

**TRƯỜNG ĐẠI HỌC SƯ PHẠM**

**KHOA TIN HỌC**

**🖎🕮✍**



**BÁO CÁO ĐỒ ÁN MÔN HỌC**

**ĐỀ TÀI**

**ỨNG DỤNG LARAVEL, MONGODB ĐỂ XÂY DỰNG WEBSITE BÁN HÀNG ĐIỆN TỬ**

**Giảng viên hướng dẫn:** **Võ Công Đình**

**Sinh viên thực hiện:** **Nhóm 5**

Đà Nẵng, 5/2022

# **BẢNG PHÂN CÔNG VIỆC**

|  |  |
| --- | --- |
| **Tên thành viên** | **Công việc** |
| **Trần Đăng Long** | Thiết kế cơ sở dữ liệu, xây dựng giao diện, xây dựng các chức năng trang chủ, chi tiết sản phẩm, thêm, sửa, xóa sản phẩm của user |
| **Võ Duy Hiền** | Thiết kế cơ sở dữ liệu, xây dựng giao diện, xây dựng các chức năng quản lý tài khoản thành viên hoặc admin, sửa xóa thông tin tài khoản, lịch sử mua hàng user |
| **Đồng Vũ Tiến Nghĩa** | Thiết kế cơ sở dữ liệu, xây dựng giao diện, xây dựng các chức năng tìm kiếm, thêm sản phẩm vào giỏ hàng, xử lý giỏ hàng, thanh toán |
| **Lê Nguyễn Anh Khoa** | Thiết kế cơ sở dữ liệu, xây dựng giao diện, xây dựng các chức năng: cập nhật profile admin, xóa sản phẩm ở phía admin, lịch sử mua hàng của admin |
| **Hồ Duy Mạnh** | Thiết kế cơ sở dữ liệu, xây dựng giao diện, xây dựng các chức năng đăng nhập và đăng kí tài khoản khách hàng |
| **Nguyễn Hữu Sang** | Thiết kế cơ sở dữ liệu, xây dựng giao diện, xây dựng các chức năng: thêm, sửa, xóa, : Blog,comments |
| **Trần Mạnh Cường** | Thiết kế cơ sở dữ liệu, xây dựng giao diện, xây dựng các chức năng : thêm, sửa, xóa : Brand, Category, Country |

*Đà Nẵng, ngày … tháng … năm …*

**Trưởng nhóm**

# **BẢNG TIẾN ĐỘ**

|  |  |
| --- | --- |
| **Thời gian thực hiện** | **Nội dung thực hiện** |
| 3/4/2022 - 8/4/2022 | Chọn đề tài, phân tích yêu cầu về phần chức năng và cách thực hiện |
| 8/4/2022 - 15/4/2022 | Phân tích yêu cầu về phần chức năng và thiết kế cơ sở dữ liệu |
| 15/4/2022 - 22/4/2022 | Tìm hiểu về framework Laravel để sử dụng cho việc lập trình các chức năng |
| 22/4/2022 - 24/4/2022 | Kết nối Laravel và Monggodb, thiết kế giao diện cho website |
| 24/4/2022 - 14/5/2022 | Thiết kế và Tạo lập các chức năng cho website |
| 14/5/2022 - 21/5/2022 | Tổng hợp lại sau đó tiến hành sửa những lỗi đang gặp phải |
| 14/5/2022 - 28/5/2022 | Viết báo cáo và chuẩn bị slide báo cáo cho buổi bảo vệ vào ngày 4/6/2022 |

**NHẬN XÉT CỦA CÁN BỘ HƯỚNG DẪN**

*Đà Nẵng, ngày … tháng … năm …*

**Cán bộ hướng dẫn**

**NHẬN XÉT CỦA HỘI ĐỒNG PHẢN BIỆN**

*Đà Nẵng, ngày … tháng … năm …*

**Hội đồng phản biện**

# **LỜI CAM ĐOAN**

Nhóm em xin cam đoan đây là kết quả sau quá trình nghiên cứu đề tài dưới sự hướng dẫn của Thầy giáo Võ Công Đình. Những nhận định được nêu ra trong đề tài cũng là kết quả từ sự nghiên cứu trực tiếp, nghiêm túc, dựa vào các cơ sở tìm kiếm, hiểu biết và nghiên cứu các tài liệu khoa học hay bản dịch khác đã được công bố và mọi sự tham khảo đều được trích dẫn, chỉ rõ nguồn tham khảo theo đúng quy định.

Nhóm thực hiện

**Nhóm 5**

**LỜI CẢM ƠN**

Để đề tài này đạt kết quả tốt đẹp, nhóm em đã nhận được sự hỗ trợ, giúp đỡ của các cơ quan, tổ chức, cá nhân. Với tình cảm sâu sắc, chân thành, cho phép nhóm em được bày tỏ lòng biết ơn sâu sắc đến tất cả các cá nhân và cơ quan đã tạo điều kiện giúp đỡ nhóm em trong quá trình học tập và nghiên cứu đề tài này.

Trước hết nhóm em xin gửi tới các Quý Thầy/Cô trong khoa Tin học Trường Đại học Sư phạm – Đại học Đà Nẵng lời chào trân trọng, lời chúc sức khỏe và lời cảm ơn sâu sắc. Với sự quan tâm, dạy dỗ, chỉ bảo tận tình chu đáo của thầy cô thời gian vừa qua mà nhóm em mới trang bị được những kiến thức chuyên ngành để có thể thực hiện tiếp chặng đường học tập, vận dụng và sáng tạo ra những sản phẩm hữu ích góp phần phục vụ các lĩnh vực khác nhau, đến nay nhóm em đã có thể hoàn thành bài báo cáo Đồ án chuyên nghành cho đề tài: "Ứng dụng PHP, Bootstrap vào xây dựng website kinh doanh đồ ăn nhanh".

Đặc biệt em xin gửi lời cảm ơn chân thành nhất tới Thầy giáo Võ Công Đình đã luôn quan tâm giúp đỡ, hướng dẫn tận tình để cho nhóm em hoàn thành tốt Đồ án chuyên ngành này trong suốt quá trình thực hiện từ lúc vừa mới bắt đầu lên ý tưởng cho tới ngày hôm nay.

Mặc dù nhóm em đã rất cố gắng nhưng với điều kiện thời gian cũng như kinh nghiệm còn hạn chế của sinh viên, Đồ án chuyên ngành này không thể tránh được những thiếu sót. nhóm Em rất mong nhận được sự chỉ bảo, đóng góp ý kiến của các thầy cô để nhóm em có điều kiện bổ sung, hoàn thiện, nâng cao ý thức của mình để phục vụ tốt hơn cho những môn học hay là công việc sau này.

Nhóm em xin chân thành cảm ơn!

Nhóm thực hiện

**Nhóm 5**

# **MỤC LỤC**

[BẢNG PHÂN CÔNG VIỆC 2](#_Toc104936218)

[BẢNG TIẾN ĐỘ 3](#_Toc104936219)

[LỜI CAM ĐOAN 6](#_Toc104936220)

[LỜI CẢM ƠN 7](#_Toc104936221)

[MỤC LỤC 8](#_Toc104936222)

[DANH MỤC BẢNG BIỂU 11](#_Toc104936223)

[DANH MỤC HÌNH VẼ 12](#_Toc104936224)

[LỜI MỞ ĐẦU 13](#_Toc104936225)

[**1. Đặt vấn đề 13**](#_Toc104936226)

[**2. Mục tiêu nghiên cứu 13**](#_Toc104936227)

[**3. Đối tượng và phạm vi nghiên cứu 14**](#_Toc104936228)

[**4. Phương pháp nghiên cứu 14**](#_Toc104936229)

[**5. Nhiệm vụ và kết quả đạt được 14**](#_Toc104936230)

[**6. Bố cục của đồ án 14**](#_Toc104936231)

[**7. Đóng góp của đồ án 15**](#_Toc104936232)

[CHƯƠNG I: CƠ SỞ LÝ THUYẾT 16](#_Toc104936233)

[**1.1 Khái niệm cơ bản về Internet 16**](#_Toc104936234)

[**1.1.1 Giới thiệu chung 16**](#_Toc104936235)

[**1.1.2 Phân loại 16**](#_Toc104936236)

[**1.1.3 World Wide Web 16**](#_Toc104936237)

[**1.2 Ngôn ngữ sử dụng 19**](#_Toc104936238)

[**1.2.1 Ngôn ngữ lập trình PHP 19**](#_Toc104936239)

[**1.2.2 Ngôn ngữ lập trình JavaScript 24**](#_Toc104936240)

[**1.2.3 Tổng quan về Laravel Framework 26**](#_Toc104936241)

[**1.3 Hệ quản trị cơ sở dữ liệu MongoDB 28**](#_Toc104936242)

[**1.3.1 Tổng quan về hệ quản trị cơ sở dữ liệu MongoDB 28**](#_Toc104936243)

[**1.3.2 Mô hình dữ liệu 33**](#_Toc104936244)

[**1.3.3 Thiết kế mô hình dữ liệu 34**](#_Toc104936245)

[**1.3.4 Chỉ mục (Index) 36**](#_Toc104936246)

[**1.3.5 Mô hình truy vấn 38**](#_Toc104936247)

[CHƯƠNG II: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 41](#_Toc104936248)

[**2.1 Phân tích yêu cầu đề tài 41**](#_Toc104936253)

[**2.1.1 Phát biểu bài toán 41**](#_Toc104936254)

[**2.1.2 Yêu cầu bài toán 42**](#_Toc104936255)

[**2.2 Nghiên cứu hiện trạng 43**](#_Toc104936256)

[**2.3 Đặc tả hệ thống 43**](#_Toc104936257)

[**2.4 Sơ đồ use-case 43**](#_Toc104936258)

[**2.4.1 Sơ đồ use-case của khách hàng 44**](#_Toc104936259)

[**2.4.2 Sơ đồ use-case của khách hàng – thành viên 44**](#_Toc104936260)

[**2.4.3 Sơ đồ use-case của người quản lí 45**](#_Toc104936261)

[**2.5 Biểu đồ phân rã chức năng 45**](#_Toc104936262)

[**2.6 Biểu đồ tuần tự 46**](#_Toc104936263)

[**2.6.1 Biểu đồ tuần tự đăng nhập 46**](#_Toc104936264)

[**2.6.2 Biểu đồ tuần tự đăng kí thành viên 47**](#_Toc104936265)

[**2.6.3 Biểu đồ tuần tự tìm kiếm sản phẩm 47**](#_Toc104936266)

[**2.6.4 Biểu đồ tuần tự đặt hàng 48**](#_Toc104936267)

[**2.6.5 Biểu đồ tuần tự xem thông tin sản phẩm 48**](#_Toc104936268)

[**2.6.6 Biểu đồ tuần tự thay đổi thông tin cá nhân 49**](#_Toc104936269)

[**2.6.7 Biểu đồ tuần tự quản lí khách hàng 49**](#_Toc104936270)

[**2.7 Thiết kế cơ sở dữ liệu 50**](#_Toc104936271)

[**2.7.1 Mô hình cơ sở dữ liệu 54**](#_Toc104936272)

[CHƯƠNG III: CÀI ĐẶT CHƯƠNG TRÌNH VÀ KẾT QUẢ 54](#_Toc104936273)

[**3.1. Kết quả chương trình 54**](#_Toc104936274)

[**3.1.1. Giao diện trang chủ 54**](#_Toc104936275)

[**3.1.2. Giao diện chi tiết sản phẩm 55**](#_Toc104936276)

[**3.1.3. Giao diện đăng nhập 56**](#_Toc104936277)

[**3.1.4. Giao diện quản lý tài khoản 57**](#_Toc104936278)

[**3.1.5. Giao diện quản lý của Admin 57**](#_Toc104936279)

[**3.1.6. Giao diện giỏ hàng 58**](#_Toc104936280)

[**3.1.7. Giao diện quản lí sản phẩm 58**](#_Toc104936281)

[**3.1.8. Giao diện thêm sản phẩm 59**](#_Toc104936282)

[**3.1.9. Giao diện chỉnh sửa thông tin sản phẩm 59**](#_Toc104936283)

[**3.1.10. Giao diện quản lí danh mục sản phẩm 60**](#_Toc104936284)

[**3.1.11. Giao diện quản lí thương hiệu sản phẩm 60**](#_Toc104936285)

[**3.1.12. Giao diện quản lí thành viên 61**](#_Toc104936286)

[KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 61](#_Toc104936287)

[**1. Kết quả đạt được 61**](#_Toc104936288)

[**2. Hạn chế của đề tài 62**](#_Toc104936289)

[**3. Hướng phát triển 62**](#_Toc104936290)

[TÀI LIỆU THAM KHẢO 63](#_Toc104936291)

**DANH MỤC BẢNG BIỂU**

[Bảng 1: Đặc tả hệ thống 43](#_Toc104936144)

[Bảng 2 : Đặc tả dữ liệu blog 50](#_Toc104936145)

[Bảng 3: Thực thể danh mục sản phẩm 50](#_Toc104936146)

[Bảng 4: Thực thể thương hiệu sản phẩm 51](#_Toc104936147)

[Bảng 5 Thực thể giỏ hàng 51](#_Toc104936148)

[Bảng 6: Bảng thực thể user 52](#_Toc104936149)

[Bảng 7 : Thực thể migratons 52](#_Toc104936150)

[Bảng 8: Lịch sử mua hàng 52](#_Toc104936151)

[Bảng 9: Thực thể bảng comment 53](#_Toc104936152)

[Bảng 10: Thực thể bảng Country 53](#_Toc104936153)

[Bảng 11: Thực thế bảng Blog 53](#_Toc104936154)

**DANH MỤC HÌNH VẼ**

[Hình 1: Cách hoạt động của trang web tĩnh 16](file:///C:\Users\Admin\Downloads\Trần_Đăng_Long-19CNTT2-ĐACN%20(1).docx#_Toc92790943)

[Hình 2: Cách hoạt động của trang web động 17](#_Toc92790944)

[Hình 3: Sơ đồ Use-case của khách hàng 31](#_Toc92790945)

[Hình 4: Sơ đồ Use-case của khách hàng-thành viên 31](#_Toc92790946)

[Hình 5: Sơ đồ Usse-case của người quản lí 32](#_Toc92790947)

[Hình 6: Biểu đồ phân cấp chức năng hệ thống 33](#_Toc92790948)

[Hình 7: Biêu đồ tuần tự đăng nhập 33](#_Toc92790949)

[Hình 8: Biểu đồ tuần tự đăng kí thành viên 34](#_Toc92790950)

[Hình 9: Biểu đồ tuần tự tìm kiếm sản phẩm 34](#_Toc92790951)

[Hình 10: Biểu đồ tuần tự đặt hàng 35](#_Toc92790952)

[Hình 11: Biểu đồ tuần tự xem thông tin sản phẩm 35](#_Toc92790953)

[Hình 12: Biểu đồ tuần tự thay đổi thông tin cá nhân 36](#_Toc92790954)

[Hình 13: Biểu đồ tuần tự quản lí khách hàng 36](#_Toc92790955)

[Hình 14: Sơ đồ quan hệ Cơ sở dữ liệu 39](file:///C:\Users\Admin\Downloads\Trần_Đăng_Long-19CNTT2-ĐACN%20(1).docx#_Toc92790956)

[Hình 15: Giao diện trang chủ 40](file:///C:\Users\Admin\Downloads\Trần_Đăng_Long-19CNTT2-ĐACN%20(1).docx#_Toc92790957)

[Hình 16: Giao diện chi tiết sản phẩm 41](file:///C:\Users\Admin\Downloads\Trần_Đăng_Long-19CNTT2-ĐACN%20(1).docx#_Toc92790958)

[Hình 17: Giao diện đăng nhập 41](file:///C:\Users\Admin\Downloads\Trần_Đăng_Long-19CNTT2-ĐACN%20(1).docx#_Toc92790959)

[Hình 18: Giao diện đăng kí 42](file:///C:\Users\Admin\Downloads\Trần_Đăng_Long-19CNTT2-ĐACN%20(1).docx#_Toc92790960)

[Hình 19: Giao diện quản lí tài khoản 43](#_Toc92790961)

[Hình 20: Giao diện quản lí Admin 43](file:///C:\Users\Admin\Downloads\Trần_Đăng_Long-19CNTT2-ĐACN%20(1).docx#_Toc92790962)

[Hình 21: Giao diện giỏ hàng 44](file:///C:\Users\Admin\Downloads\Trần_Đăng_Long-19CNTT2-ĐACN%20(1).docx#_Toc92790963)

[Hình 22: Giao diện quản lí sản phẩm 45](file:///C:\Users\Admin\Downloads\Trần_Đăng_Long-19CNTT2-ĐACN%20(1).docx#_Toc92790964)

[Hình 23: Giao diện thêm sản phẩm 45](file:///C:\Users\Admin\Downloads\Trần_Đăng_Long-19CNTT2-ĐACN%20(1).docx#_Toc92790965)

[Hình 24: Giao diện chỉnh sửa thông tin sản phẩm 45](file:///C:\Users\Admin\Downloads\Trần_Đăng_Long-19CNTT2-ĐACN%20(1).docx#_Toc92790966)

[Hình 25: Giao diện thêm danh mục sản phẩm 46](file:///C:\Users\Admin\Downloads\Trần_Đăng_Long-19CNTT2-ĐACN%20(1).docx#_Toc92790967)

[Hình 26: Giao diện quản lí danh mục sản phẩm 46](file:///C:\Users\Admin\Downloads\Trần_Đăng_Long-19CNTT2-ĐACN%20(1).docx#_Toc92790968)

[Hình 27: Giao diện thêm thương hiệu sản phẩm 46](file:///C:\Users\Admin\Downloads\Trần_Đăng_Long-19CNTT2-ĐACN%20(1).docx#_Toc92790969)

[Hình 28: Giao diện quản lí thương hiệu sản phẩm 47](file:///C:\Users\Admin\Downloads\Trần_Đăng_Long-19CNTT2-ĐACN%20(1).docx#_Toc92790970)

[Hình 29: Giao diện quản lí thành viên 47](file:///C:\Users\Admin\Downloads\Trần_Đăng_Long-19CNTT2-ĐACN%20(1).docx#_Toc92790971)

**LỜI MỞ ĐẦU**

1. **Đặt vấn đề**

Ngày nay, chúng ta đang sống trong một xã hội hiện đại với công nghệ phát triển như vũ bão thì việc đưa nó vào trong đời sống không còn quá xa lạ. Ứng dụng công nghệ thông tin và việc tin học hóa được xem là một trong những yếu tố mang tính quyết định trong hoạt động của các chính phủ, tổ chức cũng như của các công ty. Nó đóng vai trò hết sức quan trọng có thể tạo ra những bước đột phá mạnh mẽ. Là một phần của công nghệ thông tin, công nghệ web đang có được sự phát triển mạnh mẽ và phổ biến rất nhanh bởi những lợi ích mà nó mang lại cho cộng đồng là rất lớn.

