**TRƯỜNG ĐẠI HỌC MỞ HÀ NỘI**

**KHOA CÔNG NGHỆ ĐIỆN TỬ-THÔNG TIN**

-----□□□□-----



**BÁO CÁO ĐỒ ÁN**

**TỐT NGHIỆP ĐẠI HỌC**

**Đề tài : PHÂN TÍCH THIẾT KẾ VÀ XÂY DỰNG WEBSITE BÁN ĐIỆN THOẠI**

**Giảng viên hướng dẫn : ThS.PHẠM TIẾN HUY**

**Sinh viên thực hiện : ĐÀM VĂN THỊNH**

**Lớp : K21A – ĐT2**

**Khóa : 2018-2022**

**Hệ : CHÍNH QUY**

**Hà Nội, ngày 23 tháng 2 năm 2022**

-----□□□□-----

|  |  |
| --- | --- |
| TRƯỜNG ĐẠI HỌC MỞ HÀ NỘI | CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM  ***Độc lập - Tự do – Hạnh phúc*** |
| **KHOA CN ĐIỆN TỬ - THÔNG TIN** |

**ĐỀ TÀI ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC**

**Họ và tên sinh viên:** Đàm Văn Thịnh

**Lớp:**K21-ĐT2 **Khoá:** 2018-2022

**Ngành đào tạo**: Công nghệ Kỹ thuật điện tử-viễn thông

**Hệ đào tạo**: ĐHCQ

***1/ Tên đề tài ĐATN:***

Phân tích thiết kế và xây dựng website bán điện thoại

***2/ Nội dung chính:***

1. Chương 1: Giơi thiệu đề tài
2. Chương 2: Giới thiệu về công nghệ và mục tiêu thiết kế LTE.
3. Chương 3: Kiến trúc giao thức 4G-LTE

***3/ Ngày giao :*** **……………….**

***4/ Ngày nộp:* ………………**

|  |  |
| --- | --- |
| **TRƯỞNG KHOA** | **GIÁO VIÊN HƯỚNG DẪN** |
| *(Ký, ghi rõ họ tên)* | *(Ký, ghi rõ họ tên)* |

# **LỜI MỞ ĐẦU**

Hiện nay ngành viễn thông có những bước phát triển rất nhanh, hình thành môi trường cạnh tranh lớn giữa các nhà khai thác mạng phục vụ nhu cầu về thông tin cho con người ngày càng đòi hỏi cao hơn. Chất lượng các dịch vụ ngày càng trở thành chìa khóa để có thể dẫn tới thành công. Song song với xu thế này, nhu cầu ngày càng gia tăng đối với các dịch vụ truyền thông mới, đủ khả năng đáp ứng việc cung cấp dịch vụ hoặc tăng tính cạnh tranh.

ITU công bố chuẩn IMT-2000 cho hệ thống 3G năm 1992, tuy nhiên sau đó 4G mới là đích đến của truyền tải dữ liệu không dây.

LTE được xem như “người kế thừa” xuất sắc của thế hệ công nghệ mạng 3G hiện tại, dựa trên nền tảng WCDMA, HSDPA, HSUPA, và HSPA. LTE cập nhật công nghệ UMTS để cải thiện một cách đáng kể tốc độ truyền dữ liệu hai chiều.

Nhận thấy công nghệ LTE là công nghệ mới triển vọng nhất cho việc tiến lên 4G nên em chọn đề tài tìm hiểu về LTE: “Tìm hiểu kiến trúc 4G-LTE”. Nội dung báo cáo gồm có:

Chương 1: Tiến trình phát triển của các hệ thống thông tin di động từ 1G lên 4G.

Chương 2: Giới thiệu về công nghệ và mục tiêu thiết kế LTE.

Chương 3: Kiến trúc giao thức 4G-LTE

**LỜI CẢM ƠN**

Để hoàn thành chuyên đề báo cáo thực tập này trước hết em xin gửi đến quý thầy, cô giáo trong khoa Điện tử - Viên thông trường Đại Học Mở Hà Nôi lời cảm ơn chân thành. Đặc biệt, em xin gửi đến thầy Phạm Tiến Huy, người đã tận tình hướng dẫn, giúp đỡ em hoàn thành hoàn thành đề tài này lời cảm ơn sâu sắc nhất.

Em cũng xin gửi lời tri ân đến các thầy cô trong khoa Điện tử- Viễn Thông trường Đại Học Mở Hà Nội đã tận tình dạy dỗ và giảng dạy cho chúng em những kiến thức cơ bản đến nâng cao tạo cho chúng em một cơ sở kiến thức vững vàng để hoàn thành đề tài này.

Và cuối cùng chúng em xin cảm ơn các anh, chị đi trước và bạn bè đã có những lời khuyên, lời góp ý chân thành để đề tài của chúng em có thể hoàn thiện hơn.

Một lần nữa, chúng em xin cảm ơn tất cả mọi người đã luôn ở bên cạnh và giúp đỡ chúng em, tạo động lực để chúng em hoàn thành tốt đề tài này.

Xin chân thành cảm ơn!

# **NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN**

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

…………………………………………………………………………………....

…………………………………………………………………………………....

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

…………………………………………………………………………………….

…………………………………………………………………………………….

…………………………………………………………………………………….

…………………………………………………………………………………….

………….Điểm…………..(bằng chữ………)……………………………………

Ngày … tháng 2 năm 2022

Giảng viên hướng dẫn

**ThS.Phạm Tiến Huy**

# **MỤC LỤC**

[LỜI MỞ ĐẦU 2](#_Toc96593511)

[MỤC LỤC 5](#_Toc96593512)

[DANH MỤC HÌNH VẼ, BẢNG BIỂU 6](#_Toc96593513)

[CÁC THUẬT NGỮ VIẾT TẮT 7](#_Toc96593514)

[CHƯƠNG 1: TIẾN TRÌNH PHÁT TRIỂN CỦA CÁC HỆ THỐNG THÔNG TIN DI ĐỘNG TỪ 1G LÊN 4G. 9](#_Toc96593515)

[**1.1.** **Mạng thông tin di động 1G:** 9](#_Toc96593516)

[**1.2. Mạng thông tin di động 2G:** 9](#_Toc96593517)

[**1.3. Mạng thông tin di động 2,5G** 10](#_Toc96593518)

[**1.4. Mạng thông tin di động 3G:** 11](#_Toc96593519)

[**1.5. Mạng thông tin di động thế hệ 3.5G:** 12](#_Toc96593520)

[**1.6. Mạng thông tin di động 4G:** 13](#_Toc96593521)

[CHƯƠNG 2: GIỚI THIỆU VỀ CÔNG NGHỆ VÀ MỤC TIÊU THIẾT KẾ LTE 14](#_Toc96593522)

[**2.1. Giới thiệu về công nghệ LTE:** 15](#_Toc96593523)

[**2.2. Mục tiêu thiết kế LTE:** 15](#_Toc96593524)

[**2.3. Lộ trình phát triển công nghệ LTE và các công nghệ khác:** 17](#_Toc96593525)

[CHƯƠNG 3: KIẾN TRÚC GIAO THỨC CỦA 4G-LTE 22](#_Toc96593526)

[**3.1. PDCP: Packet Data Convergence Protocol - Giao thức hội tụ số liệu gói** 22](#_Toc96593527)

[**3.2. RLC: radio link control - điều khiển liên kết vô tuyến** 23](#_Toc96593528)

[**3.3. Điều khiển truy nhập môi trường MAC (Medium Access Control)** 24](#_Toc96593529)

[**3.4. PHY: physical layer - lớp vật lý** 33](#_Toc96593530)

[KẾT LUẬN 35](#_Toc96593531)

[TÀI LIỆU THAM KHẢO: 36](#_Toc96593532)

**DANH MỤC HÌNH VẼ, BẢNG BIỂU**

[Hình 1: Lộ trình phát triển thông tin di động từ 1G lên 4G. 13](#_Toc96845622)

[Bảng 1: Các tham số cơ bản của hệ thống 4G 14](#_Toc96845623)

[Hình 2.1: Kiến trúc mô hình LTE 15](#_Toc96845624)

[Bảng 2: So sánh công nghệ kỹ thuật dùng trong WiMAX và 3G LTE 18](#_Toc96845625)

[Bảng 3. Dịch vụ và ứng dụng của LTE 21](#_Toc96845626)

[Hình 3.1: Thí dụ về sắp xếp các kênh logic lên các kênh truyền tải 26](#_Toc96845627)

[Hình 3.2 – Giao thức hybrid-ARQ đồng bộ và không đồng bộ. 31](#_Toc96845628)

# **CÁC THUẬT NGỮ VIẾT TẮT**

|  |  |  |
| --- | --- | --- |
| AMPS | Advanced Mobile Phone Sytem | Hệ thống điện thoại di động tiên tiến |
| BCCH | Broadcast Control Channel | Kênh điều khiển quảng bá |
| BCH | Broadcast Channel | Kênh quảng bá |
| DCCH | Dedicated Control Channel | Kênh điều khiển dành riêng |
| DL-SCH | Downlink Shared Channel | Kênh chia sẻ đường xuống |
| DTCH | Dedicated Traffic Channel | Kênh lưu lượng dành riêng |
| EDGE | Enhanced Data Rates for GSM Evolution | Tốc độ số liệu tăng cường để phát triển GSM |
| GPRS | General Packet Radio Service | Dịch vụ vô tuyến gói chung |
| HSCSD | High Speed Circuit Switched Data | Số liệu chuyển mạch kênh tốc độ cao |
| HSDPA | High Speed Downlink Package Access | Truy nhập gói đường xuống tốc độ cao |
| MAC | Medium Access Control | Điều khiển truy cập môi trường |
| MCCH | Multicast Control Channel | Kênh điều khiển multicast |
| MCH | Multicast Channel | Kênh multicast |
| MIMO | Multiple input Multiple Output | Đa nhập đa xuất |
| MTCH | Multicast Traffic Channel | Kênh lưu lượng multicast |
| PCCH | Paging Control Channel | Kênh điều khiển tìm gọi |
| PCH | Paging channe | Kênh tìm gọi |
| PDCP | Packet Data Convergence Protocol | Giao thức hội tụ số liệu gói |
| PDU | Protocol Data Unit | Đơn vị dữ liệu giao thức |
| PHY | Physical layer | Lớp vật lý |
|  |  |  |
| ITU | International Telecommunications Union | Liên minh Viễn thông Quốc tế |
| LTE | **Long Term Evolution** | Tiến hóa dài hạn |
| RLC | Radio Link Control | Điều khiển liên kết vô tuyến |
| SDU | Service Data Unit | Đơn vị dữ liệu dịch vụ |
| TACS | Total Access Communication Sytem | Hệ thống giao tiếp truy cập tổng hợp |
| UL-SCH | Uplink shared channel | Kênh chia sẻ đường lên |

1. **GIỚI THIỆU VỀ ĐỀ TÀI:**
2. **Lý do chọn đề tài:**

Công nghệ ngày càng phát triển, việc tiếp cận và sử dụng công nghệ hiện nay đã trở nên rất phổ biến. Chỉ với một chiếc máy tính có kết nối mạng Internet là đã có thể xem và tìm kiếm rất nhiều tài nguyên mạng. Các doanh nghiệp, cửa hàng hiện nay cũng có rất nhiều sự cạnh tranh, và cần mở rộng thị trường mua sắm.

Với những hình thức quản lý cửa hàng nói chung hay những cửa hàng bán điện thoại đi động và phụ kiện điện thoại nói riêng, việc quản lý theo hình thức truyền thống là gây khó khăn và tốn kém nhân lực trong rất nhiều khâu quản lý như tìm kiếm, lưu trữ thông tin, thống kê các mặt hàng… Và với hình thức truyền thống thì những thông tin về cửa hàng, cũng như các mặt hàng điện thoại và phụ kiện mà cửa hàng bán sẽ không được tiếp cận nhanh chóng đến người dùng.

Để giải quyết những vấn đề nan giải này, em quyết định chọn đề tài “Phân tích thiết kế và xây dựng website bán điện thoại đi động ”. Nhằm giúp những người quản lý hay chủ cửa hàng có thể mở rộng thị trường mua bán, quản lý cửa hàng, doanh nghiệp dễ dàng và nhanh chóng hơn.

1. **Nhiệm vụ của đề tài:**

* Nghiên cứu và khảo sát thực trạng các cửa hàng bán điện thoại di động và phụ kiện điện thoại, để rút ra các yêu cầu mà đồ án cần đạt được.
* Nêu ra các lý thuyết áp dụng, phân tích và thiết kế hệ thống cần triển khai theo mô hình hiện đại, mà vẫn đảm bảo các nghiệp vụ của hệ thống cũ.
* Triển khai, lập trình website bán điện thoại và phụ kiện theo các yêu cầu đã đặt ra sau khi khảo sát và dựa theo phân tích thiết kế hệ thống đã tiến hành.

1. **Dự kiến kết quả đạt được:**

* Tìm hiểu nhu cầu, nghiệp vụ của các cửa hàng bán điện thoại đi động
* Hiểu về ngôn ngữ lập trình REACTJS và áp dụng xây dựng website.
* Hoàn thành quản lý website với các tính năng:

- Quản lý: Quản lý hệ thống với các mục quản lý sản phẩm, quản lý thông tin khách hàng, quản lý các đơn hàng, …

- Khách hàng: Có thể thực hiện chức năng tìm kiếm sản phẩm, xem các thông tin chi tiết của các sản phẩm điện thoại , đăng ký hay đăng nhập tài khoản…

1. **Ý nghĩa của đề tài:**

* Ý nghĩa khoa học:

Vận dụng các kiến thức đã học để xây dựng website bán điện thoại :

- Nghiên cứu và lập trình với các ngôn ngữ thiết kế website HTML, CSS, cùng với các ngôn ngữ lập trình REACTJS, Nodejs.

