**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HƯNG YÊN**

**ĐẶNG NGỌC NHÂN**

**THIẾT KẾ WEBSITE BÁN ĐỒ GIA DỤNG**

**ĐỒ ÁN MÔN HỌC 2**

**HƯNG YÊN - 2022**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HƯNG YÊN**

**ĐẶNG NGỌC NHÂN**

**THIẾT KẾ WEBSITE BÁN GIA DỤNG**

NGÀNH: CÔNG NGHỆ THÔNG TIN

CHUYÊN NGÀNH: PHÁT TRIỂN ỨNG DỤNG IOT

**ĐỒ ÁN MÔN HỌC 2**

**NGƯỜI HƯỚNG DẪN**

**TS. PHẠM NGỌC HƯNG**

**HƯNG YÊN - 2022**

Nhận xét của giảng viên 1 đánh giá quá trình:

Ký và ghi họ tên

Nhận xét của giảng viên 2 đánh giá quá trình:

Ký và ghi họ tên

Nhận xét của giảng viên hướng dẫn:

Ký và ghi họ tên

**LỜI CAM ĐOAN**

Tôi xin cam đoan đồ án “Thiết kế website bán đồ gia dụng” là công trình nghiên cứu của bản thân. Những nội dung sử dụng trong đồ án không sao chép của bất cứ tài liệu nào. Những nội dung trích dẫn được thực hiện đúng theo quy định về vi phạm bản quyền. Các kết quả trình bày trong đồ án hoàn toàn là kết quả do bản thân tôi và các thành viên trong nhóm thực hiện, nếu sai tôi xin chịu hoàn toàn trách nhiệm trước khoa và nhà trường.

*Hưng yên, ngày … tháng … năm…...*

Sinh viên

…………………………………..

**LỜI CÁM ƠN**

Để có thể hoàn thành đồ án này, lời đầu tiên em xin phép gửi lời cảm ơn tới bộ môn Hệ thống thông tin, Khoa Công nghệ thông tin – Trường Đại học Sư phạm Kỹ thuật Hưng yên đã tạo điều kiện thuận lợi cho em thực hiện đồ án môn học này.

Đặc biệt em xin chân thành cảm ơn thầy Phạm Ngọc Hưng đã rất tận tình hướng dẫn, chỉ bảo em trong suốt thời gian thực hiện đồ án vừa qua.

Em cũng xin chân thành cảm ơn tất cả các Thầy, các Cô trong Trường đã tận tình giảng dạy, trang bị cho em những kiến thức cần thiết, quý báu để giúp em thực hiện được đồ án này.

Mặc dù em đã có cố gắng, nhưng với trình độ còn hạn chế, trong quá trình thực hiện đề tài không tránh khỏi những thiếu sót. Em hi vọng sẽ nhận được những ý kiến nhận xét, góp ý của các Thầy giáo, Cô giáo về những kết quả triển khai trong đồ án.

Em xin trân trọng cảm ơn!

**MỤC LỤC**

[DANH SÁCH HÌNH VẼ 7](#_Toc106835286)

[DANH SÁCH BẢNG BIỂU 9](#_Toc106835287)

[DANH SÁCH TỪ VIẾT TẮT 11](#_Toc106835288)

[CHƯƠNG I: TỔNG QUAN VỀ ĐỒ ÁN 12](#_Toc106835289)

[1.1. Lý do chọn đồ án 12](#_Toc106835290)

[1.2. Mục tiêu của đồ án 13](#_Toc106835291)

[1.3. Giới hạn và phạm vi của đồ án 13](#_Toc106835292)

[1.3.1. Đối tượng nghiên cứu 13](#_Toc106835293)

[1.3.2. Phạm vi nghiên cứu 13](#_Toc106835294)

[1.4. Nội dung thực hiện 13](#_Toc106835295)

[1.5. Phương pháp tiếp cận 14](#_Toc106835296)

[CHƯƠNG II: CƠ SỞ LÝ THUYẾT 15](#_Toc106835297)

[2.1. Quy trình phát triển phần mềm 15](#_Toc106835298)

[2.2. Thiết kế giao diện web với HTML, CSS, JavaScript 16](#_Toc106835299)

[2.2.1. Giới thiệu HTML 16](#_Toc106835300)

[2.2.2. Giới thiệu CSS 19](#_Toc106835301)

[2.2.3. Giới thiệu JavaScript 27](#_Toc106835302)

[CHƯƠNG III: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 37](#_Toc106835303)

[3.1. Đặc tả yêu cầu Website 37](#_Toc106835304)

[3.2. Các yêu cầu chức năng 38](#_Toc106835305)

[3.2.1. Biểu đồ dòng dữ liệu 39](#_Toc106835306)

[3.3. Thiết kế cơ sở dữ liệu cho hệ thống 46](#_Toc106835307)

[3.3.1. Thiết kế mô hình thực thể liên kết 46](#_Toc106835308)

[3.3.2. Mô hình thực thể liên kết – ERM 50](#_Toc106835309)

[3.3.3. Thiết kế các bảng dữ liệu 50](#_Toc106835310)

[3.3.4. Liên kết giữa các bảng Relationships 56](#_Toc106835311)

[3.4. Thiết kế giao diện cho hệ thống 57](#_Toc106835312)

[3.4.1. Đối với admin 57](#_Toc106835313)

[3.4.2. Đối với khách hàng 61](#_Toc106835314)

[Chương IV: TRIỂN KHAI WEBSITE 69](#_Toc106835315)

[4.1. Kết nối CSDL 69](#_Toc106835316)

[4.1.1. Giới thiệu ngôn ngữ lập trình PHP 69](#_Toc106835317)

[4.1.2. Kết nối với Database 70](#_Toc106835318)

[4.2. Triển khai các chức năng cho hệ thống 71](#_Toc106835319)

[4.2.1. Đối với khách hàng 71](#_Toc106835320)

[4.2.2. Đối với nhà quản trị 77](#_Toc106835321)

[KẾT LUẬN 89](#_Toc106835322)

[TÀI LIỆU THAM KHẢO 90](#_Toc106835323)

# 

# DANH SÁCH HÌNH VẼ

[Hình 2.1: Các thuộc tính đối tượng Document 32](#_Toc106835324)

[Hình 3.1: Biểu đồ phân ra chức năng 38](#_Toc106835325)

[Hình 3.2: Biểu đồ dòng dữ liệu ngữ cảnh 39](#_Toc106835326)

[Hình 3.3: Biểu đồ dòng dữ liệu mức 0 40](#_Toc106835327)

[Hình 3.4: Mức 1 chức năng quản lý người dùng 42](#_Toc106835328)

[Hình 3.5: Mức 1 chức năng quản lý bán hàng 43](#_Toc106835329)

[Hình 3.6: Chức năng quản lý sản phẩm 44](#_Toc106835330)

[Hình 3.7: Chức năng thống kê 45](#_Toc106835331)

[Hình 3.8: Mô hình thực thể liên kết - ERM 50](#_Toc106835332)

[Hình 3.9: Liên kết giữa các bảng Relationships 56](#_Toc106835333)

[Hình 3.10: Giao diện trang đăng nhập admin 57](#_Toc106835334)

[Hình 3.11: Giao diện trang chủ admin 57](#_Toc106835335)

[Hình 3.12: Giao diện trang thêm danh mục 58](#_Toc106835336)

[Hình 3.13: Giao diện trang danh sách danh mục sản phẩm 58](#_Toc106835337)

[Hình 3.14: Giao diện trang thêm loại sản phẩm 59](#_Toc106835338)

[Hình 3.15: Giao diện trang danh sách loại sản phẩm 59](#_Toc106835339)

[Hình 3.16: Giao diện trang thêm sản phẩm 60](#_Toc106835340)

[Hình 3.17: Giao diện trang danh sách sản phẩm 60](#_Toc106835341)

[Hình 3.18: Giao diện trang chủ 61](#_Toc106835342)

[Hình 3.19: Giao diện trang sản phẩm 62](#_Toc106835343)

[Hình 3.20: Giao diện trang chi tiết sản phẩm 63](#_Toc106835344)

[Hình 3.21: Giao diện trang giỏ hàng 64](#_Toc106835345)

[Hình 3.22: Giao diện trang giao hàng 65](#_Toc106835346)

[Hình 3.23: Giao diện trang thanh toán 66](#_Toc106835347)

[Hình 3.24: Giao diện trang đặt hàng thành công 67](#_Toc106835348)

[Hình 3.25: Giao diện trang chi tiết đơn hàng 68](#_Toc106835349)

[Hình 4.1: Giới thiệu ngôn ngữ lập trình PHP 69](#_Toc106835350)

[Hình 4.2: Lưu đồ thuật toán hiển thị sản phẩm 72](#_Toc106835351)

[Hình 4.3: Lưu đồ thuật toán hiển thị chi tiết sản phẩm 73](#_Toc106835352)

[Hình 4.4: Lưu đồ thuật toán chức năng tạo giỏ hàng 74](#_Toc106835353)

[Hình 4.5: Lưu đồ thuật toán chức năng xóa giỏ hàng 75](#_Toc106835354)

[Hình 4.6: Lưu đồ thuật toán chức năng tạo đơn hàng 76](#_Toc106835355)

[Hình 4.7: Lưu đồ thuật toán hiển thị tình trạng đơn hàng 77](#_Toc106835356)

[Hình 4.8: Lưu đồ thuật toán đăng nhập quản trị 78](#_Toc106835357)

[Hình 4.9: Lưu đồ thuật toán thêm danh mục sản phẩm 79](#_Toc106835358)

[Hình 4.10: Lưu đồ thuật toán sửa danh mục sản phẩm 80](#_Toc106835359)

[Hình 4.11: Lưu đồ thuật toán xóa danh mục sản phẩm 80](#_Toc106835360)

[Hình 4.12: Lưu đồ thuật toán thêm loại sản phẩm 81](#_Toc106835361)

[Hình 4.13: Lưu đồ thuật toán sửa loại sản phẩm 82](#_Toc106835362)

[Hình 4.14: Lưu đồ thuật toán xóa loại sản phẩm 82](#_Toc106835363)

[Hình 4.15: Lưu đồ thuật toán thêm sản phẩm 83](#_Toc106835364)

[Hình 4.16: Lưu đồ thuật toán sửa sản phẩm 84](#_Toc106835365)

[Hình 4.17: Lưu đồ thuật toán xóa sản phẩm 85](#_Toc106835366)

[Hình 4.18: Lưu đồ thuật toán thông kê đơn hàng 86](#_Toc106835367)

[Hình 4.19: Lưu đồ thuật toán xác nhận đơn hàng 87](#_Toc106835368)

[Hình 4.20: Lưu đồ thuật toán xóa đơn hàng 88](#_Toc106835369)

# DANH SÁCH BẢNG BIỂU

[Bảng 2. 1: Các bộ chọn CSS 21](#_Toc106824250)

[Bảng 3.1: Danh sách các thực thể của hệ thống 46](#_Toc106835370)

[Bảng 3.2: Thực thể tbl\_admin 47](#_Toc106835371)

[Bảng 3.3: Thực thể tbl\_danhmuc 47](#_Toc106835372)

[Bảng 3.4: Thực thể tbl\_loaisanpham 47](#_Toc106835373)

[Bảng 3.5: Thực thể tbl\_sanpham 47](#_Toc106835374)

[Bảng 3.6: Thực thể tbl\_sanpham\_anh 48](#_Toc106835375)

[Bảng 3.7: Thực thể tbl\_cart 48](#_Toc106835376)

[Bảng 3.8: Thực thể tbl\_diachi 48](#_Toc106835377)

[Bảng 3.9: Thực thể tbl\_order 49](#_Toc106835378)

[Bảng 3.10: Thực thể tbl\_payment 49](#_Toc106835379)

[Bảng 3.11: Thực thể tbl\_carta 49](#_Toc106835380)

[Bảng 3.12: Bảng tbl\_admin 50](#_Toc106835381)

[Bảng 3.13: Bảng tbl\_danhmuc 51](#_Toc106835382)

[Bảng 3.14: Bảng tbl\_loaisanpham 51](#_Toc106835383)

[Bảng 3.15: Bảng tbl\_sanpham 51](#_Toc106835384)

[Bảng 3.16: Bảng tbl\_sanpham\_anh 52](#_Toc106835385)

[Bảng 3.17: Bảng tbl\_cart 52](#_Toc106835386)

[Bảng 3.18: Bảng tbl\_diachi 54](#_Toc106835387)

[Bảng 3.19: Bảng tbl\_order 54](#_Toc106835388)

[Bảng 3.20: Bảng tbl\_payment 55](#_Toc106835389)

[Bảng 3.21: Bảng tbl\_carta 55](#_Toc106835390)

[Bảng 4.1: Chức năng hiển thị sản phẩm theo loại sản phẩm 72](#_Toc106835391)

[Bảng 4.2: Chức năng chi tiết sản phẩm 72](#_Toc106835392)

[Bảng 4.3: Chức năng tạo giỏ hàng 73](#_Toc106835393)

[Bảng 4.4: Chức năng xóa giỏ hàng 74](#_Toc106835394)

[Bảng 4.5: Chức năng tạo đơn hàng 75](#_Toc106835395)

[Bảng 4.6: Chức năng hiển thị tình trạng đơn hàng 76](#_Toc106835396)

[Bảng 4.7: Chức năng đăng nhập quản trị 77](#_Toc106835397)

[Bảng 4.8: Chức năng thêm danh mục sản phẩm 78](#_Toc106835398)

[Bảng 4.9: Chức năng sửa danh mục sản phẩm 79](#_Toc106835399)

[Bảng 4.10: Chức năng xóa danh mục sản phẩm 80](#_Toc106835400)

[Bảng 4.11: Chức năng thêm loại sản phẩm 81](#_Toc106835401)

[Bảng 4.12: Chức năng sửa loại sản phẩm 81](#_Toc106835402)

[Bảng 4.13: Chức năng xóa loại sản phẩm 82](#_Toc106835403)

[Bảng 4.14: Chức năng thêm sản phẩm 82](#_Toc106835404)

[Bảng 4.15: Chức năng sửa thông tin sản phẩm 83](#_Toc106835405)

[Bảng 4.16: Chức năng xóa sản phẩm 84](#_Toc106835406)