Hiện nay, hầu hết tại các cửa hàng điện tử đã sử dụng rộng rãi ứng dụng công nghệ thông tin, giúp người dùng dễ dàng đặt mua đồ ăn, dễ dàng quản lí thông tin cá nhân,… Bên cạnh đó, sự hỗ trợ mạnh mẽ của các ngôn ngữ lập trình, cùng với sự ra đời của các framework đã thúc đẩy em chọn đề tài **“Ứng dụng Laravel và MongoDB để xây dựng website bán hàng điện tử”** để thực hiện đồ án chuyên ngành nhằm đáp ứng theo chương trình đào tạo đồng thời mong muốn ứng dụng công nghệ đó vào việc tìm hiểu và đưa ra một giải pháp tốt nhằm giải quyết công việc. Với đề tài này, website sẽ cho phép người sử dụng thực hiện một cách nhanh chóng, chính xác việc tìm kiếm thông tin, các sản phẩm, giá cả của chúng và tiến hành việc đặt mua trực tuyến, … với giao diện làm việc thân thiện, tiện dụng với người sử dụng hệ thống, người tiêu dùng.

Được sự hướng dẫn tận tình của Thầy giáo Võ Công Đình, em đã chọn đề tài **“Ứng dụng Laravel, MongoDB để xây dựng website bán hàng điện tử”** sử dụng ngôn ngữ Php, JavaScript, Laravel framework và hệ quản trị cơ sở dữ liệu NoSQL-MongoDB làm đề tài cho học phần này.

1. **Mục tiêu nghiên cứu**

Mục tiêu của đồ án là tìm ra giải pháp hiệu quả để giải quyết công việc trực tuyến cho các cửa hàng. Các mục tiêu cụ thể của đồ án gồm:

* Tìm ra phương pháp hiệu quả trong việc đặt đồ ăn nhanh trực tuyến cho khách hàng và việc quản lí thông tin cho người quản lí
* Ứng dụng NoSQL-Mongo DB và xây dựng website bán hàng điện tử.

1. **Đối tượng và phạm vi nghiên cứu**

*Đối tượng nghiên cứu gồm:*

* Các khách hàng của cửa hàng điện tử
* Vai trò của người quản lí.
* Các món ăn của cửa hàng.
* Các chi nhánh của cửa hàng.
* Giờ hoạt động ứng với các chi nhánh.

*Giới hạn phạm vi nghiên cứu trong đồ án gồm:*

* Tập trung xây dựng hệ thống giúp người dùng đặt đồ ăn trực tuyến và quản lí thông tin của các mặt hang, khách hàng là thành viên, chủ cửa hàng với vai trò người quản lí.
* Đề tài được xây dựng bởi các ngôn ngữ lập trình như: PHP, JavaScript, HTML, CSS và sử dụng hệ quản trị cơ sở dữ liệu MongoDB.

1. **Phương pháp nghiên cứu**

* Phương pháp tài liệu: Nghiên cứu các tài liệu liên quan đến các nội dung nghiên cứu: Tài liệu về các ngôn ngôn ngữ lập trình, framework Laravel, các nhu cầu của người dùng, tài liệu về thông tin các thiết bị, khung giờ mở cửa.
* Phương pháp thực nghiệm: Nghiên cứu, xây dựng các phương pháp đặt hàng và quản lí hiệu quả cho cửa hàng. Phát triển phần mềm kinh doanh thiết bị điện tử cho cửa hàng và đánh giá kết quả.

1. **Nhiệm vụ và kết quả đạt được**

* Nghiên cứu và tìm hiểu một số phần mềm kinh doanh thiết bị điện tử trực tuyến.
* Phân tích yêu cầu và đề xuất của phần mềm.
* Thiết kế đặc tả hệ thống.
* Xây dựng được cơ sở dữ liệu.
* Lập trình các module của hệ thống.

1. **Bố cục của đồ án**

Trên cơ sở các nội dung nghiên cứu, để đạt mục tiêu đề ra và đảm bảo tính logic, ngoài phần *mở đầu* và phần *kết luận*, đồ án được tổ chức thành các chương như sau:

*Chương 1. Cơ sở lý thuyết*. Chương này trình bày các cơ sở lí thuyết về internet và website; trình bày kết quả nghiên cứu, tìm hiểu về các ngôn ngữ lập trình: PHP, Javascript và Laravel được sử dụng trong việc xây dựng đề tài.

*Chương 2: Phân tích và thiết kế hệ thống.* Chương này giới thiệu bài toán và đưa ra các mô hình hệ thống; đề xuất các ý tưởng, chức năng cho phần mềm; trình bày các mô hình thực thể quan hệ, sơ đồ phân cấp chức năng. Phân tích chi tiết các chức năng của người sử dụng và người quản lí; thiết kế giao diện người dùng và thiết dữ liệu, chức năng cho hệ thống.

*Chương 3: Cài đặt chương trình và kết quả.* Trong chương này, nêu kết quả thực hiện được của đề tài và trình bày cách cài đặt chương trình.

1. **Đóng góp của đồ án**

Trong đồ án chuyên ngành này, em đã vận dụng linh hoạt những kiến thức đã được học trên giảng đường để thực hiện đề tài. Bên cạnh đó, em cũng tự tìm tòi, học hỏi, nghiên cứu thêm các kiến thức mới, ngôn ngữ lập trình mới nhằm xây dựng đồ án ngày một nhanh chóng, hoàn thiện hơn. Những đóng góp chính của đồ án:

* *Phân tích rõ ràng vai trò của người trong hệ thống.* Đối với người quản lí, khách hàng đã là thành viên hay chưa là thành viên đều được giới hạn một số chức năng nhất định.
* *Đưa ra các chức năng điều khiển dành cho người quản lí.* Nghiên cứu và triển khai thực nghiệm các chức năng nhằm đáp ứng nhu cầu cho người quản lí như quản lí các thông tin mặt hàng, thông tin các thành viên của cửa hàng.
* *Đưa ra chức năng đặt hàng cho khách hàng đã là thành viên.* Nghiên cứu và triển khai thực nghiệm chức năng đặt hàng trực tuyến với đầy đủ các thông tin liên quan như danh sách thiết bị, khung giờ mở cửa, giá cả hay nắm được thông tin các mặt hàng đã và chưa hết để lựa cho mình thích hợp nhất.

# **CHƯƠNG I: CƠ SỞ LÝ THUYẾT**

* 1. **Khái niệm cơ bản về Internet**
     1. **Giới thiệu chung**

Internet là hệ thống thông tin toàn cầu có thể truy nhập công cộng gồm các mạng máy tính được liên kết với nhau. Nguồn gốc của Internet bắt nguồn từ sự phát triển của chuyển mạch gói và nghiên cứu của bộ quốc phòng Hoa Kỳ ủy quyền thực hiện vào những năm 1960 để cho phép chia sẻ thời gian của máy tính [1].

Vào mạng Internet, bạn có tìm thấy các tài liệu để nghiên cứu, học tập, trao đổi thư từ, mặt hàng, mua bán, … Một trong những mục tiêu của Internet là chia sẻ thông tin giữa những người dùng trên nhiều lĩnh vực khác nhau.

* + 1. **Phân loại**

**Các máy tính trên mạng được chia làm 2 nhóm Client và Server.**

* Client: Máy khách trên mạng, chứa chương trình Client.
* Server: Máy phục vụ-máy chủ. Chứa các chương trình Server, tài nguyên (tập tin, tài liệu, …) dùng chung cho nhiều máy khách. Server luôn ở trạng thái chờ yêu cầu và đáp ứng yêu cầu của Client

**Các dịch vụ thường dùng trong Internet:**

* Dịch vụ World Wide Web (gọi tắt Web)
* Dịch vụ Electronic Mail (gọi tắt là Mail)
* Dịch vụ FTP

Để truyền thông với những máy tính khác, mọi máy tính trên Internet phải hỗ trợ giao thức chung TCP/IP (Transmission Control Protocol/Internet Protocol - giao thức điều khiển truyền dữ liệu và giao thức Internet), là một giao thức đồng bộ, cho truyền thông điệp từ nhiều nguồn và tới nhiều đích khác nhau. Ví dụ có thể vừa lấy thư từ trong hộp thư, đồng thời vừa truy cập vào trang web. TCP đảm bảo tính an toàn dữ liệu, IP là giao thức chi phối phương thức vận chuyển dữ liệu trên Internet [1].

* + 1. **World Wide Web**

**Các khái niệm cơ bản về World Wide Web**

World Wide Web (WWW) hay còn gọi là Web là một dịch vu phổ biến nhất hiện nay trên Internet, 85% các giao dịch trên Internet ước lượng thuộc về WWW. Ngày nay số Website trên thế giới đã đạt tới con số khổng lồ. WWW cho bạn truy xuất thông tin văn bản, hình ảnh, âm thanh, video trên toàn thế giới. Thông qua Website, các quý công ty có thể giảm thiểu tối đa chi phí in ấn và phân phát tài liệu cho khách hàng ở nhiều nơi [2].

**Cách tạo trang web**

*Có nhiều cách để tạo trang Web:*

* Có thể tạo trang web trên bất kì chương trình xử lý văn bản nào.
* Tạo web bằng cách viết mã nguồn bởi một trình soạn thảo văn bản như:
* Notpad, Wordpad, v.v. là những chương trình soạn thảo có sẵn trong window.
* Thiết kế bằng các phần mềm chuyên nghiệp như: Fronpage, Dreamweaver, Visual Studio…

Để xây dựng một ứng dụng Web hoàn chỉnh và có tính thương mại, bạn cần kết hợp cả Client Script (Kịch bản trình khách) và Server Script (Kịch bản trình chủ) với một loại cơ sở dữ liệu nào đó, chẳng hạn như: MS Access, SQL Server, Oracle, ở đây em dùng MYSQL.

Khi bạn muốn triển khai ứng dụng Web trên mạng, ngoài các điều kiện về cấu hình phần cứng, bạn cần có trình chủ web thường gọi là Web Server.

**Trình duyệt web (Web Client hay Web Browser)**

Trình duyệt web là công cụ truy xuất dữ liệu trên mạng, là phần mềm giao diện trực tiếp với người sử dụng. Nhiệm vụ của Web Browser là nhận dữ liệu từ người dùng, gửi các yêu cầu đó qua mạng tới các Web Server và nhận các dữ liệu cần thiết từ Server để hiển thị lên màn hình. Để sử dụng dịch vụ WWW, Client cần có một chương trình duyệt Web, kết nối vào Internet thông qua một ISP. Các trình duyệt thông dụng hiện nay là Microsoft Internet Explorer, Mozilla Firefox, Google Chrome, … [2]

**WebServer**

WebServer: Là một máy tính được nối vào Internet và chạy các phần mềm được thiết kế. WebServer đóng vai trò một chương trình xử lý các nhiệm vụ xác định, như tìm trang thích hợp, xử lý tổ hợp dữ liệu, kiểm tra dữ liệu hợp lệ,… WebServer cũng là nơi lưu trữ cơ sở dữ liệu, là phần mềm đảm nhiệm vai trò Server cung cấp dịch vụ Web [2].

**WebServer hỗ trợ các công nghệ khác nhau:**

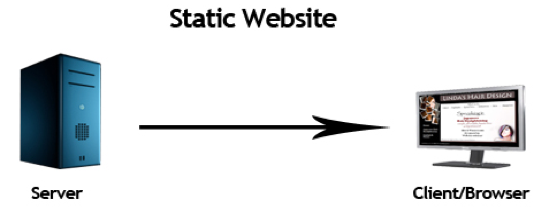
* IIS (Internet Information Server): Hỗ trợ ASP, mở rộng hỗ trợ PHP.
* Apache: Hỗ trợ PHP.
* Tomcat: Hỗ trợ JPS (Java Servlet Page).

**Phân loại Website**

* + - * **Website tĩnh**
* Tài liệu được phân phát rất đơn giản từ hệ thống file Server**.**
* Định dạng các trang web tĩnh là các siêu liên kết, các trang định dạng Text, các hình ảnh đơn giản.

**Ưu điểm:** CSDL nhỏ nên việc phân phát dữ liệu có hiệu quả rõ ràng, Server có thể đáp ứng nhu cầu Client một cách nhanh chóng. Ta nên sử dụng Web tĩnh khi không thay đổi thông tin trên đó [3].

**Nhược điểm:** không đáp ứng được yêu cầu phức tạp của người sử dụng, không linh hoạt [3].

Hoạt động của trang web tĩnh được thể hiện như sau:

Hình 1: Cách hoạt động của trang web tĩnh

* + - * **Webite động**

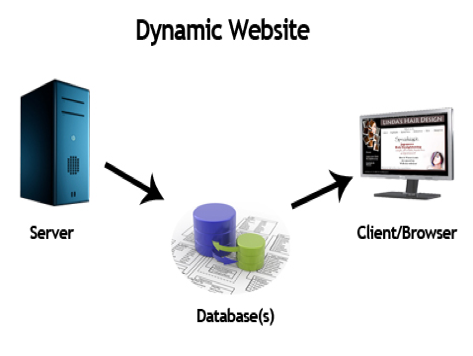
Website động là tập hợp của những trang web mà có nội dung có khả năng thay đổi. Sự thay đổi có thể là tùy theo thời gian, tùy theo người dùng, tùy theo ngữ cảnh. Nội dung của website động được lấy từ cơ sở dữ liệu hoặc hệ thống quản lý nội dung (CMS). Do đó, khi bạn cập nhật nội dung của cơ sở dữ liệu thì nội dung của trang web cũng được cập nhật theo [3].

Để tạo được website động, chúng ta thường sử dụng đến các ngôn ngữ phía server (server-side), chẳng hạn như Servlet, JSP, PHP, Python, C#... [3].

Một website động thường được đặt trên một máy chủ dịch vụ web (Web Server). Website động sử dụng kịch bản lập trình phía máy khách (Client) hoặc máy chủ (Server) hoặc cả hai để tạo ra nội dung động [3].

Khi người dùng sử dụng máy tính của mình thao tác trên trang web sẽ tạo ra một yêu cầu (request) tới máy chủ. Phía máy chủ nhận và xử lý yêu cầu máy khách dựa vào kịch bản trên máy chủ để tạo ra một trang đơn giản và gửi lại tới người dùng. Trình duyệt trên máy khách tải xuống và xử lý mã trong trang để hiển thị thông tin cho người dùng [3].

Hoạt động của Web động được thể hiện như sau:



Hình 2: Cách hoạt động của trang web động

* 1. **Ngôn ngữ sử dụng**
     1. **Ngôn ngữ lập trình PHP**
* **Giới thiệu về ngôn ngữ lập trình PHP**

**Lịch sử phát triển PHP**

PHP: Hypertext Preprocessor, thường được viết tắt thành PHP là một ngôn ngữ lập trình kịch bản hay một loại mã lệnh chủ yếu được dùng để phát triển các ứng dụng viết cho máy chủ, mã nguồn mở, dùng cho mục đích tổng quát. Nó rất thích hợp với web và có thể dễ dàng nhúng vào trang HTML. Do được tối ưu hóa cho các ứng dụng web, tốc độ nhanh, nhỏ gọn, cú pháp giống C và Java, dễ học và thời gian xây dựng sản phẩm tương đối ngắn hơn so với các ngôn ngữ khác nên PHP đã nhanh chóng trở thành một ngôn ngữ lập trình web phổ biến nhất thế giới [4].

