩm yêu thích được thêm vào giỏ hàng: xem, thêm, xóa bỏ sản phẩm

- ***Xem lịch sử mua hàng:***

Chức năng giúp cho người dùng xem lịch sử mua hàng: chi tiết đơn hàng, trạng thái đơn hàng.

- ***Đặt hàng:***

Chức năng cho phép người dùng đặt hàng các sản phẩm, áp dụng mã giảm giá, xem thông tin đơn hàng, ngày giao hàng dự kiến, tổng giá trị đơn hàng.

- ***Chức năng đánh giá sản phẩm đã mua:***

Chức năng cho phép người dùng đánh giá, nhận xét các sản phẩm đã mua.

- **Chức năng đặt câu hỏi về sản phẩm:**

Chức năng cho phép người dùng đặt câu hỏi nếu có bất kì thắc mắc nào về sản phẩm.

➤ **Khách có các chức năng sau:**

- **Xem thông tin các sản phẩm:**

Chức năng cho phép người dùng xem toàn bộ sản phẩm, thông tin chi tiết từng sản phẩm, xem nhận xét đánh giá, tham khảo giải đáp câu hỏi từ nhà hàng.

- **Tìm kiếm thông tin của các sản phẩm:**

Chức năng cho phép người dùng tìm kiếm, lọc sản phẩm theo tên, nhà cung cấp và giá bán

- **Đăng kí tài khoản:**

Là chức năng dành cho người dùng thật sự có nhu cầu mua hàng, đăng kí tài khoản với hệ thống.

### 2.2.2. Biểu đồ ca sử dụng

Các trường hợp sử dụng chính trong hệ thống được mô hình hóa thông qua các biểu đồ ca sử dụng bên dưới, qua đó cũng làm rõ yêu cầu và kiến trúc của các chức năng chính của hệ thống.

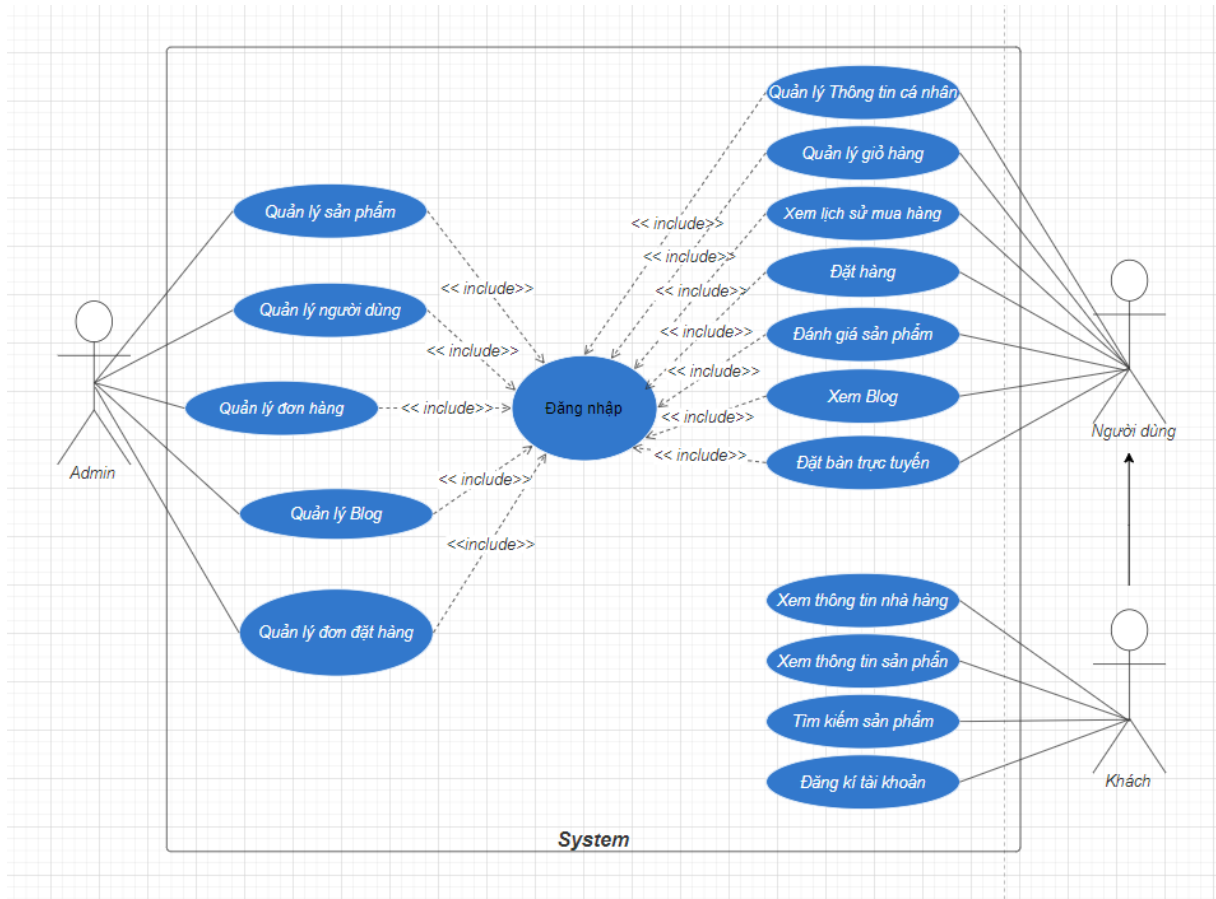
➤ **Các tác nhân**

Ứng dụng có 3 tác nhân gồm Admin quản lý hệ thống, Người dùng có đăng kí tài khoản, Khách chưa đăng kí tài khoản.