- Sử dụng các công cụ như thư viện Meterial UI, .

- Nghiên cứu phân tích và thiết kế hệ thống.

- Nghiên cứu xây dựng giao diện website.

* Ý nghĩa thực tiễn:

Website bán điện thoại và phụ kiện giúp việc quản lý nhanh chóng, lưu trữ lượng dữ liệu lớn, còn có giao diện thân thiện, dễ sử dụng… Website bán điện thoại và phụ kiện mang lại rất nhiều lợi ích cho người dùng:

- Giúp quản lý các sản phẩm điện thoại và phụ kiện với nhiều thao tác (thêm, sửa, xóa, tìm kiếm) nhanh chóng.

- Quản lý, theo dõi các đơn hàng dễ dàng, cập nhật các trạng thái liên tục.

- Các sản phẩm điện thoại và phụ kiện được cập nhật thường xuyên.

- Khách hàng dễ dàng chọn mua những sản phẩm mình yêu thích thông qua chức năng tìm kiếm, và đặt mua nhanh chóng với chức năng đặt hàng.

1. **Bố cục của đồ án:**

* ***Chương I: Giới thiệu về đề tài:***

Giới thiệu tổng quan về đề tài bao gồm nêu lý do chọn đề tài, nhiệm vụ của đề tài, dự kiến các kết quả sẽ đạt được khi xây dựng đề tài, ý nghĩa đề tài mang lại.

* ***Chương II: Kiến thức nền tảng:***

Nêu các cơ sở lý thuyết như ngôn ngữ HTML, CSS, ngôn ngữ lập trình REACTJS, Nodejs… và công cụ sử dụng như thư viện Meterial UI, … mà đề tài sử dụng.

* ***Chương III: Phân tích và thiết kế hệ thống:***

Khảo sát thực trạng của đề tài và nêu các yêu cầu của đề tài đặt ra, từ đó triển khai việc phân tích các chức năng qua các biểu đồ và thiết kế cơ sở dữ liệu của hệ thống.

* ***Chương IV: Giao diện phần mềm:***

Từ những phân tích và thiết kế ở trên, tiến hành xây dựng triển khai giao diện phần mềm của hệ thống.

* ***Chương V: Kết luận:***

Trình bày những kết quả đạt được, hạn chế và hướng phát triển của đề tài trong tương lai.

* ***Chương VI: Tài liệu tham khảo:***

Nêu các tài liệu tham khảo trong quá trình xây dựng đồ án.

1. **KIẾN THỨC NỀN TẢNG:**
2. **Tổng quan về nền tảng Web:**
   1. **Internet:**

* *Ỉnternet là gì?*

Internet là một liên mạng máy tính toàn cầu được hình thành từ các mạng nhỏ hơn, liên kết hàng triệu máy tính trên thế giới thông qua cơ sở hạ tầng viễn thông. Internet là mạng của các mạng máy tính. Trong mạng này, các máy tính và thiết bị mạng giao tiếp với nhau bằng một ngôn ngữ thống nhất.

* *Lịch sử hình thành:*
* Internet được hình thành từ năm 1969, từ một dự án nghiên cứu của Bộ quốc phòng Mỹ. Lúc đó Internet chỉ liên kết 4 địa điểm: Viện Nghiên cứu Standford, Trường Đại học tổng hợp California ở LosAngeles, UC – Santa Barbara và Trường Đại học Tổng hợp Utah. Mạng này được biết đến dưới cái tên ARPANET.
* ARPANET càng phát triển khi có nhiều máy nối vào – rất nhiều trong số này là từ các cơ quan của Bộ quốc phòng Mỹ hoặc những trường đại học nghiên cứu với các đầu nối vào Bộ quốc phòng. Đây là những giao điểm trên mạng. Trong khi ARPANET đang cố gắng chiếm lĩnh mạng quốc gia thì một nghiên cứu tại Trung tâm nghiên cứu Palo Alto của công ty Xerox đã phát triển một kỹ thuật được sử dụng trong mạng cục bộ là Ethernet.
* Theo thời gian, Ethernet trở thành một trong những chuẩn quan trọng để kết nối trong các mạng cục bộ. Cũng trong thời gian này, DARPA (đặt lại tên từ ARPA) chuyển sang hợp nhất TCP/IP (giao thức được sử dụng trong việc truyền thông trên Internet) vào phiên bản hệ điều hành UNIX của trường đại học tổng hợp California ở Berkeley. Với sự hợp nhất như vậy, những trạm làm việc độc lập sử dụng UNIX đã tạo nên một thế mạnh trên thị trường, TCP/IP cũng có thể dễ dàng tích hợp vào phần mềm hệ điều hành. TCP/IP trên Ethernet đã trở thành một cách thức thông dụng để trạm làm việc nối đến trạm khác.
* Trong thập kỷ 1980, máy tính cá nhân được sử dụng rộng rãi trong các công ty và trường Đại học trên thế giới. Mạng Ethernet kết nối các máy tính cá nhân (PC) trở thành phổ biến. Các nhà sản xuất phần mềm thương mại cũng đưa ra những chương trình cho phép các máy PC và máy UNIX giao tiếp cùng một ngôn ngữ trên mạng.
* Vào giữa thập kỷ 1980, giao thức TCP/IP được dùng trong một số kết nối khu vực – khu vực (liên khu vực) và cũng được sử dụng cho các mạng cục bộ và mạng liên khu vực. Thuật ngữ "Internet" xuất hiện lần đầu vào khoảng 1974 trong khi mạng vẫn được gọi là ARPANET. Vào thời điểm này, ARPANET (hay Internet) còn ở qui mô rất nhỏ.
* Mốc lịch sử quan trọng của Internet được chọn vào giữa thập kỷ 1980, khi Quỹ khoa học quốc gia Mỹ NSF (National Science Foundation) thành lập mạng liên kết các trung tâm máy tính lớn với nhau gọi là NSFNET. Mạng này chính là mạng Internet. Điểm quan trọng của NSFNET là cho phép mọi người cùng sử dụng. Trước NSFNET, chỉ các nhà khoa học, chuyên gia máy tính và nhân viên các cơ quan Chính phủ được kết nối Internet. Nhiều doanh nghiệp đã chuyển từ ARPANET sang NSFNET. Chính vì vậy, sau gần 20 năm ARPANET trở nên không còn hiệu quả và đã ngừng hoạt động vào khoảng năm 1990.
* Ngày nay, mạng Internet phát triển mạnh mẽ hơn các phương tiện truyền thông truyền thống khác như phát thanh và truyền hình, do sự cải tiến và phát triển không ngừng. Các công nghệ đang áp dụng trên Internet giúp cho Internet trở thành mạng liên kết vô số kho tin toàn cầu, có dịch vụ phong phú về nội dung, hình thức.
  1. **Ngôn ngữ HTML và CSS:**



***Hình 2.1. Minh họa trang web sử dụng HTML và CSS***

* *Giới thiệu HTML:*

HTML(Hypertext Markup Language) là mã được dùng để xây dựng nên cấu trúc và nội dung của trang web. Ví dụ, nội dung có thể được cấu thành bởi một loạt các đoạn văn, một danh sách liệt kê, hoặc sử dụng những hình ảnh và bảng biểu...

* *Tính năng mới trong HTML5:*

*-* Phần tử <small> dùng để chỉ hiển thị cỡ nhỏ, nó có thể được ký hiệu là trình bao bọc chính xác cho thông tin này.

- Việc sử dụng dấu ngoặc kép có thể phụ thuộc vào người dùng.

- Cho phép người dùng chỉnh sửa bất kỳ văn bản nào có trong phần tử, bao gồm cả phần tử con của nó.

- HTML5 loại bỏ hoàn toàn thuộc tích <type>.

- Cung cấp các đầu vào email cho phép hướng dẫn trình duyệt chỉ cho phép các chuỗi xác nhận địa chỉ email.

- Dễ dàng tạo thanh trượt với phạm vi đầu vào.

* *Giới thiệu CSS:*

CSS (viết tắt của Cascading Style Sheets) dùng để miêu tả cách trình bày các tài liệu viết bằng ngôn ngữ HTML và XHTML. CSS được hiểu một cách mà chúng ra thêm các kiểu hiển thị (font chữ, kích thước, màu sắc,…) cho một tài liệu Web.

* *Tính năng mới trong CSS3:*

- Bo tròn ở các góc ảnh.

- Xây dựng phần tử động, tạo đối tườn động animation.

- Biến đổi phần tử bằng cách quay, dịch chuyển…

- Đổ bóng cho phần tử với box-shadow.

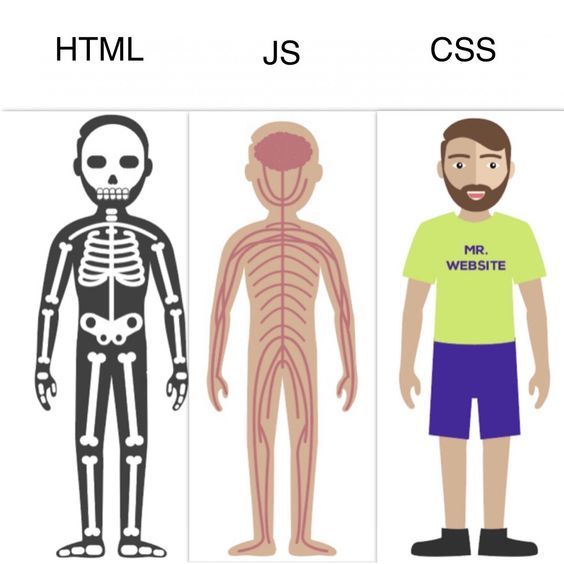
- Tạo gradient với hai kiểu linear và radial.

- Bo trong các đường viền ở 4 góc của phần tử box.

* 1. **Ngôn ngữ JavaScript:**
* *Giới thiệu JavaScript:*

JavaScriptlà một ngôn ngữ lập trình của HTML và ứng dụng Web. Nó được sử dụng phổ biến nhất như một phần của các trang web, chúng cho phép Client-Side script tương tác với người sử dụng và tạo các trang web động. Nó là một ngôn ngữ chương trình thông dịch với các khả năng hướng đối tượng.

* *JavaScript kết hợp với HTML và CSS:*



***Hình 2.2. Minh họa JavaScript kết hợp với HTML và CSS***

Một website thường có 3 phần cơ bản gồm HTML, CSS và Javascript. Trong đó, HTML quyết định nội dung và cấu trúc trang web; CSS quyết định màu sắc, hình dáng, kiểu chữ,… Hầu hết các thay đổi của HTML và CSS đều được thể hiện dưới dạng tĩnh, không thể thực hiện các hành động với chuyển động bắt mắt như xoay hình, kiểm tra thông tin hợp lệ, hiển thị thông báo người dùng...

Tất cả các hành động này được chuyển thể từ trạng thái tĩnh sang trạng thái động nhờ vào thành phần thứ 3 là Javascript. Ba thành phần này kết hợp với nhau tạo nên website hoàn chỉnh với giao diện (UI) và trải nghiệm người dùng (UX) chất lượng.

* 1. **ReactJS:**
* *Giới thiệu ReactJs:*

Là một thư viện JavaScript được phát triển bởi Facebook và Instagram, được dùng để xây dựng giao diện người dùng (Front-end).

***Một số thành phần liên quan:***

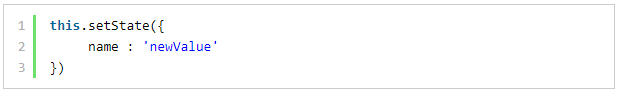
* React Components giúp phân chia các UI (giao diện người dùng) thành các phần nhỏ để dễ dàng quản lý và tái sử dụng, là một khối đóng gói các thẻ HTML.
* Props là một object được truyền vào trong mỗi components, mỗi components sẽ nhận vào props và trả về react element (phần tử react). Props cho phép các component giao tiếp với nhau bằng cách truyền tham số qua lại

Cách truyền props giống như cách thêm một attributes (thuộc tính) cho một element HTML (phần tử HTML)

* State là một object có thể được sử dụng để chứa dữ liệu hoặc thông tin về components. State có thể được thay đổi bất cứ khi nào mong muốn. Khác với props, props có thể truyền sang các components khác nhau còn state chỉ tồn tại trong phạm vi của một component chứa nó, mỗi khi state thay đổi thì component chứa nó sẽ được render lại.
* Khởi tạo một state:



* Cập nhật state:



* JSX (Javascript + XML) là cú pháp mở rộng giống như XML của Javascript. React sử dụng JSX làm quy chuẩn khi lập trình thay cho Java Script thông thường. Cho phép viết HTML trong React.
* Sử dụng JSX