Ngôn ngữ, các thư viện, tài liệu gốc của PHP được xây dựng bởi cộng đồng và có sự đóng góp rất lớn của Zend Inc., công ty do các nhà phát triển cốt lõi của PHP lập nên nhằm tạo ra một môi trường chuyên nghiệp để đưa PHP phát triển [4].

PHP được phát triển từ một sản phẩm có tên là PHP/FI. PHP/FI do Rasmus Lerdorf tạo ra năm 1994, ban đầu được xem như là một tập con đơn giản của các mã kịch bản Perl để theo dõi tình hình truy cập đến bản sơ yếu lý lịch của ông trên mạng. Ông đã đặt tên cho bộ mã kịch bản này là 'Personal Home Page Tools'. Khi cần đến các chức năng rộng hơn, Rasmus đã viết ra một bộ thực thi bằng C lớn hơn để có thể truy vấn tới các cơ sở dữ liệu và giúp cho người sử dụng phát triển các ứng dụng web đơn giản. Rasmus đã quyết định công bố mã nguồn của PHP/FI cho mọi người xem, sử dụng cũng như sửa các lỗi có trong nó đồng thời cải tiến mã nguồn [4].

PHP/FI, viết tắt từ "Personal Home Page/Forms Interpreter", bao gồm một số các chức năng cơ bản cho PHP như ta đã biết đến chúng ngày nay. Nó có các biến kiểu như Perl, thông dịch tự động các biến của form và cú pháp HTML nhúng. Cú pháp này giống như của Perl, mặc dù hạn chế hơn nhiều, đơn giản và có phần thiếu nhất quán

Vào năm 1997, PHP/FI 2.0, lần viết lại thứ hai của phiên bản C, đã thu hút được hàng ngàn người sử dụng trên toàn thế giới với xấp xỉ 50.000 tên miền đã được ghi nhận là có cài đặt nó, chiếm khoảng 1% số tên miền có trên mạng Internet. Tuy đã có tới hàng nghìn người tham gia đóng góp vào việc tu chỉnh mã nguồn của dự án này thì vào thời đó nó vẫn chủ yếu chỉ là dự án của một người [4].

PHP/FI 2.0 được chính thức công bố vào tháng 11 năm 1997, sau một thời gian khá dài chỉ được công bố dưới dạng các bản beta. Nhưng không lâu sau đó, nó đã được thay thế bởi các bản alpha đầu tiên của PHP 3.0 [4].

**Các tính năng của PHP**

Hiệu suất: Chương trình viết bằng PHP thực thi nhanh hơn các chương trình được viết bằng các ngôn ngữ khác như JSP, ASP [5].

Mã nguồn mở: PHP là một ngôn ngữ lập trình mã nguồn mở, bạn có thể phát triển tất cả phiên bản PHP theo yêu cầu của mình mà không phải trả bất kỳ chi phí nào

Nền tảng độc lập: PHP có sẵn cho hệ điều hành WINDOWS, MAC, LINUX & UNIX. Một ứng dụng PHP được phát triển trong một hệ điều hành có thể dễ dàng được thực thi trong hệ điều hành khác [5].

Khả năng tương thích: PHP tương thích với hầu hết các máy chủ cục bộ được sử dụng ngày nay như Apache, IIS, v.v [5].

Nhúng: Mã PHP có thể dễ dàng được nhúng trong các script và thẻ HTML [5].

**Các phiên bản của Php**

**PHP 3:** Là phiên bản đầu tiên cho chúng ta thấy một hình ảnh gần gũi với các phiên bản PHP mà chúng ta được biết ngày nay. Nó đã được Andi Gutmans và Zeev Suraski tạo ra năm 1997 sau khi viết lại hoàn toàn bộ mã nguồn trước đó. Lý do chính mà họ đã tạo ra phiên bản này là do họ nhận thấy PHP/FI 2.0 hết sức yếu kém trong việc phát triển các ứng dụng thương mại điện tử mà họ đang xúc tiến trong một dự án của trường đại học. Trong một nỗ lực hợp tác và bắt đầu xây dựng dựa trên cơ sở người dùng đã có của PHP/FI, Andi, Rasmus và Zeev đã quyết định hợp tác và công bố PHP 3.0 như là phiên bản thế hệ kế tiếp của PHP/FI 2.0, và chấm dứt phát triển PHP/FI 2.0.

Một trong những sức mạnh lớn nhất của PHP 3.0 là các tính năng mở rộng mạnh mẽ của nó. Ngoài khả năng cung cấp cho người dùng cuối một cơ sở hạ tầng chặt chẽ dùng cho nhiều cơ sở dữ liệu, giao thức và API khác nhau, các tính năng mở rộng của PHP 3.0 đã thu hút rất nhiều nhà phát triển tham gia và đề xuất các mô đun mở rộng mới. Hoàn toàn có thể kết luận được rằng đây chính là điểm mấu chốt dẫn đến thành công vang dội của PHP 3.0. Các tính năng khác được giới thiệu trong PHP 3.0 gồm có hỗ trợ cú pháp hướng đối tượng và nhiều cú pháp ngôn ngữ nhất quán khác.

Ngôn ngữ hoàn toàn mới đã được công bố dưới một cái tên mới, xóa bỏ mối liên hệ với việc sử dụng vào mục đích cá nhân hạn hẹp mà cái tên PHP/FI 2.0 gợi nhắc. Nó đã được đặt tên ngắn gọn là 'PHP', một kiểu viết tắt hồi quy của "PHP: Hypertext Preprocessor".

Vào cuối năm 1998, PHP đã phát triển được con số cài đặt lên tới hàng chục ngàn người sử dụng và hàng chục ngàn Web site báo cáo là đã cài nó. Vào thời kì đỉnh cao, PHP 3.0 đã được cài đặt cho xấp xỉ 10% số máy chủ Web có trên mạng Internet.

PHP 3.0 đã chính thức được công bố vào tháng 6 năm 1998, sau thời gian 9 tháng được cộng đồng kiểm nghiệm [4].

**PHP 4:** Vào mùa đông năm 1998, ngay sau khi PHP 3.0 chính thức được công bố, Andi Gutmans và Zeev Suraski đã bắt đầu bắt tay vào việc viết lại phần lõi của PHP. Mục đích thiết kế là nhằm cải tiến tốc độ xử lý các ứng dụng phức tạp, và cải tiến tính mô đun của cơ sở mã PHP. Những ứng dụng như vậy đã chạy được trên PHP 3.0 dựa trên các tính năng mới và sự hỗ trợ khá nhiều các cơ sở dữ liệu và API của bên thứ ba, nhưng PHP 3.0 đã không được thiết kế để xử lý các ứng dụng phức tạp như thế này một cách có hiệu quả.

Một động cơ mới, có tên 'Zend Engine' (ghép từ các chữ đầu trong tên của Zeev và Andi), đã đáp ứng được các nhu cầu thiết kế này một cách thành công, và lần đầu tiên được giới thiệu vào giữa năm 1999. PHP 4.0, dựa trên động cơ này, và đi kèm với hàng loạt các tính năng mới bổ sung, đã chính thức được công bố vào tháng 5 năm 2000, gần 2 năm sau khi bản PHP 3.0 ra đời. Ngoài tốc độ xử lý được cải thiện rất nhiều, PHP 4.0 đem đến các tính năng chủ yếu khác gồm có sự hỗ trợ nhiều máy chủ Web hơn, hỗ trợ phiên làm việc HTTP, tạo bộ đệm thông tin đầu ra, nhiều cách xử lý thông tin người sử dụng nhập vào bảo mật hơn và cung cấp một vài các cấu trúc ngôn ngữ mới.

Với PHP 4, số nhà phát triển dùng PHP đã lên đến hàng trăm nghìn và hàng triệu site đã công bố cài đặt PHP, chiếm khoảng 20% số tên miền trên mạng Internet.

Nhóm phát triển PHP cũng đã lên tới con số hàng nghìn người và nhiều nghìn người khác tham gia vào các dự án có liên quan đến PHP như PEAR, PECL và tài liệu kĩ thuật cho PHP [4].

**PHP 5:** Sự thành công hết sức to lớn của PHP 4.0 đã không làm cho nhóm phát triển PHP tự mãn. Cộng đồng PHP đã nhanh chóng giúp họ nhận ra những yếu kém của PHP 4 đặc biệt với khả năng hỗ trợ lập trình hướng đối tượng (OOP), xử lý XML, không hỗ trợ giao thức máy khách mới của MySQL 4.1 và 5.0, hỗ trợ dịch vụ web yếu. Những điểm này chính là mục đích để Zeev và Andi viết Zend Engine 2.0, lõi của PHP 5.0. Một thảo luận trên Slashdot đã cho thấy việc phát triển PHP 5.0 có thể đã bắt đầu vào thời điểm tháng 12 năm 2002 nhưng những bài phỏng vấn Zeev liên quan đến phiên bản này thì đã có mặt trên mạng Internet vào khoảng tháng 7 năm 2002. Ngày 29 tháng 6 năm 2003, PHP 5 Beta 1 đã chính thức được công bố để cộng đồng kiểm nghiệm. Đó cũng là phiên bản đầu tiên của Zend Engine 2.0. Phiên bản Beta 2 sau đó đã ra mắt vào tháng 10 năm 2003 với sự xuất hiện của hai tính năng rất được chờ đợi: Iterators, Reflection nhưng namespaces một tính năng gây tranh cãi khác đã bị loại khỏi mã nguồn. Ngày 21 tháng 12 năm 2003: PHP 5 Beta 3 đã được công bố để kiểm tra với việc phân phối kèm với Tidy, bỏ hỗ trợ Windows 95, khả năng gọi các hàm PHP bên trong XSLT, sửa chữa nhiều lỗi và thêm khá nhiều hàm mới. PHP năm bản chính thức đã ra mắt ngày 13 tháng 7 năm 2004 sau một chuỗi khá dài các bản kiểm tra thử bao gồm Beta 4, RC 1, RC2, RC3. Mặc dù coi đây là phiên bản sản xuất đầu tiên nhưng PHP 5.0 vẫn còn một số lỗi trong đó đáng kể là lỗi xác thực HTTP.

Ngày 14 tháng 7 năm 2005, PHP 5.1 Beta 3 được PHP Team công bố đánh dấu sự chín muồi mới của PHP với sự có mặt của PDO, một nỗ lực trong việc tạo ra một hệ thống API nhất quán trong việc truy cập cơ sở dữ liệu và thực hiện các câu truy vấn. Ngoài ra, trong PHP 5.1, các nhà phát triển PHP tiếp tục có những cải tiến trong nhân Zend Engine 2, nâng cấp mô đun PCRE lên bản PCRE 5.0 cùng những tính năng và cải tiến mới trong SOAP, streams và SPL [4].

**PHP 6:** Phiên bản PHP 6 được kỳ vọng sẽ lấp đầy những khiếm khuyết của PHP ở phiên bản hiện tại, ví dụ: hỗ trợ namespace; hỗ trợ Unicode; sử dụng PDO làm API chuẩn cho việc truy cập cơ sở dữ liệu, các API cũ sẽ bị đưa ra thành thư viện PECL... Phiên bản 6 này chỉ dùng ở việc nghiên cứu và thử nghiệm. Sau này PHP bỏ hẳn phiên bản 6 và lên 7 [4].

**PHP 7:** Với việc sử dụng bộ nhân Zend Engine mới PHPNG cho tốc độ nhanh gấp 2 lần. Ngoài ra ở phiên bản này còn thêm vào rất nhiều cú pháp, tính năng mới giúp cho PHP trở nên mạnh mẽ hơn. Những tính năng mới quan trọng có thể kể đến như:

Khai báo kiểu dữ liệu cho biến.

Xác định kiểu dữ liệu sẽ trả về cho 1 hàm.

Thêm các toán tử mới (??, <=>, ...) [4].

**Môi trường lập trình PHP:**

Có thể sử dụng bất kì một chương trình soạn thảo văn bản dạng text bất kì để viết mã nguồn PHP.

**Các bước cơ bản để xây dựng và thực thi một chương trình PHP:**

* **Soạn thảo:** Mã nguồn chương trình được viết bằng một phần mềm soạn thảo văn bản dạng text và lưu trên ổ đĩa. Ta có thể dùng những phần mềm soạn thảo văn bản đơn giản nhất như Notepad (trong môi trường Windows) hay emacs (trong môi trường Unix/Linux), hoặc các công cụ soạn thảo trong môi trường tích hợp để viết mã nguồn chương trình. Mã nguồn PHP đặt trong các file với tên có phần mở rộng là .php (ví dụ: index.php).
* **Download phần mềm chạy Web Server cho máy tính của bạn:** Có rất nhiều phần mềm nhưng nên chọn Xampp vì tương nó tương thích với nhiều hệ điều hành hiện nay.
* **Cài đặt phần mềm chạy Web Server:** Hãy chắc chắn rằng bạn đã cài đặt đầy đủ cả Apache và MySQL khi cài đặt Xampp.
* **Chạy ứng dụng:** Bật Apache của Xampp lên. Để kiểm tra mọi thứ đã được cài đặt đúng thì mở trình duyệt và nhập “localhost” vào trong địa chỉ URL. Nếu mọi thứ đều ổn, thì trang chào mừng của Xampp sẽ được hiển thị.
* **Nạp và chạy:** Di chuyển file có phần mở rộng .php đã tạo trước đó (ví dụ index.php) đến thư mục htdocs của Xampp sau đó mở trình duyệt lên và đi đến địa chỉ localhost/index.php và kết quả sẽ được hiển thị ngay trên trình duyệt.
  + 1. **Ngôn ngữ lập trình JavaScript**
* **Giới thiệu chung về ngôn ngữ JavaScript**

JavaScript là ngôn ngữ lập trình của Web. Phần lớn các trang web hiện đại sử dụng JavaScript và tất cả các trình duyệt web hiện đại trên máy tính để bàn, trò chơi bảng điều khiển, máy tính bảng và điện thoại thông minh bao gồm trình thông dịch JavaScript, làm cho Java Script trở thành ngôn ngữ lập trình phổ biến nhất trong lịch sử. JavaScript là một phần của bộ ba công nghệ mà tất cả các nhà phát triển Web phải học: HTML để chỉ định nội dung của các trang web, CSS để chỉ định cách trình bày của các trang web và JavaScript để chỉ định hoạt động của các trang web. Nếu bạn đã quen thuộc với các ngôn ngữ lập trình khác, nó có thể giúp bạn biết rằng JavaScript là một ngôn ngữ lập trình thông dịch cấp cao, động, không định kiểu rất phù hợp với phong cách lập trình hướng đối tượng và chức năng. JavaScript lấy cú pháp của nó từ Java, các hàm hạng nhất của nó từ Scheme, và kế thừa dựa trên nguyên mẫu của nó từ Self. Nhưng bạn không cần biết bất kỳ ngôn ngữ nào trong số đó, hoặc làm quen với các thuật ngữ đó, để sử dụng cuốn sách này và học JavaScript. Tên “JavaScript” thực sự hơi gây hiểu lầm. Ngoại trừ sự giống nhau về cú pháp bề ngoài, JavaScript hoàn toàn khác với ngôn ngữ lập trình Java. Và JavaScript từ lâu đã phát triển nhanh hơn nguồn gốc ngôn ngữ kịch bản của nó để trở thành một ngôn ngữ có mục đích chung mạnh mẽ và hiệu quả [6].

**Tại sao nên sử dụng JavaScript**

* **Dễ phát triển dự án lớn:** Với việc sử dụng các kỹ thuật mới nhất và lập trình hướng đối tượng nên JavaScript giúp chúng ta phát triển các dự án lớn một cách dễ dàng.
* **Nhiều Framework và thư viện để lựa chọn:** Hiện nay các thư viện và framework Javascript rất nhiều và đã được phát triển sẵn có thể sử dụng trực tiếp trong quá trình phát triển ứng dụng để tiết kiệm thời gian như: AngularJS, ReactJS, jQuery, …
* **Javascript có ở khắp mọi nơi:** Nó được tích hợp sẵn trên mọi trình duyệt web hiện đại và bạn thực sự không cần bất kỳ thiết lập môi trường đặc biệt nào. Ví dụ: Chrome, Mozilla Firefox, Safari và mọi trình duyệt phổ biến cho đến ngày hôm nay, đều hỗ trợ Javascript.
* **Javascript có cộng đồng lớn:** StackOverflow là một trang web Hỏi Đáp về lập trình rất phổ biến hiện nay. Javascript đã vượt qua Java về quy mô cộng đồng và có khả năng nó sẽ tiếp tục phát triển. Một trong những điều mà các nhà phát triển Javascript thích nhất là cộng đồng. Javascript liên tục đổi mới các công cụ, tiện lợi, có thể rất thú vị và vui nhộn. Có hơn 3100 dự án GitHub với hơn 500 sao. Tuy nhiên, nhiều công cụ cũng có tốc độ khuấy cao, vì vậy theo kịp tất cả những điều mới có thể hơi mệt mỏi

**Những ưu điểm của JavaScript**

* **Ít tương tác với máy chủ**

Bạn có thể xác thực đầu vào của người dùng trước khi gửi dữ liệu đến máy chủ. Điều này giúp tiết kiệm lưu lượng máy chủ, có nghĩa là tải ít hơn trên máy chủ của bạn. [6].