[Bảng 4.17: Chức năng thống kê đơn hàng 85](#_Toc106835407)

[Bảng 4.18: Chức năng xác nhận đơn hàng 86](#_Toc106835408)

[Bảng 4.19: Chức năng xóa đơn hàng 87](#_Toc106835409)

# DANH SÁCH TỪ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| **Từ viết tắt** | **Nghĩa tiếng Anh** | **Nghĩa tiếng Việt** |
| CMS | Content Management System | Hệ quản trị nội dung |
| CSDL | Database | Cơ sở dữ liệu |
| SQL | Structed Query Language | Ngôn ngữ truy vấn có cấu trúc |
| HTML | Hypertext Markup Language | Ngôn ngữ đánh dấu siêu văn bản |
| CSS | Cascading Style Sheets | Các tệp tin định dạnh theo kiểu tầng |
| JS | Javascript | Ngôn ngữ kịch bản |
| TMĐT |  | Thương mại điện tử |
| PTTK |  | Phân tích thiết kế |

# CHƯƠNG I: TỔNG QUAN VỀ ĐỒ ÁN

## 1.1. Lý do chọn đồ án

Ngày nay, kinh tế phát triển đời sống sinh hoạt của mọi người cũng nâng cao  
hơn, tất bậc hơn. Con người đòi hỏi những sản phẩm tiện nghi nhằm đáp ứng nhu cầu sử dụng thường xuyên cho sinh hoạt hằng ngày đối với gia đình, hộ gia đình. Vì vậy những sản phẩm gia dụng là rất cần thiết cho mỗi gia đình, các sản phẩm này mang lại sự tiện lợi, nhanh chóng khi sử dụng, giúp tiết kiện được thời gian, công sức trong công việc như: nấu ăn, giặc quần áo, dọn dẹp nhà cửa, bảo quản thực phẩm…Giúp cho mọi người đặc biệt là các bà mẹ làm việc nhà nhanh hơn, các ông bố có thể làm công việc nhà một cách dễ dàng hơn để phụ giúp gia đình…Hiện nay người tiêu dùng không chỉ muốn mua những sản phẩm có mẫu mã đẹp, chất lượng tốt mà giá thành phải hợp lý đáp ứng được nhu cầu mua sắm thường xuyên của khách hàng.

Cùng với sự phát triển của kinh tế thì công nghệ thông tin cũng không ngừng  
phát triển và được ứng dụng rộng rãi trên nhiều lĩnh vực. Việc ứng dụng Internet vào đời sống ngày càng phổ biến, truy cập Internet giúp cho mọi cá nhân, tổ chức có được một kho thông tin khổng lồ phục vụ nhu cầu, mục đích của mọi người mà ít tốn thời gian và chi phí. Nhận thức được nhu cầu tìm hiểu thông tin, giải trí cũng như mua sắm của xã hội thì hàng loạt các Website với mục đích giải trí, tin tức, thương mại,…được ra đời.Hiện nay thương mại điện tử trên thế giới đang phát triển rất mạnh mẽ. Kỹ thuật số giúp chúng ta tiết kiệm được đáng kể các chi phí nhờ vào chi phí vận chuyển trung gian, chi phí giao dịch đặc biệt là giúp tiết kiệm được thời gian để con người đầu tư vào các hoạt động khác. Hơn nữa thương mại điện tử còn giúp cho các cá nhân, tổ chức có thể quảng bá, giới thiệu và bán những sản phẩm của mình đến cho tất cả mọi người ở khắp mọi nơi một cách nhanh chóng và ít tốn chi phí về nhiều mặt.

Xuất phát từ những nhu cầu đó tôi chọn đề tài:

“**Xây dựng website bán đồ gia dụng**”. Tại đây, khách hàng có thể dễ dàng xem được thông tin chi tiết của từng sản phẩm một cách đầy đủ và chính xác, cập nhật được những sản phẩm mới, tin tức mới thường xuyên, cũng như tìm kiếm được những sản phẩm mình cần tìm một cách nhanh nhất. Giúp khách hàng tiết kiệm được thời gian, công sức, chi phí, có được nhiều sự lựa chọn hơn và mua được những sản phẩm với giá cả hợp lý, chất lượng.

## 1.2. Mục tiêu của đồ án

* Nắm được cơ sở lý thuyết chung về thương mại điện tử và các vấn đề liên quan đến thương mại điện tử
* Tìm hiểu các Website thương mại điện tử lớn đã được triển khai để nắm được  
  cách thức hoạt động và những yêu cầu cần thiết đối với ứng dụng thương mại điện tử.
* Xác định đối tượng khách hàng mà mình hướng tới để phục vụ.
* Tính toán, đưa ra phương pháp và kế hoạch thực hiện dự án trong thời gian  
  ngắn nhất và chi phí thấp nhất có thể

## 1.3. Giới hạn và phạm vi của đồ án

### 1.3.1. Đối tượng nghiên cứu

* Tìm hiểu thực trạng TMĐT tại Việt Nam.
* Các hệ thống website kinh doanh đồ gia dụng.
* Các quy trình mua hàng, thêm sản phẩm, tin tức, …
* Quy trình phân tích thiết kế hệ thống, xây dựng cơ sở dữ liệu

### 1.3.2. Phạm vi nghiên cứu

* Các công cụ ngôn ngữ xây dựng website.
* Chức năng của website kinh doanh đồ gia dụng.
* Qui trình xây dựng và thiết kế website.

## 1.4. Nội dung thực hiện

* Chương 1: Mở đầu
* Chương 2: Cơ sở lý thuyết
* Chương 3: Phân tích và thiết kế hệ thống
* Chương 4: Xây dựng ứng dụng

## 1.5. Phương pháp tiếp cận

* Tiến hành thu thập và phân tích những thông tin, tài liệu liên quan đến đề tài  
  kinh doanh Đồ gia dụng trực tuyến để hình thành nên những ý tưởng tổng quan (mục đích cần đạt đến của Website, đối tượng cần hướng đến là ai? thông tin gì đã có trong tay và sử dụng chúng như thế nào?).
* Xác định các yêu cầu phân tích thiết kế hệ thống chương trình cho phù hợp.
* Xây dựng chương trình theo những yêu cầu đã đặt ra.
* Triển khai chương trình và đánh giá kết quả đạt được.

# CHƯƠNG II: CƠ SỞ LÝ THUYẾT

## 2.1. Quy trình phát triển phần mềm

Quy trình phát triển phần mềm là một cấu trúc bao gồm tập hợp các thao tác và các kết quả tương quan sử dụng trong việc phát triển để sản xuất ra một sản phẩm phần mềm. Các thuật ngữ tương tự là vòng đời phần mềm và quy trình phần mềm. Đây được coi là một thành phần tập con của vòng đời phát triển hệ thống. Có một số mô hình cho việc xây dựng các quy trình này, mỗi mô hình mô tả các phương thức cũng như các nhiệm vụ hoặc thao tác cần được thực hiện trong cả quá trình. Nhiều người coi mô hình vòng đời là một thuật ngữ phạm vi rộng và quy trình phát triển phần mềm là một thuật ngữ ở mức chi tiết cụ thể hơn. Ví dụ, có rất nhiều quy trình phát triển phần mềm tuân theo mô hình vòng đời xoắn ốc. ISO/IEC 12207 là một tiêu chuẩn quốc tế cho các quy trình vòng đời phần mềm, mục đích là trở thành một tiêu chuẩn định nghĩa tất cả các công việc cần thực hiện để xây dựng và bảo trì sản phẩm phần mềm.

Một quá trình kéo dài hàng thập kỷ với mục tiêu tìm ra được các quy trình có tính lặp lại và có thể dự đoán trước được để cải thiện hiệu suất lao động và chất lượng sản phẩm. Một số người đã cố gắng hệ thống hóa hoặc hình thức hóa các nhiệm vụ viết phần mềm vốn không tuân theo quy tắc nào cả. Một số khác áp dụng các kỹ thuật quản lý dự án để viết phần mềm. Nếu như không có quản lý dự án, thì các dự án phần mềm có thể sẽ dễ bị chuyển giao chậm hoặc vượt quá ngân sách. Với một số lượng lớn các dự án phần mềm không đáp ứng được kỳ vọng về chức năng, chi phí hoặc kế hoạch chuyển giao đã cho thấy một thực tế là do đang thiếu các phương thức quản lý dự án hiệu quả.

Có 4 thao tác là nền tảng của hầu hết các quy trình phần mềm là:

* Đặc tả phần mềm: Các chức năng của phần mềm và điều kiện để nó hoạt động phải được định nghĩa.
* Sự phát triển phần mềm: Để phần mềm đạt được đặc tả thì phải có quy trình phát triển này.
* Đánh giá phần mềm: Phần mềm phải được đánh giá để chắc chắn rằng nó làm những gì mà khách hàng muốn.
* Sự tiến hóa của phần mềm: Phần mềm phải tiến hóa để thỏa mãn sự thay đổi các yêu cầu của khách hàng.

## 2.2. Thiết kế giao diện web với HTML, CSS, JavaScript

### 2.2.1. Giới thiệu HTML

1. ***Tổng quan về HTML[4]***

HTML(viết tắt của từ HyperText Markup Language, hay là "Ngôn ngữ Đánh dấu Siêu văn bản") là một [ngôn ngữ đánh dấu](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_%C4%91%C3%A1nh_d%E1%BA%A5u) được thiết kế ra để tạo nên các [trang web](https://vi.wikipedia.org/wiki/Website) trên World Wide Web. Nó có thể được trợ giúp bởi các công nghệ như [CSS](https://vi.wikipedia.org/wiki/CSS) và các ngôn ngữ kịch bản giống như [JavaScript](https://vi.wikipedia.org/wiki/JavaScript).

Các [trình duyệt web](https://vi.wikipedia.org/wiki/Tr%C3%ACnh_duy%E1%BB%87t_web) nhận tài liệu HTML từ một [web server](https://vi.wikipedia.org/wiki/M%C3%A1y_ch%E1%BB%A7_web) hoặc một kho lưu trữ cục bộ và [render](https://vi.wikipedia.org/w/index.php?title=Browser_engine&action=edit&redlink=1) tài liệu đó thành các trang web đa phương tiện. HTML mô tả cấu trúc của một [trang web](https://vi.wikipedia.org/wiki/Trang_web) về mặt ngữ nghĩa và các dấu hiệu ban đầu được bao gồm cho sự xuất hiện của tài liệu.

Các [phần tử HTML](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_t%E1%BB%AD_HTML) là các khối xây dựng của các trang HTML. Với cấu trúc HTML, hình ảnh và các đối tượng khác như biểu mẫu tương tác có thể được nhúng vào trang được hiển thị. HTML cung cấp một phương tiện để tạo tài liệu có cấu trúc bằng cách biểu thị ngữ nghĩa cấu trúc cho văn bản như headings, paragraphs, lists, [links](https://vi.wikipedia.org/wiki/Si%C3%AAu_li%C3%AAn_k%E1%BA%BFt), quotes và các mục khác. Các phần tử HTML được phân định bằng các tags, được viết bằng dấu ngoặc nhọn. Các tags như <img /> và <input /> giới thiệu trực tiếp nội dung vào trang. Các tags khác như <p> bao quanh và cung cấp thông tin về văn bản tài liệu và có thể bao gồm các thẻ khác làm phần tử phụ. Các trình duyệt không hiển thị các thẻ HTML, nhưng sử dụng chúng để diễn giải nội dung của trang.

HTML có thể nhúng các chương trình được viết bằng [scripting](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_k%E1%BB%8Bch_b%E1%BA%A3n) như [JavaScript](https://vi.wikipedia.org/wiki/JavaScript), điều này ảnh hưởng đến hành vi và nội dung của các trang web. Việc bao gồm CSS xác định giao diện và bố cục của nội dung. [World Wide Web Consortium](https://vi.wikipedia.org/wiki/World_Wide_Web_Consortium) (W3C), trước đây là đơn vị bảo trì HTML và là người duy trì hiện tại của các tiêu chuẩn CSS, đã khuyến khích việc sử dụng CSS trên HTML trình bày rõ ràng kể từ năm 1997.

1. ***Mục tiêu***

* Hiểu về HTML và vận dụng cách sử dụng HTML tạo lập 1 trang web cơ bản.
* Sử dụng các thẻ, các kỹ thuật và các cách xử lý thẻ cơ bản như: xử lý đoạn văn bản, khối, bảng, hình ảnh, danh sách, siêu liên kết, đồ họa ….
* Áp dụng nhiều nhất các kỹ thuật có thể sử dụng để tạo 1 trang web giới thiệu về HTML.

1. ***Các kỹ thuật quan trọng***

**+ Một số thẻ định dạng văn bản:**

* **<b>** in đậm
* **<strong>** văn bản quan trọng
* **<i>** in nghiêng
* **<em>** nhấn mạnh
* **<mark>** văn bản đánh dấu
* **<small>** chữ nhỏ
* **<dell>** văn bản bị xóa
* **<ins>** văn bản được chèn vào
* **<sub**> văn bản chỉ số dưới
* **<sup>** văn bản chỉ số trên

**+ Một số thẻ định dạng bảng:**

* **<thead>:** dành cho tiêu đề, tên bảng, các miêu tả bắt đầu cho bảng.
* **<tbody>:** dành cho bảng dữ liệu chính.
* **<tfoot>:** dành cho tổng hợp, thống kê, bản quyền, lời kết.

**<thead>, <tbody>, <tfoot>** đều có thể định nghĩa:

**<tr>:** table-row - dòng dữ liệu. Trong khi đó ở mỗi dòng có thể định nghĩa từng ô.

**<th>:** dành cho các ô tiêu đề.

**<td>:** dành cho các ô dữ liệu.

*\*****<th>****và****<td>****về mặc cú pháp sử dụng giống nhau, nhưng được khuyến khích sử dụng đúng ngữ nghĩa của nó.****<th>****nên dành cho tiêu đề và****<td>****dành cho dữ liệu.*

**+ Một số thẻ định dạng danh sách:**

Danh sách có thứ tự (Ordered List) được định nghĩa bởi cặp thẻ đóng mở **<ol>...</ol>.** Mỗi phần tử trong **<ol>** được định nghĩa bởi thẻ **<li></li>** (List Item).