*Bảng 2. 1 Danh sách và mô tả các tác nhân hệ thống*

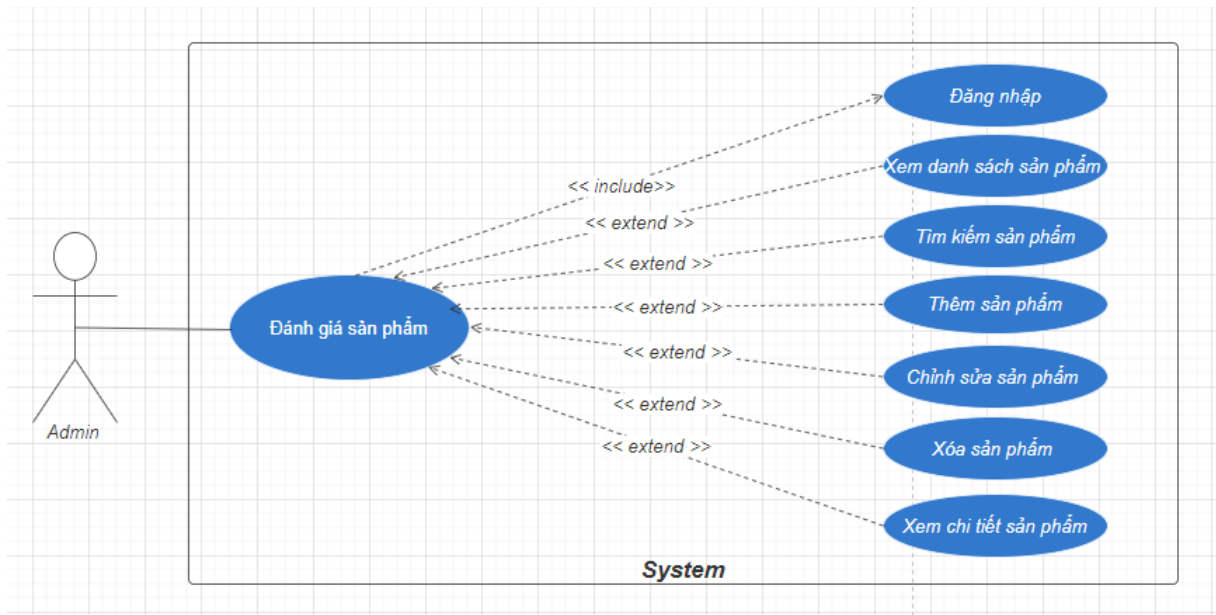
Tác nhân	Mô tả
Admin	Quản lý người dùng, các yếu tố liên quan đến sản phẩm như danh mục, khuyến mãi, nhận xét, quản lý đơn hàng
Người dùng	Quản lý tài khoản cá nhân, giỏ hàng, thực hiện mua hàng
Khách	Tham quan website, xem, tìm kiếm thông tin sản phẩm, đăng kí tài khoản