* Không sử dụng JSX



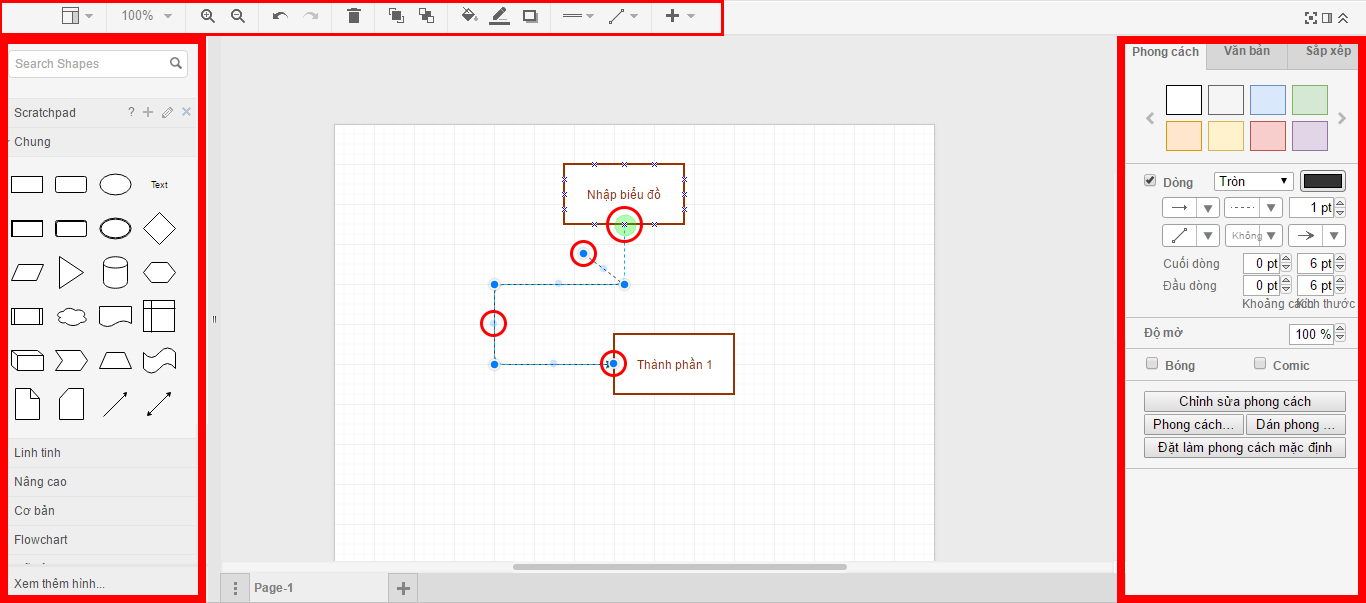
* 1. **Hệ quản trị cơ sở dữ liệu:**
* *Giới thiệu Cơ sở dữ liệu:*

Cơ sở dữ liệu (Database) là một tập hợp các dữ liệu có tổ chức, thường được lưu trữ và truy cập điện tử từ hệ thống máy tính. Khi cơ sở dữ liệu phức tạp hơn, chúng thường được phát triển bằng cách sử dụng các kỹ thuật thiết kế và mô hình hóa chính thức.

* *Giới thiệu Hệ quản trị cơ sở dữ liệu:*

Hệ quản trị cơ sở dữ liệu (viết tắt của Database Management System) là hệ thống được thiết kế để quản lý một khối lượng dữ liệu nhất định một cách tự động và có trật tự. Các hành động quản lý bao gồm chỉnh sửa, xóa, lưu và tìm kiếm thông tin trong một nhóm dữ liệu nhất định.

1. **Công cụ hỗ trợ:**
   1. **Phần mềm trực tuyến Draw.io:**



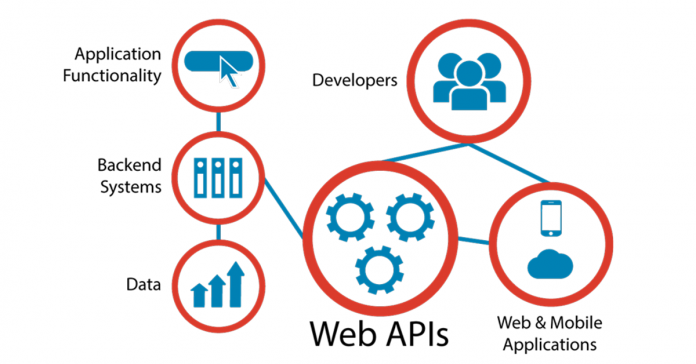
***Hình 2.3. Giao diện phần mềm trực tuyết Draw.io***

Draw.io là một công cụ vẽ sơ đồ rất mạnh mẽ, hỗ trợ nhiều hình khối, chạy online không cần cài đặt mà lại miễn phí và không bị giới hạn số biểu đồ như nhiều tool vẽ web khác.

Draw.io cho phép vẽ hàng nghìn sơ đồ thiết kế phần mềm, phần cứng và hệ thống. Nó có thư viện template rất phong phú để bạn có thể bắt đầu nhanh hơn, không phải tự mình vẽ từ đầu.

* 1. **WEB API:**
* *Giới thiệu Web API:*

API là các phương thức, giao thức kết nối với các thư viện và ứng dụng khác. API là viết tắt của Application Programming Interface - giao diện lập trình ứng dụng, cung cấp khả năng truy xuất đến một tập các hàm hay dung, từ đó có thể trao đổi dữ liệu giữa các ứng dụng. Nhiều loại hệ thống và ứng dụng thực hiện API như các hệ thống đồ họa, cơ sở dữ liệu, mạng, dịch dụ web và một số ứng dụng game.



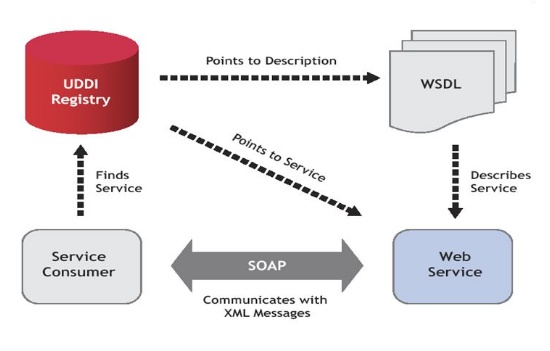
Hình 1.5: Mô hình hoạt động của các thành phần

.

* *Những đặc điểm nổi bật API:*

Web API hỗ trợ restful đầy đủ các phương thức: Get/Post/Put/Delete dữ liệu. Giúp xây dựng các HTTP service một cách rất đơn giản và nhanh chóng, có khả năng hỗ trợ đầy đủ các thành phần HTTP: URI, request/response headers, caching, versioning, content format.

* Khả năng tích hợp linh động
* API cho phép lấy nội dung từ bất kỳ Website hoặc ứng dụng nào một cách dễ dàng nếu được cho phép, tăng trải nghiệm người dùng. API hoạt động như một chiếc cổng, cho phép các công ty chia sẻ thông tin được chọn nhưng vẫn tránh được những yêu cầu không mong muốn.
* Cập nhật thông tin thời gian thực
* API có chức năng thay đổi và cập nhật thay đổi theo thời gian thực. Với công nghệ này, dữ liệu sẽ được truyền đi tốt hơn, thông tin chính xác hơn, dịch vụ cung cấp linh hoạt hơn.
* API dùng chung nên khi thay đổi bên Back-end thì các ứng dụng khác (mobile, ...) khi gọi đến API đó đều đc thay đổi, còn đối với mô hình MVC thì khi thay đổi thì phải thay đổi ở các ứng dụng khác (Ví dụ: Khi thay đổi xử lý ở Website thì phải thay đổi ở bên Mobile web)
* *Cách hoạt động Web API*
* Xây dựng URL API để bên thứ ba có thể gửi request dữ liệu đến máy chủ cung cấp nội dung, dịch vụ thông qua giao thức HTTP hoặc HTTPS.
* Tại web server cung cấp nội dung, các ứng dụng nguồn sẽ thực hiện kiểm tra xác thực nếu có và tìm đến tài nguyên thích hợp để tạo nội dung trả về kết quả.
* Server trả về kết quả theo định dạng JSON hoặc XML thông qua giao thức HTTP/HTTPS.
* Tại nơi yêu cầu ban đầu là ứng dụng web hoặc ứng dụng di động, dữ liệu JSON/XML sẽ được parse để lấy data. Sau khi có được data thì thực hiện tiếp các hoạt động như lưu dữ liệu xuống cơ sở dữ liệu, hiển thị dữ liệu…
  1. **Mô hình web services:**
* *Giới thiệu web services:*
* Web Services (hay “dịch vụ web”) là một mô-đun phần mềm được thiết kế để thực hiện một nhóm các tác vụ nhất định, là sự kết hợp giữa các máy tính cá nhân với các thiết bị khác, các cơ sở dữ liệu và các mạng máy tính để tạo ra một cơ cấu máy tính ảo mà người sử dụng có thể làm việc thông qua các trình duyệt mạng. Web services có thể được truy cập và sử dụng thông qua mạng Internet dưới dạng dịch vụ. Khi đó, web service sẽ cung cấp các chức năng của nó cho máy khách để người dùng đạt được các mục tiêu sử dụng nhất định.
* Các ứng dụng phần mềm được viết bằng các ngôn ngữ lập trình khác nhau và chạy trên các nền tảng khác nhau, có thể sử dụng các dịch vụ web để trao đổi dữ liệu qua mạng máy tính.
* Web service hoạt động một cách độc lập không phụ thuộc bất kỳ ngôn ngữ nào. Các ứng dụng java, .net hoặc PHP… có thể giao tiếp với các ứng dụng khác thông qua web service.
* Web service truyền thông bằng cách sử dụng các giao thức mở, tài nguyên phần mềm có thể xác định bằng địa chỉ URL, thực hiện các chức năng và đưa ra các thông tin người dùng yêu cầu, các ứng dụng độ lập và tự mô tả chính nó.
* Nền tảng cơ bản của web service là XML và HTTP. Bất cứ một ứng dụng nào cũng đều có thể có một thành phần Web service. Web service có thể được tạo ra bằng bất kỳ một ngôn ngữ lập trình nào.
* Một Web Service đơn thuần là một API được bao bọc trong HTTP. API không phải lúc nào cũng cần dựa trên web. Một API bao gồm một bộ đầy đủ các quy tắc và thông số kỹ thuật cho một chương trình phần mềm làm theo để tạo điều kiện tương tác..
* *Cách Thức hoạt động :*
* XML (eXtensible Markup Language): Là ngôn ngữ đánh dấu mở rộng do W3C đề nghị với mục đích tạo ra các ngôn ngữ đánh dấu khác. XML cung cấp một ngôn ngữ mà có thể được sử dụng giữa ngôn ngữ lập trình và các nền tảng khác. Đồng thời, nó còn có thể được dùng để mô tả những thông điệp và chức năng phức tạp. Do web service là sự kết hợp của nhiều thành phần khác nhau, do đó web serices sử dụng các tính năng và đặc trưng ủa các thành phần này để giao tiếp với nhau. Vì vậy XML là một công cụ chính yếu để giải quyết vấn đề này. Web service tận dụng khả năng giải quyết vấn đề của các ứng dụng lớn trên các hệ điều hành khác nhau cho chúng giao tiếp với nhau.
* HTTP (HyperText Transfer Protocol): Là giao thức truyền tải siêu văn bản, bao gồm:
* URLs (Uniform Resource Locators) dùng để tham chiếu tới tài nguyên trên Internet
* Header: Mô tả dữ liệu, chứa các thông tin của thiết bị thực hiện Request
* Body: Phần thân chứa dữ liệu
* Method: GET (Lấy dữ liệu)/ POST (Gửi dữ liệu)/ PUT (Cập nhật dữ liệu)/ DELETE (Xóa dữ liệu)
* Nền tảng Web service cơ bản là XML HTTP. Tất cả các web service chuẩn đều hoạt động bằng các thành phần sau:
* SOAP (là viết tắt của Simple Object Access Protocol) – giao thức truy cập đối tượng đơn giản: SOAP là một giao thức dựa trên XML đơn giản cho phép các ứng dụng trao đổi thông tin qua HTTP.
* UDDI (Universal Description, Discovery and Integration): UDDI là một tiêu chuẩn dựa trên XML để mô tả, xuất bản và tìm kiếm các dịch vụ web.
* WSDL (Web Services Description Language) – ngôn ngữ định nghĩa web service: WSDL là một ngôn ngữ dựa trên XML để mô tả các dịch vụ web và cách truy cập chúng.



Hình 1.4: Mô hình hoạt động của các thành phần

* *Đặc điểm :*
* Sử dụng Internet để kết nối các thành phần (FE, BE, BD) hoặc các thiết bị khác nhau (máy tính, điện thoại, …) với nhau, các thành phần kết nối không phụ thuộc vào vị trí máy tính được đặt.
* Cho phép client và server tương tác ngay cả trong môi trường khác nhau. (Ví dụ server chạy linux, client chạy windows).
* Phần lớn được xây dựng dựa trên mã nguồn mở và phát triển các chuẩn đã được công nhận. (Ví dụ XML).
* Có thể triển khai bởi một phần mềm ứng dùng phía server
  1. **MonggooDB:**
* *NoSQL:*
* NoSQL là 1 dạng CSDL mã nguồn mở và được viết tắt bởi: None-Relational SQL hay có nơi thường gọi là Not-Only SQL.
* NoSQL được phát triển trên Javascript Framework với kiểu dữ liệu là JSON và dạng dữ liệu theo kiểu key và value.
* NoSQL ra đời như là 1 mảnh vá cho những khuyết điểm và thiếu xót cũng như hạn chế của mô hình dữ liệu quan hệ RDBMS (Relational Database Management System - Hệ quản trị cơ sở dữ liệu quan hệ) về tốc độ, tính năng, khả năng mở rộng, ...
* Với NoSQL bạn có thể mở rộng dữ liệu mà không lo tới những việc như tạo khóa ngoại, khóa chính, kiểm tra ràng buộc
* NoSQL bỏ qua tính toàn vẹn của dữ liệu và transaction để đổi lấy hiệu suất nhanh và khả năng mở rộng.
* *Ưu điểm của MonggooDB:*
* Do MongoDB sử dụng lưu trữ dữ liệu dưới dạng Document JSON nên mỗi một collection sẽ có các kích cỡ và các document khác nhau, linh hoạt trong việc lưu trữ dữ liệu.
* Dữ liệu trong MongoDB không có sự ràng buộc lẫn nhau, không có join như trong RDBMS nên khi insert, xóa hay update nó không cần phải mất thời gian kiểm tra xem có thỏa mãn các ràng buộc dữ liệu như trong RDBMS.
* MongoDB rất dễ mở rộng (Horizontal Scalability). Trong MongoDB có một khái niệm cluster là cụm các node chứa dữ liệu giao tiếp với nhau, khi muốn mở rộng hệ thống ta chỉ cần thêm một node với vào cluster.
* Trường dữ liệu “\_id” luôn được tự động đánh index (chỉ mục) để tốc độ truy vấn thông tin đạt hiệu suất cao nhất.
* Khi có một truy vấn dữ liệu, bản ghi được cached lên bộ nhớ Ram, để phục vụ lượt truy vấn sau diễn ra nhanh hơn mà không cần phải đọc từ ổ cứng.
* Hiệu năng cao: Tốc độ truy vấn (find, update, insert, delete) của MongoDB nhanh hơn hẳn so với các hệ quản trị cơ sở dữ liệu quan hệ (RDBMS). Với một lượng dữ liệu đủ lớn thì thử nghiệm cho thấy tốc độ insert của MongoDB có thể nhanh tới gấp 100 lần so với MySQL.
* *Nhược điểm của MonggooDB:*
* Một ưu điểm của MongoDB cũng chính là nhược điểm của nó. MongoDB không có các tính chất ràng buộc như trong RDBMS nên khi thao tác với MongoDB cần phải chú ý.
* Tốn bộ nhớ do dữ liệu lưu dưới dạng key-value, các collection chỉ khác về value do đó key sẽ bị lặp lại. Không hỗ trợ join nên dễ bị dữ thừa dữ liệu.
* Khi insert/update/remove bản ghi, MongoDB sẽ chưa cập nhật ngay xuống ổ cứng, mà sau 60 giây MongoDB mới thực hiện ghi toàn bộ dữ liệu thay đổi từ RAM xuống ổ cứng điều này sẽ là nhược điểm vì sẽ có nguy cơ bị mất dữ liệu khi xảy ra các tình huống như mất điện...