* **Phản hồi ngay lập tức cho khách truy cập**

Họ không phải đợi tải lại trang để kiểm tra xem họ có quên nhập nội dung nào không mà nó sẽ hiển thị ngay lập tức [6].

* **Giao diện phong phú hơn**

Bạn có thể sử dụng JavaScript để kéo thả các thành phần, để tạo slider, ... cho khách truy cập trang web của bạn. [6].

* **Tăng tính tương tác**

Bạn có thể tạo các giao diện tương tác khi người dùng di chuột qua chúng hoặc kích hoạt chúng thông qua bàn phím.

* **Tương đối nhanh với người dùng cuối**

Khi mã được thực hiện trên máy tính của người dùng, kết quả và quá trình xử lý được hoàn thành hầu như ngay lập tức tùy thuộc vào nhiệm vụ (nhiệm vụ trong javascript trên các trang web thường rất đơn giản để tránh bị hog bộ nhớ) Không cần phải được xử lý trong máy chủ web của trang web và gửi lại cho người dùng tiêu tốn băng thông của địa phương cũng như máy chủ.

* **Khá dễ dàng cho việc tìm hiểu**

Javascript là tương đối dễ dàng để tìm hiểu và bao gồm các cú pháp đó là gần tiếng Anh. Nó sử dụng mô hình DOM cung cấp nhiều tính năng được viết sẵn cho các đối tượng khác nhau trên các trang giúp bạn dễ dàng phát triển một kịch bản để giải quyết một mục đích tùy chỉnh.

**Các nhược điểm còn tồn tại của JavaScript**

* **Các vấn đề về bảo mật**

Các đoạn mã Javascript, một khi đã được nối vào các trang web thực hiện trên các máy chủ khách hàng ngay lập tức và do đó cũng có thể được sử dụng để khai thác hệ thống của người dùng. Mặc dù một số hạn chế nhất định được thiết lập bởi các tiêu chuẩn web hiện đại trên các trình duyệt, mã độc hại vẫn có thể được thực hiện tuân thủ các quy định hạn chế.

* **Thiếu các tính năng quan trọng sau**

JavaScript phía máy khách không cho phép đọc hoặc ghi tệp. Điều này đã được giữ vì lý do an ninh.

JavaScript không thể được sử dụng cho các ứng dụng mạng vì không có hỗ trợ như vậy.

JavaScript không có bất kỳ khả năng đa luồng hoặc đa xử lý nào.

* + 1. **Tổng quan về Laravel Framework**

**Tổng quan về Laravel**

[Laravel là PHP Web Framework miễn phí, mã nguồn mở, dành cho việc phát triển các ứng dụng web theo mô hình kiến trúc mô hình MVC và dựa trên Symfony PHP Framework. Một số tính năng của Laravel như là sử dụng hệ thống đóng gói module, quản lý package (Composer), hỗ trợ nhiều hệ quản trị CSDL quan hệ (MySQL, MariaDB, SQLite, PostgreSQL,…), các tiện ích hỗ trợ triển khai và bảo trì ứng dụng.](https://docs.google.com/spreadsheets/d/1DUF2isFWsqVSYhbaACYtbgcLi_YjDqpE3GLQIVgkKQg/edit#gid=69851113)

Laravel được tạo ra bởi Taylor Otwell với phiên bản đầu tiên được ra mắt vào tháng 6 năm 2011. Từ đó cho đến này, Laravel đã phát triển một cách mạnh mẽ, vượt qua những framework khác và vươn lên trở thành framework PHP có thể nói được ưa chuộc và được cộng đồng sử dụng nhiều nhất khi phát triển web với PHP.

**Những thuận lợi của Laravel**

* **Lợi ích của Laravel**

[Nguồn tài nguyên vô cùng lớn và sẵn có: nguồn tài nguyên của Laravel rất thân thiện với các lập trình viên với đa dạng tài liệu khác nhau để tham khảo. Các phiên bản được phát hành đều có nguồn tài liệu phù hợp với ứng dụng của mình.](https://docs.google.com/spreadsheets/d/1DUF2isFWsqVSYhbaACYtbgcLi_YjDqpE3GLQIVgkKQg/edit#gid=69851113)

[Tích hợp với dịch vụ mail: Lavarel là framework được trang bị API sạch trên thư viện SwiftMailer, do đó, bạn có thể gửi thư qua các dịch vụ dựa trên nền tảng đám mây hoặc local.](https://docs.google.com/spreadsheets/d/1DUF2isFWsqVSYhbaACYtbgcLi_YjDqpE3GLQIVgkKQg/edit#gid=69851113)

[Tốc độ xử lý nhanh: Laravel hỗ trợ hiệu quả cho việc tạo lập website hay các dự án lớn trong thời gian ngắn. Vì vậy, nó được các công ty công nghệ và lập trình viên sử dụng rộng rãi để phát triển các sản phẩm của họ.](https://docs.google.com/spreadsheets/d/1DUF2isFWsqVSYhbaACYtbgcLi_YjDqpE3GLQIVgkKQg/edit#gid=69851113)

[Dễ sử dụng: Laravel được đón nhận và trở nên sử dụng phổ biến vì nó rất d sử dụng. Thường chỉ mất vài giờ, bạn có thể bắt đầu thực hiện một dự án nhỏ với vốn kiến thức cơ bản nhất về lập trình với PHP.](https://docs.google.com/spreadsheets/d/1DUF2isFWsqVSYhbaACYtbgcLi_YjDqpE3GLQIVgkKQg/edit#gid=69851113)

[Tính bảo mật cao: Laravel cung cấp sẵn cho người dùng các tính năng bảo mật mạnh mẽ để người dùng hoàn toàn tập trung vào việc phát triển sản phẩm của mình](https://docs.google.com/spreadsheets/d/1DUF2isFWsqVSYhbaACYtbgcLi_YjDqpE3GLQIVgkKQg/edit#gid=69851113)

* **Những tính năng đặc biệt khi sử dụng Laravel**

***Composer****:* sử dụng để nâng cấp, cài đặt…

***Eloquent ORM***:  thao tác với cú pháp đẹp mắt và đơn giản.

***Restful API***: hỗ trợ biến Laravel thành một web service API.

***Artisan***: cung cấp các lệnh cần thiết để phát triển ứng dụng.

***View***: giúp code sạch sẽ hơn rất nhiều.

***Migrations***: hỗ trợ tạo các trường trong cơ sở dữ liệu, thêm các cột trong bảng, tạo mối quan hệ giữa các bảng, hỗ trợ quản lý cơ sở dữ liệu.

***Authentication***: cung cấp sẵn các tính năng đăng nhập, đăng ký, quên mật khẩu…

***Unit Testing***: hỗ trợ test lỗi để sửa chữa.

* 1. **Hệ quản trị cơ sở dữ liệu MongoDB**
     1. **Tổng quan về hệ quản trị cơ sở dữ liệu MongoDB**

1. Giới thiệu về MongoDB

**Khái niệm:** MongoDB là một chương trình cơ sở dữ liệu mã nguồn mở được thiết kế theo kiểu hướng đối tượng trong đó các bảng được cấu trúc một cách linh hoạt cho phép các dữ liệu lưu trên bảng không cần phải tuân theo một dạng cấu trúc nhất định nào. Chính do cấu trúc linh hoạt này nên MongoDB có thể được dùng để lưu trữ các dữ liệu có cấu trúc phức tạp và đa dạng và không cố định.

Lợi thế của MongoDB so với các CSDL dạng quan hệ(RDBMS)

* Ít Schema hơn: MongoDB là một cơ sở dữ liệu dựa trên Document, trong đó một Collection giữ các Document khác nhau. Số trường, nội dung và kích cỡ của Document này có thể khác với Document khác.
* Cấu trúc của một đối tượng là rõ ràng.
* Không có các Join phức tạp.
* Khả năng truy vấn sâu hơn. MongoDB hỗ trợ các truy vấn động trên các Document bởi sử dụng một ngôn ngữ truy vấn dựa trên Document mà mạnh mẽ như SQL.
* MongoDB dễ dàng để mở rộng.
* Việc chuyển đổi/ánh xạ của các đối tượng ứng dụng đến các đối tượng cơ sở dữ liệu là không cần thiết.
* Sử dụng bộ nhớ nội tại để lưu giữ phần công việc, giúp truy cập dữ liệu nhanh hơn.

1. Lịch sử phát triển của MongoDB

* MongoDB được tạo bởi Dwight Merriman và Eliot Horowitz – những người đã gặp các vấn đề về phát triển và khả năng mở rộng với cách tiếp cận cơ sở dữ liệu quan hệ truyền thống trong khi xây dựng các ứng dụng trong [thiết kế web](https://websolutions.com.vn/) tại – để hỗ trợ một lượng lớn dữ liệu.
* Merriman và Horowitz đã giúp thành lập 10Gen Inc. Vào 2007 để thương mại hóa MongoDB và các phần mềm liên quan. Công ty đã được đổi tên thành MongoDB Inc. vào năm 2013 và ra mắt vào 10/2017 với ký hiệu MDB.
* DBMS được phát hành dưới dạng phần mềm nguồn mở vào năm 2009 và có các điều khoản của Phiên bản 3.0 của Giấy phép [GNU](https://vi.wikipedia.org/wiki/GNU), bên cạnh các giấy phép thương mại được cung cấp bởi MongoDB Inc.

1. Đặc điểm của MongoDB

* Kho lưu định hướng Document: Dữ liệu được lưu trong các tài liệu kiểu JSON.
* Lập chỉ mục trên bất kỳ thuộc tính nào.
* Các truy vấn đa dạng.
* Cập nhật nhanh hơn.

1. Tính năng của hệ quản trị MongoDB

MongoDB có các tính năng sau:

**Truy vấn ad hoc**

Đây là tính năng tốt nhất của MongoDB. Nó có tác dụng giúp các trường truy vấn phạm vi, kết hợp cùng việc tìm kiếm biểu thức nhằm trả về kết quả tài liệu cụ thể với kích thước nhất định. Ngoài ra, các trường dùng trong MongoDB còn được dùng để lập chỉ mục chính, phụ.

**Nhân rộng**

MongoDB cung cấp Replica Set cho phép nhân 2 hoặc nhiều bản sao của dữ liệu.

Đồng thời, mỗi bản sao lại đóng vai trò chính và phụ.

* Khi nhân rộng, toàn bộ dữ liệu khi ghi và đọc được thực hiện trên bản sao chính.
* Bản sao thứ cấp sẽ dùng bản sao tích hợp để có thể duy trì các bản sao dữ liệu. Trong trường hợp có bất kỳ bản sao chính nào bị thất bại thì Replica set sẽ chọn một bản sao thứ cấp để thay thế làm bản sao chính tiếp theo. Trong quá trình nhân rộng, Replica thứ cấp được tùy ý chọn các hoạt động nhưng dữ liệu cuối cùng vẫn phải tuân theo mặc định.

**Cân bằng tải**

MongoDB cân bằng tải bằng cách dựa vào các Shard key để chia dữ liệu thành các phạm vi và phân phối đồng đều. Chúng có thể chạy trên nhiều máy chủ khác nhau và thực hiện chức năng sao chép dữ liệu hay cân bằng tải nhằm giữ hệ thống hoạt động liên tục trong trường hợp phát sinh lỗi về phần cứng.

**Lưu trữ tệp**

Chúng ta thấy rằng tính năng lưu trữ tệp được dùng như một hệ thống tệp (gọi là GridFS) đóng vai trò cân bằng tải, đồng thời, sao chép dữ liệu trên nhiều máy tính. Cụ thể, GridFS chia một tệp ra làm nhiều phần và lưu trữ thành các tài liệu riêng biệt. Sau đó, người dùng dễ dàng truy cập GridFS thông qua Mongofiles hay các plugin sử dụng cho Nginx và Lighttpd.

**Tập hợp**

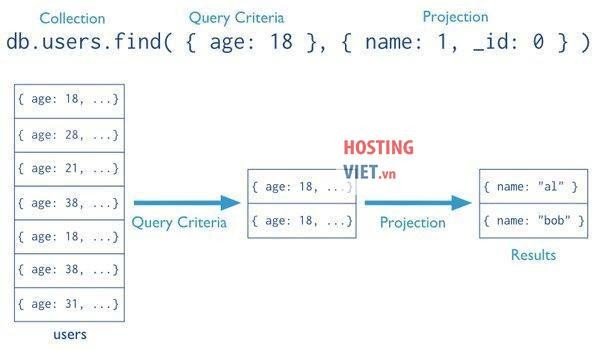
Tính năng này chính là chương trình mang đến ba giải pháp để thực hiện tập hợp gồm Aggregation Pipeline, Mapreduce và Single-purpose Aggregation. Trong đó, Aggregation Pipeline được đánh giá là có hiệu suất tốt nhất.

**Giới hạn kích thước collection**

Các collection được MongoDB hỗ trợ thường có kích thước cố định nên gọi chúng là collection giới hạn. Với kích cỡ cố định, kết hợp cùng việc theo sau thứ tự chèn giúp tăng hiệu suất của các hoạt động liên quan đến dữ liệu. Và khi dữ liệu vượt giới hạn thì những tài liệu cũ hơn sẽ tự động bị xóa mà bạn không cần thực hiện thao tác thêm bất kỳ dòng lệnh nào.

**Giao dịch**

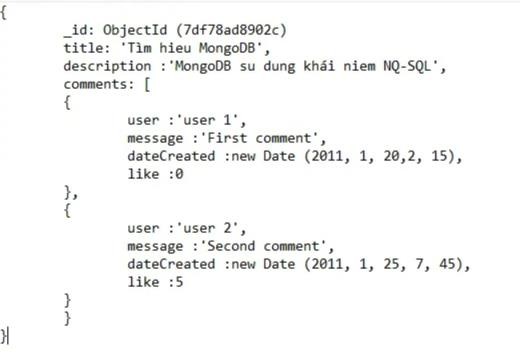
Bắt đầu từ phiên bản 4.0 MongoDB được bổ sung thêm tính năng hỗ trợ giao dịch ACID đa tài liệu.



1. Một số câu lệnh cơ bản trên MongoDB

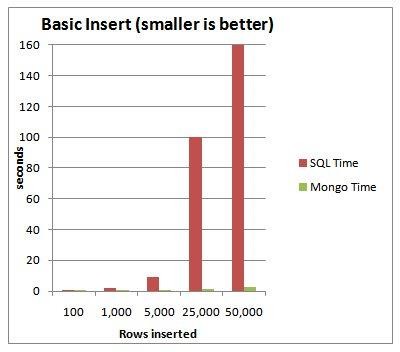
Một số câu lệnh cơ bản của MongoDB như sau:

* Tạo bảng: db. createCollection (‘students’);
* Tạo bảng ghi: db. students. Insert ({name:’thanh’,gender:’male’});
* Cập nhật: db. students. Update ({\_id:1},{$set:{name:’thanh update’}});
* Xóa bản ghi: db. students. Remove ({\_id:1});
* Tìm kiếm: db. students. Find ({name:’thanh’});
* Tìm kiếm tất cả: db. students. Find ({});



1. Ưu điểm, nhược điểm của MongoDB

* Ưu điểm của MongoDB
* 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, nên bạn muốn gì thì cứ insert vào thoải mái.
* 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. 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 CSDL 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 MongoDB
* 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 thì phải hết sức cẩn thận.
* 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 ít giây MongoDB mới thực hiện ghi toàn bộ dữ liệu thay đổi từ RAM xuống ổ cứng điêù này sẽ là nhược điểm vì sẽ có nguy cơ bị mất dữ liệu khi xảy ra các tình huống như mất điện...

* + 1. **Mô hình dữ liệu**

Dữ liệu trong [MongoDB](https://quantrimang.com/hoc-mongodb) có một Schema linh động. Các Document trong cùng Collection không cần thiết phải có cùng tập hợp các trường hoặc cấu trúc, và các trường chung trong các Document của Collection có thể giữ các kiểu dữ liệu khác nhau.

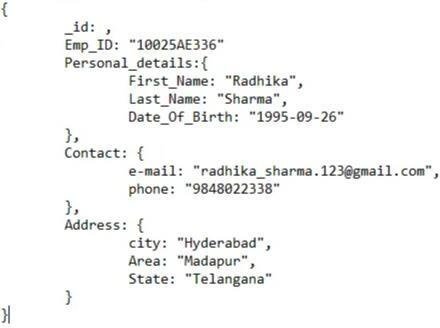
* + 1. **Thiết kế mô hình dữ liệu**

MongoDb cung cấp 2 kiểu dữ liệu mô hình : Mô hình dữ liệu nhúng và mô hình dữ liệu chuẩn hoá. Tuỳ vào trường hợp, bạn có thể sử dụng model phù hợp trong quá trình chuẩn bị document

1. Mô hình dữ liệu nhúng

Trong mô hình này, bạn có thể có (nhúng) tất cả dữ liệu liên quan vào một tài liệu duy nhất, nó còn được gọi là mô hình dữ liệu đã được chuẩn hóa.