➤ **Biểu đồ ca sử dụng**

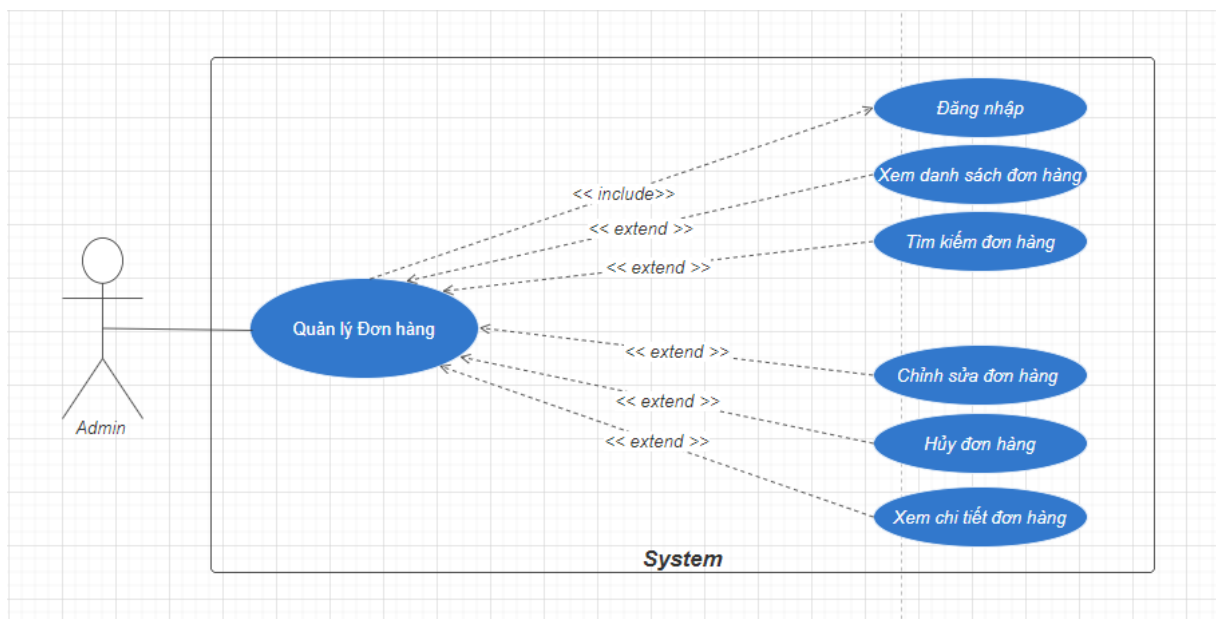


Hình 2. 3 Sơ đồ use-case tổng quát

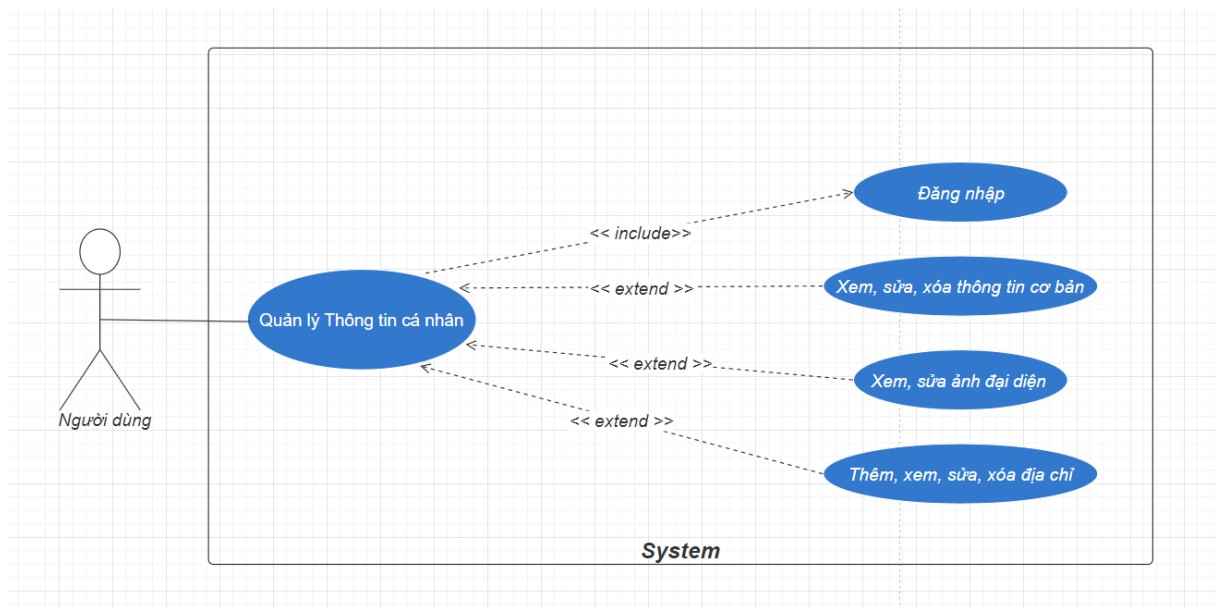




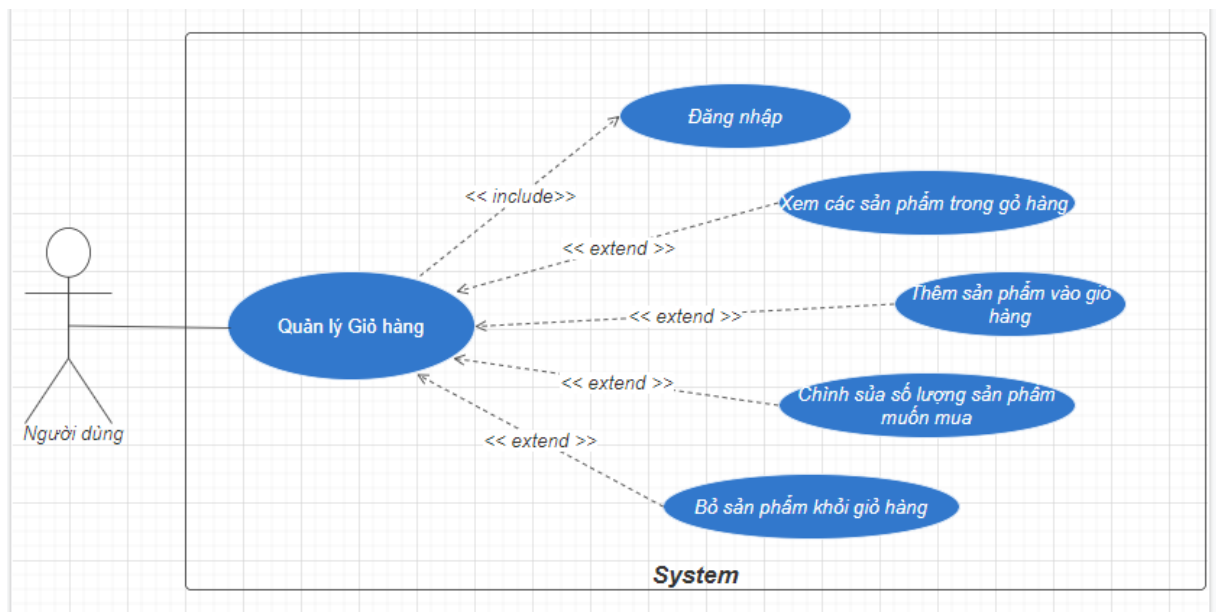
Hình 2. 4 Sơ đồ use-case chức năng quản lý sản phẩm của Admin



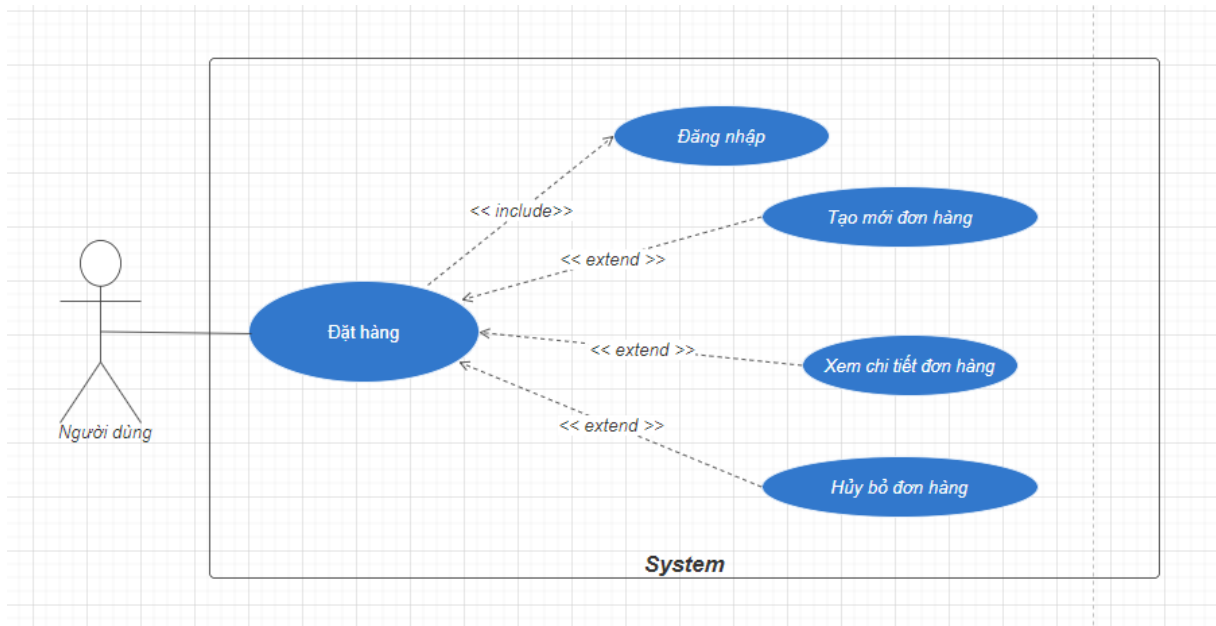
Hình 2. 5 Sơ đồ use-case chức năng quản lý đơn hàng của Admin