1. **PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG:**
2. **Khảo sát hệ thống:**
   1. **Khảo sát thực trạng:**

Cửa hàng Cheapstore cung cấp các sản phẩm điện thoại và phụ kiện chất lượng cho người dùng, và đang có chiến lược mở rộng sang hình thức kinh doanh online. Cửa hàng phục vụ nhiều loại sản phẩm điện thoại và phụ kiện phong cách mới lạ, với nhiều người tiêu dùng khác nhau, bên cạnh đó cửa hàng còn có nhiều hình thức quan trọng là buôn bán, giao dịch... Vì vậy, cần có một hệ thống phần mềm chuyên nghiệp để giảm thiểu các rủi ro, lưu trữ lượng lớn dữ liệu và dễ dàng trong việc quản lý.

Người quản lý cần quản lý các thông tin về các loại sản phẩm của cửa hàng, và các sản phẩm cụ thể của cửa hàng đang bày bán như điện thoại Oppo điện thoại XiaoMi,…. Lưu các thông tin về sản phẩm như tên sản phẩm, loại sản phẩm, giá, hình ảnh minh họa... để người dùng lựa chọn theo các tiêu chí của bản thân.

Quản lý cũng cần lưu lại các thông tin menu và bài viết của cửa hàng để quảng cáo cho các thông tin sản phẩm mà cửa hàng đang đẩy mạnh. Có các trang quảng cáo cho những sản phẩm nổi bật, được khách hàng sử dụng nhiều, mang lại doanh thu lớn cho cửa hàng, và tiếp cận nhiều hơn tới khách hàng mới.

Khi có thắc mắc hay phản hồi về sản phẩm đã sử dụng, khách hàng liên hệ với cửa hàng. Khi khách hàng mua hàng của cửa hàng, các thông tin cơ bản của khách hàng như họ tên, số điện thoại, địa chỉ… sẽ được lưu lại để tiện cho việc liên lạc, giao hàng, cũng như thực hiện tri ân khách hàng nhân các dịp lễ; và lưu các thông tin giao dịch như các sản phẩm khách hàng mua, số lượng, giá tiền, giảm giá và tổng tiền….

Khách hàng sau khi sử dụng sản phẩm, có thể phản hồi các ý kiến tới cửa hàng, để cửa hàng nắm bắt được chất lượng các sản phẩm và nhu cầu người dùng của mình.

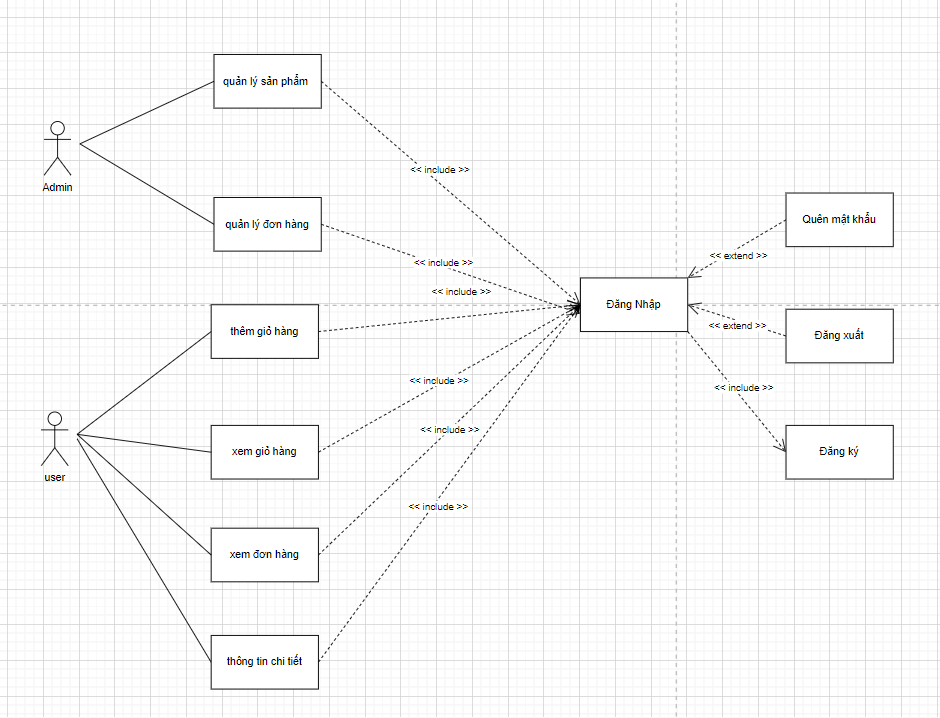
* 1. **Yêu cầu của hệ thống:**

Từ những khảo sát ở trên, có thể rút ra các yêu cầu mà hệ thống mới xây dựng cần đáp ứng:

* Hệ thống phân quyền chức năng với 2 nhóm người: người quản trị và khách hàng.
* Có các chức năng đăng ký, xem sản phẩm, xem chi tiết các sản phẩm, quản lý đơn hàng và thông tin cá nhân… với khách hàng.
* Đặc quyền quản lý các chức năng: quản lý thành viên, danh mục, sản phẩm, đơn đặt hàng,… với các người quản trị viên của hệ thống.
* Giao diện hệ thống thân thiện, hài hòa với người dùng.
* Giao diện dễ sử dụng với quản trị viên, dễ bảo trì.

1. **UseCase của hệ thống:**

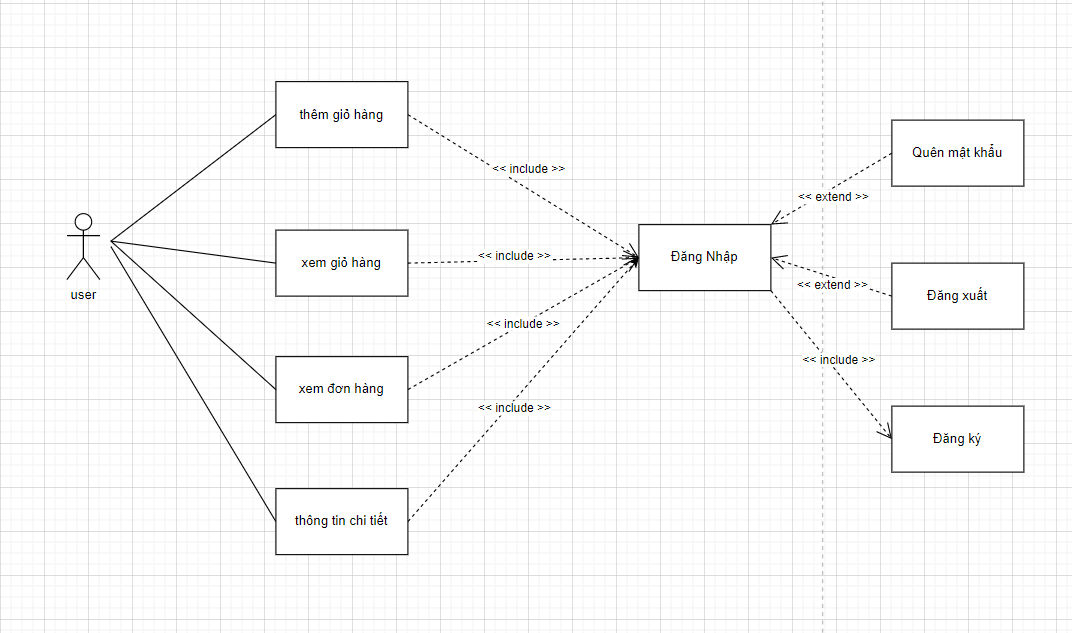
### 2.1. UseCase hệ thống:

**

### *Hình 3.1. UseCase hệ thống*

* ***Đăng ký:***
  + Tác nhân: Khách hàng.
  + Mô tả tổng quát: Khách hàng truy cập website có thể đăng ký thành viên, xem các sản phẩm và chi tiết sản phẩm, tìm kiếm các sản phẩm mong muốn.
  + Điều kiện đầu vào: Khi khách hàng truy cập với nhu cầu xem các sản phẩm.
  + Dòng sự kiện chính:
    - Khách hàng chọn đăng ký.
    - Nhập các trường dữ liệu, và chọn đăng ký.
    - Hệ thống kiểm tra các trường dữ liệu:
      * Nếu đúng thì lưu tài khoản của khách hàng.
      * Nếu sai thực hiện luồng sự kiện phụ A1.
  + Dòng sự kiện rẽ nhánh A1:
    - Hệ thống thông báo việc nhập dữ liệu không hợp lệ.
    - Nhập lại thông tin.
    - Quay lại trang đăng ký.
* Điều kiện đầu ra: Lưu tài khoản khách hàng đã đăng ký.
* ***Đăng nhập:***
* Tác nhân: Người quản lý và khách hàng.
* Mô tả tổng quát: Người quản lý và khách hàng đăng nhập vào hệ thống.
* Điều kiện đầu vào: UseCase sử dụng khi người dùng muốn đăng nhập vào hệ thống.
* Dòng sự kiện chính:
* Hệ thống yêu cầu nhập tài khoản và mật khẩu.
* Người dùng nhập tài khoản và mật khẩu.
* Hệ thống kiểm tra tên và mật khẩu vừa nhập:
  + - Nếu đúng sẽ kiểm tra quyền, và hiển thị quyền tương ứng của tài khoản.
    - Nếu sai thực hiện luồng sự kiện phụ A1.
* Dòng sự kiện phụ A1:
* Hệ thống sẽ thông báo việc nhập dữ liệu không hợp lệ.
* Người dùng nhập lại tài khoản và mật khẩu.
* Quay lại bước 3 của luồng sự kiện chính, hoặc hủy bỏ việc đăng nhập, khi đó ca sử dụng kết thúc.
* ***Đăng xuất:***
* Tác nhân: Người quản lý và khách hàng.
* Mô tả tổng quát: Người quản lý và khách hàng muốn thoát tài khoản khỏi hệ thống.
* Điều kiện đầu vào: UseCase sử dụng khi người quản trị và thành viên chọn đăng xuất.
* Dòng sự kiện chính:
* Hệ thống hủy việc lưu thông tin đăng nhập.
* Quay về màn hình đăng nhập.

