KHOA KỸ THUẬT VÀ CÔNG NGHỆ

**BỘ MÔN CÔNG NGHỆ THÔNG TIN**

****

**THỰC TẬP ĐỒ ÁN CƠ SỞ NGÀNH**

**HỌC KỲ I, NĂM HỌC 2024-2025**

**THIẾT KẾ ỨNG DỤNG WEB**

**GIỚI THIỆU SẢN PHẨM PHỤ KIỆN**

**ĐIỆN THOẠI CỬA HÀNG QUỐC BẢO**

|  |  |
| --- | --- |
| *Giáo viên hướng dẫn*  ThS. Nguyễn Ngọc Đan Thanh | *Sinh viên thực hiện:*  Họ tên: La Thuấn Khang  Mã số sinh viên: 110122090  Lớp: DA22TTD |

***Trà Vinh, Tháng 12 Năm 2024***

KHOA KỸ THUẬT VÀ CÔNG NGHỆ

**BỘ MÔN CÔNG NGHỆ THÔNG TIN**

****

**THỰC TẬP ĐỒ ÁN CƠ SỞ NGÀNH**

**HỌC KỲ I, NĂM HỌC 2024-2025**

**THIẾT KẾ ỨNG DỤNG WEB**

**GIỚI THIỆU SẢN PHẨM PHỤ KIỆN**

**ĐIỆN THOẠI CỬA HÀNG QUỐC BẢO**

|  |  |
| --- | --- |
| *Giáo viên hướng dẫn*  ThS. Nguyễn Ngọc Đan Thanh | *Sinh viên thực hiện:*  Họ tên: La Thuấn Khang  Mã số sinh viên: 110122090  Lớp: DA22TTD |

***Trà Vinh, Tháng 12 Năm 2024***

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

*Trà Vinh, ngày … tháng … năm 2024*

**Giáo viên hướng dẫn**

*(Ký và ghi rõ họ tên)*

**NHẬN XÉT CỦA THÀNH VIÊN HỘI ĐỒNG**

*Trà Vinh, ngày … tháng … năm 2024*

**Thành viên hội đồng**

*(Ký và ghi rõ họ tên)*

**LỜI CẢM ƠN**

Trước tiên, tôi xin cảm ơn sâu sắc đến trường Đại học Trà Vinh đã tạo điều kiện để tôi thực hiện dự án này, những người đã đưa ra những sự hỗ trợ, góp ý, cũng như là sự giúp đỡ cô Nguyễn Ngọc Đan Thanh – người đóng vai trò quan trọng trong việc phát triển và làm cho dự án thành công tốt đẹp.

Tôi biết ơn tất cả những người đã dành thời công sức, kiến thức và thời gian quý báo của mình để hỗ trợ tôi. Điều đó sẽ là nguồn động lực giúp tôi phát triển bản thân, những ý kiến, lời khuyên đó sẽ góp phần tạo nên một môi trường làm việc tích cực

Xin chân thành cô Nguyễn Ngọc Đan Thanh và trường Đại học Trà Vinh đã giúp đỡ , do đây là dự án đầu tiên cũng như hạn chế về mặt kiến thức lẫn kinh nghiệm nên chắc chắn sẽ xảy ra những sai sót nhỏ, vậy nên tôi mong rằng sẽ nhận được những góp ý và lời khuyên, tôi sẽ đưa những lời khuyên, góp ý đó vào dự án này để tạo nên một thành công tốt đẹp .

Trân trọng,

**MỤC LỤC**

[CHƯƠNG 1. TỔNG QUAN NGHIÊN CỨU 1](#_Toc187094921)

[1.1. Giới thiệu đề tài 1](#_Toc187094922)

[1.2. Mục đích nghiên cứu 1](#_Toc187094923)

[1.3. Đối tượng nghiên cứu 2](#_Toc187094924)

[1.3.1 Thông tin sản phẩm phụ kiện điện thoại 2](#_Toc187094925)

[1.3.2 Công nghệ thiết kế và phát triển web 3](#_Toc187094926)

[1.3.3 Hành vi và nhu cầu tra cứu sản phẩm của người dùng 3](#_Toc187094927)

[1.3.4 Quy trình quản lý dữ liệu và xử lý thông tin 3](#_Toc187094928)

[1.4. Phạm vi nghiên cứu 3](#_Toc187094929)

[1.4.1 Công nghệ sử dụng 3](#_Toc187094930)

[1.4.2 Chức năng ứng dụng 4](#_Toc187094931)

[1.4.3 Quy mô nghiên cứu 4](#_Toc187094932)

[1.4.4 Khả năng nghiên cứu và áp dụng 4](#_Toc187094933)