Hình 2. 6 Sơ đồ use-case chức năng quản lý tài khoản của người dùng



Hình 2. 7 Sơ đồ use-case chức năng quản lý giỏ hàng của người dùng



Hình 2. 8 Sơ đồ use-case chức năng đặt hàng của người dùng

### ➤ Đặc tả ca sử dụng

Đặc tả ca sử dụng là quá trình mô tả cách các tác nhân tương tác, thực hiện ca sử dụng cho phép người dùng hiểu rõ hơn về cách sử dụng chức năng đó.

Bảng 2. 2 Ca sử dụng đăng nhập

<b>Mã ca sử dụng</b>	<b>CA.1</b>	
<b>Tên ca sử dụng</b>	<b>Đăng nhập thủ công</b>	
<b>Tác nhân</b>	User, Admin.	
<b>Mô tả</b>	Cho phép Admin, User đăng nhập vào website.	
<b>Điều kiện trước</b>	Người dùng nhấn hiện trang đăng nhập ở trên thanh header hoặc nhấn vào các đường dẫn yêu cầu đăng nhập.	
<b>Điều kiện sau</b>	Người dùng đăng nhập vào hệ thống thành công Hệ thống ghi nhận trạng thái đăng nhập thành công, đưa đến trang chủ	
<b>Các bước thực hiện</b>	<b>Tác nhân</b>	<b>Hệ thống</b>
	Người dùng nhập các thông tin email và mật khẩu.	Hệ thống kiểm tra dữ liệu nhập là đúng, đưa đến trang chủ.
<b>Trường hợp lỗi</b>	1. Người dùng không nhập các thông tin email và mật khẩu.	

	2. Hệ thống kiểm tra dữ liệu nhập email và mật khẩu sai. 3. Thông báo lỗi ra màn hình.
--	---

*Bảng 2. 3 Ca sử dụng admin xem chi tiết người dùng*

<b>Mã ca sử dụng</b>	<b>CA.2</b>	
<b>Tên ca sử dụng</b>	<b>Xem chi tiết người dùng</b>	
<b>Tác nhân</b>	Admin.	
<b>Mô tả</b>	Cho phép Admin xem thông tin chi tiết người dùng.	
<b>Điều kiện trước</b>	Admin đã đăng nhập thành công vào hệ thống, muốn xem thông tin người dùng	
<b>Điều kiện sau</b>	Hiển thị trang thông tin người dùng	
<b>Các bước thực hiện</b>	<b>Tác nhân</b>	<b>Hệ thống</b>
	1. Admin chọn quản lý người dùng. 2. Admin chọn người dùng cụ thể	1. Hệ thống trả về danh sách người dùng, hiển thị thành danh sách 2. Hệ thống trả về thông tin chi tiết người dùng, hiển thị chi tiết người dùng
<b>Trường hợp lỗi</b>	1. Phiên đăng nhập của Admin hết hạn 2. Admin xem người dùng đã bị xóa, hoặc không tồn tại	

*Bảng 2. 4 Ca sử dụng admin thêm sản phẩm*

<b>Mã ca sử dụng</b>	<b>CA.3</b>	
<b>Tên ca sử dụng</b>	<b>Admin thêm sản phẩm</b>	
<b>Tác nhân</b>	Admin.	
<b>Mô tả</b>	Cho phép Admin thêm thông tin sản phẩm.	
<b>Điều kiện trước</b>	Admin đã đăng nhập, muốn thêm sản phẩm	
<b>Điều kiện sau</b>	Lưu sản phẩm thành công, cập nhật trang danh sách sản phẩm	
<b>Các bước thực hiện</b>	<b>Tác nhân</b>	<b>Hệ thống</b>
	1. Admin chọn quản lý sản phẩm 2. Admin chọn thêm sản phẩm 3. Admin nhập thông tin vào form, nhấn Save	1. Hệ thống trả về giao diện danh sách sản phẩm 2. Hệ thống chuyển đến trang tạo mới sách 3. Hệ thống kiểm tra thông tin, lưu và cập nhật trang danh sách sản phẩm
<b>Trường hợp lỗi</b>	1. Hệ thống kiểm tra các thông tin bắt buộc thiếu hoặc không hợp lệ. 2. Phiên đăng nhập của Admin đã hết hạn	