Danh sách không có thứ tự (Unordered List) được định nghĩa bởi cặp thẻ đóng mở **<ul></ul>** với mỗi phần tử trong danh sách được định nghĩa bởi thẻ **<li>** (List Item).

Mỗi **<li>** có thể chứa một danh sách **<ol>** hoặc **<ul>.**

**+ Định dạng siêu liên kết:**

Thẻ liên kết có chức năng điều hướng giữa các trang web trong website, ngoài ra còn được dùng để làm bookmark, được định nghĩa bởi thẻ **<a>** có cấu trúc như sau:

<a href="liên kết" target="\_blank">Nội dung hiển thị</a>

Các thuộc tính của thẻ **<a>**

* **href**: đường dẫn đến website cụ thể.
* **target**: cách thức mở 1 trang web.
* **\_self**: mở với tab hiện tại, nếu không có thuộc tính **target**, mặc định sẽ là **\_self**.
* **\_blank**: mở với tab mới.
* **\_parent**: mở với frame chứa frame của trang web hiện tại (nếu trang này được tải bởi iframe).
* **\_top**: vượt đến cấp cao nhất của frame, mở với frame gốc (tab hiện tại) (nếu trang được mở bởi các iframe lồng nhau).

**+ Một số thẻ xử lý khối:**

**<div>** Đây là một thẻ cấp khối rất quan trọng đóng vai trò lớn trong việc tạo nhóm các thẻ HTML khác và áp dụng CSS trong nhóm các phần tử. Ngay trong hiện tại thì thẻ **<div>** có thể được sử dụng để tạo cách bố trí (layout) cho web, tại đây chúng ta xác định các phần khác nhau (trái, phải, trên,…) của trang bằng cách sử dụng thẻ này.

**<span>** là một phần tử nội tuyến và có thể được sử dụng để tạo nhóm nội tuyến trong tài liệu HTML. Thẻ này cũng không cung cấp bất cứ sự thay đổi về thị giác trên khối, nhưng nó có nhiều ý nghĩa hơn khi sử dụng cùng với CSS.

**+ Xử lý hình ảnh:**

Trong HTML, hình ảnh được định nghĩa bởi thẻ **<img>** có cấu trúc như sau:

<img src="liên kết" alt="nội dung thay thế" >

Thuộc tính **src** xác định đường dẫn đến hình ảnh và thuộc tính **alt** là xác định một đoạn nội dung thay thế khi hình ảnh không được hiển thị do đường dẫn lỗi hoặc tải không được ảnh.

Các thuộc tính của **<img>**

* **src**: liên kết đến file hình ảnh.
* **alt**: nội dung thay thế nếu không tải được ảnh.
* **width**: độ rộng của ảnh (px, %).
* **height**: độ cao của ảnh (px, %).

### 2.2.2. Giới thiệu CSS

1. ***Tổng quan về CSS[5]***

Trong [tin học](https://vi.wikipedia.org/wiki/Tin_h%E1%BB%8Dc), các tập tin định kiểu theo tầng – dịch từ [tiếng Anh](https://vi.wikipedia.org/wiki/Ti%E1%BA%BFng_Anh) là Cascading Style Sheets (CSS) – được dùng để miêu tả cách trình bày các tài liệu viết bằng ngôn ngữ [HTML](https://vi.wikipedia.org/wiki/HTML) và [XHTML](https://vi.wikipedia.org/wiki/XHTML). Ngoài ra ngôn ngữ định kiểu theo tầng cũng có thể dùng cho [XML](https://vi.wikipedia.org/wiki/XML), [SVG](https://vi.wikipedia.org/wiki/SVG), [XUL](https://vi.wikipedia.org/w/index.php?title=XUL&action=edit&redlink=1). Các đặc điểm kỹ thuật của CSS được duy trì bởi [World Wide Web Consortium](https://vi.wikipedia.org/wiki/W3C) ([W3C](http://www.w3c.org/)). Thay vì đặt các thẻ quy định kiểu dáng cho văn bản [HTML](https://vi.wikipedia.org/wiki/HTML) (hoặc [XHTML](https://vi.wikipedia.org/wiki/XHTML)) ngay trong [nội dung](https://vi.wikipedia.org/w/index.php?title=N%E1%BB%99i_dung&action=edit&redlink=1) của nó, bạn nên sử dụng CSS.

Tác dụng của CSS:

* Hạn chế tối thiểu việc làm rối mã [HTML](https://vi.wikipedia.org/wiki/HTML) của trang [Web](https://vi.wikipedia.org/wiki/World_Wide_Web) bằng các thẻ quy định kiểu dáng (chữ đậm, chữ in nghiêng, chữ có gạch chân, chữ màu), khiến mã nguồn của trang Web được gọn gàng hơn, tách nội dung của trang [Web](https://vi.wikipedia.org/wiki/World_Wide_Web) và định dạng hiển thị, dễ dàng cho việc cập nhật nội dung.
* Tạo ra các kiểu dáng có thể áp dụng cho nhiều trang Web, giúp tránh phải lặp lại việc định dạng cho các trang [Web](https://vi.wikipedia.org/wiki/World_Wide_Web) giống nhau.

1. ***Mục tiêu***

* Hiểu về CSS và vận dụng cách sử dụng CSS tạo lập 1 trang web cơ bản.
* Áp dụng các kỹ thuật về cú pháp, cách tạo CSS, các bộ chọn, định dạng để tạo 1 trang web có nhúng CSS.
* Sử dụng CSS trong HTML.
* Kết hợp sử dụng Bootstrap để hoàn thiện giao diện, tăng tính tương tác với người dùng.

1. ***Các kỹ thuật quan trọng***

**+ Cú pháp và cách tạo CSS**

Có 3 cách để sử dụng CSS:

* “Inline CSS”: Áp dụng trực tiếp trên một đối tượng nhất định bằng thuộc tính style:
* “Internal CSS”: Đặt CSS ở đầu trang Web để áp dụng kiểu dáng cho toàn bộ trang ấy, khi đó chỉ cần đặt đoạn CSS vào trong cặp thẻ <style> rồi đặt vào trong phần header của Web (giữa <head> và </head>):
* “External CSS”: Đặt các thuộc tính CSS vào một tệp tin riêng biệt (\*.css), khi đó có thể tham chiếu đến từ nhiều trang Web khác nhau.

**+ Cấu trúc một đoạn CSS**

Một đoạn CSS bao gồm :

vùng chọn {

    thuộc tính : giá trị;

    thuộc tính: giá trị;

    .....

}

**+ Các loại bộ chọn:**

**Bảng** 2.1: Các bộ chọn CSS

|  |  |  |  |
| --- | --- | --- | --- |
| **Bộ chọn** | **Ví dụ** | **Mô tả các ví dụ** | **CSS** |
| .*class* | .intro | Chọn tất cả các phần tử có class=”intro” | 1 |
| #*id* | #firstname | Chọn tất cả các phần tử có id=”firstname” | 1 |
| \* | \* | Chọn tất cả các phần tử | 2 |
| *element* | p | Chọn tất cả các phần tử  <p> | 1 |
| *element,element* | div, p | Chọn tất cả các phần tử <div>và phần tử <p> | 1 |
| *element* *element* | div p | Chọn tất cả các phần tử <p> và bên trong phần tử  <div> | 1 |
| *element*>*element* | div > p | Chọn tất cả các phần tử <p> có phần tử cha là <div> | 2 |
| *element*+*element* | div + p | Chọn tất cả các phần tử <p> được đặt phía sau phần tử  <div> | 2 |
| *element1*~*element2* | p ~ ul | Chọn tất cả các phần tử  <ul> được đặt trước bởi một phần tử  <p> | 3 |
| [*attribute*] | [target] | Chọn tất cả các phần tử có cùng thuộc tính | 2 |
| [*attribute*=*value*] | [target=\_blank] | Chọn tất cả các phần tử có thuộc tính bằng giá trị( target=”\_blank”) | 2 |
| [*attribute*~=*value*] | [title~=flower] | Chọn tất cả các phần tử có tiêu đề của thuộc tính có chứa từ “flower” | 2 |
| **Bộ chọn** | **Ví dụ** | **Mô tả các ví dụ** | **CSS** |
| [*attribute*^=*value*] | a[href^=”https”] | Chọn tất cả các phần tử  <a> có giá trị thuộc tính “href” bắt đầu bằng “https” | 3 |
| [*attribute*$=*value*] | a[href$=”.pdf”] | Chọn tất cả các phần tử <a> có giá trị thuộc tính “href” kết thúc bằng”.pdf” | 3 |
| [*attribute*\*=*value*] | a[href\*=”timoday”] | Chọn tất cả các phần tử <a> có giá trị thuộc tính “href” chứa chuỗi”timoday” | 3 |
| :active | a:active | Chọn tất cả các liên kết được kích hoạt | 1 |
| ::after | p::after | Chèn thêm nội dung ngay phía sau của các phần tử <p> | 2 |
| ::before | p::before | Chèn thêm nội dung ngay phía trước của các phần tử <p> | 2 |
| :checked | input:checked | Chọn tất cả các phần tử <input> đang được chọn (selected) | 3 |
| :disabled | input:disabled | Chọn tất cả các phần tử <input> đang được vô hiệu hoá (disabled) | 3 |
| :empty | p:empty | Chọn tất cả các phần tử  <p> không chứa phần tử con (bao gồm cả các nút văn bản) | 3 |
| :enabled | input:enabled | Chọn tất cả các phần tử <input> đang được kích hoạt | 3 |
| :first-child | p:first-child | Chọn các phần tử  <p> có phần tử đầu tiên của phần tử cha chứa nó | 2 |
| **Bộ chọn** | **Ví dụ** | **Mô tả các ví dụ** | **CSS** |
| ::first-line | p::first-line | Chọn dòng đầu tiên của các phần tử <p> | 1 |
| :first-of-type | p:first-of-type | Chọn tất cả các phần tử <p> có phần tử đầu tiên <p> là phần tử cha | 3 |
| :focus | input:focus | Chọn các phần tử <input> nhận focus | 2 |
| :hover | a:hover | Chọn các liên kết khi chuột di chuyển qua | 1 |
| :in-range | input:in-range | Chọn phần tử <input> có giá trị  trong phạm vi nhất định | 3 |
| :invalid | input:invalid | Chọn tất cả các phần tử <input> có giá trị không hợp lệ | 3 |
| :lang(*language*) | p:lang(it) | Chọn tất cả các phần tử  <p> có giá trị thuộc tính “lang” bằng “it” | 2 |
| :last-child | p:last-child | Chọn tất cả các phần tử <p> là phần tử con cuối cùng của phần tử cha | 3 |
| :last-of-type | p:last-of-type | Chọn tất cả các phần tử <p> là thuộc tính cuối cùng của phần tử cha | 3 |
| :link | a:link | Chọn tất cả các liên kết khi chưa được click | 1 |
| :not(*selector*) | :not(p) | Chọn tất cả các phần tử không phải là một phần tử <p> | 3 |
| :nth-child(*n*) | p:nth-child(2) | Chọn tất cả các phần tử <p> là phần tử thứ hai của phần tử cha | 3 |
| **Bộ chọn** | **Ví dụ** | **Mô tả các ví dụ** | **CSS** |
| :nth-last-child(*n*) | p:nth-last-child(2) | Chọn tất cả các phần tử <p> là phần tử con thứ hai của phần tử cha, tính từ phần tử con cuối cùng | 3 |
| :nth-last-of-type(*n*) | p:nth-last-of-type(2) | Chọn tất cả các phần tử <p>là phần tử thuộc tính thứ hai của phần tử cha, tính từ phần tử thuộc tính con cuối cùng | 3 |
| :nth-of-type(*n*) | p:nth-of-type(2) | Chọn tất cả các phần tử <p> là phần tử thuộc tính con thứ hai của phần tử cha | 3 |
| :only-of-type | p:only-of-type | Chọn tất cả các phần tử <p> là thuộc tính duy nhất của phần tử cha | 3 |
| :only-child | p:only-child | Chọn tất cả các phần tử <p> là con duy nhất của phần tử cha | 3 |
| :optional | input:optional | Chọn tất cả các phần tử đầu vào không có thuộc tính “required” | 3 |
| :out-of-range | input:out-of-range | Chọn tất cả các phần tử đầu vào có giá trị ngoài một phạm vi nhất định | 3 |
| :read-only | input:read-only | Chọn tất cả các phần tử đầu vào có thuộc tính xác định “readonly” | 3 |
| :read-write | input:read-write | Chọn tất cả các phần tử đầu vào có thuộc tính không xác định “readonly” | 3 |
| **Bộ chọn** | **Ví dụ** | **Mô tả các ví dụ** | **CSS** |
| :required | input:required | Chọn tất cả các phần tử đầu vào có thuộc tính  “required” xác định | 3 |
| :root | :root | Chọn các phần tử gốc của văn bản | 3 |
| ::selection | ::selection | Chọn các phần tử được người dùng lựa chọn |  |
| :target | #news:target | Chọn các phần tử đang hoạt động hiện tại (click trong các liên kết  anchor name) | 3 |
| :valid | input:valid | Chọn tất cả các phần tử đầu vào có một giá trị hợp lệ | 3 |
| :visited | a:visited | Chọn tất cả các liên kết được truy cập | 1 |

**+ Cách dùng CSS trong HTML**

Để áp dụng CSS tác dụng vào các phần HTML có ba cách nhúng CSS: Inline, **Internal** và **External.**

1. Inline: mã CSS viết tại thuộc tính style của phần tử HTML
2. Internal: mã CSS trong chính văn bản HTML, nằm trong khối thẻ **<style>**
3. External: mã CSS ở một file riêng biệt sau đó nạp vào HTML bằng phần tử <link>

**+ Giới thiệu Bootstrap và cách sử dụng**

***Bootstrap là:***

- Một framework của css và javascript. Hay nói cách khác, nó là những đoạn code viết sẵn.

- Định nghĩa sẵn các class css và các hàm trong javascript. Chúng ta chỉ việc sử dụng các class này để tùy chỉnh thêm

***Ưu điểm của bootstrap:***

- Giảm thiểu bớt việc viết code, đóng vai trò như bộ khung nền, giúp phát triển web nhanh hơn.

- Hỗ trợ đa dạng như slide, responsive, menu nhiều cấp.