Ở ví dụ dưới đây, giả sử chúng ta lấy chi tiết nhân viên trong 3 documents khác nhau,cụ thể là : Personal\_details, Contact và, Address. Bạn có thể nhúng hoàn toàn 3 documents như sau :

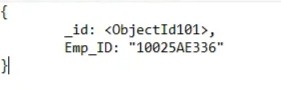


1. Mô hình dữ liệu chuẩn hóa

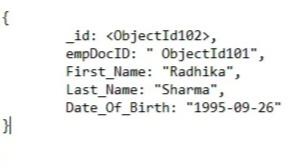
Ở mô hình này, bạn có thể tham khảo các tài liệu phụ trong tài liệu gốc, sử dụng tài liệu tham khảo.

Ví dụ: bạn có thể viết lại tài liệu trên trong mô hình chuẩn hóa dưới dạng:

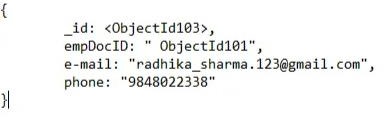
**Employee**



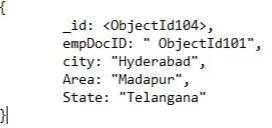
**Personal\_details**



**Contact**



**Address**



* + 1. **Chỉ mục (Index)**

1. Tổng quan, các loại chỉ mục

* Trong MongoDB, Index là một cấu trúc dữ liệu đặc biệt, nó lưu trữ một phần nhỏ dữ liệu của collection ở một dạng dễ dàng duyệt được. Index lưu trữ giá trị của một trường cụ thể hoặc 1 tập hợp các trường của collection và được sắp xếp theo giá trị của các trường đó. Thứ của các index entries hỗ trợ cho các tác vụ truy vấn bằng và truy vấn theo khoảng. Thêm vào đó, MongoDB có thể trả về tập kết quả đã được sắp xếp bằng việc sử dụng thứ tự trong index.
* Index hỗ trợ thực thi có hiệu quả các câu truy vấn trong MongoDB. Nếu không có index, MongoDB phải thực hiện collection scan, tức là quét qua hết tất cả các document trong collection để tìm được những document khớp với điều kiện truy vấn.
* Tuy nhiên, nếu có index, thay vì phải scan qua từng document, thì MongoDB chỉ cần tìm trong trong index. Với 1 cặp key-value được lưu trữ, key là giá trị của trường được đánh chỉ mục, và value là chính document, ta sẽ giảm được số lượng document cần phải duyệt qua để thỏa mãn câu truy vấn. Từ đó tối ưu và tăng tốc độ truy vấn dữ liệu.
* MongoDB sử dụng cấu trúc dữ liệu b-tree để hiện thực index.
* Về cơ bản, index trong MongoDB cũng tương tự index trong các hệ cơ sở dữ liệu khác. MongoDB cũng hỗ trợ define index ở collection level, và ta có thể tạo chỉ mục index ở bất kì field hay sub-field nào của collection trong MongoDB.

1. Thuộc tính chỉ mục

**- Unique index:** Khi ta thêm thuộc tính unique vào index thì MongoDB sẽ reject những giá trị trùng lặp trên field đã được đánh index

Ví dụ:

*$ db.user.createIndex({ name: 1 }, {unique: true})*

Khi ta thêm 1 document mà có thiếu field được đánh index thì ta vẫn tạo được document với field đó bị thiếu. Tuy nhiên index sẽ trả về lỗi nếu ta cố tình tiếp tục thêm document thiếu indexed field vào collection đã có document thiếu indexed field

- **Partially index**: Chúng ta có thể đánh index trên một số document của collection mà thỏa mãn một số điều kiện nhất định. Bằng việc đánh index trên 1 tập con của các document trong collection, partially index chỉ yêu cầu bộ nhớ ít.

Để tạo Partially index, ta cần thêm partialFilterExpression option khi tạo index

*$ db. restaurant. createIndex (*

*{*

*cuisine: 1, name: 1},*

*{*

*partialFilterExpression: {rating: {$gt: 5 } } } ,*

*)*

=> Chỉ mục được đánh trên field cuisine và name của các document thuộc có collection restaurand những có rating > 5

* **Sparse index**: Sparse index là index chỉ chứa những document có field được đánh index có giá trị (kể cả giá trị null). Để tạo sparse index, ta thêm option sparse: true khi tạo index.

*$ db.student.createIndex( {score: 1}, {sparse: true} )*

Trong ví dụ trên, những document nào có field score mới được đánh index

* + 1. **Mô hình truy vấn**

a, Lấy tất cả dữ liệu trong Collection

Để lấy tất cả dữ liệu ở trong collection chúng ta sử dụng phương thức **find()** với cú pháp:

db.collectionName.find()

Với collectionName là tên của collection mà các bạn muốn truy vấn.

* Tuy nhiên, khi chỉ sử dụng mỗi phương thức find thì dữ liệu trả về sẽ dưới dạng object nhưng không theo một cấu trúc nào cả.

b, Truy vấn có điều kiện trong MongoDB

Để truy vấn có điều kiện trong MongoDB thì bạn cũng sử dụng cú pháp tương tự như phần 1, nhưng lúc này chúng ta sẽ chèn thêm điều kiện vào trong hàm find() với cú pháp sau:

db.collection.find(condition)

Trong đó:

* collectionName là tên của collection mà các bạn muốn truy vấn.
* condition là object chứa mệnh đề điều kiện. Theo các cú pháp sau đây:

|  |  |  |  |
| --- | --- | --- | --- |
| Phép Toán | Cú Pháp | Ví dụ | Câu lệnh tương ứng trong SQL |
| Bằng (Equality) | {key: value} | db.admin.find({name: "Vu Thanh Tai"  }).pretty() | .. WHERE  name = "Vu Thanh Tai" |
| Nhỏ hơn (Less Than ) | {key: {$lt: value}} | db.admin.find({age: {  $lt: 18}).pretty() | ... WHERE  age < 18 |
| Nhỏ hơn bằng (Less Than Equals) | {key: {$lte: value}} | db.admin.find({age: {  $lte: 18}).pretty() | ... WHERE  age <= 18 |
| Lơn hơn (Greater Than) | {key: {$gt: value}} | db.admin.find({age: {  $gt: 12}).pretty() | ... WHERE  age > 12 |
| Lớn hơn bằng (Greater Than Equals) | {key: {$gte: value}} | db.admin.find({age: {  $gte: 12}).pretty() | ... WHERE  age >= 12 |
| Khác (Not Equals) | {key: {$ne: value}} | db.admin.find({age: {  $ne: 12}).pretty() | ... WHERE  age != 12 |
| Trong ( In) | {key: {$in: [value1, value2,..]}} | db.admin.find({age: {  $in: [12, 18]}).pretty() | ... WHERE  age IN |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | (12, 18) |
| Không Thuộc (Not In) | {key: {$nin: [value1, value2,..]}} | db.admin.find({age: {  $nin: [12, 18]}).pretty() | ... WHERE  age NOT IN (12,  18) |

c, Truy vấn nhiều điều kiện trong MongoDB

+ Trong MongoDB cũng có hỗ trợ chúng ta truy vấn nhiều điều kiện trên một lần khai báo, với các toán tử AND, OR Như trong SQL.

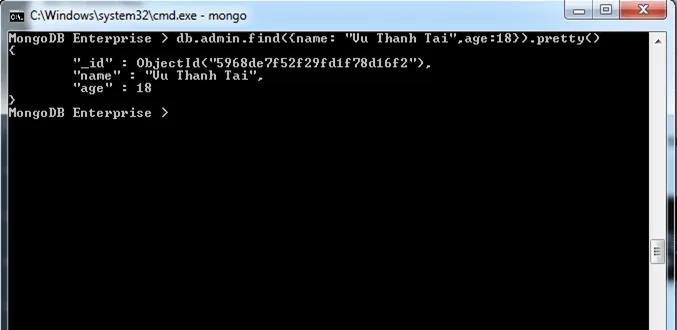
**AND**

+ Để thực hiện phép toán này thì các bạn chỉ cần thêm các điều kiện cảu câu truy vấn vào trong object chứa điều kiện bind vào trong phương thức find.

**VD**: Lấy ra admin có tên Vũ Thanh Tài và có tuổi là 18 trong collection admin.

**db. Admin. Find ({**

**name: “Vu Thanh Tài” age: 18**

*3.7: Truy vấn nhiều điều kiện trong MongoDB*

* Tương tự, nếu như bạn muốn and bao nhiêu điều kiện thì thêm bấy nhiêu vào trong object.

**OR**

* Để sử dụng mệnh đề or (hoặc) trong MongoDB thì chúng ta cần phải truyền một key scope có tên là $or vào làm key chứa mảng các điều kiện hoặc, theo cú pháp: vào làm key chứa mảng các điều kiện hoặc, theo cú pháp:

**db. CollectionName.find({**

**$or: [**

**{key1: value1},**

**{key2: value2},**

**…,**

**{keyN: valueN}**

**}). Pretty ()**

Kết hợp cả **AND** và **OR**.

* Để kết hợp giữa **AND** và **OR** thì bạn chỉ cần làm tương tự như cách thực hiện truy vấn **AND**, và nếu truy vấn nào là or thì object đó lại làm tương tự như truy vấn **OR**.

**CHƯƠNG II: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG**

4. 1. **Phân tích yêu cầu đề tài**
      1. **Phát biểu bài toán**

Tên bài toán: “Ứng dụng PHP, Bootstrap vào xây dựng website kinh doanh đồ ăn nhanh”

Đây là một loại hình kinh doanh qua mạng bằng cách xây dựng một trang web bán hang điện tử để mọi người có thể vào đó lựa mua những mặt hàng cần thiết,lựa chọn số lượng và sau đó đặt hàng. Với trang kinh doanh đồ ăn nhanh trực tuyến, cửa hàng sẽ đưa thông tin các thiết bị điện tử đang có và sắp ra mắt của mình. Việc đưa thông tin phải theo một trật tự hợp lý. Ví dụ, khi đưa thông tin thiết bị điện tử thì các thông tin phải đầy đủ gồm: Hình ảnh, tên thiết bị, thông số kỹ thuật, giá tiền, số lượng… Người quản trị trang web cũng có quyền thêm mới, sửa đổi thông tin thiết bị, hoặc xóa đi những thiết bị đã dừng kinh doanh. Mọi sự thay đổi đều được ghi nhận trong cơ sở dữ liệu để đến với người sử dụng.

Với khách hàng thì được xem trang web, xem thông tin các món ăn đang kinh doanh và sắp sắp ra mắt nhưng không có quyền đặt hàng. Để có thể đặt hàng, khách hàng cần phải đăng kí là thành viên của trang web. Khi đã là thành viên, khách hàng đăng nhập vào website theo email và mật khẩu của mình. Khi đó, khách hàng sẽ có quyền đặt hàng. Tùy theo cửa hàng ở những thành phố khác nhau, khách hàng có quyền chọn cửa hàng, thành phố mình muốn đặt hàng. Bên cạnh đó, khách hàng cũng có thể chọn mặt hàng theo ý thích và nắm được thiết bị nào đang bán chạy. Khi hoàn thành việc đặt hàng, nhân viên sẽ giao hang tận nơi theo địa chỉ mà khách hang đã đăng kí trên hệ thống.

Khi khách hàng đặt hàng thành công, thông tin đơn hàng sẽ được ghi nhận trong hệ thống và chuyển đến màn hình của người quản lý cửa hàng sau đó lên đơn hàng và sẽ được nhân viên của cửa hàng vận chuyển đến địa chỉ của khách hàng đã đăng kí trên hệ thống.

* + 1. **Yêu cầu bài toán**

Hệ thống có chức năng ủy quyền người dùng, do đó:

**Đối với người dùng là khách hàng - không phải thành viên:**

* Được xem thông tin các thiết bị, khuyến mãi, giá, các mặt hàng hiện có.
* Được đăng ký tài khoản, đăng nhập.

**Đối với người dùng là khách hàng - thành viên:**

* Được xem thông tin các thiết bị, khuyến mãi, giá, các mặt hàng đang có.
* Được đăng ký tài khoản, đăng nhập.
* Được đặt hàng, bán hang và chọn các thiết bị yêu thích nhận khuyến mãi.
* Được quản lí tài khoản thành viên của mình

**Đối với người dùng là chủ cửa hàng hoặc quản lý:**

* Được thêm thông tin các thiết bị: Khi có hàng mới được nhập về, admin sẽ thêm mới thông tin về thiết bị đó vào danh sách.
* Được xóa các thiết bị: Có thể vì một lí do nào đó mà ngừng kinh doanh và buộc xóa khỏi hệ thống thì người quản lí cũng có thể xóa được.
* Được thêm tài khoản các thành viên: Khi có khách hàng có nhu cầu trở thành thành viên mới, người quản lí có thể trực tiếp thêm, tạo thành viên mới giúp khách hàng.
* Được xóa thông tin các thành viên: Khi khách hàng vi phạm các điều luật của cửa hàng hoặc vì một lí do nào đó yêu cầu xóa tài khoản, người quản lí có thể xóa toàn bộ thông tin liên quan đến tài khoản đó.
* Được chỉnh sửa danh sách những thiết bị trong ngày: Khi có sự thay đổi về các thiết bị, người quản lí có thể cập nhật lại mới trên hệ thống
  1. **Nghiên cứu hiện trạng**

Sau khi nghiên cứu, khảo sát hiện trạng, em đã nắm bắt được các thông tin sau:

* Quản lí mặt hàng: Mỗi mặt hàng được quản lý các thông tin sau: Hình ảnh, tên, thông số kỹ thuật, thành phần, giá cả, số lượng, khuyến mãi.
* Quản lí thành viên: Mỗi thành viên được quản lí các thông tin sau: Mã thành viên, họ và tên thành viên, tài khoản người dùng, mật khẩu, email, chứng mình nhân dân, ngày sinh, giới tính, số điện thoại, địa chỉ, ngày đăng kí tài khoản, tình trạng tài khoản.
* Đặt hàng trực tuyến: Mỗi đơn hàng khi được đặt cần đáp ứng đầy đủ các thông tin sau: thông tin của món ăn (bao gồm thông tin món ăn, ngày đặt, giờ đặt, và cửa hàng), thông tin của tài khoản thành viên (bao gồm mã thành viên, họ và tên thành viên, tài khoản người dùng, …) và phương thức thanh toán.
  1. **Đặc tả hệ thống**

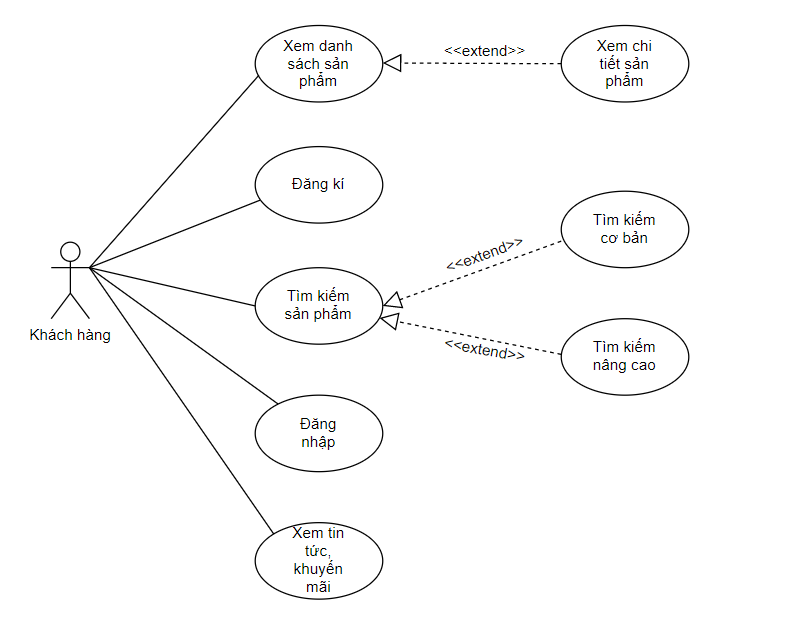
|  |  |  |
| --- | --- | --- |
| **Vai trò** | | **Mô tả** |
| Người quản lí | | Là người có vai trò cao nhất trong hệ thống, người quản lí có thể thêm, sửa, xóa mọi thông tin của hệ thống |
| Khách hàng | Không phải thành viên | Thao tác trực tiếp trên hệ thống người dùng, với các chức năng: xem danh sách mặt hàng, xem khuyến mãi, xem giá, thiết bị hiện có, đăng nhập, đăng ký tài khoản |
| Thành viên | Vận hành các chức năng của khách hàng - không phải thành viên, ngoài ra khách hàng thành viên có thể đặt hàng, xem thông tin của tài khoản. |

#### Bảng 1: Đặc tả hệ thống

* 1. **Sơ đồ use-case**

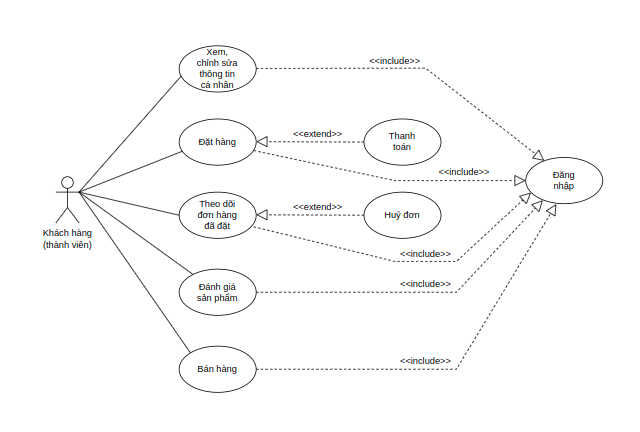
Sơ đồ ca sử dụng (use-case)là kỹ thuật được dùng trong kỹ thuật phần mềm và hệ thống nhằm nắm bắt những yêu cầu chức năng của hệ thống. Use case mô tả sự tương tác đặc trưng giữa người dùng bên ngoài và hệ thống. Use case cũng mô tả các yêu cầu đối với hệ thống.