*Bảng 2. 5 Ca sử dụng Admin chỉnh sửa đơn hàng*

<b>Mã ca sử dụng</b>	<b>CA.4</b>	
<b>Tên ca sử dụng</b>	<b>Admin chỉnh sửa đơn hàng</b>	
<b>Tác nhân</b>	Admin.	
<b>Mô tả</b>	Cho phép Admin sửa đơn hàng.	
<b>Điều kiện trước</b>	Khách tạo mới đơn hàng thành công, Admin đã đăng nhập thành công và muốn quản lý đơn hàng	
<b>Điều kiện sau</b>	Thông báo cập nhật thành và cập nhật danh sách đặt hàng	
<b>Các bước thực hiện</b>	<b>Tác nhân</b>	<b>Hệ thống</b>
	1. Admin chọn quản lý đơn hàng 2. Admin chọn sửa ở 1 đơn hàng cụ thể. 3. Admin nhập, sửa các thông tin cần thiết, bấm Save	1. Hệ thống hiển thị danh sách đơn hàng 2. Chuyển sang trang chỉnh sửa đơn hàng. 3. Hệ thống kiểm tra dữ liệu hợp lệ, lưu, cập nhật danh sách đơn hàng
<b>Trường hợp lỗi</b>	1. Hệ thống kiểm tra các thông tin bắt buộc, khả năng đáp ứng đơn hàng không đảm bảo. 2. Thông báo yêu cầu admin nhập các thông tin bắt buộc, hoặc đơn hàng không thành công.	

*Bảng 2. 6 Ca sử dụng sử đổi thông tin cá nhân*

<b>Mã ca sử dụng</b>	<b>CA.5</b>	
<b>Tên ca sử dụng</b>	<b>Sửa đổi thông tin cá nhân</b>	
<b>Tác nhân</b>	Admin, Người dùng.	
<b>Mô tả</b>	Cho phép Admin, Người dùng chỉnh sửa các thông tin cá nhân.	
<b>Điều kiện trước</b>	Admin, Người dùng đã đăng nhập thành công, muốn chỉnh sửa thông tin cá nhân	
<b>Điều kiện sau</b>	Thông báo lưu thông tin thành công và cập nhật lại trang thông tin cá nhân	
<b>Các bước thực hiện</b>	<b>Tác nhân</b>	<b>Hệ thống</b>
	1. Admin, Người dùng chọn quản lý thông tin cá nhân 2. Người dùng thay đổi bao gồm quản lý thông tin cơ bản, đổi mật khẩu và bấm Lưu.	1. Hệ thống trả về thông tin cá nhân, hiển thị trên giao diện. 2. Trả lại thông báo thành công và cập nhật trang quản lý thông tin cá nhân.
<b>Trường hợp lỗi</b>	1. Lưu các thông tin thay đổi không thành công. 2. Hệ thống kiểm tra dữ liệu sai. 3. Thông báo lỗi ra màn hình.	

*Bảng 2. 7 Ca sử dụng quản lý giỏ hàng*

<b>Mã ca sử dụng</b>	<b>CA.6</b>	
<b>Tên ca sử dụng</b>	<b>Xóa sản phẩm khỏi giỏ hàng</b>	
<b>Tác nhân</b>	User	
<b>Mô tả</b>	Cho phép User bỏ sản phẩm khỏi giỏ hàng.	
<b>Điều kiện trước</b>	Khách hàng đã đăng nhập thành công và có từ 1 sản phẩm trong giỏ hàng	
<b>Điều kiện sau</b>	Sản phẩm được bỏ khỏi giỏ hàng, cập nhật danh sách giỏ hàng	
<b>Các bước thực hiện</b>	<b>Tác nhân</b>	<b>Hệ thống</b>
	1. User chọn biểu tượng giỏ hàng 2. User chọn Xóa ở 1 sản phẩm cụ thể.	1. Hệ thống trả về danh sách giỏ hàng. 2. Xóa sản phẩm khỏi giỏ hàng, cập nhật danh sách sản phẩm trong giỏ hàng.
<b>Trường hợp lỗi</b>	1. Người dùng chọn sản phẩm đã hết hàng. 2. Thông báo đã hết hàng, trở về trang chi tiết sản phẩm.	