***Cách sử dụng:***

Sử dụng bootstrap thì bạn phải**tải bootstrap**về.

Trong đó:

**CSS**

* boostrap.css: Chứa bộ core đầy đủ CSS của Boostrap
* boostrap.min.css: Chứa bộ core đầy đủ CSS của Boostrap nhưng đã được loại bỏ các khoảng trắng, chú thích,…để giảm dung lượng file
* bootstrap-theme.css: Chứa các hiệu ứng của Boostrap (gradient, animation)
* bootstrap-theme.min.css: Chứa các hiệu ứng của Boostrap (gradient, animation) nhưng đã được loại bỏ các khoảng trắng, chú thích,…để giảm dung lượng file

**JS**

* bootstrap.js: Chứa bộ core đầy đủ Javascript của Bootstrap
* bootstrap.min.js: Chứa bộ core đầy đủ Javascript của Bootstrap nhưng đã được loại bỏ các khoảng trắng, chú thích để giảm dung lượng file

**Fonts**

* Thư mục chứa toàn bộ fonts của thư viện glyphicons giúp việc trang trí website bằng bộ font icon
* Lưu ý: Khi nhúng vào website, bạn nên sử dụng file .min để website load nhanh hơn.

**Cách nhúng Bootstrap và Website**

***Cách 1: Nhúng từ liên kết từ Bootstrap***

Bootstrap có cung cấp sẵn cho chúng ta các liên kết để nhúng trực tiếp vào website nhưng chúng ta hạn chế sử dụng nó vì khi đó website phải load liên kết bên ngoài, dẫn đến việc website bị chậm.

***Cách 2: Nhúng bằng cách tự host***

### 2.2.3. Giới thiệu JavaScript

1. ***Tổng quan về JavaScript[6]***

JavaScript, theo phiên bản hiện hành, là một [ngôn ngữ lập trình thông dịch](https://vi.wikipedia.org/wiki/Interpreted_language) được phát triển từ các ý niệm [nguyên mẫu](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_d%E1%BB%B1a_tr%C3%AAn_nguy%C3%AAn_m%E1%BA%ABu). Ngôn ngữ này được dùng rộng rãi cho các [trang web](https://vi.wikipedia.org/wiki/Website) (phía người dùng) cũng như phía máy chủ (với Nodejs). Nó vốn được phát triển bởi [Brendan Eich](https://vi.wikipedia.org/wiki/Brendan_Eich) tại [Hãng truyền thông Netscape](https://vi.wikipedia.org/wiki/Netscape) với cái tên đầu tiên *Mocha*, rồi sau đó đổi tên thành *LiveScript*, và cuối cùng thành JavaScript. Giống [Java](https://vi.wikipedia.org/wiki/Java_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)), JavaScript có [cú pháp](https://vi.wikipedia.org/wiki/C%C3%BA_ph%C3%A1p_h%E1%BB%8Dc) tương tự [C](https://vi.wikipedia.org/wiki/C_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)), nhưng nó gần với [Self](https://vi.wikipedia.org/w/index.php?title=Self_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)&action=edit&redlink=1) hơn Java. .js là [phần mở rộng](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%9F_r%E1%BB%99ng_t%C3%AAn_t%E1%BA%ADp_tin) thường được dùng cho [tập tin](https://vi.wikipedia.org/wiki/T%E1%BA%ADp_tin) [mã nguồn](https://vi.wikipedia.org/wiki/M%C3%A3_ngu%E1%BB%93n) JavaScript.

1. ***Mục tiêu***

* Hiểu về JavaScript và vận dụng cách sử dụng JavaScript tạo lập 1 trang web cơ bản.
* Áp dụng các kỹ thuật về các câu lệnh, xử lý hàm, đối tượng trong JavaSript để tạo 1 trạng web có nhúng JavaScript.
* Sử dụng JavaScript trong HTML.
* Tìm hiểu thêm về XML, JSON, AJAX.

1. ***Các kỹ thuật quan trọng***

**+ Các lệnh điều khiển**

**Câu lệnh if . . . else**

Câu lệnh này dùng để kiểm tra điều kiện, nó thực hiện việc tính toán trên một biểu thức. Nếu điệu kiện là đúng (true) thì khối lệnh được thực thi.

if (condition) { statements; }

Ta cũng có thể chỉ ra khối lệnh cần thực hiện khi điều kiện là sai (false) bằng

việc dùng mệnh đề else

if (condition) { statements; }

else { statements2; }

Nếu điều kiện là sai khối lệnh sau else được thực thi.

**Lệnh switch**

Khi ta có nhiều tuỳ chọn If...else thì tốt hơn nên sử dụng lệnh switch. Lệnh này còn được xem là lệnh case, lệnh switch thực thi một trong các khối lệnh tuỳ thuộc vào giá trị của biểu thức. Nếu không tìm thấy một giá trị nào trong danh sách các case của nó, khối lệnh trong phần default sẽ được thực hiện. Lệnh break dùng để thoát ra khỏi câu lệnh switch.

switch (expression){

case label:

statement;

break;

case label:

statement;

break;

...

default: statement;

}

**+ Các lệnh vòng lặp**

Các cấu trúc điều khiển việc thực hiện lặp đi lặp lại trong một chương trình được gọi là vòng lặp. Có nhiều loại vòng lặp:

* vòng lặp thực hiện lặp đi lặp lại các lệnh cho đến khi điều kiện là False
* vòng lặp thực hiện lặp đi lặp lại các lệnh cho đến khi điều kiện là True
* vòng lặp thực hiện lặp đi lặp lại các lệnh theo một số lần nhất định

**Vòng lặp “for”**

Vòng lặp for sẽ thực hiện lặp đi lặp lại khối lệnh cho đến khi điều kiện là false. Số lần thực hiện của vòng lặp thường được điều khiển thông qua một biến đếm.

Lệnh for bao gồm ba phần, cách nhau bởi dấu chấm phẩy. Cả ba phần đó đều không bắt buộc phải có, và chúng điều khiển việc thực hiện của vòng lặp for.

for (lệnh khởi tạo; điều kiện; lệnh tăng)

{ statements; }

**Do . . . while**

Vòng lặp Do...while được dùng để thực thi một khối lệnh cho đến khi điều kiên là false. Cú pháp là:

do

{ statements;}

while (condition)

**Lệnh while**

Lệnh while là một cấu trúc lặp khác trong JavaScript. Nó được dùng để thực hiện một khối các câu lệnh chừng nào điều kiện là true. Nếu có nhiều câu lệnh thực hiện trong thân của vòng lặp chương trình phải sử dụng cặp dấu { và } để chứa các câu lệnh trong đó.

Khác biệt chính giữa vòng lặp while và do…while là các lệnh trong thân vòng lặp while có thể khộng được thực hiện một lần nào vì có thể ngay từ ban đầu điều kiện đã là false. Tuy nhiên vòng lặp do…while bao giờ cũng được thực hiện ít nhất một lần. Cú pháp là:

while (condition)

{ statements; }

**Câu lệnh break & continue**

Vòng lặp while loop và for sẽ kết thúc thực hiện khi điều kiện là false.Tuy nhiên ta cũng có thể kết thúc vòng lặp nếu muốn. Lệnh break dùng để kết thúc việc thực thi của một câu lênh. Khi được sử dụng trong một vòng lặp, lệnh break làm dừng ngay vòng lặp đó và không thực hiện thêm nữa.

Một lệnh đặc biệt khác cũng có thể được sử dụng trong vòng lặp là lệnh continue. Continue dừng ngay lần lặp hiện tại và quay lại kiểm tra điều kiện để thực hiện lần lặp tiếp theo.

**for...in**

Câu lệnh for . . in được dùng để duyệt các thuộc tính của một đối tượng hay các phần tử của một mảng. Ví dụ, chúng ta có thể muốn thực hiện một khối các câu lệnh cho mỗi phần tử của mảng.

Chú pháp là:

for (variable in object)

{ statements; }

**With**

Câu lệnh “with” được dùng để thực thi một tập các lệnh cùng tham chiếu đến một đối tượng xác định. Đó là đối tựong được chỉ ra trong câu lệnh « with »

Chú pháp:

with (object)

{ statements; }

**+ Hàm và các sử dụng hàm**

Function hay còn gọi là hàm, là tập hợp một đoạn code dùng để xử lý một nhiệm vụ nào đó. Code bên trong function không được biên dịch cho tới khi được gọi đến. Chính vì vậy khi sử dụng function sẽ giúp chương trình được linh hoạt hơn.

Cú pháp:

|  |
| --- |
| function name\_of\_function(var1, var2, var3, ...)  {  // Some code  }  Trong đó: |

* function: là từ khóa của javascript nên bắt buộc phải như vậy
* name\_of\_function: là tên của hàm, thông thường chúng ta tạo những tên có ý nghĩa như find\_max, find\_min, ...
* var1, var2 var3, ... là các tham số truyền vào hàm. Ví dụ viết hàm kiểm tra số chẵn hay lẽ thì ta sẽ có một tham số đó là số cần kiểm tra. Đương nhiên bạn có thể tạo một hàm không có tham số.

Để gọi hàm thì ta chỉ cần gọi đến tên hàm, sau đó truyền vào các tham số cần thiết.Hàm có return là hàm có sử dụng từ khóa **return** để đặt ở cuối hàm với mục đích trả kết quả về để sử dụng tiếp ở những đoạn code bên ngoài.

Có một số trường hợp bạn muốn một tham số nào đó có thể được truyền hoặc không cần truyền vào đều được, lúc này chúng ta phải sử dụng nó như một tham số mặc định. Javascript không có cú pháp gán giá trị mặc định như PHP mà thay vào đó chúng ta sử dụng toán tử ||.

**+ Một số đối tượng**

**Đối tượng DOM**

Chúng ta có thể thấy tất cả các thẻ HTML sẽ được quản lý trong đối tượng document (DOM), thẻ cao nhất là thẻ html, tiếp đến là phân nhánh body và head. Bên trong head thì có những thẻ như style, title,... và bên trong body chứa bất kì một thẻ nào đó là thành phần của HTML. Như vậy ta có thể hiểu trong Javascript để thao tác được với các thẻ HTML ta phải thông qua đối tượng documnent (DOM).

Với DOM, JavaScript được tất cả sức mạnh cần thiết để tạo ra HTML động:

* JavaScript có thể thay đổi tất cả các phần tử HTML trong trang
* JavaScript có thể thay đổi tất cả các thuộc tính HTML trong trang
* JavaScript có thể thay đổi tất cả các phong cách CSS trong trang
* JavaScript có thể loại bỏ các yếu tố HTML và thuộc tính hiện tại
* JavaScript có thể thêm các yếu tố HTML mới và các thuộc tính
* JavaScript có thể phản ứng với tất cả các sự kiện HTML hiện trong trang
* JavaScript có thể tạo ra các sự kiện HTML mới trong trang

Document Object Model - DOM ("Mô hình Đối tượng Tài liệu"), là một giao diện lập trình ứng dụng (API). DOM được dùng để truy xuất các tài liệu dạng HTML và XML, có dạng một cây cấu trúc dữ liệu, và thông thường mô hình DOM độc lập với hệ điều hành và dựa theo kỹ thuật lập trình hướng đối tượng để mô tả tài liệu.

Thời kì sơ khai các thành phần trong một tài liệu HTML mô tả bằng các phiên bản khác nhau của DOM được hiển thị bởi các chương trình duyệt web thông qua JavaScript vì chưa có một chuẩn thống nhất nào. Điều này buộc World Wide Web Consortium (W3C) phải đưa ra một loạt các mô tả kĩ thuật về tiêu chuẩn cho DOM để thống nhất mô hình này.

**Đối tượng JavaScript**

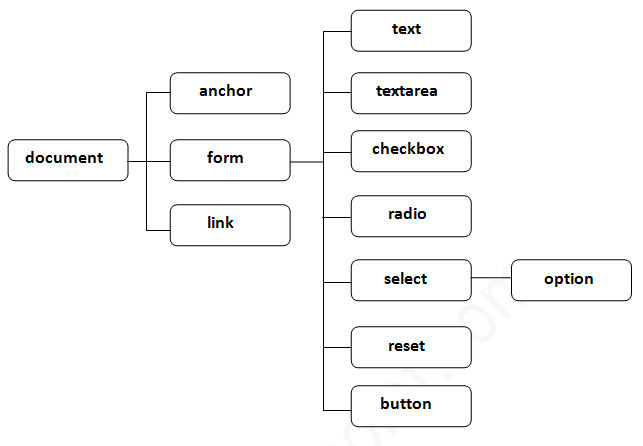
Javascript có 5 kiểu dữ liệu Number, String, Boolean, Undefined và Null và còn 1 kiểu khác nữa đó là Object (kiểu dữ liệu phức hợp). Kiểu Object là kiểu được sử dụng nhiều nhất vì tính linh hoạt cực kỳ mạnh mẽ của nó trong việc xử lý dữ liệu

**+ Document**

Đối tượng document là thuộc tính của đối tượng window, vì vậy nó có thể được truy cập bằng cách:

**window.document**

Các thuộc tính của đối tượng Document:



**Hình** 2.1: Các thuộc tính đối tượng Document

Các phương thức của đối tượng document

Chúng ta có thể truy cập và thay đổi nội dung của trang web bằng các phương thức của đối tượng document.

Các phương thức quan trọng của đối tượng document như sau:

|  |  |
| --- | --- |
| **Phương thức** | **Mô tả** |
| write("string") | Viết chuỗi đã cho trên doucment. |
| writeln("string") | Viết chuỗi đã cho trên doucment với ký tự newline ở cuối. |
| getElementById() | Trả về phần tử có giá trị id đã cho. |
| getElementsByName() | Trả về tất cả các phần tử có giá trị name đã cho. |
| getElementsByTagName() | Trả về tất cả các phần tử có tên thẻ đã cho. |
| getElementsByClassName() | Trả về tất cả các phần tử có class đã cho. |

**+ Math**

Đối tượng Math là định nghĩa sẵn trong JS, nó chứa các thuộc tính và phương thức cho phép thi hành một số tác vụ về toán học.