* + 1. **Sơ đồ use-case của khách hàng**



Hình 3: Sơ đồ Use-case của khách hàng

* + 1. **Sơ đồ use-case của khách hàng – thành viên**



Hình 4: Sơ đồ Use-case của khách hàng-thành viên

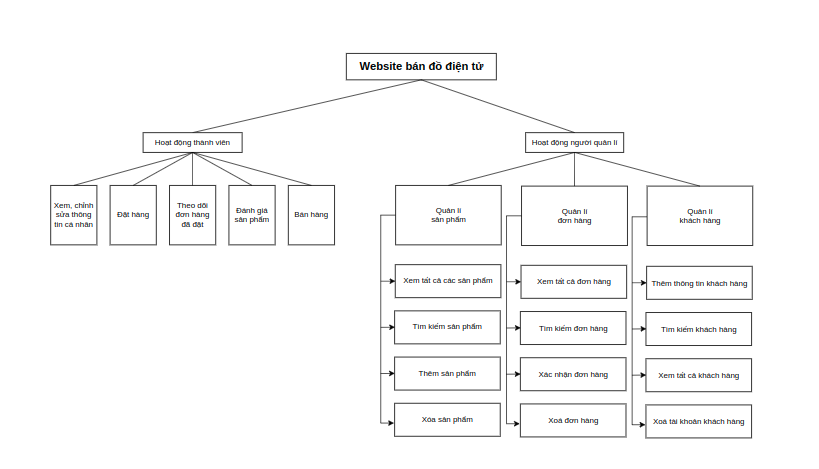
* + 1. **Sơ đồ use-case của người quản lí**



Hình 5: Sơ đồ Usse-case của người quản lí

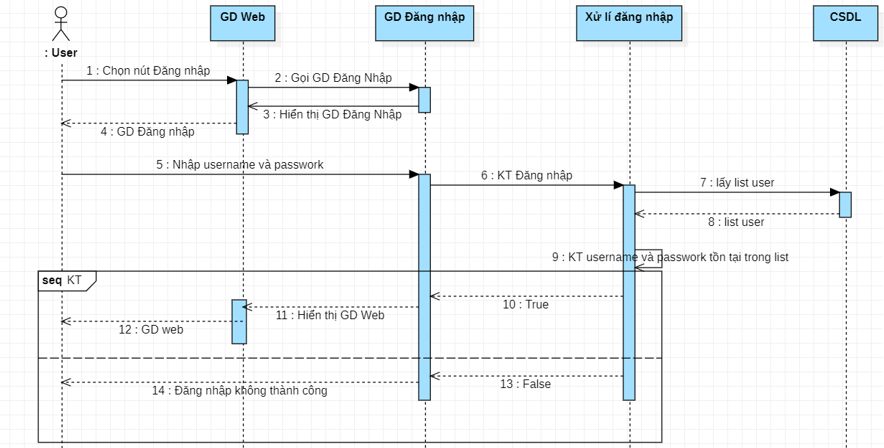
* 1. **Biểu đồ phân rã chức năng**

Mô hình phân rã chức năng (BFD – Business Function Diagram) là công cụ biểu diễn việc phân rã có thứ bậc đơn giản các công việc cần thực hiện. Mỗi công việc được chia ra làm các công việc con, số mức chia ra phụ thuộc kích cỡ và độ phức tạp của hệ thống.



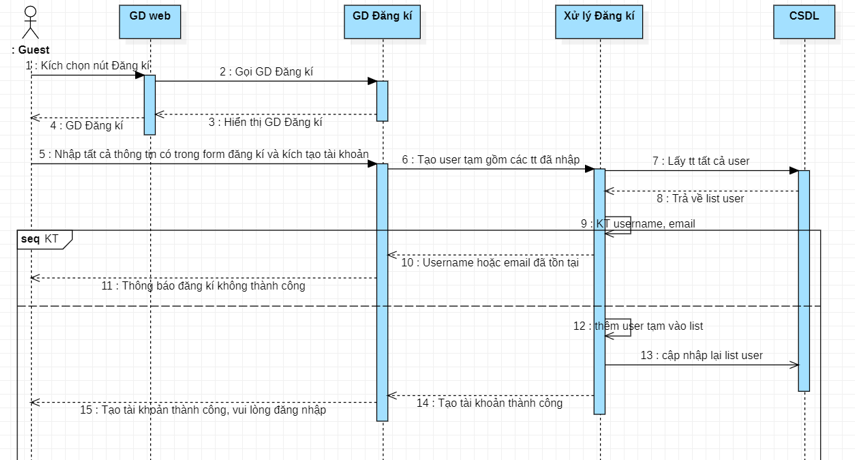
Hình 6: Biểu đồ phân cấp chức năng hệ thống

* 1. **Biểu đồ tuần tự**
     1. **Biểu đồ tuần tự đăng nhập**

****

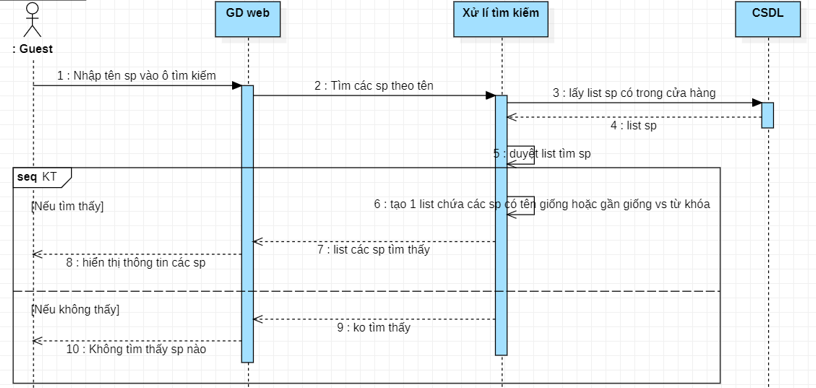
Hình 7: Biêu đồ tuần tự đăng nhập

* + 1. **Biểu đồ tuần tự đăng kí thành viên**



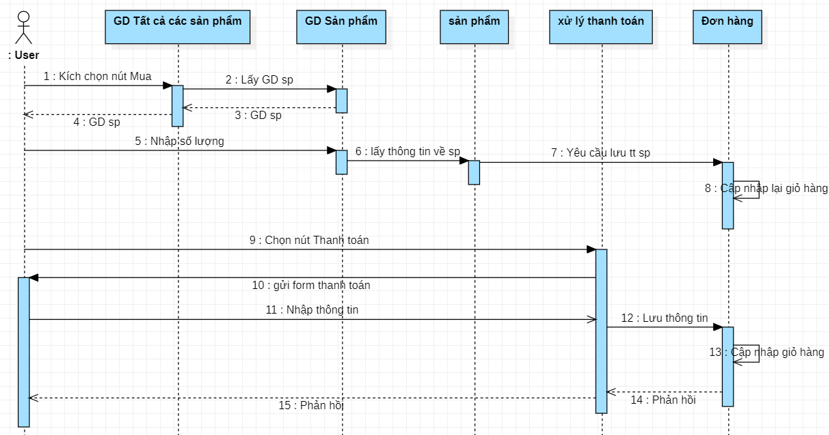
Hình 8: Biểu đồ tuần tự đăng kí thành viên

* + 1. **Biểu đồ tuần tự tìm kiếm sản phẩm**



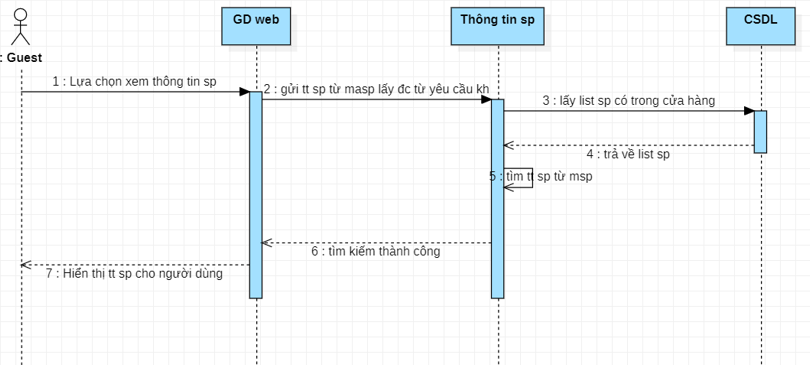
Hình 9: Biểu đồ tuần tự tìm kiếm sản phẩm

* + 1. **Biểu đồ tuần tự đặt hàng**



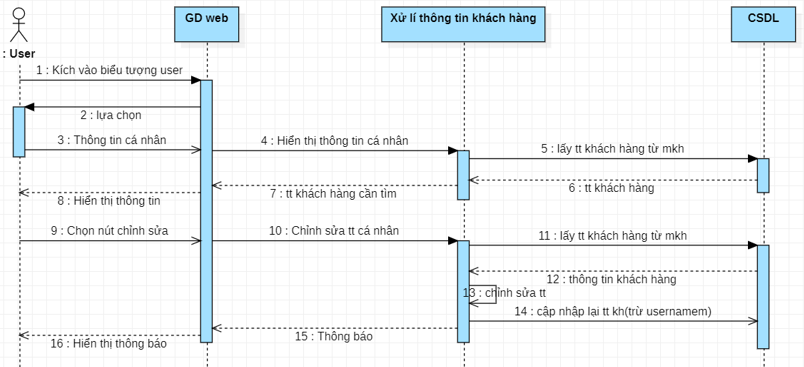
Hình 10: Biểu đồ tuần tự đặt hàng

* + 1. **Biểu đồ tuần tự xem thông tin sản phẩm**



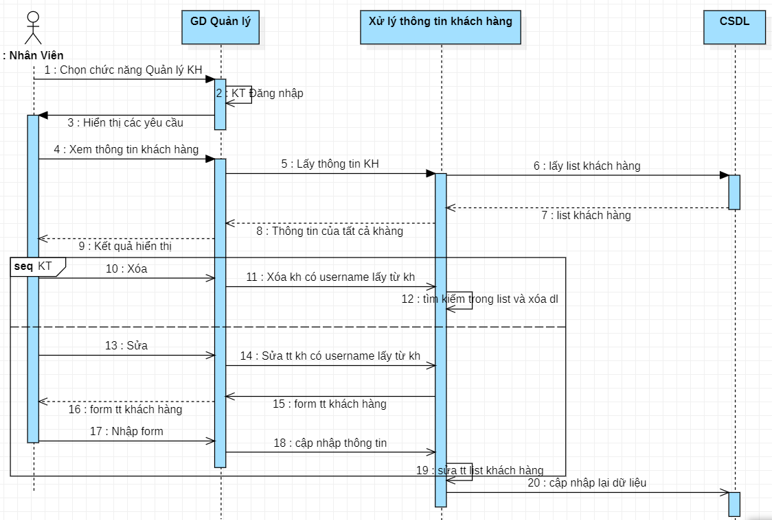
Hình 11: Biểu đồ tuần tự xem thông tin sản phẩm

* + 1. **Biểu đồ tuần tự thay đổi thông tin cá nhân**



Hình 12: Biểu đồ tuần tự thay đổi thông tin cá nhân

* + 1. **Biểu đồ tuần tự quản lí khách hàng**



Hình 13: Biểu đồ tuần tự quản lí khách hàng

* 1. **Thiết kế cơ sở dữ liệu**

Khi xây dựng một ứng dụng, việc thiết kế cơ sở dữ liệu đúng chuẩn và đáp ứng được truy cập là rất quan trọng. Một cơ sở dữ liệu tốt là cơ sở dữ liệu đủ tiêu chuẩn đáp ứng được tối thiểu chuẩn 3NF. Trong ứng dụng này, cơ sở dữ liệu sử dụng sẽ là MySQL.

Sau đây là các bảng dữ liệu được thiết kế:

Graphical user interface, text, application, email

Description automatically generated

Bảng 2 : Đặc tả dữ liệu blog

Graphical user interface, text, application, email

Description automatically generatedBảng 3: Thực thể danh mục sản phẩm

Graphical user interface, text, application, email

Description automatically generatedGraphical user interface, text, application, email

Description automatically generatedBảng 4: Thực thể thương hiệu sản phẩm

Bảng 5 Thực thể giỏ hàng

Graphical user interface, text, application, email

Description automatically generated

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Users** | | | | | |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Khóa chính** | **Not Null** | **Khóa ngoại** |
| 1 | id | int(11) | **X** | **X** |  |
| 2 | ip\_address | varchar(255) |  | **X** |  |
| 3 | name | text |  | **X** |  |
| 4 | email | varchar(255) |  | **X** |  |
| 5 | password | varchar(100) |  | **X** |  |
| 6 | country | text |  | **X** |  |
| 7 | city | text |  | **X** |  |
| 8 | contact | Varchar(45) |  | **X** |  |
| 9 | user\_address | BIGINT |  | **X** |  |
| 10 | image | varchar(255) |  | **X** |  |
| 11 | role | varchar(50) |  | **X** |  |

Bảng 6: Bảng thực thể user

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **migrations** | | | | | |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Khóa chính** | **Not Null** | **Khóa ngoại** |
| 1 | migra\_id | int(100) | X | X |  |
| 2 | migra\_title | varchar(100) |  | X |  |

Bảng 7 : Thực thể migratons

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **history** | | | | | |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Khóa chính** | **Not Null** | **Khóa ngoại** |
| 1 | id | int(11) | **X** | **X** |  |
| 2 | ip\_address | varchar(255) |  | **X** |  |
| 3 | name | text |  | **X** |  |
| 4 | email | varchar(255) |  | **X** |  |
| 5 | phone | varchar(100) |  | **X** |  |
| 6 | country | text |  | **X** |  |

Bảng 8: Lịch sử mua hàng

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **comment** | | | | | |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Khóa chính** | **Not Null** | **Khóa ngoại** |
| 1 | id | int(11) | **X** | **X** |  |
| 2 | ip\_address | varchar(255) |  | **X** |  |
| 3 | name | text |  | **X** |  |
| 4 | content | varchar(255) |  | **X** |  |
| 5 | avatar | varchar(100) |  | **X** |  |

Bảng 9: Thực thể bảng comment

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **country** | | | | | |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Khóa chính** | **Not Null** | **Khóa ngoại** |
| 1 | brand\_id | int(100) | X | X |  |
| 2 | brand\_title | varchar(100) |  | X |  |
| 3 | name | text |  | X |  |

Bảng 10: Thực thể bảng Country

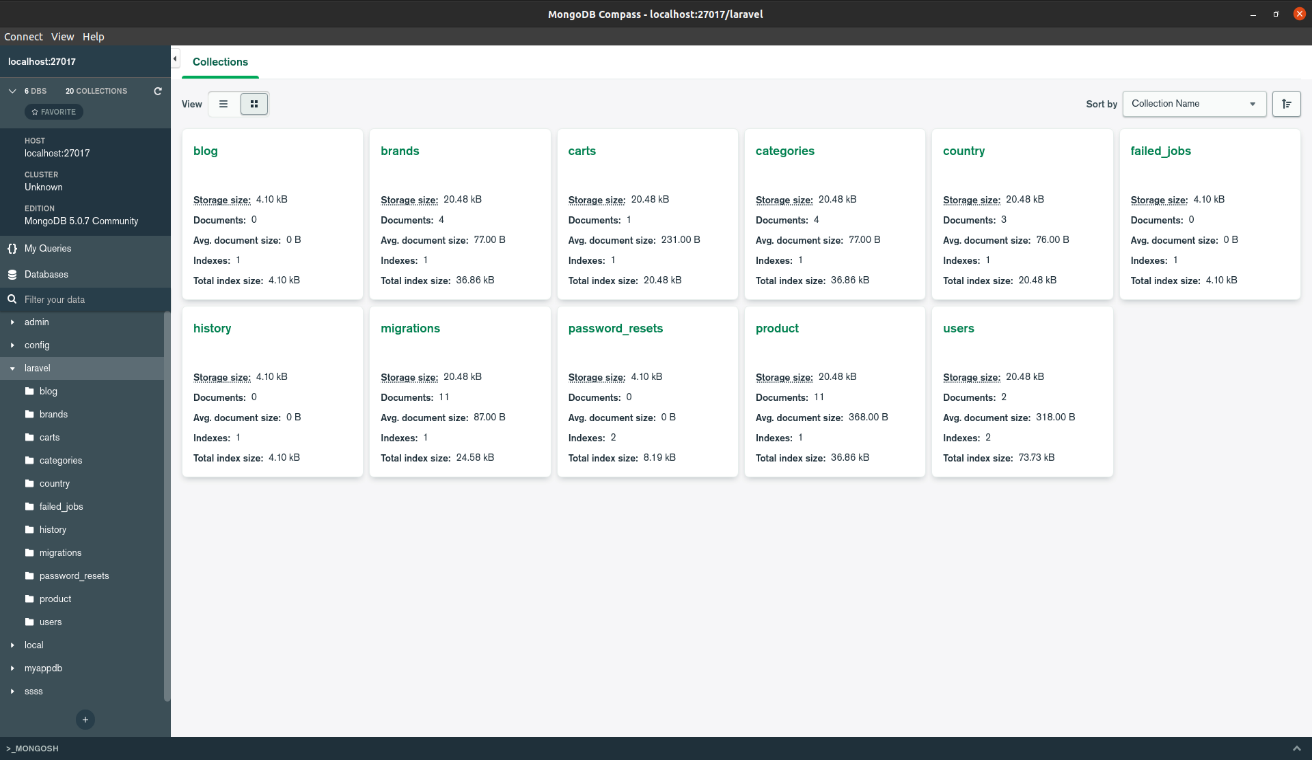
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Blog** | | | | | |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Khóa chính** | **Not Null** | **Khóa ngoại** |
| 1 | id | int(11) | **X** | **X** |  |
| 2 | ip\_address | varchar(255) |  | **X** |  |
| 3 | Blog\_title | varchar(100) |  | **X** |  |
| 4 | image | string |  | **X** |  |
| 5 | phone | number |  | **X** |  |
| 6 | country | text |  | **X** |  |