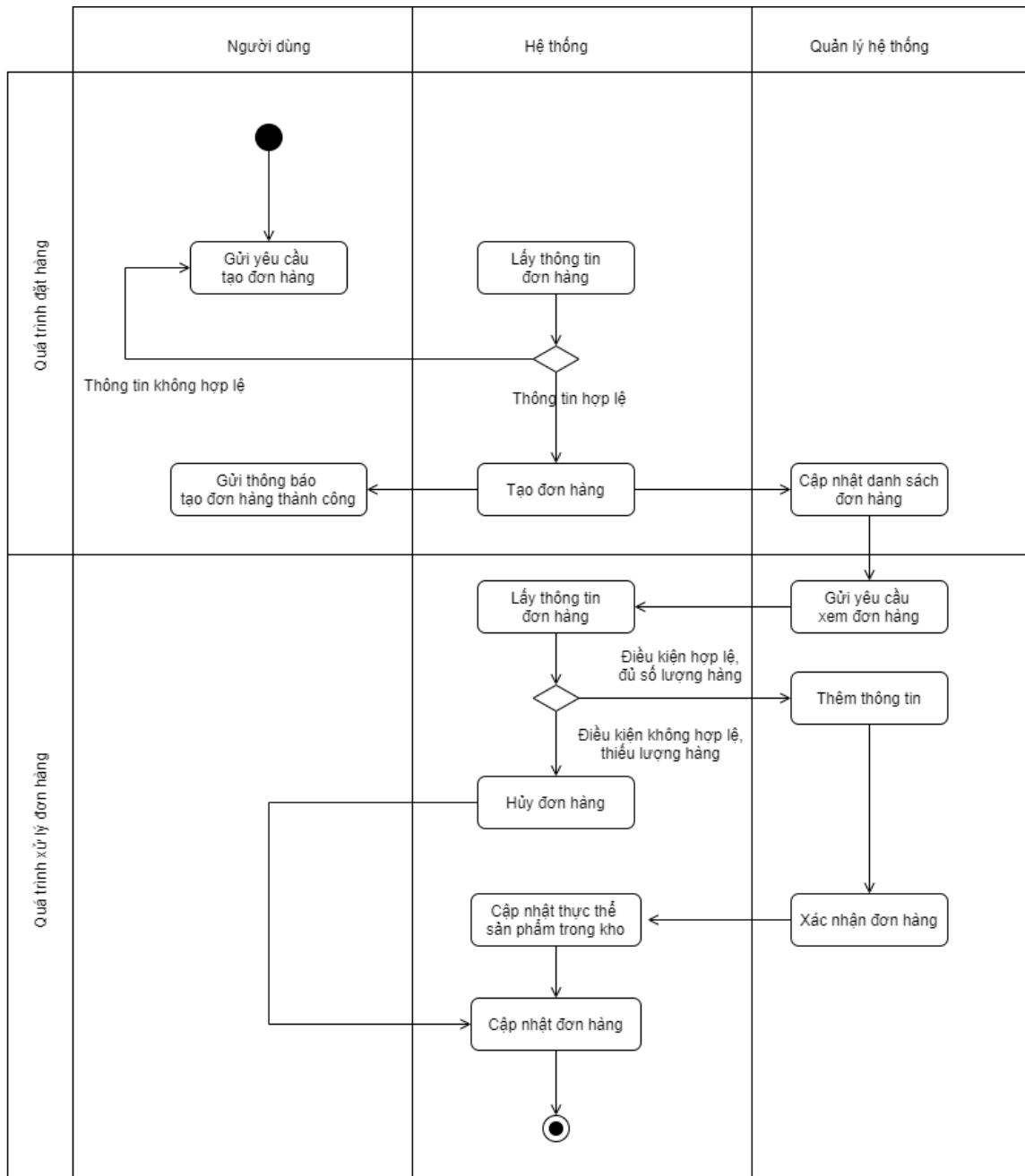


*Bảng 2. 8 Ca sử dụng đặt hàng*

<b>Mã ca sử dụng</b>	<b>CA.7</b>	
<b>Tên ca sử dụng</b>	<b>Đặt hàng</b>	
<b>Người dùng</b>	Người dùng	
<b>Mô tả</b>	Cho phép Người dùng đặt hàng giao đến địa chỉ cụ thể.	
<b>Điều kiện trước</b>	Người dùng nhấn chọn các sản phẩm từ giỏ hàng rồi nhấn “Thanh toán” hoặc chọn sản phẩm và nhấn “Mua ngay”, hệ thống chuyển đến trang thanh toán	
<b>Điều kiện sau</b>	Hệ thống thông báo và gửi về chi tiết đơn hàng	
<b>Các bước thực hiện</b>	<b>Tác nhân</b>	<b>Hệ thống</b>
	1. Người dùng chọn địa điểm giao hàng, số điện thoại và các thông tin đơn hàng khác, nhấn “Thanh toán đơn hàng”	1. Hệ thống kiểm tra thông tin hệ lệ, trả về thông báo hiển thị lại chi tiết đơn hàng.
<b>Trường hợp lỗi</b>	1. Người dùng không nhập thông tin cho đơn hàng. 2. Người dùng nhập sai định dạng các thông tin bắt buộc. 3. Thông báo lỗi ra màn hình.	

### 2.2.3. Sơ đồ hoạt động

Sơ đồ thể hiện việc đặt hàng của người dùng và quá trình xử lý đơn hàng của hệ thống. Khi đơn hàng được tạo bởi người dùng, đơn các mới sẽ được cập nhật trong danh sách đơn hàng với trạng thái chờ xử lý. Sau đó quản trị viên sẽ kiểm tra lại từng đơn hàng xem lại điều kiện có hợp lệ hay không (số lượng hàng đủ cho đơn hàng không,..) và quyết định hủy hoặc xác nhận đơn hàng. Nếu đơn hàng được xác nhận hệ thống sẽ cập nhật lại số thực thể sản phẩm trong kho. Cả hai trường hợp đều cập nhật lại trạng thái đơn hàng và thông báo cho người dùng.



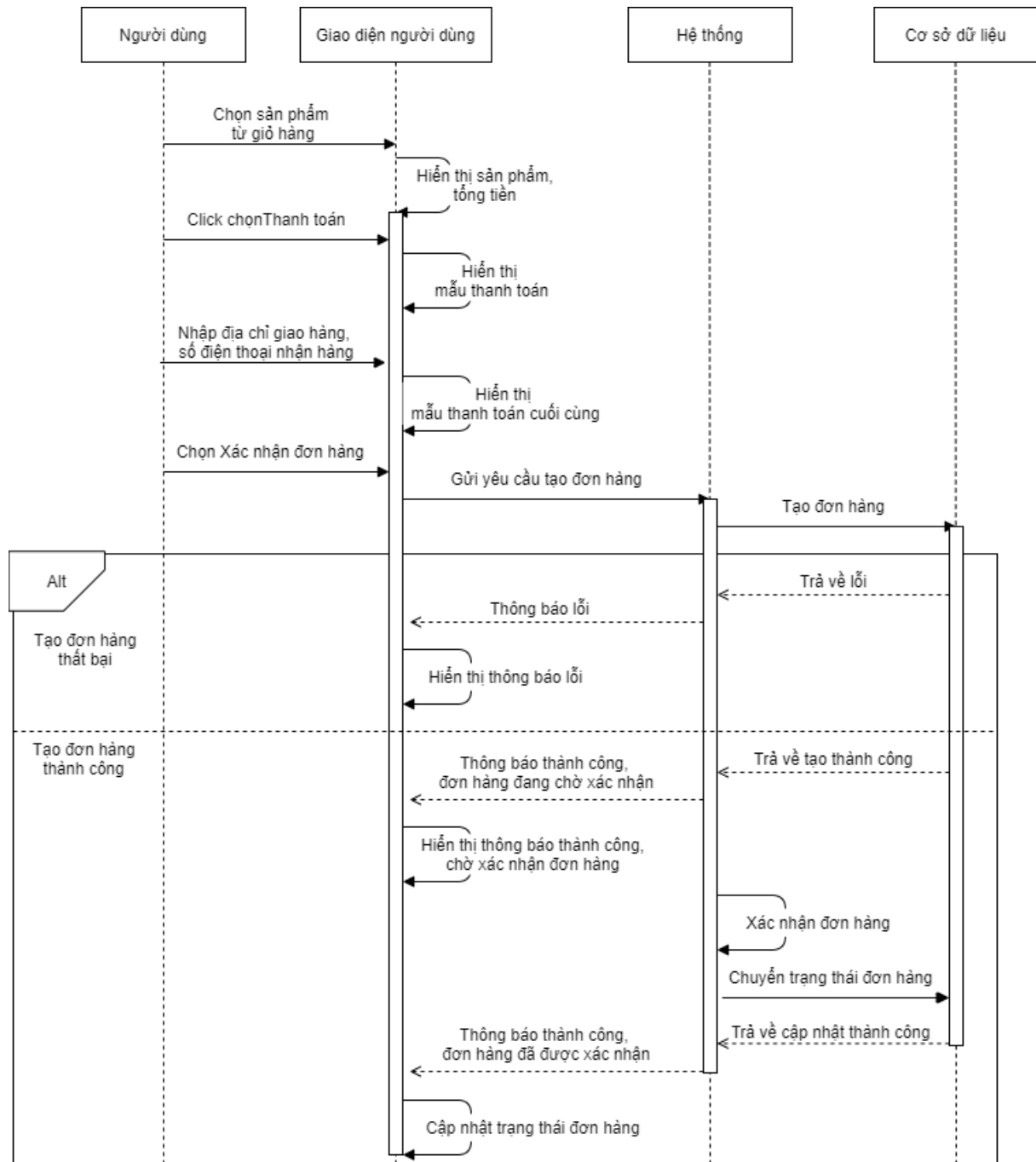
Hình 2. 9 Sơ đồ hoạt động việc đặt hàng và xác nhận đơn hàng