Đối tượng Math luôn có sẵn để sử dụng ngay mà không cần tạo mới. VD:

document.write(Math.PI);

//In ra 3.141592653589793

**+ String**

Đối tượng **String** giúp bạn làm việc với một dãy các ký tự; nó giúp xử lý các kiểu dữ liệu chuỗi gốc trong JavaScript với một số phương thức giúp đỡ.

Khi JavaScript tự động biến đổi giữa chuỗi gốc và các đối tượng String, bạn có thể gọi bất kỳ phương thức giúp đỡ nào của đối tượng String trên một chuỗi gốc.

Cú pháp

Sử dụng cú pháp sau để tạo một đối tượng String:

**var val = new String(string);**

Tham số **string** là một dãy các ký tự mà đã được mã hóa một cách chính xác.

**+ Array**

Các phương thức, thuộc tính của array trong JavaScript:

|  |  |  |
| --- | --- | --- |
| concat() | [includes()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#includes) | [reduce()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#reduce) |
| [copyWithin()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#copyWithin) | [indexOf()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#indexOf) | [reduceRight()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#reduceRight) |
| [entries()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#entries) | [isArray()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#isArray) | [reverse()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#reverse) |
| [every()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#every) | [isArray()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#isArray) | [shift()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#shift) |
| [fill()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#fill) | [keys()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#keys) | [slice()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#slice) |
| [filter()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#filter) | [length](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#length) | [some()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#some) |
| [find()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#find) | [lastIndexOf()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#lastIndexOf) | [sort()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#sort) |
| [findIndex()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#findIndex) | [map()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#map) | [splice()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#splice) |
| [forEach()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#forEach) | [pop()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#pop) | [toString()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#toString) |
| [from()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#from) | [push()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#push) | [unshift()](https://phambinh.net/bai-viet/lam-viec-voi-array-trong-javascript/#unshift) |

Nhóm thêm, xóa vào đầu, cuối mảng

* pop(): Xóa phần tử cuối mảng
* push(): Thêm vào cuối mảng
* shift(): Xóa phần tử đầu mảng
* unshift(): Thêm phần tử vào đầu mảng
* Nhóm check mảng:
* every(): Kiểm tra tất cả phần tử mảng phải thỏa mãn điều kiện
* some(): Kiểm tra một trong số các phần tử mảng có phần tử nào thỏa mãn điều kiện.
* includes(): Kiểm tra trong mảng có phần tử bạn cần tìm hay không?
* Nhóm tìm cái gì đó trong mảng:
* find(): Tìm phần tử mảng đầu tiên thỏa mãn điều kiện
* findIndex(): Tìm vị trí của phần tử mảng đầu tiên thỏa mãn điều kiện
* indexOf(): Tìm vị trí đầu tiên của phần tử xuất hiện trong mảng.
* lastIndexOf(): Tìm vị trí cuối cùng của phần tử xuất hiện trong mảng.

***+ Tóm tắt về XML, JSON, AJAX***

**XML** - eXtensible Markup Language, tức "Ngôn ngữ đánh dấu mở rộng": là [ngôn ngữ đánh dấu](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_%C4%91%C3%A1nh_d%E1%BA%A5u) với mục đích chung do [W3C](https://vi.wikipedia.org/wiki/W3C) đề nghị, để tạo ra các ngôn ngữ đánh dấu khác. Đây là một tập con đơn giản của [SGML](https://vi.wikipedia.org/wiki/SGML), có khả năng mô tả nhiều loại [dữ liệu](https://vi.wikipedia.org/wiki/D%E1%BB%AF_li%E1%BB%87u) khác nhau. Mục đích chính của XML là đơn giản hóa việc chia sẻ dữ liệu giữa các hệ thống khác nhau, đặc biệt là các hệ thống được kết nối với [Internet](https://vi.wikipedia.org/wiki/Internet).Các ngôn ngữ dựa trên XML (Ví dụ: [RDF](https://vi.wikipedia.org/wiki/RDF), [RSS](https://vi.wikipedia.org/wiki/RSS_(%C4%91%E1%BB%8Bnh_d%E1%BA%A1ng_t%E1%BA%ADp_tin)), [MathML](https://vi.wikipedia.org/wiki/MathML), [XHTML](https://vi.wikipedia.org/wiki/XHTML), [SVG](https://vi.wikipedia.org/wiki/SVG), [GML](https://vi.wikipedia.org/wiki/GML) và [cXML](https://vi.wikipedia.org/wiki/CXML)) được định nghĩa theo cách thông thường, cho phép các chương trình sửa đổi và kiểm tra hợp lệ bằng các ngôn ngữ này mà không cần có hiểu biết trước về hình thức của chúng.

**JSON** - JavaScript Object Notation: là một kiểu dữ liệu mở trong [JavaScript](https://vi.wikipedia.org/wiki/JavaScript). Kiểu dữ liệu này bao gồm chủ yếu là [text](https://vi.wikipedia.org/wiki/V%C4%83n_b%E1%BA%A3n), có thể đọc được theo dạng cặp "thuộc tính - giá trị". Về cấu trúc, nó mô tả một vật thể bằng cách bọc những vật thể con trong vật thể lớn hơn trong dấu ngoặc nhọn ({ }). JSON là một kiểu dữ liệu trung gian, chủ yếu được dùng để vận chuyển thông tin giữa các thành phần của một chương trình.

**AJAX** - Asynchronous JavaScript and XML, nghĩa là "JavaScript và XML không đồng bộ": là một nhóm các công nghệ [phát triển web](https://vi.wikipedia.org/w/index.php?title=Ph%C3%A1t_tri%E1%BB%83n_web&action=edit&redlink=1) được sử dụng để tạo các [ứng dụng web](https://vi.wikipedia.org/wiki/%E1%BB%A8ng_d%E1%BB%A5ng_web) động hay các [ứng dụng giàu tính Internet](https://vi.wikipedia.org/w/index.php?title=%E1%BB%A8ng_d%E1%BB%A5ng_gi%C3%A0u_t%C3%ADnh_Internet&action=edit&redlink=1) (rich Internet application). Từ Ajax được ông Jesse James Garrett đưa ra và dùng lần đầu tiên vào tháng 2 năm 2005 để chỉ kỹ thuật này, mặc dù các hỗ trợ cho Ajax đã có trên các chương trình duyệt từ 10 năm trước. Ajax là một kỹ thuật phát triển [web](https://vi.wikipedia.org/wiki/World_Wide_Web) có tính tương tác cao bằng cách kết hợp các ngôn ngữ:

* [HTML](https://vi.wikipedia.org/wiki/HTML) (hoặc [XHTML](https://vi.wikipedia.org/wiki/XHTML)) với [CSS](https://vi.wikipedia.org/wiki/CSS) trong việc hiển thị thông tin
* Mô hình DOM ([Document Object Model](https://vi.wikipedia.org/wiki/DOM)), được thực hiện thông qua [JavaScript](https://vi.wikipedia.org/wiki/JavaScript), nhằm hiển thị thông tin động và tương tác với những thông tin được hiển thị
* Đối tượng XMLHttpRequest để trao đổi dữ liệu một cách không đồng bộ với máy chủ web. (Mặc dù, việc trao đổi này có thể được thực hiện với nhiều định dạng như HTML, văn bản thường, [JSON](https://vi.wikipedia.org/wiki/JavaScript_Object_Notation) và thậm chí [EBML](https://vi.wikipedia.org/w/index.php?title=EBML&action=edit&redlink=1), nhưng [XML](https://vi.wikipedia.org/wiki/XML) là ngôn ngữ thường được sử dụng).
* XML thường là định dạng cho dữ liệu truyền, mặc dù bất cứ định dạng nào cũng có thể dùng, bao gồm HTML định dạng trước, văn bản thuần (plain text), JSON và ngay cả EBML.

# CHƯƠNG III: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

## 3.1. Đặc tả yêu cầu Website

**Mô tả quy trình nghiệp vụ bán hàng qua mạng**

1. ***Hoạt động của khách hàng***

Khi khách hàng quyết định tham gia vào hoạt động mua hàng của “cửa hàng  
ảo” thì mỗi khách hàng sẽ được cung cấp một giỏ hàng tương ứng. Khách hàng tự do lựa chọn loại hàng hóa nào mà mình thích vào giỏ hàng của mình. Thông qua các chức năng tìm kiếm của Website khách hàng có thể nhanh chóng tìm được loại hàng mà mình mong muốn mua nhanh nhất.

Khách hàng có thể cập nhật hay thêm các mặt hàng vào giỏ hàng của mình nếu muốn mua. Có thể xóa mặt hàng trong giỏ hàng mà mình không muốn mua.  
Nếu khách hàng chấp nhận thanh toán thì thì hệ thống yêu cầu nhập thông tin  
chi tiết của khách hàng như họ tên,địa chỉ Email, username, password...nếu khách  
hàng chưa đăng ký, đăng nhập, ngược lại nếu khách hàng nào đã đăng ký rồi thì họ chỉ cần đăng nhập bằng username và password của mình.

Khi khách hàng đã hoàn thành quá trình đặt hàng với hệ thống thì khách hàng tiến hành thanh toán, khách hàng có thể lựa chọn các hình thức thanh toán khác trên Website. Sau khi đã hoàn thành quá trình thanh toán khách hàng có thể thực hiện quá trình mua hàng của mình nếu khách hàng muốn mua thêm hàng hóa. Khi đã đặt hàng rồi khách hàng có thể tiếp tục tình trạng đơn hàng của mình qua website nếu có băn khoăn, nếu có những gì không đồng ý thì có thể gửi đơn khiếu nại và thông tin phản hồi hệ thống qua chức năng liên hệ giành cho khách hàng.

*b) Hoạt động nhà quản trị*

Cung cấp giỏ hàng cho khách hàng giúp khách hàng cảm thấy an tâm khi mua hàng mà không bị một số lỗi làm gián đoạn quá trình mua hàng của khách hàng, giúp quá trình mua hàng của họ trở nên hiệu quả hơn cho đến khi kết thúc việc mua hàng cũng như ấn định số lượng mua hàng.

Theo dõi khách hàng: cần phải biết khách hàng tham gia mua là ai và họ cần  
mua gì trong số khách hàng tham gia vào việc mua hàng. Thống kê đơn hàng mua của khách hàng, doanh thu từng mặt hàng của đơn hàng… mặt hàng nào bán nhiều nhất hiện tại.

Sau khi kết thúc phiên giao dịch của khách hàng, các đơn hàng khách hàng mua sẽ được tổng kết lại và chuyển sang bộ phận khác để xử lý như bộ phận kế toán, kinh doanh và bộ phận chuyển giao cho khách hàng.

Để thu hút khách hàng thường xuyên tham gia vào trang Web, nhà quản trị cần có biện pháp để xúc tiến đăng tải các chương trình khuyến mãi, giảm giá trong những diệp cần thiết để lôi kéo khách hàng về phía mình và có các ưu đãi với các khách hàng thường xuyên mua với số lượng lớn. Có các hoạt động quản lý và xử lý đơn hàng như: thông tin hiển thị các mặt hàng đã giao hay chưa?

## 3.2. Các yêu cầu chức năng

*Biểu đồ phân rã chức năng*

Diagram

Description automatically generated

**Hình** 3.1: Biểu đồ phân ra chức năng

***Các yêu cầu phi chức năng***

* Website đáp ứng được khả năng truy cập nhanh, thông tin tin cậy, chính xác.
* Giao diện thân thiện, dễ sử dụng, tính thẩm mỹ và tính ổn định cao.
* Website đảm bảo được tính bảo mật cao, phân quyền một cách hợp lý.
* Website có thể hoạt động 24/7/365.
* Website có khả năng lưu trữ, xử lý khối lượng lớn thông tin nhưng tốn ít tài nguyên trong hệ thống.

### 3.2.1. Biểu đồ dòng dữ liệu

#### *Biểu đồ luồng dữ liệu ngữ cảnhText Description automatically generated with medium confidence*

**Hình** 3.2: Biểu đồ dòng dữ liệu ngữ cảnh

* **Chú thích:**
  + 1,3: Gửi yêu cầu
  + 2,4: Đáp ứng yêu cầu

#### *Biểu đồ luồng dữ liệu mức 0*

Diagram, schematic

Description automatically generated

**Hình** 3.3: Biểu đồ dòng dữ liệu mức 0

* **Chú thích:**

|  |  |
| --- | --- |
| 1. Khách hàng đăng ký, đăng nhập 2. Cập nhật, kiểm tra tài khoản 3. Thông báo kết quả 4. Xác nhận đăng ký, đăng nhập 5. Quản trị website đăng nhập 6. Xác nhận đăng nhập 7. Yêu cầu tạo đơn hàng, xử lý đơn hàng 8. Lưu thông tin đơn hàng 9. Lấy kết quả từ CSDL 10. Đáp ứng yêu cầu 11. Yêu cầu tạo đơn hàng, xử lý đơn hàng 12. Đáp ứng yêu cầu 13. Yêu cầu cập nhật danh mục sản phẩm, sản phẩm, tìm kiếm sản phẩm 14. Đẩy thông tin vào CSDL 15. Lấy thông tin từ CSDL 16. Đáp ứng yêu cầu 17. Yêu cầu xem, tìm kiếm sản phẩm 18. Đẩy thông tin vào CSDL | 19. Lấy kết quả từ CSDL 20. Hiển thị kết quả 21. Yêu cầu xem tin tức 22. Truy vấn dữ liệu tin tức 23. Lấy kết quả từ CSDL 24. Hiển thị tin tức 25. Gủi liên hệ 26. Lưu thông tin vào CSDL 27. Kết quả 28. Thông báo kết quả 29. Yêu cầu xử lý liên hệ 30. Kết quả 31. Yêu cầu cập nhật tin tức 32. Kết quả 33. Yêu cầu thống kê 35. Kết quả thống kê người dùng 36. Kết quả thống kê sản phẩm 37. Kết quả thống kê đơn hàng |

#### Diagram Description automatically generatedB*iểu đồ luồng dữ liệu mức 1*

**Hình** 3.4: Mức 1 chức năng quản lý người dùng

* **Chú thích:**

|  |  |
| --- | --- |
| 1. Yêu cầu đăng ký tài khoản 2. Lưu đăng ký 3. Kết quả 4. Kết quả đăng ký 5. Yêu cầu đăng nhập 6. Thông tin tài khoản 7. Xác nhận đăng nhập 8. Yêu cầu quản trị | 9. Xác nhận quản trị 10. Xác nhận khách hàng 11. Yêu cầu quên mật khẩu 12. Thông tin tài khoản 13. Kết quả 14. Đáp ứng yêu cầu 15. Yêu cầu đăng xuất 16. Đáp ứng yêu cầu |