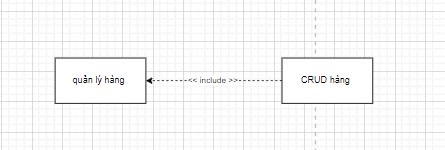
### 2.2. UseCase của Customer:

**

### *Hình 3.2. UseCase của Customer*

* ***Xem chi tiết sản phẩm:***
* Tác nhân: Khách hàng.
* Mô tả tổng quát: Khách hàng muốn xem chi tiết một sản phẩm.
* Điều kiện đầu vào: UseCase sử dụng khi khách hàng chọn một sản phẩm.
* Dòng sự kiện chính:
* Hệ thống lấy thông tin chi tiết của sản phẩm theo mã.
* Hiển thị form chi tiết sản phẩm.
* ***Đặt hàng:***
* Tác nhân: Khách hàng.
* Mô tả tổng quát: Các khách hàng muốn đặt mua sản phẩm của cửa hàng.
* Điều kiện đầu vào: UseCase sử dụng khi bắt đầu đăng nhập thành công hệ thống.
* Dòng sự kiện chính:
* Ca sử dụng bắt đầu khi khách hàng đăng nhập vào hệ thống.
* Chọn giỏ hàng.
* Hệ thống hiển thị chức năng: Cập nhật số lượng trong giỏ hàng, xóa sản phẩm trong giỏ và thanh toán đơn hàng:
  + - Cập nhật số lượng trong giỏ hàng:
      * Người dùng chọn số lượng sản phẩm mong muốn.
      * Chọn cập nhật.
      * Hệ thống kiểm tra số lượng sản phẩm còn lại trong cửa hàng:
        + Nếu còn hàng sẽ cập nhật số lượng trong giỏ hàng.
        + Nếu sai, thực hiện luồng sự kiện rẽ nhánh A1.
    - Xóa sản phẩm trong giỏ hàng:
      * Người dùng chọn sản phẩm muốn xóa.
      * Chọn xóa sản phẩm.
      * Hệ thống xóa sản phẩm khỏi giỏ hàng của khách hàng.
    - Thanh toán đơn hàng:
      * Khách hàng chọn thanh toán các sản phẩm trong giỏ hàng.
      * Hệ thống hiển thị form thanh toán.
      * Khách hàng nhập các thông tin cần thiết, và chọn đặt hàng.
      * Hệ thống kiểm tra các trường thông tin:
        + Nếu thỏa mãn thực hiện cập nhật thông tin đơn hàng và chi tiết vào cơ sở dữ liệu.
        + Nếu không thì yêu cầu nhập lại.
* Dòng sự kiện rẽ nhánh A1:
* Hệ thống báo số lượng sản phẩm không đủ.
* Quay lại bước 2 của dòng sự kiện chính.
* ***Quản lý đơn hàng cá nhân:***
* Tác nhân: Khách hàng.
* Mô tả tổng quát: Khách hàng có thể xem lại các đơn hàng mình đã đặt tại cửa hàng.
* Điều kiện đầu vào: Khách hàng đăng nhập thành công hệ thống, chọn mục các đơn hàng.
* Dòng sự kiện chính:
* Người dùng đăng nhập hệ thống.
* Truy cập hệ thống chọn mục các đơn hàng.
* Hệ thống hiển thị danh sách các đơn hàng, cùng với các chức năng người dùng có thể thực hiện: Xóa đơn hàng và xem chi tiết đơn hàng:
  + - Xóa đơn hàng:
      * Chọn đơn hàng muốn Xóa.
      * Chọn xóa đơn hàng.
      * Kiểm tra đơn hàng:
        + Nếu đơn hàng chưa giao dịch, thì thực hiện xóa đơn hàng.
        + Nếu đơn hàng đã có giao dịch, thực hiện luồng sự kiện rẽ nhánh A1.
    - Xem chi tiết đơn hàng:
      * Chọn đơn hàng muốn xem chi tiết.
      * Chọn xem chi tiết đơn hàng.
      * Hệ thống chuyển sang form chi tiết đơn hàng.
* Dòng sự kiện rẽ nhánh A1:
* Thông báo lỗi.
* Chuyển tới bước 3 của dòng sự kiện chính.
* Điều kiện đầu ra: Các thông tin về đơn hàng được cập nhật vào cơ sở dữ liệu.
* ***Cập nhật thông tin cá nhân:***
  + Tác nhân: Khách hàng.
  + Mô tả tổng quát: Khách hàng cập nhật thông tin cá nhân của mình.
  + Điều kiện đầu vào: Khi khách hàng đăng nhập hệ thống thành công và chọn thông tin cá nhân.
    - Khách hàng chọn cập nhật thông tin cá nhân.
    - Hệ thống hiển thị form cập nhật thông tin.
    - Khách hàng nhập các thông tin muốn sửa đổi, và chọn cập nhật.
    - Hệ thống kiểm tra các trường dữ liệu:
* Nếu đầy đủ và đúng định dạng thì chuyển qua bước tiếp theo.
* Nếu không thì thực hiện dòng sự kiện rẽ nhánh A1.
* Hệ thống thông báo cập nhật thông tin cá nhân thành công.
* Hệ thống cập nhật thông tin cá nhân vào cơ sở dữ liệu.
  + Dòng sự kiện rẽ nhánh A1:
    - Hệ thống thông báo việc nhập dữ liệu không hợp lệ.
    - Nhập lại thông tin.
    - Quay lại bước 4 của sự kiện chính.
* Điều kiện đầu ra: Thông tin của khách hàng được cập nhật vào cơ sở dữ liệu.

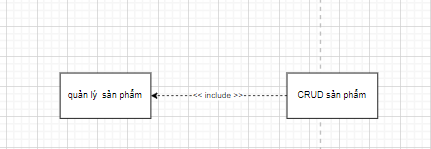
### 2.3. UseCase Quản lý hàng:

******

### *Hình 3.3. UseCase Quản lý hàng*

* Tác nhân: Người quản trị.
* Mô tả tổng quát: Người quản trị quản lý các danh mục, giúp quản lý các sản phẩm dễ dàng.
* Điều kiện đầu vào: Người dùng đăng nhập thành công, với quyền quản trị.
* Dòng sự kiện chính:
* Quản trị đăng nhập hệ thống.
* Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, và chọn quản lý danh mục.
* Hệ thống hiển thị danh sách danh mục, và các chức năng: thêm, sửa và xóa và tìm kiếm danh mục:
  + - Thêm danh mục:
      * Hệ thống hiển thị form nhập thông tin danh mục.
      * Người quản lý nhập thông tin danh mục.
      * Nhập nút thêm danh mục.
      * Hệ thống kiểm tra các trường dữ liệu:
        + Nếu thỏa mãn, thì thực hiện bước tiếp theo.
        + Nếu không, thì thực hiện luồng sự kiện A1.
      * Hệ thống thông báo thêm danh mục thành công.
      * Hệ thống lưu lại thông tin danh mục.
    - Sửa danh mục:
      * Hệ thống hiển thị form cập nhật thông tin danh mục.
      * Người quản lý nhập thông tin cần thay đổi.
      * Nhập nút cập nhật danh mục.
      * Hệ thống kiểm tra các trường dữ liệu:
        + Nếu thỏa mãn, thì thực hiện bước tiếp theo.
        + Nếu không, thì thực hiện luồng sự kiện A1.
      * Hệ thống thông báo cập nhật danh mục thành công.
      * Hệ thống cập nhật thông tin danh mục.
    - Xóa danh mục:
      * Người quản lý chọn thông tin danh mục cần xóa.
      * Hệ thống kiểm tra:
        + Nếu không có ràng buộc, thực hiện bước tiếp theo.
        + Ngược lại, hủy xóa.
* Dòng sự kiện rẽ nhánh A1:
* Hệ thống báo lỗi.
* Quản trị thưc hiện nhập lại dữ liệu.
* Quay lại bước 3 của sự kiện phụ.
* Điều kiện đầu ra: Danh sách các danh mục được cập nhật đầy đủ vào cơ sở dữ liệu.

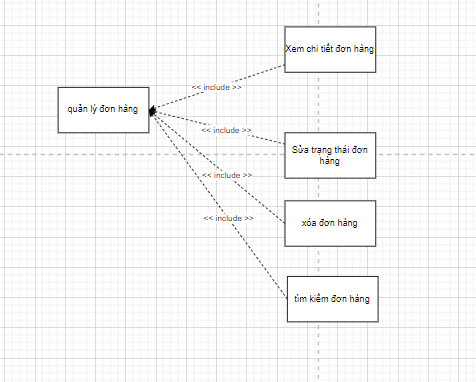
2.4. UseCase Quản lý sản phẩm:



### *Hình 3.4. UseCase Quản lý sản phẩm*

* Tác nhân: Người quản trị.
* Mô tả tổng quát: Người quản trị quản lý các sản phẩm của cửa hàng.
* Điều kiện đầu vào: Người dùng đăng nhập thành công, với quyền quản trị.
* Dòng sự kiện chính:
* Quản trị đăng nhập hệ thống.
* Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, và chọn quản lý sản phẩm.
* Hệ thống hiển thị danh sách sản phẩm và các chức năng: thêm, sửa và xóa và tìm kiếm sản phẩm:
  + - Thêm thông tin sản phẩm:
      * Hệ thống hiển thị form nhập thông tin sản phẩm.
      * Người quản lý nhập thông tin sản phẩm.
      * Nhập nút thêm sản phẩm.
      * Hệ thống kiểm tra các trường dữ liệu:
        + Nếu thỏa mãn, thì thực hiện bước tiếp theo.
        + Nếu không, thì thực hiện luồng sự kiện A1.
      * Hệ thống thông báo thêm sản phẩm thành công.
      * Hệ thống lưu lại thông tin sản phẩm.
    - Sửa thông tin sản phẩm:
      * Hệ thống hiển thị form cập nhật thông tin sản phẩm.
      * Người quản lý nhập thông tin cần thay đổi.
      * Nhập nút cập nhật sản phẩm.
      * Hệ thống kiểm tra các trường dữ liệu:
        + Nếu thỏa mãn, thì thực hiện bước tiếp theo.
        + Nếu không, thì thực hiện luồng sự kiện A1.
      * Hệ thống thông báo cập nhật sản phẩm thành công.
      * Hệ thống cập nhật thông tin sản phẩm.
    - Tìm kiếm sản phẩm:
      * Nhập từ khóa cần tìm.
      * Nhấn nút tìm kiếm.
      * Hệ thống kiểm tra thông tin sản phẩm:
        + Nếu có hiển thị danh sách sản phẩm.
        + Nếu không, thực hiện sự kiện rẽ nhánh A1.
    - Xóa sản phẩm:
      * Người quản lý chọn thông tin sản phẩm cần xóa.
      * Hệ thống kiểm tra:
        + Nếu không có ràng buộc, thực hiện bước tiếp theo.
        + Ngược lại, hủy xóa.
* Dòng sự kiện rẽ nhánh A1:
* Hệ thống báo lỗi.
* Quản trị thưc hiện nhập lại dữ liệu.
* Quay lại bước 3 của sự kiện phụ.
* Điều kiện đầu ra: Danh sách các sản phẩm được cập nhật đầy đủ vào cơ sở dữ liệu.

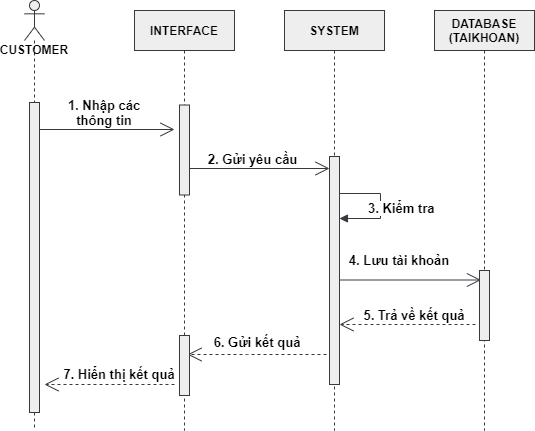
### 2.5. UseCase Quản lý đơn hàng:



### *Hình 3.5. UseCase Quản lý đơn hàng*

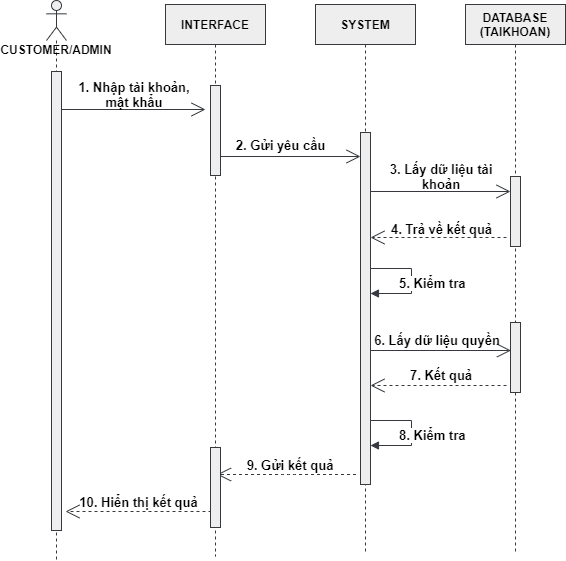
* Tác nhân: Người quản trị.
* Mô tả tổng quát: Người quản trị quản lý các đơn hàng của thành viên.
* Điều kiện đầu vào: Người dùng đăng nhập thành công, với quyền quản trị.
* Dòng sự kiện chính:
* Quản trị đăng nhập hệ thống.
* Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, và chọn quản lý đơn hàng.
* Hệ thống hiển thị danh sách đơn hàng, và các chức năng: xem chi tiết, sửa và xóa và tìm kiếm đơn hàng:
  + - Xem chi tiết đơn hàng:
      * Quản trị chọn đơn hàng muốn xem chi tiết.
      * Chọn xem chi tiết.
      * Hệ thống hiển thị form chi tiết đơn hàng.
    - Sửa đơn hàng:
      * Hệ thống hiển thị form cập nhật thông tin đơn hàng.
      * Người quản lý nhập thông tin cần thay đổi.
      * Nhập nút cập nhật đơn hàng.
      * Hệ thống kiểm tra các trường dữ liệu:
        + Nếu thỏa mãn, thì thực hiện bước tiếp theo.
        + Nếu không, thì thực hiện luồng sự kiện A1.
      * Hệ thống thông báo cập nhật đơn hàng thành công.
      * Hệ thống cập nhật thông tin đơn hàng.
    - Tìm kiếm đơn hàng:
      * Nhập từ khóa cần tìm.
      * Nhấn nút tìm kiếm.
      * Hệ thống kiểm tra thông tin đơn hàng:
        + Nếu có hiển thị danh sách đơn hàng.
        + Nếu không, thực hiện sự kiện rẽ nhánh A1.
    - Xóa đơn hàng:
      * Người quản lý chọn thông tin đơn hàng cần xóa.
      * Hệ thống kiểm tra:
        + Nếu không có ràng buộc, thực hiện bước tiếp theo.
        + Ngược lại, hủy xóa.
* Dòng sự kiện rẽ nhánh A1:
* Hệ thống báo lỗi.
* Quản trị thưc hiện nhập lại dữ liệu.
* Quay lại bước 3 của sự kiện phụ.
* Điều kiện đầu ra: Danh sách các đơn hàng được cập nhật đầy đủ vào cơ sở dữ liệu.