#### 2.2.4. Sơ đồ tuần tự

Các biểu đồ tuần tự minh họa việc gửi và nhận thông điệp giữa các đối tượng và theo thứ tự chuỗi thời gian. Ở đây em chỉ vẽ sơ đồ tuần tự cho hai chức năng quan trọng nhất của kinh doanh online là đặt hàng và quản lý đơn hàng.

##### ➤ Sơ đồ tuần tự chức năng đặt hàng từ giỏ hàng

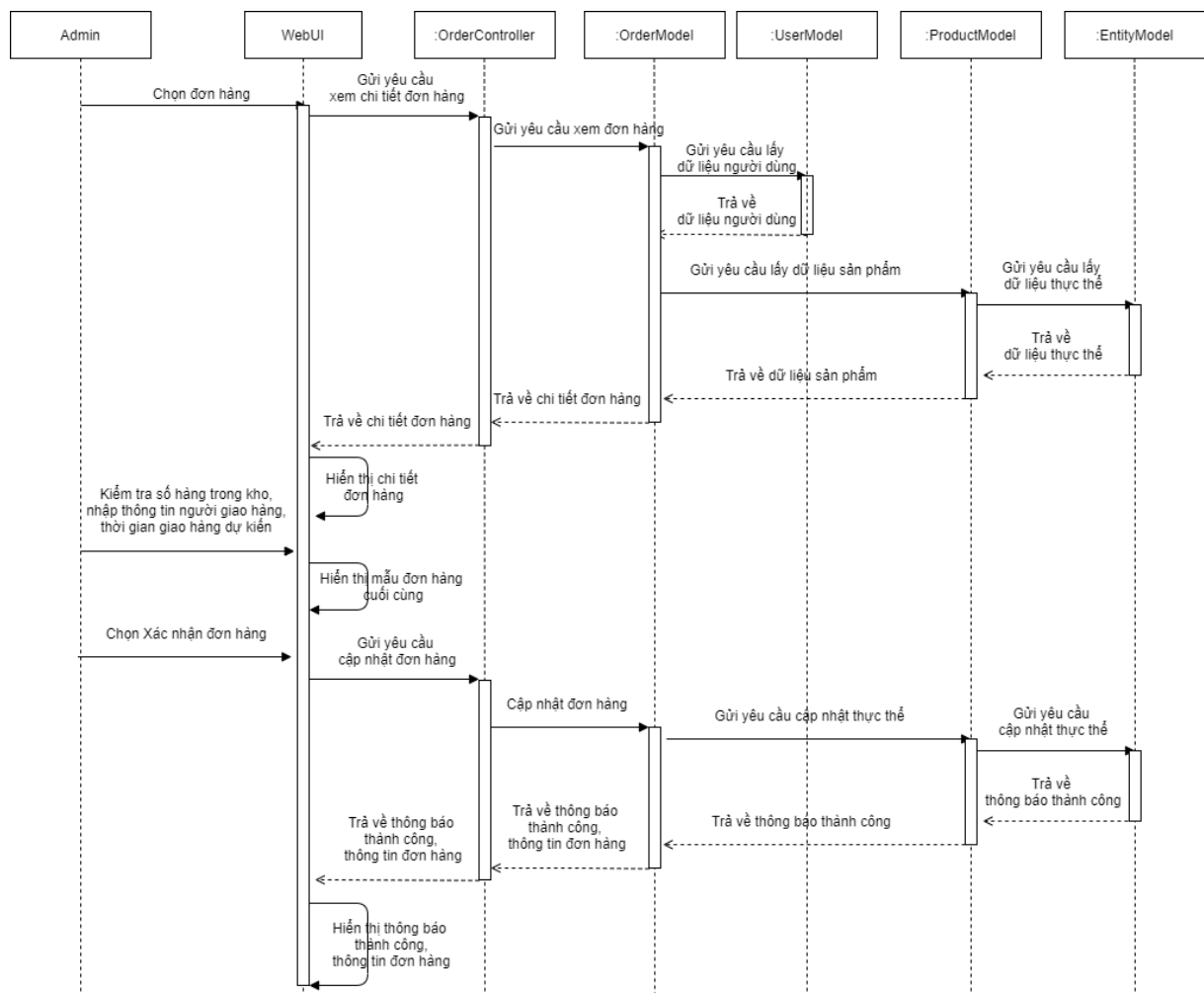
Chức năng được thực hiện bởi người dùng. Người dùng chọn thanh toán giỏ hàng, sau đó nhập dữ liệu vào Form, nhấn “Xác nhận” thì sẽ gửi yêu cầu tạo đơn hàng lên Server. Server sẽ chịu trách nhiệm kiểm tra và lưu dữ liệu vào DB và thông báo về cho người dùng thông báo tạo thành công và đơn hàng đang được xử lý hoặc thông báo tạo đơn hàng thất bại.