**Diagram, schematic

Description automatically generated**

**Hình** 3.5: Mức 1 chức năng quản lý bán hàng

* **Chú thích:**

|  |  |
| --- | --- |
| 1. Yêu cầu tạo đơn hàng 2. Thông tin đơn hàng 3. Kết quả thông tin đơn hàng 4. Đáp ứng yêu cầu 5. Yêu cầu tạo đơn hàng 6. Đẩy vào CSDL 7. Lấy thông tin từ CSDL 8. Đáp ứng yêu cầu | 9. Yêu cầu xử lý đơn hàng 10.Thông tin đơn hàng 11. Lấy thông tin đơn hàng 12. Đáp ứng yêu cầu 13. Lấy thông tin đơn hàng 14. Thanh toán 15. Đáp ứng yêu cầu 16. Đáp ứng yêu cầu |

**Diagram, schematic

Description automatically generated**

**Hình** 3.6: Chức năng quản lý sản phẩm

* **Chú thích:**

|  |  |
| --- | --- |
| 1. Yêu cầu cập nhật danh mục sản phẩm 2. Đẩy yêu cầu vào CSDL 3. Lấy thông tin từ CSDL 4. Đáp ứng yêu cầu 5. Yêu cầu cập nhật sản phẩm 6. Đẩy yêu cầu vào CSDL 7. Lấy thông tin từ CSDL 8. Đáp ứng yêu cầu 9. Yêu cầu cập nhật danh mục sản phẩm | 10. Đẩy yêu cầu vào CSDL 11. Lấy thông tin từ CSDL 12. Đáp ứng yêu cầu 13. Yêu cầu tìm kiếm sản phẩm 14. Đẩy yêu cầu vào CSDL 15. Lấy thông tin sản phẩm 16. Đáp ứng yêu cầu 17. Yêu cầu tìm kiếm sản phẩm 18. Đáp ứng yêu cầu |

Diagram

Description automatically generated**Hình** 3.7: Chức năng thống kê

* **Chú thích:**

|  |  |
| --- | --- |
| 1. Yêu cầu thống kê người dùng 2. Lưu yêu cầu 3. Lấy thông tin người dùng 4. Kết quả 5. Yêu cầu thống kê sản phẩm 6. Lưu yêu cầu 7. Lấy thông tin từ CSDL 8. Kết quả 9. Yêu cầu thống kê đơn hàng 10. Lưu yêu cầu | 11. Thông đơn hàng 12. Kết quả thống kê 13. Yêu cầu thống kê tin doanh thu 14. Lưu yêu cầu 15. Thông tin doanh thu 16. Kết quả thống kê |

## 3.3. Thiết kế cơ sở dữ liệu cho hệ thống

### 3.3.1. Thiết kế mô hình thực thể liên kết

* **Danh sách các thực thể của hệ thống**

**Bảng** 3.1: Danh sách các thực thể của hệ thống

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên thực thể** | **Mô tả** |
| **1** | Tbl\_admin | Lưu trữ tài khoản quản lý của admin |
| **2** | Tbl\_cart | Lưu thông tin sản phẩm và giỏ hàng mà người dùng muốn mua |
| **3** | Tbl\_carta | Sau khi khách hàng xác nhận mua, thông tin sản phẩm trong giỏ hàng được và lưu vào tbl\_carta để lưu lại thông tin sản phẩm khách hàng đã mua |
| **4** | Tbl\_danhmuc | Lưu danh mục sản phẩm |
| **5** | Tbl\_diachi | Lưu thông tin các địa chị các tỉnh, huyện, xã.. của Việt Nam |
| **6** | Tbl\_loaisan pham | Lưu thông tin loại sản phẩm |
| **7** | Tbl\_order | Lưu thông tin đơn hàng mà khách đã xác nhận mua |
| **8** | Tbl\_payment | Lưu phương thức thanh toán mà khách hàng đã chọn khi mua hàng |
| **9** | Tbl\_sanpham | Lưu thông tin sản phẩm |
| **10** | Tbl\_sanpham\_anh | Lưu thông tin ảnh của từng sản phẩm |

* **Chi tiết hóa các thực thể của hệ thống**

***Thực thể tbl\_admin***

**Bảng** 3.2: Thực thể tbl\_admin

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Mô tả** |
| **1** | Admin\_id | Id định danh admin |
| **2** | Admin\_name | Tên đăng nhập |
| **3** | Admin\_password | Mật khẩu đăng nhập |

***Thực thể tbl\_danhmuc***

**Bảng** 3.3: Thực thể tbl\_danhmuc

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Mô tả** |
| **1** | Danhmuc\_id | ID danh mục sản phẩm |
| **2** | Danhmuc\_ten | Tên danh mục sản phẩm |

***Thực thể tbl\_loaisanpham***

**Bảng** 3.4: Thực thể tbl\_loaisanpham

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Mô tả** |
| **1** | Loaisanpham\_id | ID loại sẩm phẩm |
| **2** | Loaisanpham\_ten | Tên loại sản phẩm |

***Thực thể tbl\_sanpham***

**Bảng** 3.5: Thực thể tbl\_sanpham

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Mô tả** |
| **1** | Sanpham\_id | ID sản phẩm |
| **2** | Sanpham\_tieude | Tên sản phẩm |
| **3** | Sanpham\_ma | Mã sản phẩm |
| **4** | Sanpham\_gia | Giá của sản phẩm |
| **5** | Sanpham\_chitiet | Mô tả thông số kỹ thuật |
| **6** | Sanphan\_baoquan | Mô tả bảo quản, giới thiệu sản phẩm |
| **7** | Sanpham\_anh | Tên ảnh của sản phẩm |
| **8** | Sanpham\_anh\_chitiet | Tên ảnh chi tiết giói thiệu sản phẩm |

***Thực thể tbl\_sanpham\_anh***

**Bảng** 3.6: Thực thể tbl\_sanpham\_anh

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Mô tả** |
| **1** | Sanpham\_anh\_id | ID sản phẩm ảnh |
| **2** | Sanpham\_anh | Tên ảnh |

***Thực thể tbl\_cart***

**Bảng** 3.7: Thực thể tbl\_cart

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Mô tả** |
| **1** | Cart\_id | ID sản phẩm |
| **2** | Session\_idA | Session\_id |
| **3** | Sanpham\_id | Id của sản phẩm định mua |
| **4** | Sanpham\_anh | Tên ảnh đại diện của sản phẩm |
| **5** | Sanpham\_tieude | Tên sản phẩm |
| **6** | Sanpham\_gia | Giá của sản phẩm |
| **7** | Quantitys | Số lượng sản phẩm |

***Thực thể tbl\_diachi***

**Bảng** 3.8: Thực thể tbl\_diachi

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Mô tả** |
| **1** | Tinh\_tp | Tên tỉnh, thành phố |
| **2** | Ma\_tinh | Mã tỉnh, thành phố |
| **3** | Quan\_huyen | Tên quận, huyện |
| **4** | Ma\_qh | Mã quận, huyện |
| **5** | Phuong\_xa | Tên phường, xã |
| **6** | Ma\_px | Mã phường, xã |

***Thực thể tbl\_order***

**Bảng** 3.9: Thực thể tbl\_order

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Mô tả** |
| **1** | Order\_id | ID của order |
| **2** | Session\_idA | Session\_id |
| **3** | Loaikhach | Loại khach mua sản phẩm (khách kẻ | thành viên,..) |
| **4** | Customer\_name | Tên khách hàng |
| **5** | Customer\_phone | Số điện thoại của khách hàng |
| **6** | Customer\_tinh | Mã tỉnh |
| **7** | Customer\_huyen | Mã huyện |
| **8** | Customer\_xa | Mã xã |
| **9** | Customer\_diachi | Địa chỉ cụ thể |

***Thực thể tbl\_payment***

**Bảng** 3.10: Thực thể tbl\_payment

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Mô tả** |
| **1** | Payment\_id | ID payment |
| **2** | Session\_idA | Session\_id |
| **3** | Giaohang | Phương thức giao hàng |
| **4** | Thanhtoan | Phương thức thanh toán |
| **5** | Order\_date | Ngày đặt hàng |
| **6** | statusA | Tình trạng đơn hàng(0-chưa giao hàng; 1-đã giao hàng) |

***Thực thể tbl\_carta (chi chiết đơn hàng)***

**Bảng** 3.11: Thực thể tbl\_carta

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Mô tả** |
| **1** | Carta\_id | ID sản phẩm |
| **2** | Session\_idA | Session\_id |
| **3** | Sanpham\_id | Id của sản phẩm định mua |
| **4** | Sanpham\_anh | Tên ảnh đại diện của sản phẩm |
| **5** | Sanpham\_tieude | Tên sản phẩm |
| **6** | Sanpham\_gia | Giá của sản phẩm |

### 3.3.2. Mô hình thực thể liên kết – ERM

Diagram

Description automatically generated

**Hình** 3.8: Mô hình thực thể liên kết - ERM

### 3.3.3. Thiết kế các bảng dữ liệu

Mô tả các bảng trong cơ sở dữ liệu:

***Bảng tbl\_admin***

**Bảng** 3.12: Bảng tbl\_admin

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| **1** | Admin\_id | Int | Khóa chính | Id định danh admin |
| **2** | Admin\_name | Varchar(255) | Not null | Tên đăng nhập |
| **3** | Admin\_password | Varchar(255) | Not null | Mật khẩu đăng nhập |

***Bảng tbl\_danhmuc***

**Bảng** 3.13: Bảng tbl\_danhmuc

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| **1** | Danhmuc\_id | Int | Khóa chính | ID danh mục sản phẩm |
| **2** | Danhmuc\_ten | Varchar(255) | Not null | Tên danh mục sản phẩm |

***Bảng tbl\_loaisanpham***

**Bảng** 3.14: Bảng tbl\_loaisanpham

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| **1** | Loaisanpham\_id | Int | Khóa chính | ID loại sẩm phẩm |
| **2** | Danhmuc\_id | Int | Tham chiếu từ danhmuc\_id từ tbldahmuc | ID danh mục sản phẩm |
| **3** | Loaisanpham | Nvarchar(255) | Not null | Tên loại sản phẩm |

***Bảng tbl\_sanpham***

**Bảng** 3.15: Bảng tbl\_sanpham

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| **1** | Sanpham\_id | Int | Khóa chính | ID sản phẩm |
| **2** | Sanpham\_tieude | Varchar(500) | Not null | Tên sản phẩm |
| **3** | Sanpham\_ma | Nvarchar(50) | Not null | Mã sản phẩm |
| **4** | Danhmuc\_id | Int | Tham chiếu từ danhmuc\_id từ tbl\_loaisanpham | ID danh mục |
| **5** | Loaisanpham\_id | Int | Tham chiếu từ loáianpham\_id từ tbl\_loaisanpham | ID loại sản phẩm |
| **6** | Sanpham\_gia | Varchar(100) | Not null | Giá của sản phẩm |
| **7** | Sanpham\_chitiet | Varchar(10000) | Not null | Mô tả thông số kỹ thuật |
| **8** | Sanphan\_baoquan | Varchar(10000) | Not null | Mô tả bảo quản, giới thiệu sản phẩm |
| **9** | Sanpham\_anh | Varchar(255) | Not null | Tên ảnh đại diện của sản phẩm |
| **10** | Sanpham\_anh\_chitiet | Varchar(255) | Not null | Tên ảnh giới thiệu sản phẩm chi tiết |

***Bảng tbl\_sanpham\_anh***

**Bảng** 3.16: Bảng tbl\_sanpham\_anh

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| **1** | Sanpham\_anh\_id | Int | Khóa chính | ID sản phẩm ảnh |
| **2** | Sanpham\_id | Int | Tham chiếu từ sanpham\_id từ tbl\_sanpham | ID sản phẩm |
| **3** | Sanpham\_anh | Nvarchar(255) | Not null | Tên ảnh |

***Bảng tbl\_cart***

**Bảng** 3.17: Bảng tbl\_cart

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| **1** | Cart\_id | Int | Khóa chính | ID sản phẩm |
| **2** | Session\_idA | Nvarchar(500) | Not null | Session\_id |
| **3** | Sanpham\_id | int | Tham chiếu từ sanpham\_id từ tbl\_sanpham | Id của sản phẩm định mua |
| **4** | Sanpham\_anh | Nvarchar(500) | Tham chiếu từ sanpham\_id từ tbl\_sanpham | Tên ảnh đại diện của sản phẩm |
| **5** | Sanpham\_tieude | Nvarchar(500) | Tham chiếu từ sanpham\_id từ tbl\_sanpham | Tên sản phẩm |
| **6** | Sanpham\_gia | Nvarchar(255) | Tham chiếu từ sanpham\_id từ tbl\_sanpham | Giá của sản phẩm |
| **7** | Quantitys | Varchar(255) | Tham chiếu từ sanpham\_id từ tbl\_sanpham | Số lượng sản phẩm |

***Bảng tbl\_diachi***

**Bảng** 3.18: Bảng tbl\_diachi

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| **1** | Tinh\_tp | Varchar(100) | Not null | Tên tỉnh, thành phố |
| **2** | Ma\_tinh | Nvarchar(10) | Not null | Mã tỉnh, thành phố |
| **3** | Quan\_huyen | Nvarchar(100) | Not null | Tên quận, huyện |
| **4** | Ma\_qh | Nvarchar(10) | Not null | Mã quận, huyện |
| **5** | Phuong\_xa | Nvarchar(100) | Not null | Tên phường, xã |
| **6** | Ma\_px | Nvarchar(10) | Not null | Mã phường, xã |