1. **Biểu đồ tuần tự:**
   1. **Biểu đồ tuần tự Đăng ký:**



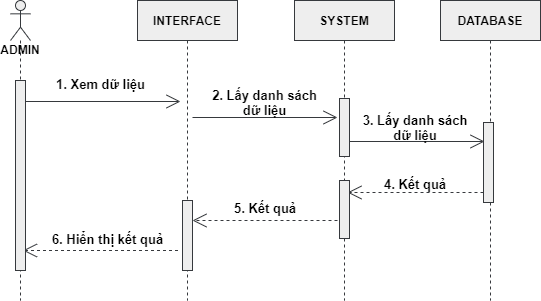
### *Hình 3.12. Biểu đồ tuần tự Đăng ký*

* 1. **Biểu đồ tuần tự Đăng nhập:**



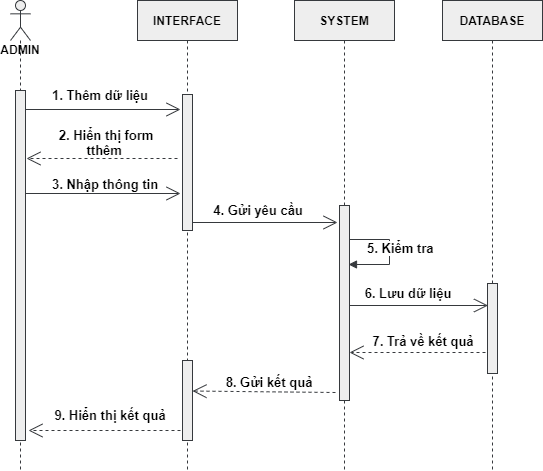
### *Hình 3.13. Biểu đồ tuần tự Đăng nhập*

* 1. **Biểu đồ tuần tự Xem dữ liệu:**



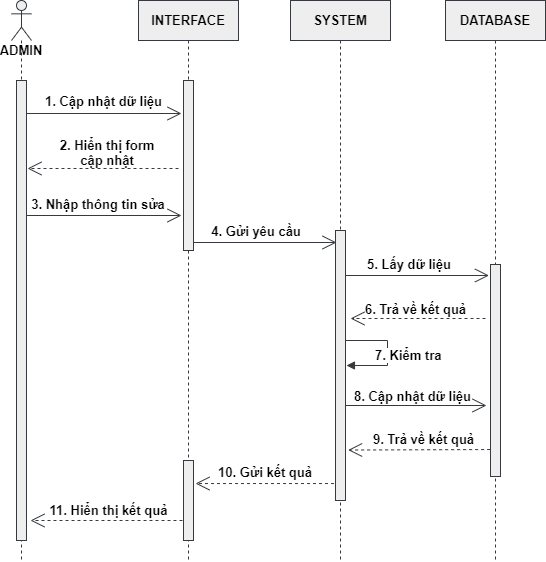
### *Hình 3.14. Biểu đồ tuần tự Xem dữ liệu*

* 1. **Biểu đồ tuần tự Thêm dữ liệu trang quản lý:**



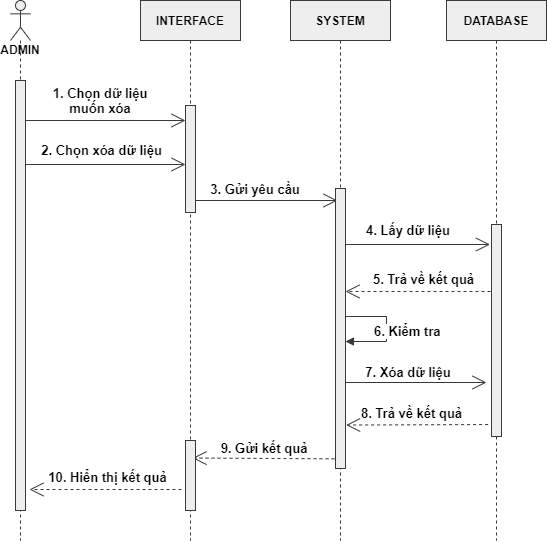
### *Hình 3.15. Biểu đồ tuần tự Thêm dữ liệu trang quản lý*

* 1. **Biểu đồ tuần tự Cập nhật dữ liệu:**



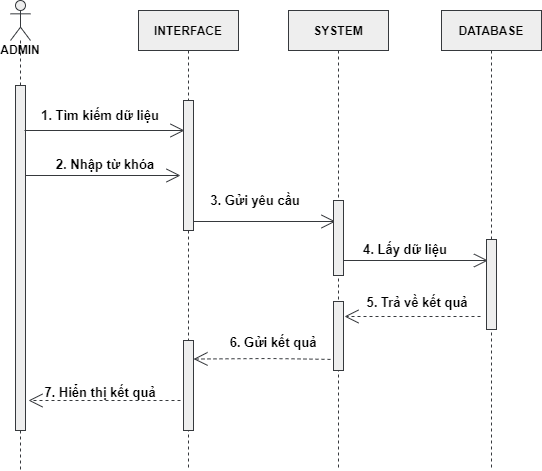
### *Hình 3.16. Biểu đồ tuần tự Cập nhật dữ liệu*

* 1. **Biểu đồ tuần tự Xóa dữ liệu:**



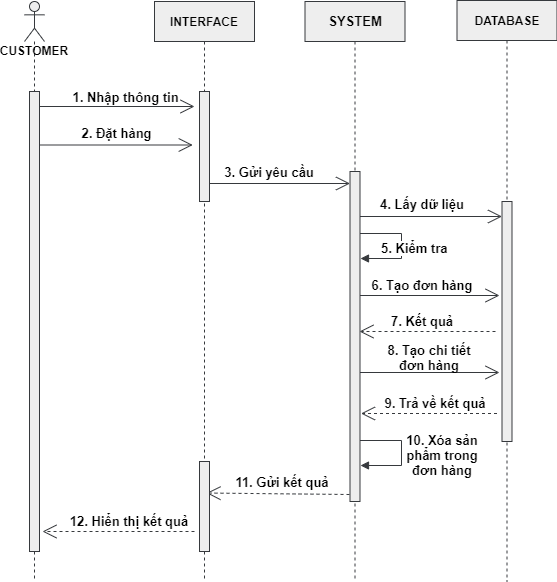
### *Hình 3.17. Biểu đồ tuần tự Xóa dữ liệu*

* 1. **Biểu đồ tuần tự Tìm kiếm dữ liệu:**



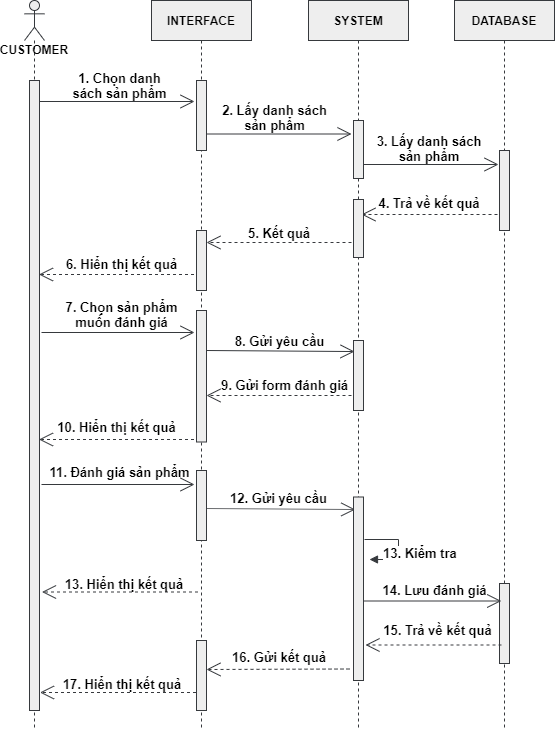
### *Hình 3.18. Biểu đồ tuần tự Tìm kiếm dữ liệu*

* 1. **Biểu đồ tuần tự Đặt hàng:**



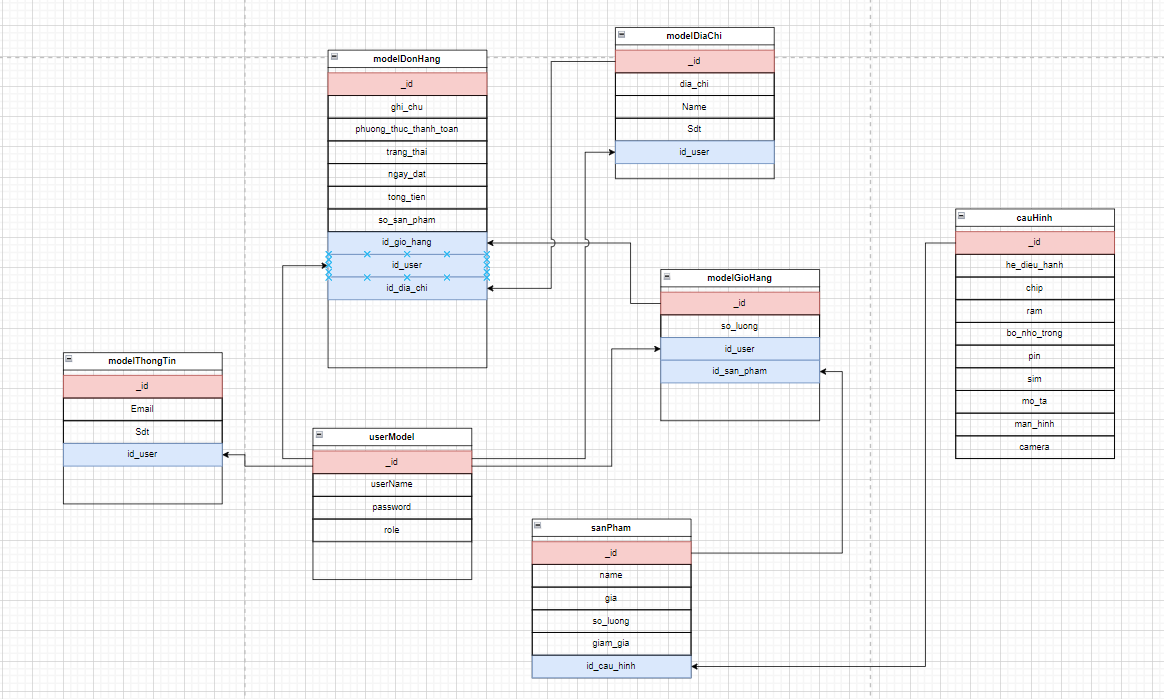
### *Hình 3.19. Biểu đồ tuần tự Đặt hàng*

* 1. **Biểu đồ tuần tự Đánh giá sản phẩm:**



### *Hình 3.20. Biểu đồ tuần tự Đánh giá sản phẩm*

1. **Thiết kế cơ sở dữ liệu:**
   1. **Biểu đồ lớp:**

****

***Hình 3.21. Biểu đồ lớp***

* 1. **Xác định các bảng thực thể:**
* *Bảng userModel:*

| **STT** | **Tên thực thể** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- |
| 1 | \_id | ObjectId | Khóa chính | Khóa chính của bảng |
| 2 | userName | String | Null | Tên đăng nhập |
| 4 | Password | String | Null | Mật khẩu |
| 5 | role | String | Null | Phân quyền |

### *Bảng 3.2. Bảng userModel*

* *Bảng modelThongTin:*

| **STT** | **Tên thực thể** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- |
| 1 | \_id | ObjectId | Khóa chính | Khóa chính của bảng |
| 2 | Email | String | Null | Tên Email |
| 3 | Sdt | Number | Null | Số điện thoại |
| 4 | id\_user | ObjectId | Khóa ngoại | \_id bảng userModel |

### *Bảng 3.3. Bảng product*

* *Bảng cauhinh:*

| **STT** | **Tên thực thể** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- |
| 1 | \_id | ObjectId | Khóa chính | Khóa chính của bảng |
| 2 | He\_dieu\_hanh | String | Null | Hệ điều hành |
| 4 | chip | String | Null | Chip |
| 5 | ram | String | Null | Ram |
| 6 | Bo\_nho\_trong | String | Null | Bộ nhớ trong |
| 7 | pin | String | Null | Pin |
| 8 | sim | String | Null | Sim |
| 9 | Mo\_ta | String | Null | Mô tả |
| 10 | Man\_hinh | String | Null | Màn Hình |
| 11 | Camera | String | Null | Camera |

### *Bảng 3.4. Bảng categories*

* *Bảng sanpham:*

| **STT** | **Tên thực thể** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- |
| 1 | \_id | ObjectId | Khóa chính | Khóa chính của bảng |
| 2 | name | String | null | Tên |
| 3 | gia | Number | Null | Giá |
| 4 | img | Array | null | Ảnh |
| 5 | So\_luong | Number | Null | Số lượng |
| 6 | Giam\_gia | Number | Null | Giảm giá |
| 7 | Id\_cau\_hinh | ObjectId | Khóa ngoại | \_id bảng cauhinh |

### *Bảng 3.5. Bảng keyword*

* *Bảng modelDiaChi:*

| **STT** | **Tên thực thể** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- |
| 1 | \_id | ObjectId | Khóa chính | Khóa chính của bảng |
| 2 | Dia\_chi | String | null | Địa chỉ nhận hàng |
| 3 | Id\_user | ObjectId | Khóa ngoại | \_id bảng userModel |
| 4 | name | String | Null | Tên người nhận |
| 5 | Sdt | Number | Null | Số điện thoại |