Hình 2. 10 Sơ đồ tuần tự chức năng đặt hàng

### ➤ Sơ đồ tuần tự chức năng xử lý đơn hàng

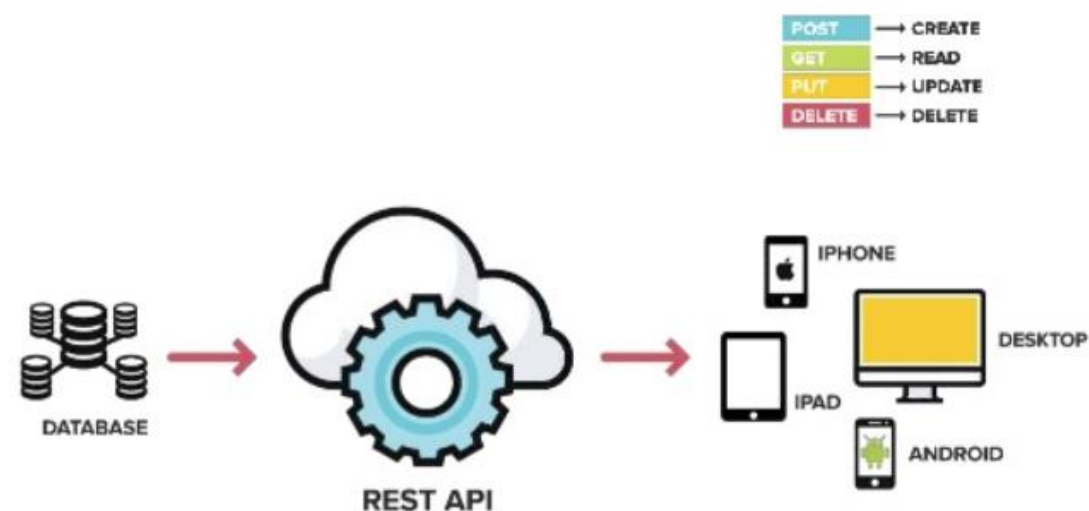
Sơ đồ biểu diễn người đang thực hiện là người quản lý hệ thống. Quản trị viên chọn nút xem chi tiết đơn hàng và gửi yêu cầu lên server. Khi đó OrderController sẽ tập hợp dữ liệu cần thiết để trả về chi tiết đơn hàng. Sau đó Quản trị viên sẽ kiểm tra và thêm các thông tin người giao hàng, thời gian giao hàng dự kiến và gửi yêu cầu xác nhận hay hủy đơn hàng, Controller sau đó sẽ bắt đầu cập nhật lại đơn hàng và hiển thị kết quả là thông tin đầy đủ của đơn hàng.



Hình 2. 11 Sơ đồ tuần tự chức năng xử lý đơn hàng

### 2.2.5. Kiến trúc hệ thống

Đây là ứng dụng có sự tương tác trên môi trường web giành cho quản lý nên hệ thống sẽ được xây dựng theo mô hình client-server và dựa trên kiến trúc RESTful. Trong đó server nắm vai trò chính trong xử lý logic nghiệp vụ và quản trị cơ sở dữ liệu, phía clients gồm có trình duyệt web.



Hình 2. 12 Kiến trúc hệ thống

Nhờ vào kiến trúc như vậy, chúng ta có thể triển khai môi trường client trên nhiều nền tảng khác nhau, các xử lý liên quan đến nghiệp vụ tập trung, dễ dàng nâng cấp hoặc thay đổi.

### 2.2.6. Thiết kế cơ sở dữ liệu

Cơ sở dữ liệu luôn giữ vai trò rất quan trọng trong mọi hệ thống. Tất cả dữ liệu trên hệ thống đều được lưu vào các bảng trong cơ sở dữ liệu. Giữa các bảng có những mối quan hệ nhất định được liên kết bằng cách tham chiếu đến khóa chính của bảng khác. Từ phân tích yêu cầu, thiết kế sơ đồ lớp, hệ thống có sơ đồ cơ sở dữ liệu quan hệ được thiết kế tương ứng như sau:



Hình 2. 13 Quan hệ cơ sở dữ liệu

### **2.3. Kết chương**

Chương này đi sâu phân tích hệ thống, thiết kế về mặt chức năng, tổ chức CSDL. Trình bày các biểu đồ ca sử dụng tổng quan và các chức năng của hệ thống. Mô tả chi tiết luồng hoạt động của một số chức năng chính, được trình bày thông qua các biểu đồ tuần tự và sơ đồ hoạt động. Ngoài ra chương còn trình bày biểu đồ lớp và thiết kế CSDL bằng mô hình quan hệ ERD và các bảng mô tả chi tiết.

## **CHƯƠNG 3. TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ**

### **3.1. Môi trường và công cụ phát triển**

### **3.2. Kết quả triển khai**

### **3.3. Đánh giá**

### **3.4. Kết chương**



## **KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

- 1. Kết quả đạt được**
- 2. Những vấn đề hạn chế**
- 3. Hướng phát triển**

## TÀI LIỆU THAM KHẢO

**Sách, giáo trình và bài báo tham khảo:**