Bảng 11: Thực thế bảng Blog

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **brands** | | | | | |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Khóa chính** | **Not Null** | **Khóa ngoại** |
| 1 | brands\_id | int(100) | X | X |  |
| 2 | brands\_title | varchar(100) |  | X |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **product** | | | | | |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Khóa chính** | **Not Null** | **Khóa ngoại** |
| 1 | id | int(11) | **X** | **X** |  |
| 2 | ip\_address | varchar(255) |  | **X** |  |
| 3 | name | text |  | **X** |  |
| 4 | image | string |  | **X** |  |
| 5 | company | varchar(100) |  | **X** |  |
| 6 | brand | varchar(100) |  | **X** |  |
| 7 | category | varchar(100) |  | **X** |  |

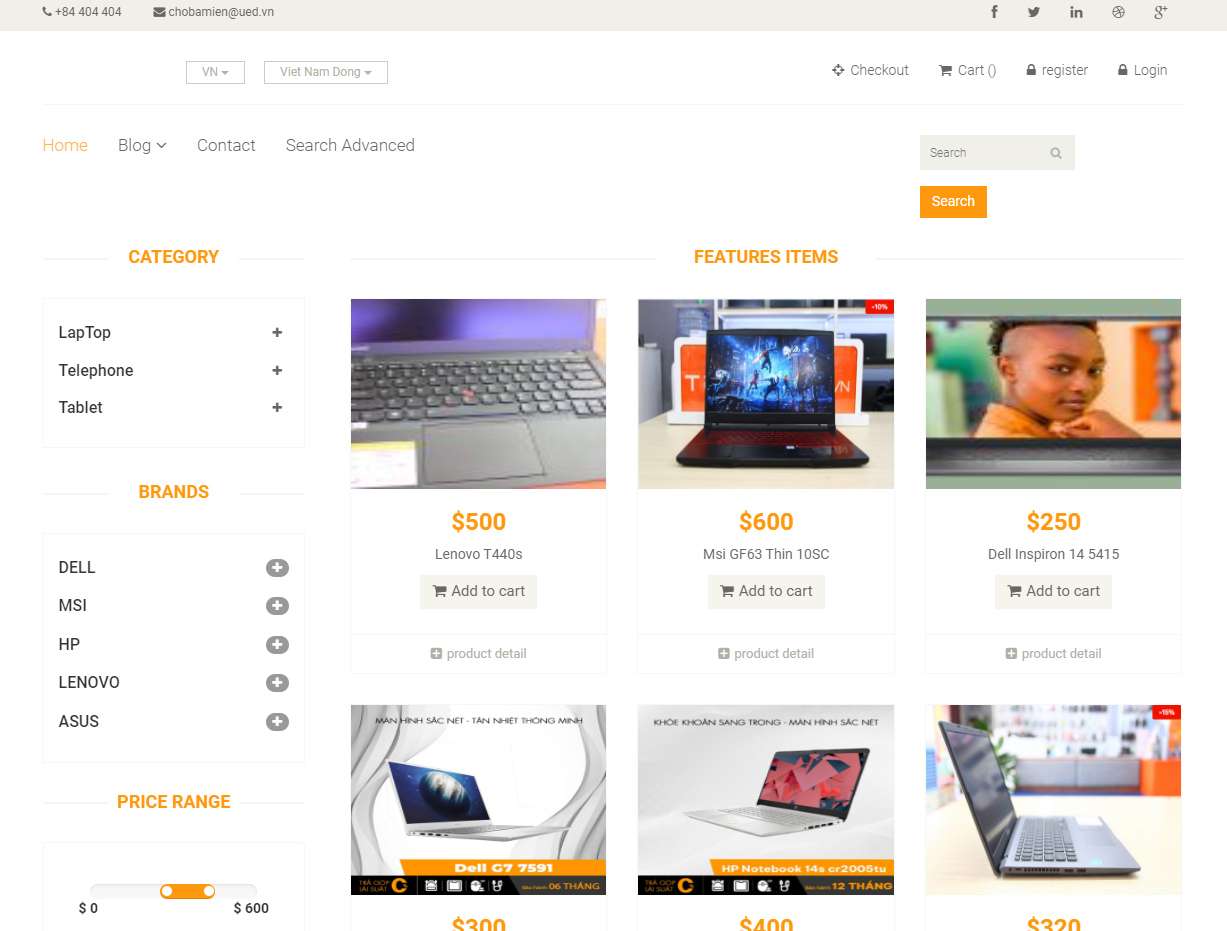
* + 1. **Mô hình cơ sở dữ liệu**



Hình 14: Sơ đồ quan hệ Cơ sở dữ liệu

**CHƯƠNG III: CÀI ĐẶT CHƯƠNG TRÌNH VÀ KẾT QUẢ**

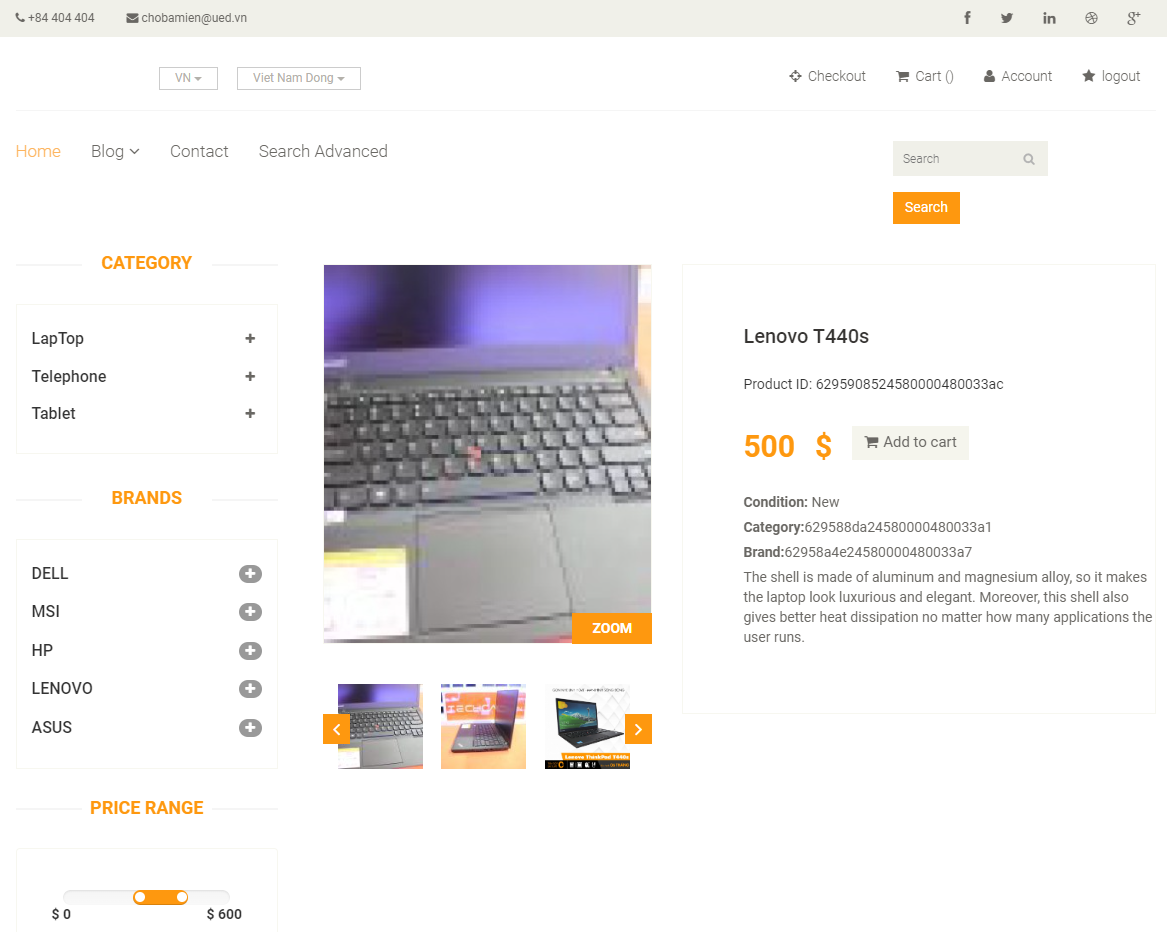
* 1. **Kết quả chương trình**
     1. **Giao diện trang chủ**

Đây là giao diện trang chủ của trang web. Với trang này người quản lí, khách hàng kể cả đã là thành viên hay chưa là thành viên đều có thể truy cập. Trang web cung cấp các thông tin khuyến mãi, các món ăn đang bán và sắp bán cũng như thông tin của cửa hàng.

Hình 15: Giao diện trang chủ

* + 1. **Giao diện chi tiết sản phẩm**

Đây là trang cung cấp thông tin chi tiết về sản phẩm được chọn mà sau khi người dùng click vào.

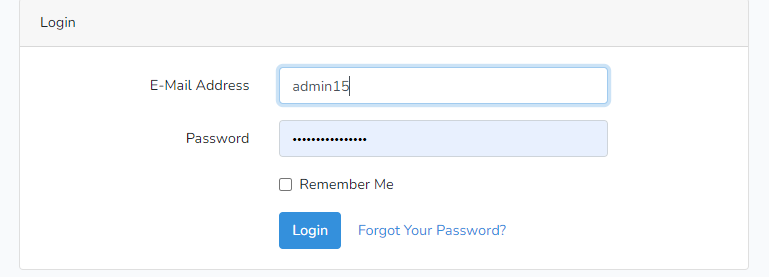


Hình 16: Giao diện chi tiết sản phẩm

* + 1. **Giao diện đăng nhập**

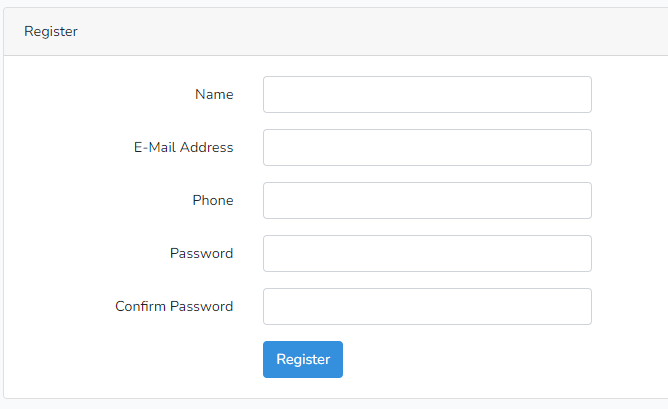
Hình 17: Giao diện đăng nhập

Người dùng sẽ đăng nhập vào tài khoản để đảm bảo việc mua hàng và thanh toán.



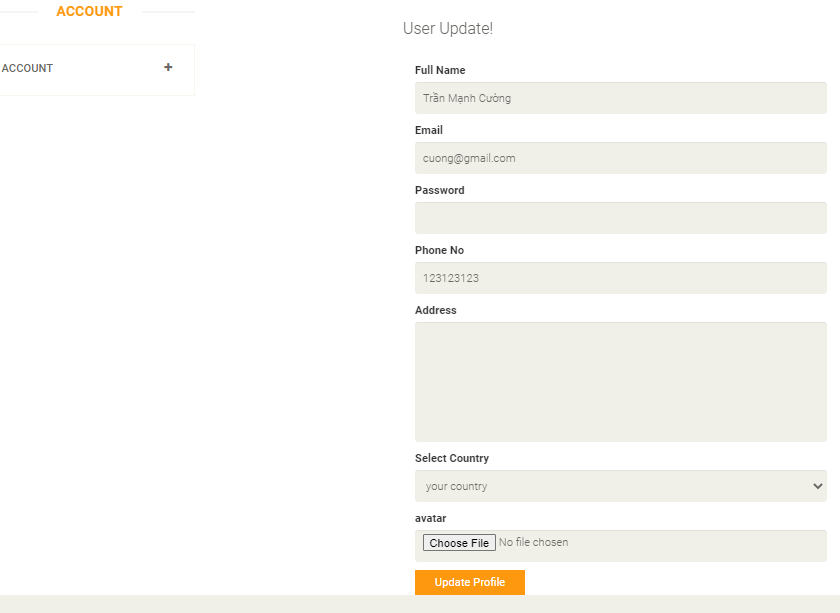
* + 1. **Giao diện đăng kí**

Người dùng sẽ đăng kí tài khoản để trở thành thành viên của cửa hàng

.

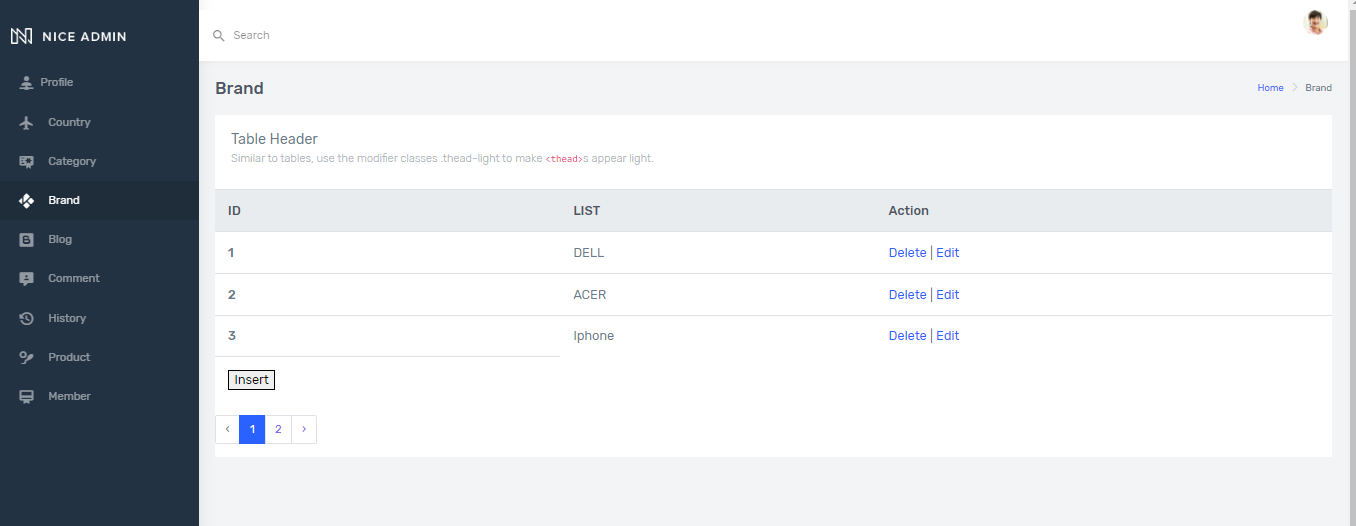
Hình 18: Giao diện đăng kí

* + 1. **Giao diện quản lý tài khoản**

Sau khi click vào quản lý tài khoản, hệ thống sẽ chuyển đến trang cá nhân của thành viên. 