### *Bảng 3.6. Bảng menu*

* *Bảng modelDonHang:*

| **STT** | **Tên thực thể** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- |
| 1 | \_Id | ObjectId | Khóa chính | Khóa chính của bảng |
| 2 | Ghi\_chu | String | null | Ghi chú |
| 3 | Phuong\_thuc\_thanh\_toan | String | null | Phương thức thanh toán |
| 4 | Trang\_thai | String | null | Trạng thái đơn hàng |
| 5 | Id\_user | ObjectId | Khóa ngoại | \_id bản userModel |
| 6 | Id\_dia\_chi | ObjectId | Khóa ngoại | \_id bảng modelDiaChi |
| 7 | Id\_gio\_hang | ObjectId | Khóa ngoại | \_id bảng modelGioHang |
| 8 | Ngay\_dat | Date | Null | Ngày đặt |
| 9 | Tong\_tien | Number | Null | Tổng tiền |
| 10 | So\_san\_pham | Number | Null | Tổng số sản phẩm |
| 11 | Name | String | Null | Tên Đơn Hàng |

### *Bảng 3.7. Bảng article*

* *Bảng modelGioHang:*

| **STT** | **Tên thực thể** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- |
| 1 | \_id | ObjectId | Khóa chính | Khóa chính của bảng |
| 2 | Id\_san\_pham | ObjectId | Khóa ngoại | \_id bảng sanpham |
| 3 | Id\_user | ObjectId | Khóa Ngoại | \_id bảng userModel |
| 4 | So\_luong | Number | Null | Số lượng sản phẩm |

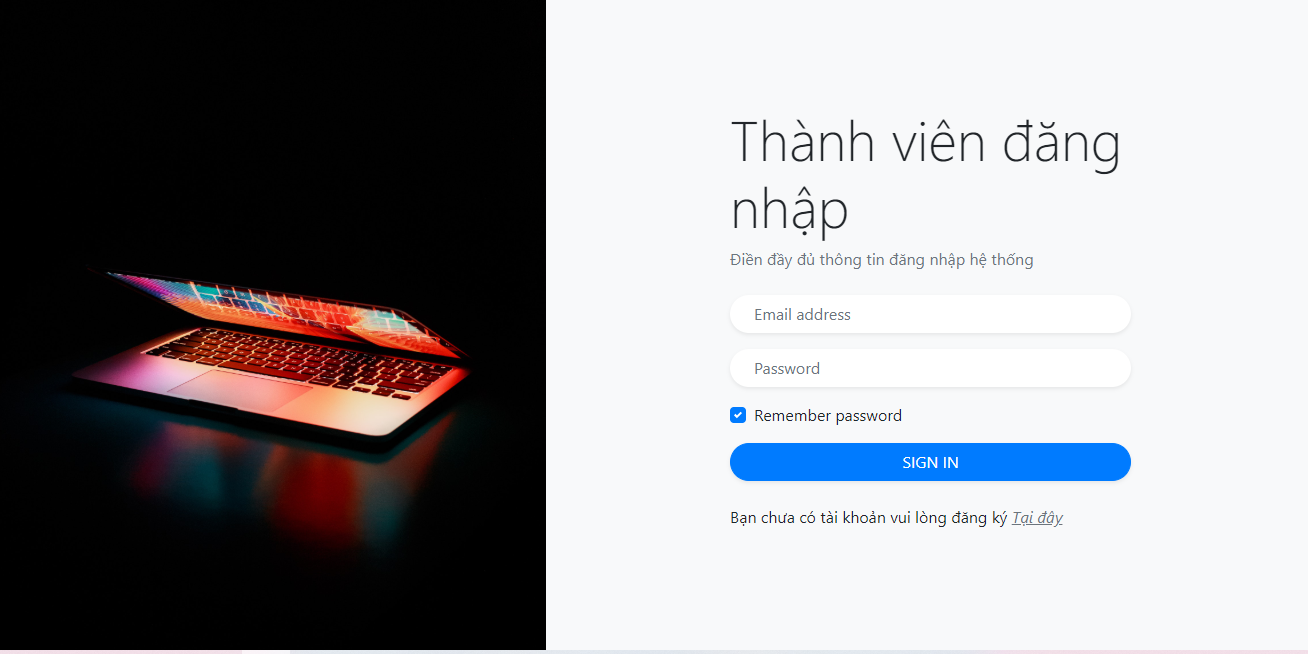
### *Bảng 3.8. Bảng slide*

1. **GIAO DIỆN PHẦN MỀM:**
2. **Giao diện người dùng:**
   1. **Giao diện trang chủ:**



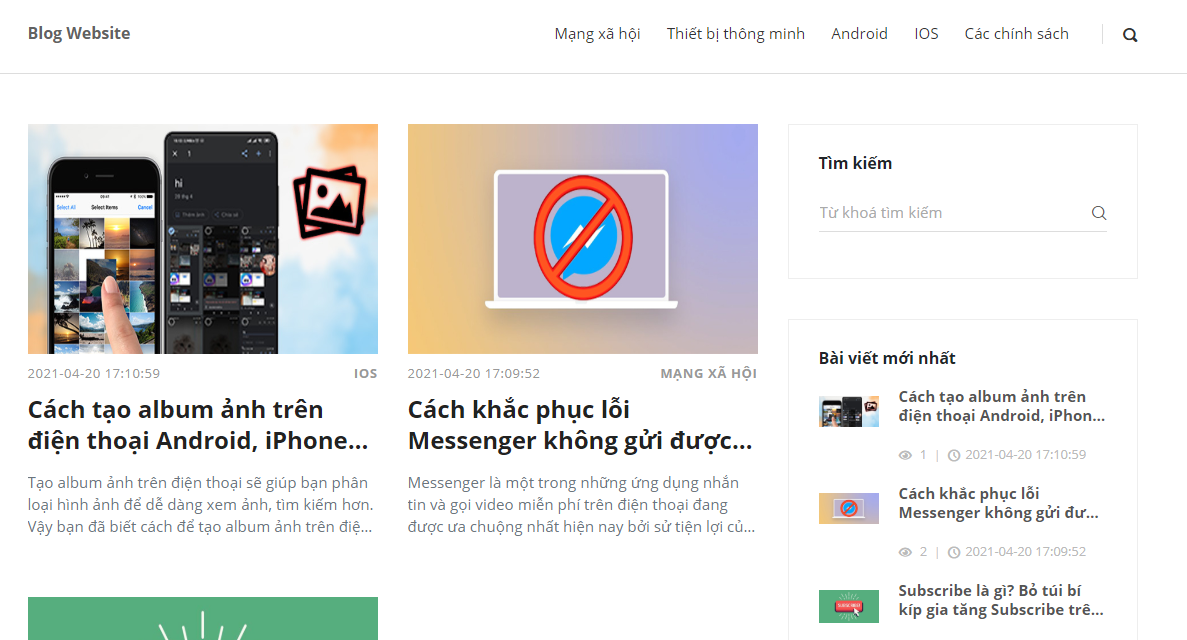
***Hình 4.1. Giao diện trang chủ người dùng***

* 1. **Giao diện trang đăng nhập:**

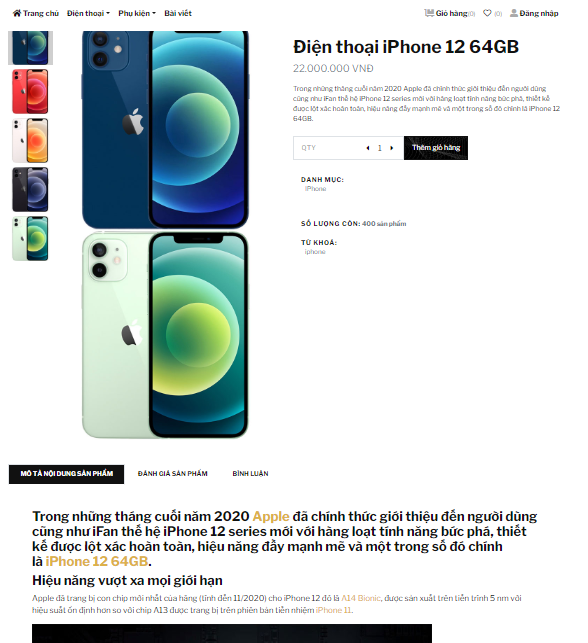


***Hình 4.2. Giao diện trang đăng nhập***

* 1. **Giao diện trang bài viết:**

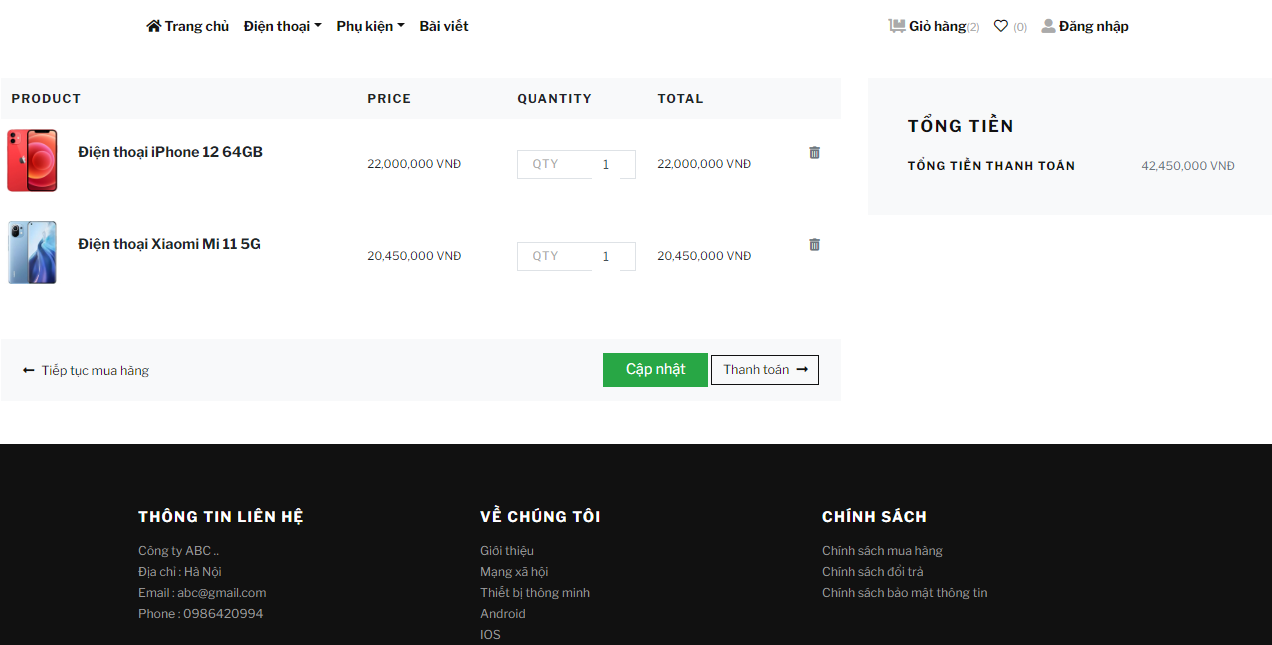


***Hình 4.3. Giao diện trang bài viết***

* 1. **Giao diện trang chi tiết sản phẩm:**

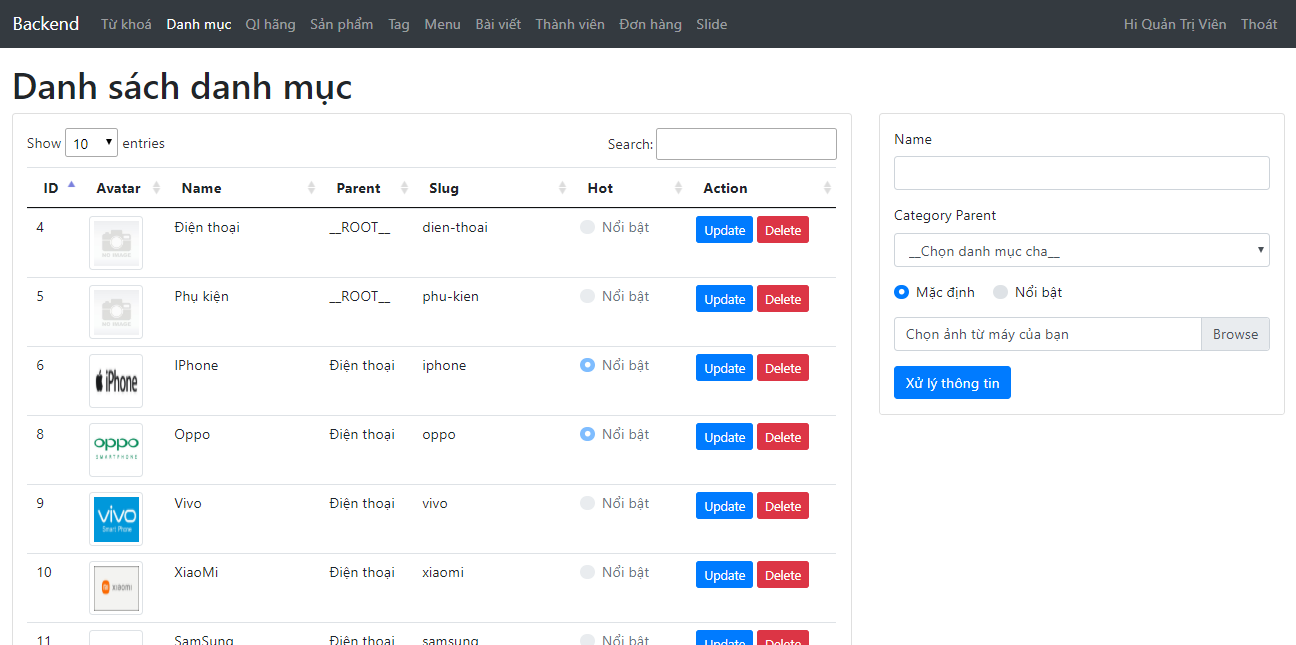
***Hình 4.4. Giao diện trang chi tiết sản phẩm***