***Bảng tbl\_order***

**Bảng** 3.19: Bảng tbl\_order

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| **1** | Order\_id | Int | Khóa chính | ID địa chỉ giao hàng |
| **2** | Session\_idA | Nvarchar(50) | Not null | Session\_id |
| **3** | Loaikhach | Nvarchar(50) | Not null | Loại khach mua sản phẩm (khách kẻ | thành viên,..) |
| **4** | Customer\_name | Nvarchar(100) | Not null | Tên khách hàng |
| **5** | Customer\_phone | Nvarchar(100) | Not null | Số điện thoại của khách hàng |
| **6** | Customer\_tinh | Nvarchar(10) | Tham chiếu từ Ma\_tinh từ tbl\_diachi | Mã tỉnh |
| **7** | Customer\_huyện | Nvarchar(10) | Tham chiếu từ Ma\_huyện từ tbl\_diachi | Mã huyện |
| **8** | Customer\_xã | Nvarchar(10) | Tham chiếu từ Ma\_px từ tbl\_diachi | Mã xã |
| **9** | Customer\_diachi | Nvarchar(500) | Not null | Địa chỉ cụ thể |

***Bảng tbl\_payment***

**Bảng** 3.20: Bảng tbl\_payment

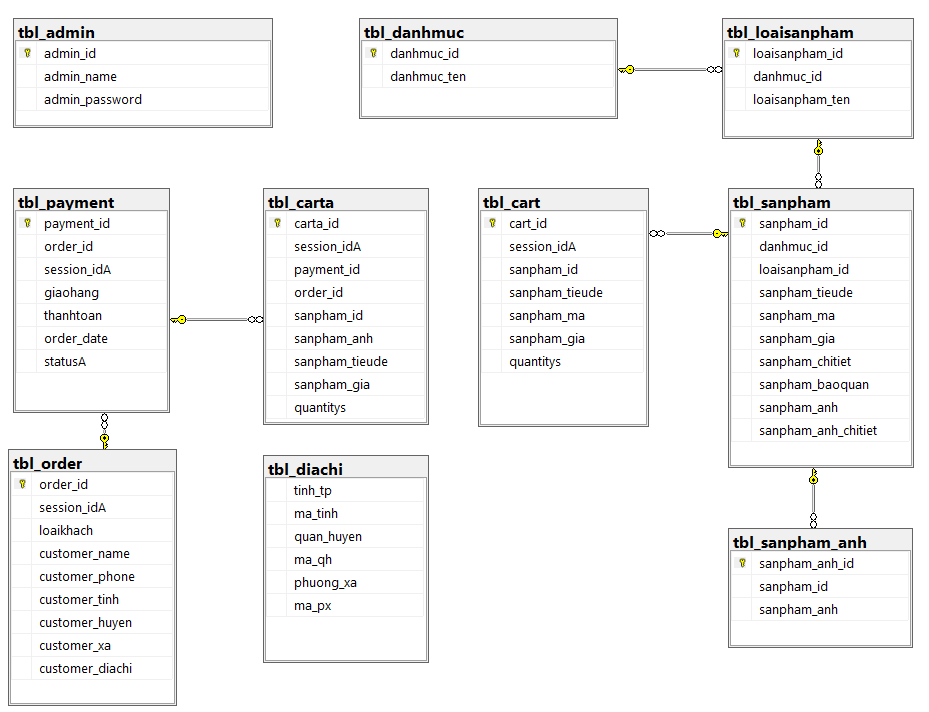
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **iểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| **1** | Payment\_id | Int | Khóa chính | ID thanh toán đơn hàng |
| **2** | Session\_idA | Varchar(50) | Not null | Session\_id |
| **3** | Giaohang | Varchar(50) | Not null | Phương thức giao hàng |
| **4** | Thanhtoan | Varchar(50) | Not null | Phương thức thanh toán |
| **5** | statusA | Int | Not null | Tình trạng đơn hàng |
| **6** | Order\_date | Varchar(50) | Not null | Ngày đặt hàng |

***Bảng tbl\_carta***

**Bảng** 3.21: Bảng tbl\_carta

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| **1** | Cart\_id | Int | Khóa chính | ID sản phẩm |
| **2** | Session\_idA | Nvarchar(50) | Not null | Session\_id |
| **3** | Order\_id | Int | Tham chiếu từ order\_id từ bảng order | ID địa chỉ giao hàng |
| **4** | Payment\_id | Int | Tham chiếu từ payment\_id từ bảng payment | Id thanh toán đơn hàng |
| **3** | Sanpham\_id | int | Tham chiếu từ sanpham\_id từ tbl\_sanpham | Id của sản phẩm mua |
| **4** | Sanpham\_anh | Nvarchar(50) | Tham chiếu từ sanpham\_id từ tbl\_sanpham | Tên ảnh đại diện của sản phẩm |
| **5** | Sanpham\_tieude | Nvarchar(500) | Tham chiếu từ sanpham\_id từ tbl\_sanpham | Tên sản phẩm |
| **6** | Sanpham\_gia | Nvarchar(20) | Tham chiếu từ sanpham\_id từ tbl\_sanpham | Giá của sản phẩm |

### 3.3.4. Liên kết giữa các bảng Relationships



**Hình** 3.9: Liên kết giữa các bảng Relationships

## 3.4. Thiết kế giao diện cho hệ thống

### 3.4.1. Đối với admin

* **A screenshot of a computer

  Description automatically generatedTrang đăng nhập admin**

**Hình** 3.10: Giao diện trang đăng nhập admin

* **A picture containing graphical user interface

  Description automatically generatedTrang chủ admin**

**Hình** 3.11: Giao diện trang chủ admin

* **Graphical user interface, text, application

  Description automatically generatedThêm danh mục sản phẩm**

**Hình** 3.12: Giao diện trang thêm danh mục

* **A screenshot of a computer

  Description automatically generatedDanh sách danh mục sản phẩm**

**Hình** 3.13: Giao diện trang danh sách danh mục sản phẩm

* **Graphical user interface, application

  Description automatically generatedThêm loại sản phẩm**

**Hình** 3.14: Giao diện trang thêm loại sản phẩm

* **A screenshot of a computer

  Description automatically generatedDanh sách loại sản phẩm**

**Hình** 3.15: Giao diện trang danh sách loại sản phẩm

* **Graphical user interface, application

  Description automatically generatedThêm sản phẩm**

**Hình** 3.16: Giao diện trang thêm sản phẩm

* **A picture containing graphical user interface

  Description automatically generatedDanh sách sản phẩm**

**Hình** 3.17: Giao diện trang danh sách sản phẩm

### 3.4.2. Đối với khách hàng

* Graphical user interface, website

  Description automatically generated**Giao diện trang chủ**

**Hình** 3.18: Giao diện trang chủ

Trang chủ: là trang đầu tiên khi người dùng truy cập vào website này, đây là trang web mặc định khi người dùng truy cập vào địa chỉ website.

* Trang chủ của website có thanh menu: thực hiện một số chức năng chính của trang web: giúp người dùng đăng nhập, đăng ký tài khoản, xem giỏ hàng, có các danh mục hàng hóa cho người dùng chọn giúp ngươi dùng mua hàng dễ dàng…
* Phần sliders: hiển thị các hiền ảnh giới thiệu về trang web, làm cho trang web sinh động
* Phần footer: hiển thị thông tin liên quan của website
* Graphical user interface, website, timeline

  Description automatically generated**Giao diện trang sản phẩm**

**Hình** 3.19: Giao diện trang sản phẩm

Trang sản phẩm: Hiển thị các loại sản phẩm mà người dùng chọn trong danh mục các sản phẩm ở trên thanh menu của website. Hiển thị các loại sản phẩm, sơ lược qua thông tin của sản phẩm cho người dùng tham khảo, chọn mua….

* Diagram, timeline

  Description automatically generated**Giao diện trang chi tiết sản phẩm**

**Hình** 3.20: Giao diện trang chi tiết sản phẩm

Trang chi tiết sản phẩm: Mô ta rõ thông tin về sản phẩm mà khách hàng muốn mua. Giới thiệu một số sản phẩm tương tự sản phẩm mà khách đang xem.

* **Graphical user interface, application, website

  Description automatically generatedGiao diện trang giỏ hàng**

**Hình** 3.21: Giao diện trang giỏ hàng

Trang giỏ hàng: Sau khi khách hàng đã chọn được mặt hàng muốn mua, hàng hóa sẽ được thêm vào giỏ hàng của khách hàng, website sẽ tính số tiền của các mặt hàng và chờ được khách hàng quết định mua và thanh toán.

* **A screenshot of a computer

  Description automatically generated with medium confidenceGiao diện trang giao hàng**

**Hình** 3.22: Giao diện trang giao hàng

Trang giao hàng: website yêu cầu khách hàng nhập thông tin cá nhân: tên, địa chỉ giao hàng …để người bán có thể giao hàng đến khách hàng.

* **Graphical user interface

  Description automatically generatedGiao diện trang thanh toán**

**Hình** 3.23: Giao diện trang thanh toán

Trang thanh toán: người dùng chọn phương thức để thanh toán rồi hoàn tất quá trình giao dịch.

* Graphical user interface, text

  Description automatically generated**Giao diện trang đặt hàng thành công**

**Hình** 3.24: Giao diện trang đặt hàng thành công

Trang thanh toán thành công: thông báo cho người dùng biết đã thanh toán thành công…

………

* Graphical user interface, website

  Description automatically generated**Giao diện trang chi tiết đơn hàng**

**Hình** 3.25: Giao diện trang chi tiết đơn hàng

Trang chi tiết đơn hàng hiển thị thông tin các đơn hàng khách hàng đã mua.

# Chương IV: TRIỂN KHAI WEBSITE

## 4.1. Kết nối CSDL

### 4.1.1. Giới thiệu ngôn ngữ lập trình PHP

PHP - viết tắt hồi quy của "Hypertext Preprocessor", là một ngôn ngữ lập trình kịch bản được chạy ở phía server nhằm sinh ra mã html trên client. PHP đã trải qua rất nhiều phiên bản và được tối ưu hóa cho các ứng dụng web, với cách viết mã rõ rãng, tốc độ nhanh, dễ học nên PHP đã trở thành một ngôn ngữ lập trình web rất phổ biến và được ưa chuộng.

PHP chạy trên môi trường Webserver và lưu trữ dữ liệu thông qua hệ quản trị cơ sở dữ liệu nên PHP thường đi kèm với Apache, MySQL và hệ điều hành Linux (LAMP).

* Apache là một phần mềm web server có nhiệm vụ tiếp nhận request từ trình duyệt người dùng sau đó chuyển giao cho PHP xử lý và gửi trả lại cho trình duyệt.
* MySQL cũng tương tự như các hệ quản trị cơ sở dữ liệu khác (Postgress, Oracle, SQL server...) đóng vai trò là nơi lưu trữ và truy vấn dữ liệu.
* Linux: Hệ điều hành mã nguồn mở được sử dụng rất rộng rãi cho các webserver. Thông thường các phiên bản được sử dụng nhiều nhất là RedHat Enterprise Linux, Ubuntu...[7]

Diagram

Description automatically generated

**Hình** 4.1: Giới thiệu ngôn ngữ lập trình PHP

### 4.1.2. Kết nối với Database

Để tạo một kết nối đến Database sẽ cần phải có các thông tin sau:

* + **host**: là tên/ địa chỉ Host, nơi lưu trữ Database.
  + **username**: tên tài khoản cho phép sử dụng Database.
  + **password**: mật khẩu truy cập của username.
  + **database**: tên database kết nối.

<?php

define("DB\_HOST", "localhost");

define("DB\_USER", "root");

define("DB\_PASS", "");

define("DB\_NAME", "website\_giadung");

?>

Class Database{

   public $host   = DB\_HOST;

   public $user   = DB\_USER;

   public $pass   = DB\_PASS;

   public $dbname = DB\_NAME;

   public $link;

   public $error;

 public function \_\_construct(){

  $this->connectDB();

 }

private function connectDB(){

   $this->link = new mysqli($this->host, $this->user, $this->pass,

    $this->dbname);

   if(!$this->link){

     $this->error ="Connection fail".$this->link->connect\_error;

    return false;

   }

 }

// Select or Read data

public function select($query){

  mysqli\_set\_charset($this->link,'UTF8');

  $result = $this->link->query($query) or

   die($this->link->error.\_\_LINE\_\_);

  if($result->num\_rows > 0){

    return $result;

  } else {

    return false;

  }

 }

## 4.2. Triển khai các chức năng cho hệ thống

### 4.2.1. Đối với khách hàng

* Hiển thị tất cả sản phẩm
* Hiển thị sản phẩm theo danh mục
* Hiển thị chi tiết sản phẩm
* Đăng ký thành viên đăng nhập thành viên
* Tạo, xóa giỏ hàng
* Tạo và gửi đơn đặt hàng
* Hiển thị tình trạng đơn hàng
* **Hiển thị sản phẩm theo loại sản phẩm**

**Bảng** 4.1: Chức năng hiển thị sản phẩm theo loại sản phẩm

|  |  |
| --- | --- |
| Mô tả | Hiển thị sản phẩm theo từng danh mục, loại sản phẩm |
| Đầu vào | Click vào danh mục --> click loại sản phẩm |
| Quá trình xử lý | Lấy tất cả sản phẩm có trong CSDL theo danh mục, loại sản phẩm. |
| Đầu ra | Hiển thị danh sách sản phẩm |

Diagram

Description automatically generated

**Hình** 4.2: Lưu đồ thuật toán hiển thị sản phẩm

* **Chức năng chi tiết sản phẩm**

**Bảng** 4.2: Chức năng chi tiết sản phẩm

|  |  |
| --- | --- |
| Mô tả | Hiển thị chi tiết sản phẩm |
| Đầu vào | Click chuột vào tên hoặc hình sản phẩm |
| Quá trình xử lý | Truy xuất vào CSDL để lấy thông tin chi tiết sản phẩm |
| Đầu ra | Hiển thị thông tin chi tiết sản phẩm |

***Diagram

Description automatically generated***

**Hình** 4.3: Lưu đồ thuật toán hiển thị chi tiết sản phẩm

* **Chức năng tạo giỏ hàng**

**Bảng** 4.3: Chức năng tạo giỏ hàng

|  |  |
| --- | --- |
| Mô tả | Mô phỏng giỏ hàng thực tế dùng để chứa các mặt hàng mà khách hàng đã chọn. |
| Đầu vào | Thông tin sản phẩm khách hàng định mua |
| Quá trình xử lý | Click vào thêm sản phẩm vào giỏ hàng --> thêm thông tin vào CSDL --> cập nhập hiển thị trang giỏ hàng |
| Đầu ra | Danh sách các sản phẩm mà khách hàng đã chọn và tổng giá tiền |