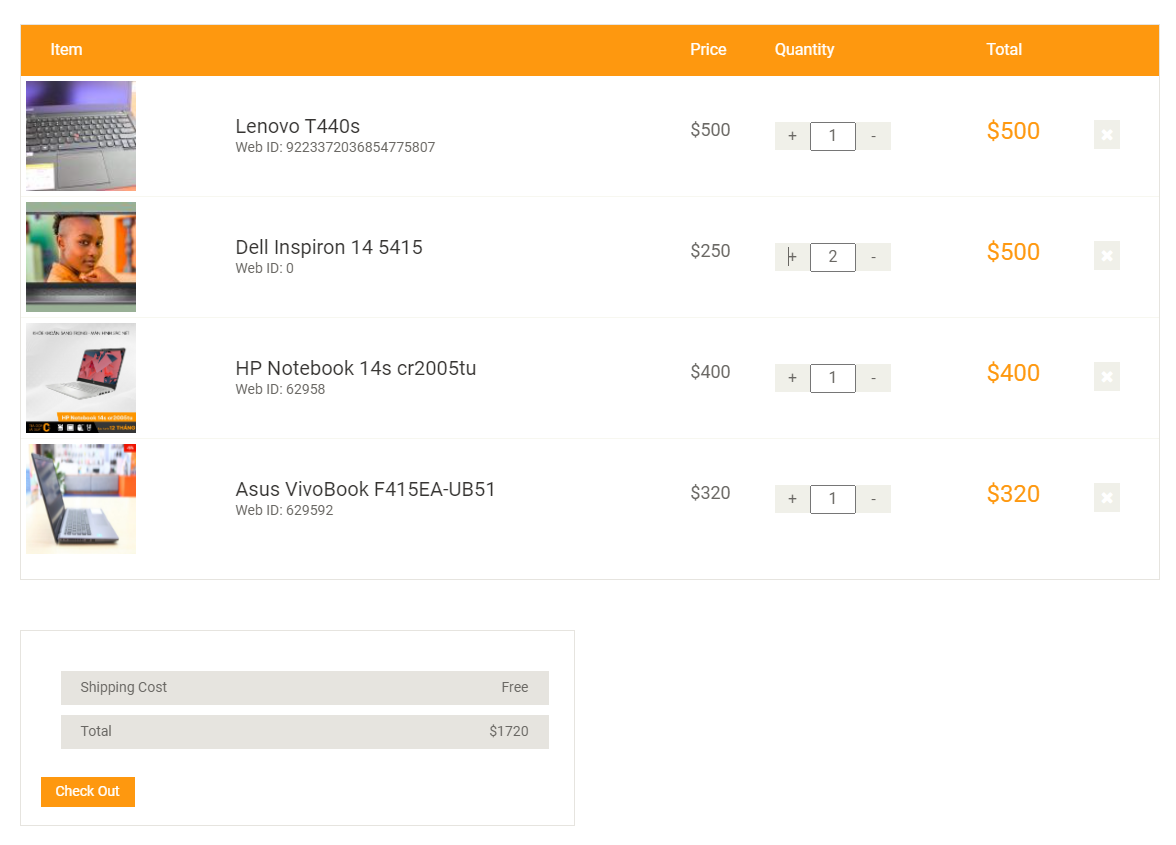
Hình 19: Giao diện quản lí tài khoản

* + 1. **Giao diện quản lý của Admin**

Nếu sau khi đăng nhập và hệ thông xác nhận có vai trò là người quản lý thì sẽ có thêm tuỳ chọn quản trị website và sau khi nhấn vào sẽ được chuyển hướng đến giao diện quản lý như phía dưới

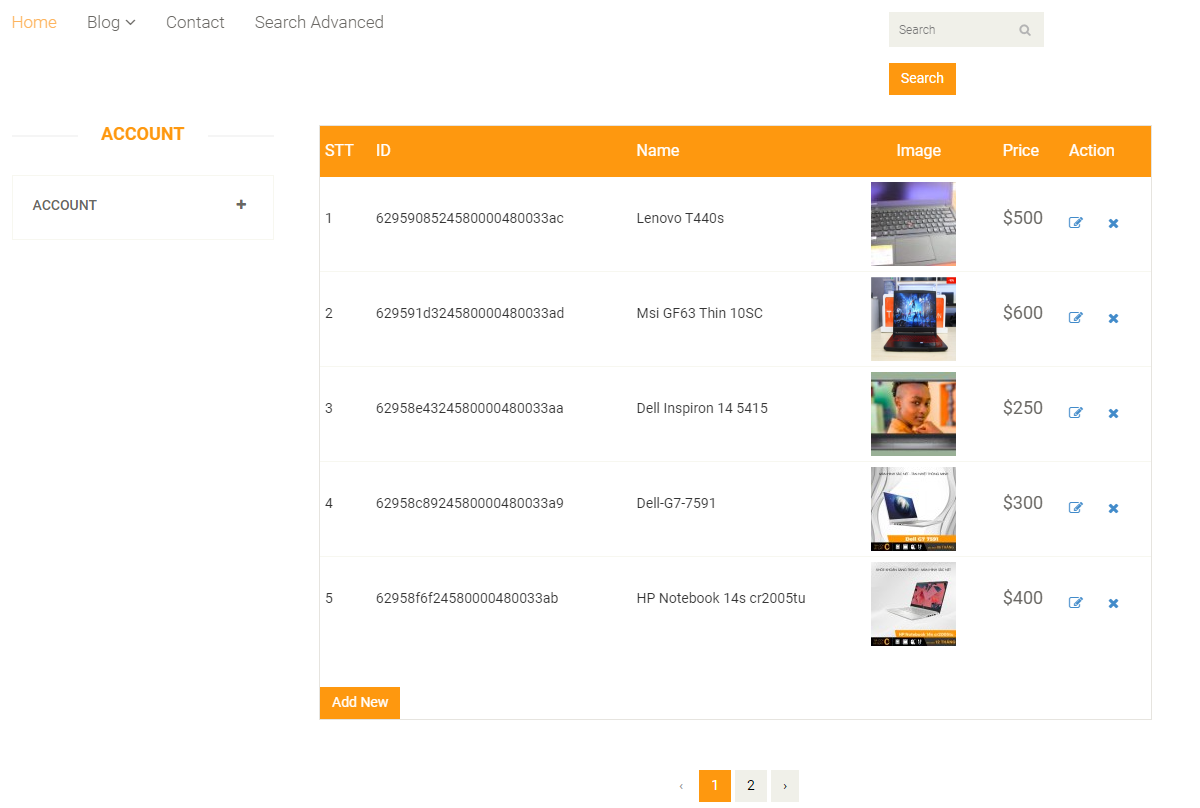
Hình 20: Giao diện quản lí Admin

* + 1. **Giao diện giỏ hàng**

Người dùng chọn sản phẩm thì nó sẽ được đưa vào giỏ hàng sau đó vào giỏ hàng dể thanh toán cũng lúc nhiều sản phẩm trong một lần giao dịch.

Hình 21: Giao diện giỏ hàng

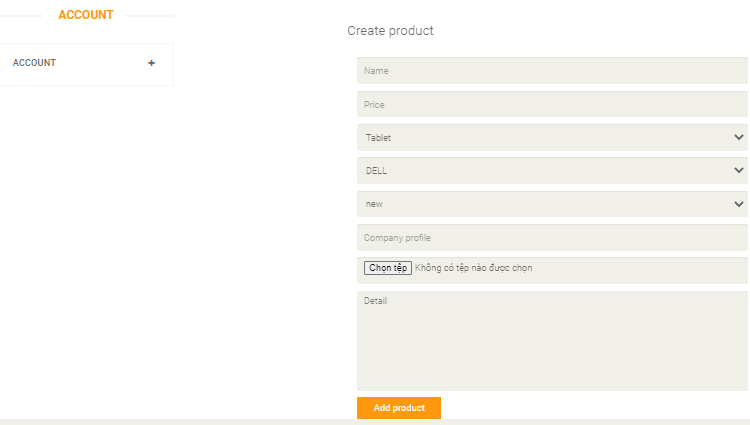
* + 1. **Giao diện quản lí sản phẩm**

Trong trang này, người quản lí kiểm soát được toàn bộ thông tin của các sản phẩm. 

Hình 22: Giao diện quản lí sản phẩm

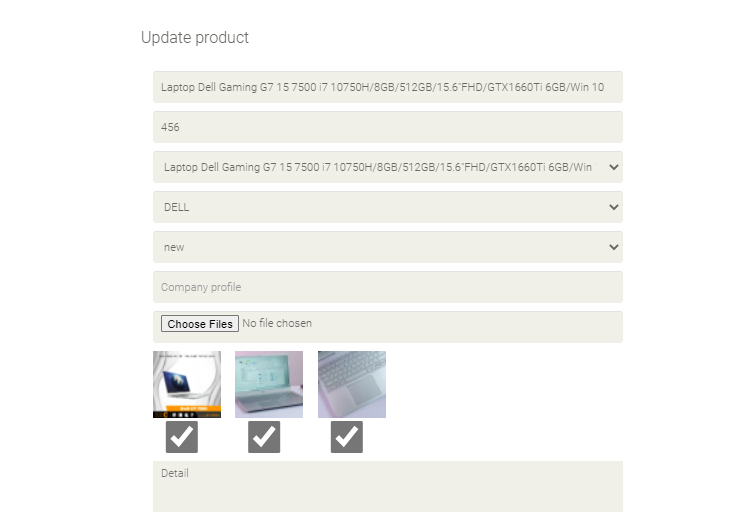
* + 1. **Giao diện thêm sản phẩm**

Trong trang này, người quản lí có thể thêm sản phẩm mới vào website.



Hình 23: Giao diện thêm sản phẩm

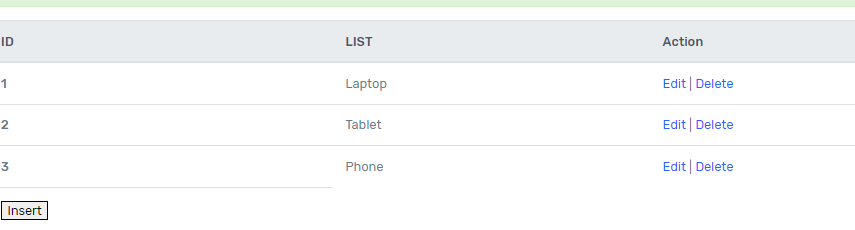
* + 1. **Giao diện chỉnh sửa thông tin sản phẩm**

Trong trang này, người quản lí có thể chỉnh sửa mọi thông tin của sản phẩm bất kì. 

Hình 24: Giao diện chỉnh sửa thông tin sản phẩm

* + 1. **Giao diện quản lí danh mục sản phẩm**

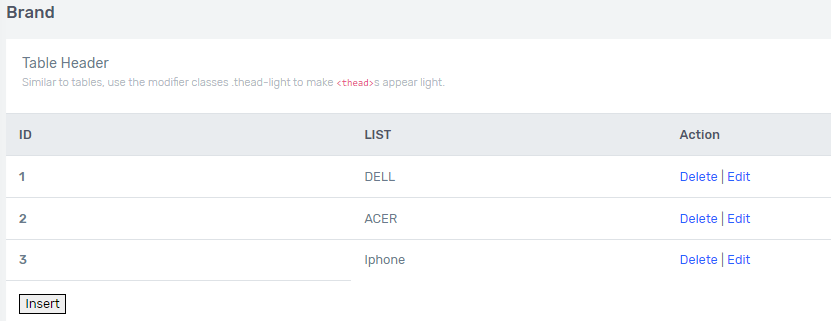
Trong trang này, người quản lí kiểm soát được toàn bộ thông tin của các danh mục sản phẩm đang có trong website với chức năng thêm, sửa, xoá.



Hình 25: Giao diện thêm danh mục sản phẩm

* + 1. **Giao diện quản lí thương hiệu sản phẩm**

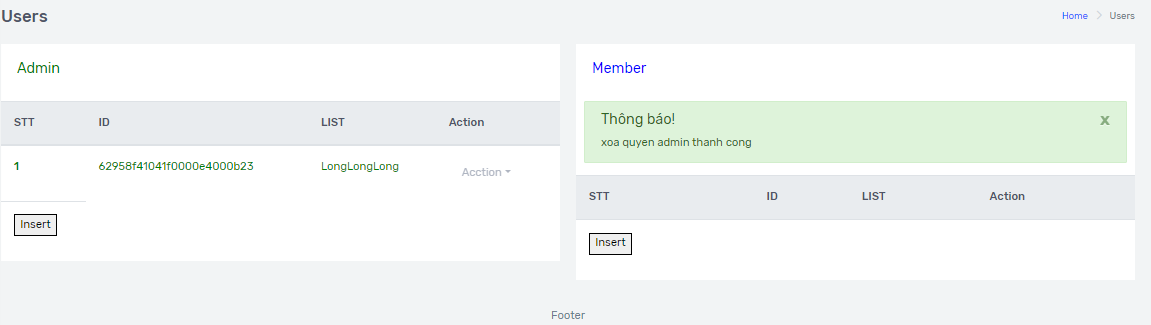
Trong trang này, người quản lí kiểm soát được toàn bộ thông tin của các thương hiệu của các sản phẩm đang hiện có với chức năng thêm, sửa, xoá.



Hình 27: Giao diện thêm thương hiệu sản phẩm

* + 1. **Giao diện quản lí thành viên**

Trong trang này, người quản lí kiểm soát được toàn bộ thông tin của các tài khoản thành viên với chức năng xoá thành viên.



Hình 28: Giao diện quản lí thương hiệu sản phẩm

# **KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

* 1. **Kết quả đạt được**

**Về kiến thức và học tập**

Đối với lần nghiên cứu này, chúng em đã xem đây là một cơ hội và cũng là một thử thách đối với bản thân để tự học hỏi, rèn luyện những điều mới lạ từ các ngôn ngữ lập trình hay framework của chúng. Khi tiếp xúc và thực hành dùng các ngôn ngữ, em đều nhận ra được sự thú vị, đặc trưng riêng biệt mà chúng hỗ trợ. Bên cạnh đó, việc áp dụng các kiến thức đã học và tự học cũng là điều thu hút em. Điển hình như ngôn ngữ PHP, HTML, CSS, Javascript… đều đã được thầy cô truyền tải cho chúng em trên giảng đường, bây giờ, nó lại trở thành kiến thức căn bản, nền tảng để em tiếp tục con đường tự tìm tòi, học hỏi. Đặc biệt, về khả năng phân tích, thiết kế hệ thống hay giải quyết các vấn đề trong quá trình nghiên cứu đều một lần nữa được củng cố, áp dụng một cách linh hoạt. Xây dựng được cơ sở dữ liệu trong quá trình nghiên cứu và phân tích hệ thống nhằm đáp ứng tốt cho việc viết chương trình. Và không quên chú trọng vào việc tìm hiểu và nghiên cứu cấu trúc của hệ thống để đảm bảo trải nghiệm tốt nhất cho người dùng.

**Về phần mềm**

Phần mềm được viết bằng ngôn ngữ PHP đóng vai trò như là một server để như xử lí logic, để tìm kiếm, lưu trữ, hoặc thay đổi dữ liệu và phục vụ trở lại người dùng. Phần mềm cũng được viết bằng ngôn ngữ Javascript (thông qua thư viện jQuery) đóng vai trò như một Client, hệ thống các giao diện người sử dụng để tương tác trực tiếp với người dùng. Phần mềm được viết rõ ràng các phần liên quan đến các ngôn ngữ nên hệ thống được vận hành một cách nhanh chóng và chính xác, đảm bảo trải nghiệm cho người dùng một cách tốt nhất. Bên cạnh đó việc tìm kiếm và đặt hàng mà mục tiêu của hệ thống nhằm hướng đến người sử dụng là khách hàng nên việc đảm bảo tốc độ tải trang, khả năng truy xuất là thực sự cần thiết. Bên cạnh đó việc thiết kế bố cục, tương phản đều được châu chuốt kĩ lưỡng để thu hút sự thích thú của người tiêu dùng. Về phía hệ thống, xử lí tốt các dữ liệu, yêu cầu người dùng gửi đến, đảm bảo tính đúng đắn, nhanh chóng. Thêm vào đó, hệ thống còn cung cấp được một số tính năng cơ bản cho người quản lí và khách hàng đúng nhu cầu.

* 1. **Hạn chế của đề tài**

Hệ thống dành cho khách hàng và cả người quản lí thật sự là một hệ thống lớn, cần đầu tư thời gian, công sức và cả sự nỗ lực để hoàn thiện. Song, do thời gian nghiên cứu có hạn, hệ thống mà nhóm đã phân tích và thiết kế vẫn còn nhiều hạn chế:

* Chưa bổ sung thêm vai trò nhân viên để có thể đặt đồ ăn giúp khách hàng.
* Chưa thực hiện được chức năng thanh toán online qua các thẻ tín dụng, ví điện tử…
* Tính bảo bảo mật cho hệ thống chưa được hoàn chỉnh.
* Kiến thức thực tế và kiến thức lập trình còn hạn chế nên tính chuyên nghiệp của chương trình chưa cao, bắt lỗi chưa kĩ ở một số chổ và một số tính năng chưa tối ưu.
* Giao diện chưa được hoàn chỉnh như ý muốn của nhóm đã đưa ra.
  1. **Hướng phát triển**
* B[ổ sung những tính năng mới, công nghệ mới vào đề tài nhằm khắc phục những hạn chế của đề tài và phát triển một hệ thống hoàn chỉnh, thân thiện với người sử dụng.](https://docs.google.com/spreadsheets/d/1DUF2isFWsqVSYhbaACYtbgcLi_YjDqpE3GLQIVgkKQg/edit#gid=69851113)
* Xây dựng website hoàn chỉnh hơn về mặt chức năng cũng như mặt giao diện

**TÀI LIỆU THAM KHẢO**

1. https://www.w3schools.com/php/default.asp
2. https://www.mongodb.com/docs/v5.0/
3. https://laravel.com/docs/7.x
4. https://stackoverflow.com/
5. https://youtu.be/ImtZ5yENzgE