* 1. **Giao diện trang giỏ hàng:**



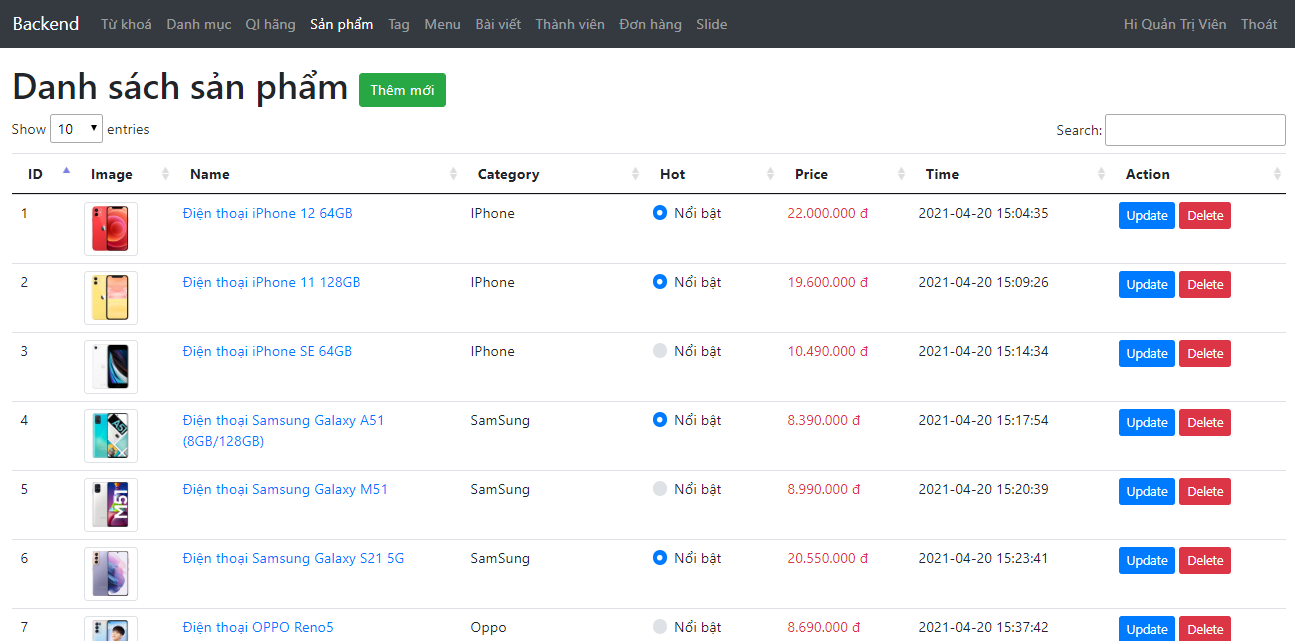
***Hình 4.5. Giao diện trang giỏ hàng***

1. **Giao diện quản lý:**
   1. **Giao diện trang quản lý danh mục:**



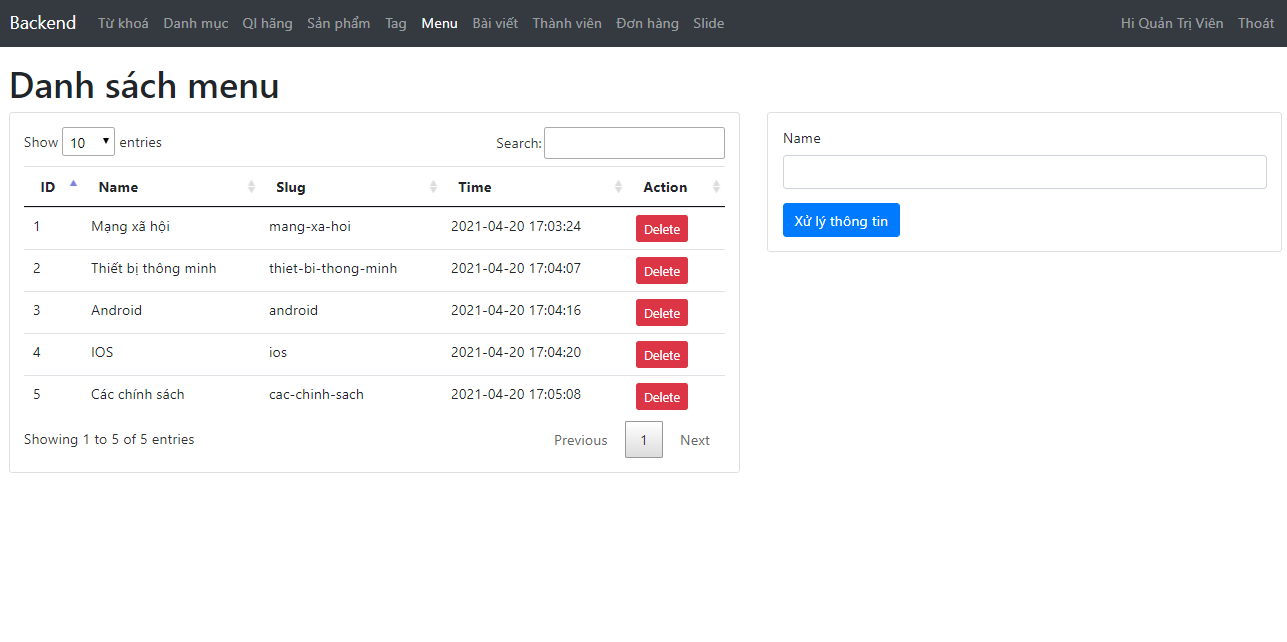
***Hình 4.6. Giao diện trang quản lý danh mục***

* 1. **Giao diện trang quản lý sản phẩm:**



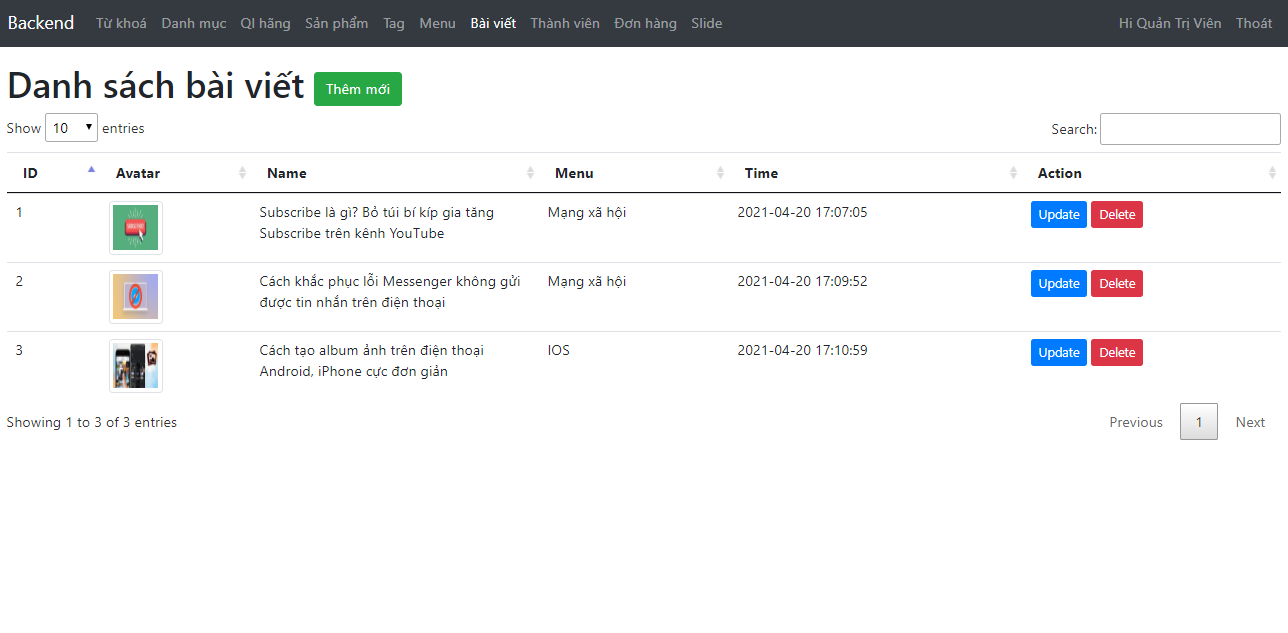
***Hình 4.7. Giao diện trang quản lý sản phẩm***

* 1. **Giao diện trang quản lý menu:**



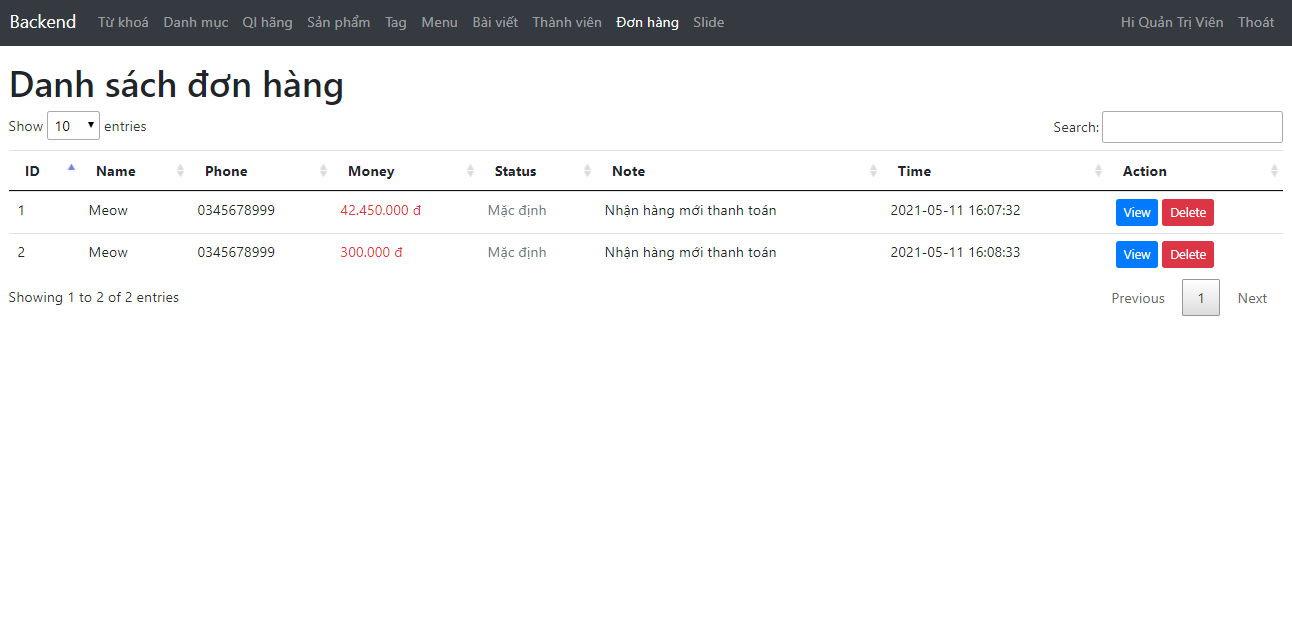
***Hình 4.8. Giao diện trang quản lý menu***

* 1. **Giao diện trang quản lý bài viết:**



***Hình 4.9. Giao diện trang quản lý bài viết***

* 1. **Giao diện quản lý đơn hàng:**



***Hình 4.10. Giao diện trang quản lý đơn hàng***

1. **KẾT LUẬN:**
   * + 1. **Kết quả đạt được:**

Từ những hướng dẫn tận tình của thầy, cùng những gì đã học được ở trường, em đã phân tích và xây dựng phần mềm với các kết quả đạt được:

* Áp dụng các kiến thức đã học xây dựng website bán mĩ phẩm, phân quyền người dùng và người quản trị, với giao diện và chức năng khác nhau.
* Quản lý các thông tin cần thiết của cửa hàng như các sản phẩm, thông tin khách hàng và các giao dịch mua hàng cửa khách hàng.
* Có hầu như đầy đủ chức năng cơ bản để quản lý cửa hàng.
  + - 1. **Hạn chế:**

Bên cạnh đó cũng có những hạn chế như:

* Chưa có nhiều kiến thức thực tế, nên việc xây dựng phần mềm áp dụng vào thực tế còn nhiều thiếu sót và chưa khả quan.
* Giao diện chưa thật sự bắt mắt.
* Một số chức năng còn hạn chế.
  + - 1. **Kết luận:**

Bài báo cáo về đề tài “xây dựng website bán điện thoại di động và phụ kiện điện thoại” được xây dựng trên mô hình MVC với framework laravel kết hợp với hệ quản trị cơ sở dữ liệu MySQL. Với các chức năng và cơ sở dữ liệu đã được nghiên cứu, khảo sát từ thực tế, ứng dụng trên mô hình cửa hàng kinh doanh mĩ phẩm, giúp cải thiện những khó khăn cho người quản lý, cũng như thuận tiện cho người dùng.

# **TÀI LIỆU THAM KHẢO:**

1. Giáo trình Thông tin di động – tác giả Nguyễn Phạm Anh Dũng
2. Bài giảng truyền dẫn vô tuyến số - tác giả Nguyễn Viết Đảm
3. Bài giảng Tiến trình phát triển hệ thống thông tin di động từ 1G lên 4G – tác giả Nguyễn Phạm Anh Dũng
4. Các trang web:
5. <https://reactjs.org/>
6. <https://mui.com/>
7. <https://nodejs.org/en/>
8. <http://www.thongtincongnghe.com/article/2745>
9. Cùng 1 số nguồn tài liệu các đồ án, chuyên đề viễn thông trên internet khác.