***Diagram

Description automatically generated***

**Hình** 4.4: Lưu đồ thuật toán chức năng tạo giỏ hàng

* **Chức năng xóa giỏ hàng**

**Bảng** 4.4: Chức năng xóa giỏ hàng

|  |  |
| --- | --- |
| Mô tả | Khách hàng có thể xóa sản phẩm đã đưa vào trong giỏ hàng |
| Đầu vào | Sau khi đã chọn sản phẩm vào giỏ hàng khách hàng có thể xóa sản phẩm khi không muốn mua |
| Quá trình xử lý | Khách hàng nhấn xóa sản phẩm trong giỏ hàng 🡪 hệ thống xóa dữ liệu thông tin sản phẩm trong giỏ hàng 🡪 cập nhập lại trang giỏ hàng |
| Đầu ra | Hiển thị lại trang giỏ hàng đã xóa sản phẩm |

***Diagram

Description automatically generated***

**Hình** 4.5: Lưu đồ thuật toán chức năng xóa giỏ hàng

* **Chức năng tạo đơn hàng**

**Bảng** 4.5: Chức năng tạo đơn hàng

|  |  |
| --- | --- |
| Mô tả | Sau khi khách hàng chọn hàng vào giỏ hàng và quyết định đặt mua hàng |
| Đầu vào | Thông tin các sản phẩm khách hàng chọn trong giỏ hàng như : tên, số lượng, đơn giá, thành tiền. |
| Quá trình xử lý | Khách hàng nhập thêm thông tin khách hàng(tên, số điện thoại, email, địa chỉ giao hàng,…phương thức thanh toán) Lưu thông tin đơn hàng vào CSDL nếu tạo đơn hàng thành công |
| Đầu ra | Thông báo khách hàng đã tạo đơn hàng thành công hay chưa. |

***Diagram

Description automatically generated***

**Hình** 4.6: Lưu đồ thuật toán chức năng tạo đơn hàng

* **Chức năng hiển thị tình trạng đơn hàng**

**Bảng** 4.6: Chức năng hiển thị tình trạng đơn hàng

|  |  |
| --- | --- |
| Mô tả | Khi khách hàng đã xác nhận mua đơn hàng, khách hàng có thể xem thông tin về đơn hàng của mình |
| Đầu vào | Thông tin các sản phẩm khách hàng chọn mua. |
| Quá trình xử lý | Thông tin đơn hàng được lấy từ CSDL 🡪 hiển thị lên website |
| Đầu ra | Hiển thị thông tin đơn hàng, tình trạng đơn hàng |

***Diagram

Description automatically generated***

**Hình** 4.7: Lưu đồ thuật toán hiển thị tình trạng đơn hàng

### 4.2.2. Đối với nhà quản trị

* Đăng nhập quyền quản trị Admin.
* Hiển thị tất cả thông tin sản phẩm.
* Quản lý đơn hàng.
* Quản lý sản phẩm.
* **Chức năng đăng nhập quản trị**

**Bảng** 4.7: Chức năng đăng nhập quản trị

|  |  |
| --- | --- |
| Mô tả | Dùng cho quản trị viên đăng nhập khi có tài khoản |
| Đầu vào | Thông tin đăng nhập + Tên đăng nhâp + Mật khẩu |
| Quá trình xử lý | Kết nối CSDL so sánh thông tin đăng nhập có đúng không |
| Đầu ra | Thông báo kết quả đăng nhập |

**Diagram

Description automatically generated**

**Hình** 4.8: Lưu đồ thuật toán đăng nhập quản trị

* **Chức năng quản lý sản phầm**
* **Thêm danh mục sản phẩm**

**Bảng** 4.8: Chức năng thêm danh mục sản phẩm

|  |  |
| --- | --- |
| Mô tả | Thêm mới danh mục sản phẩm |
| Đầu vào | Tên danh mục sản phẩm |
| Quá trình xử lý | Kiểm tra dữ liệu form 🡪 đưa thông tin vào cơ sở dữ liệu |
| Đầu ra | Hiển thị danh mục sản phẩm mới |

**Diagram

Description automatically generated**

**Hình** 4.9: Lưu đồ thuật toán thêm danh mục sản phẩm

* **Sửa danh mục sản phẩm**

**Bảng** 4.9: Chức năng sửa danh mục sản phẩm

|  |  |
| --- | --- |
| Mô tả | Sửa tên danh mục sản phẩm |
| Đầu vào | Tên danh mục sản phẩm |
| Quá trình xử lý | Kiểm tra dữ liệu form 🡪 cập nhập thông tin vào cơ sở dữ liệu |
| Đầu ra | Hiển thị tên danh mục sản phẩm mới |

**Diagram

Description automatically generated**

**Hình** 4.10: Lưu đồ thuật toán sửa danh mục sản phẩm

* **Xóa danh mục sản phẩm**

**Bảng** 4.10: Chức năng xóa danh mục sản phẩm

|  |  |
| --- | --- |
| Mô tả | Xóa toàn bộ sản phẩm của danh mục và danh mục |
| Đầu vào | Tên danh mục sản phẩm |
| Quá trình xử lý | Xóa toàn bộ sản phẩm trong loại sản phẩm thuộc danh mục --> xóa toàn bộ sản phẩm thuộc danh mục --> xóa danh mục |
| Đầu ra | Xóa thành công danh mục |

**Diagram

Description automatically generated**

**Hình** 4.11: Lưu đồ thuật toán xóa danh mục sản phẩm

* **Thêm loại sản phẩm**

**Bảng** 4.11: Chức năng thêm loại sản phẩm

|  |  |
| --- | --- |
| Mô tả | Thêm mới loại sản phẩm |
| Đầu vào | Tên danh mục, tên loại sản phẩm |
| Quá trình xử lý | Kiểm tra dữ liệu form 🡪 đưa thông tin vào cơ sở dữ liệu |
| Đầu ra | Hiển thị loại sản phẩm mới |

**Diagram

Description automatically generated**

**Hình** 4.12: Lưu đồ thuật toán thêm loại sản phẩm

* **Sửa loại sản phẩm**

**Bảng** 4.12: Chức năng sửa loại sản phẩm

|  |  |
| --- | --- |
| Mô tả | Sửa tên loại sản phẩm |
| Đầu vào | Tên danh mục sản phẩm, tên loại sản phẩm |
| Quá trình xử lý | Kiểm tra dữ liệu form 🡪 cập nhập thông tin vào cơ sở dữ liệu |
| Đầu ra | Hiển thị tên loại sản phẩm mới |

**Diagram

Description automatically generated with medium confidence**

**Hình** 4.13: Lưu đồ thuật toán sửa loại sản phẩm

* **Xóa loại sản phẩm**

**Bảng** 4.13: Chức năng xóa loại sản phẩm

|  |  |
| --- | --- |
| Mô tả | Xóa toàn bộ sản phẩm của loại sản phẩm và tên loại sản phẩm |
| Đầu vào | Danh sách sản phẩm, tên loại sản phẩm |
| Quá trình xử lý | Xóa toàn bộ sản phẩm trong loại sản phẩm --> xóa loại sản phẩm |
| Đầu ra | Xóa thành công |

**Diagram

Description automatically generated**

**Hình** 4.14: Lưu đồ thuật toán xóa loại sản phẩm

* **Thêm sản phẩm**

**Bảng** 4.14: Chức năng thêm sản phẩm

|  |  |
| --- | --- |
| Mô tả | Thêm mới một sản phẩm |
| Đầu vào | +Tên sản phẩm + Thuộc danh mục sản phẩm  + Thuộc loại sản phẩm + Mô tả + Giá + Hình sản phẩm |
| Quá trình xử lý | Kiểm tra dữ liệu form 🡪 đưa thông tin vào CSDL |
| Đầu ra | Hiển thị thông tin về sản phẩm |

**Diagram

Description automatically generated**

**Hình** 4.15: Lưu đồ thuật toán thêm sản phẩm

* **Sửa thông tin sản phẩm**

**Bảng** 4.15: Chức năng sửa thông tin sản phẩm

|  |  |
| --- | --- |
| Mô tả | Thay đổi thông tin một sản phẩm |
| Đầu vào | Chọn sản phẩm cần thay đổi thông tin |
| Quá trình xử lý | Hiển thị thông tin sản phẩm 🡪 sửa thông tin 🡪 Kết nối CSDL để cập nhật thông tin |
| Đầu ra | Hiển thị thông tin mới của sản phẩm |

**Diagram

Description automatically generated**

**Hình** 4.16: Lưu đồ thuật toán sửa sản phẩm

* **Xóa sản phẩm**

**Bảng** 4.16: Chức năng xóa sản phẩm

|  |  |
| --- | --- |
| Mô tả | Xóa một sản phẩm |
| Đầu vào | +Tên sản phẩm + Thuộc danh mục sản phẩm + Mô tả + Giá + Hình sản phẩm |
| Quá trình xử lý | Chọn sản phẩm cần xóa 🡪 click xóa 🡪 xóa dữ liệu trong CSDL |
| Đầu ra | Sản phẩm sẽ được xóa ra khỏi CSDL |

**Diagram

Description automatically generated with medium confidence**

**Hình** 4.17: Lưu đồ thuật toán xóa sản phẩm

* **Quản lý đơn hàng**
* **Thống kê đơn hàng**

**Bảng** 4.17: Chức năng thống kê đơn hàng

|  |  |
| --- | --- |
| Mô tả | Thống kê các đơn hàng của khách hàng |
| Đầu vào | + ID giao dịch + Họ tên + Địa chỉ + Số điện thoại + Ngày tạo đơn hàng + Tình trạng đơn hàng |
| Quá trình xử lý | Lấy dữ liệu trong CSDL hiển thị (đơn hàng đã giao, chưa giao, tất cả đơn hàng) |
| Đầu ra | Hiển thị thông tin về các đơn hàng |

**Diagram

Description automatically generated**

**Hình** 4.18: Lưu đồ thuật toán thông kê đơn hàng

* **Xác nhận đơn hàng**

**Bảng** 4.18: Chức năng xác nhận đơn hàng

|  |  |
| --- | --- |
| Mô tả | Thống kê các đơn hàng của khách hàng |
| Đầu vào | + ID giao dịch + Họ tên, Địa chỉ, Số điện thoại + Ngày tạo đơn hàng + Tình trạng đơn hàng |
| Quá trình xử lý | Admin có thể thay đổi tình trạng đơn hàng (đã giao hay chưa giao) 🡪 cập nhập tình trạng của đơn hàng trong CSDL 🡪 cập nhập lại website |
| Đầu ra | Hiển thị thông tin về các đơn hàng đã được sử lý và chưa xử lý |

**Diagram

Description automatically generated**

**Hình** 4.19: Lưu đồ thuật toán xác nhận đơn hàng

* **Xóa đơn hàng**

**Bảng** 4.19: Chức năng xóa đơn hàng

|  |  |
| --- | --- |
| Mô tả | Xóa đơn hàng |
| Đầu vào | + ID giao dịch + Họ tên + Địa chỉ + Số điện thoại + Ngày tạo đơn hàng + Tình trạng đơn hàng |
| Quá trình xử lý | Admin click xóa đơn hàng 🡪xóa thông tin đơn hàng trong CSDL 🡪 cập nhập lại website |
| Đầu ra | Xóa thành công đơn hàng |

**Diagram

Description automatically generated**

**Hình** 4.20: Lưu đồ thuật toán xóa đơn hàng

# KẾT LUẬN

**Kết quả đạt được của đồ án**

Sau thời gian nghiên cứu và thực hiện đề tài, các yêu cầu chính của đề tài cơ bản đã hoàn tất với những nội dung chủ yếu sau:

Chương trình có giao diện thân thiện, thuận tiện cho người dùng.

* Chức năng cập nhật (thêm, sửa, xóa), tìm kiếm được thực hiện nhanh chóng.
* Dễ dàng sử dụng và quản lý
* Đề tài đã xây dựng được các chức năng cần thiết cho hệ thống quản lý
* Tương đối đầy đủ chức năng cho một hệ thống quản lý.

Được ôn tập lại kỹ càng kiến thức HTML, CSS, JavaScript, PHP. Bên cạnh đó em đã tiếp thu và tích lũy được rất nhiều kiến thức làm website.

**Hạn chế của đồ án**

Mặc dù đã rất cố gắng nhưng do kinh nghiệm còn hạn chế nên kết quả vẫn còn một số thiếu sót cần tiếp tục được hoàn thiện để hệ thống có thể ứng dụng trong thực tiễn đạt hiệu quả tốt hơn như:

* Khả năng xử lý nghiệp vụ chưa linh hoạt nhạy bén.
* Lưu trữ dữ liệu bằng mảng làm cho ứng dụng có hạn chế về bộ nhớ
* Chưa giải quyết trọn vẹn các vấn đề nảy sinh trong quá trình quản lý
* Giao diện website chưa được đẹp mắt
* Một số chức năng còn chưa tối

**Hướng phát triển của đồ án**

Nghiên cứu sâu và nhiều hơn kỹ thuật thiết kế website để hoàn thiện chương trình tối ưu so với thực tiễn.

# TÀI LIỆU THAM KHẢO

[1]. Khoa CNTT, *Giáo trình nội bộ “Thiết kế web với HTML/CSS &JavaScript”*

[2]. J.D Gauchat, *“HTML5 for Masterminds,”* 3rd Edition, Kindle Edition, 2018

[3]. Mark Myers, *“A Smarter Way to Learn HTML & CSS: Learn it faster. Remember it longer,”* NXB, Kindle Edition, 2017 Jennifer Robbins, *“HTML5 Pocket Reference: Quick, Comprehensive, Indispensable (Pocket Reference (O'Reilly)),”* Kindle Edition, 2016

[4]. <https://www.w3schools.com/html/>

[5]. <https://www.w3schools.com/css/>

[6]. <https://www.w3schools.com/js/>

[7]. [https://vnsharebox.com/blog/cach-tao-file-php-ket-noi-den-database-trong phpmyadmin/](https://vnsharebox.com/blog/cach-tao-file-php-ket-noi-den-database-trong%20phpmyadmin/)