[1.5. Phương pháp nghiên cứu 4](#_Toc187094934)

[CHƯƠNG 2. NGHIÊN CỨU LÝ THUYẾT 5](#_Toc187094935)

[2.1. Tổng quan thiết kế ứng dụng web 5](#_Toc187094936)

[2.1.1 Kiến trúc ứng dụng web 5](#_Toc187094937)

[2.1.2 Ngôn ngữ HTML, CSS, Javascript 6](#_Toc187094938)

[2.1.2.1 HTML 6](#_Toc187094939)

[2.1.2.2 CSS 9](#_Toc187094940)

[2.1.2.3 Javascript 24](#_Toc187094941)

[2.1.3 Tìm hiểu frontend framework 25](#_Toc187094942)

[2.1.4 Ngôn ngữ PHP và MySQL 27](#_Toc187094943)

[2.1.4.1 PHP 27](#_Toc187094944)

[2.1.4.2 MySQL 32](#_Toc187094945)

[2.1.5 Mối liên hệ giữa PHP và MySQL 35](#_Toc187094946)

[CHƯƠNG 3. HIỆN THỰC HÓA NGHIÊN CỨU 37](#_Toc187094947)

[3.1. Mô tả bài toán 37](#_Toc187094948)

[3.2. Phân tích thiết kế hệ thống 37](#_Toc187094949)

[3.2.1 Đặc tả yêu cầu hệ thống 37](#_Toc187094950)

[3.2.1.1 Yêu cầu chức năng 37](#_Toc187094951)

[3.2.1.2 Yêu cầu phi chức năng 38](#_Toc187094952)

[3.2.2 Kiến trúc hệ thống 39](#_Toc187094953)

[3.2.3 Thiết kế dữ liệu 39](#_Toc187094954)

[3.2.3.1 Mô hình ERD 39](#_Toc187094955)

[3.2.3.2 Danh sách các thực thể và mối kết hợp 39](#_Toc187094956)

[3.2.3.3 Chi tiết các thực thể và mối kết hợp 39](#_Toc187094957)

[3.2.4 Thiết kế xử lý 42](#_Toc187094958)

[3.2.4.1 Mô hình DFD mức ngữ cảnh 42](#_Toc187094959)

[3.2.4.2 Mô hình DFD mức 1 43](#_Toc187094960)

[3.2.4.3 Mô hình DFD mức 2 43](#_Toc187094961)

[3.2.5 Thiết kế giao diện 44](#_Toc187094962)

[3.2.5.1 Sơ đồ website 44](#_Toc187094963)

[3.2.5.2 Giao diện trang chủ 44](#_Toc187094964)

[3.2.5.3 Giao diện trang quản trị 45](#_Toc187094965)

[3.2.5.4 Giao diện chức năng tìm kiếm sản phẩm 45](#_Toc187094966)

[3.2.5.5 Giao diện chức năng Đăng nhập / Đăng kí 46](#_Toc187094967)

[3.2.5.6 Giao diện trang chi tiết sản phẩm 47](#_Toc187094968)

[3.2.5.7 Giao diện trang giỏ hàng / thanh toán / Lịch sử đơn hàng 48](#_Toc187094969)

[CHƯƠNG 4. KẾT QUẢ NGHIÊN CỨU 49](#_Toc187094970)

[4.1. Dữ liệu thử nghiệm 49](#_Toc187094971)

[4.2. Kết quả thực nghiệm 50](#_Toc187094972)

[4.2.1 Chức năng của trang giao diện 50](#_Toc187094973)

[4.2.1.1 Chức năng tra cứu 50](#_Toc187094974)

[4.2.1.2 Chức năng đăng nhập và đăng kí 51](#_Toc187094975)

[4.2.1.3 Chức năng Giỏ Hàng và Thanh Toán 51](#_Toc187094976)

[4.2.1.4 Chức năng xem lịch sử đơn hàng 52](#_Toc187094977)

[4.2.1.5 Chức năng đánh giá sản phẩm 52](#_Toc187094978)

[4.2.2 Chức năng của trang quản trị 53](#_Toc187094979)

[4.2.2.1 Chức năng quản lí sản phẩm 53](#_Toc187094980)

[4.2.2.2 Chức năng quản lí danh mục 53](#_Toc187094981)

[4.2.2.3 Chức năng quản lí đơn hàng 53](#_Toc187094982)

[4.2.2.4 Chức năng quản lí người dùng 54](#_Toc187094983)

[4.2.2.5 Chức năng đăng nhập/đăng xuất 54](#_Toc187094984)

[CHƯƠNG 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 55](#_Toc187094985)

[5.1. Kết luận 55](#_Toc187094986)

[5.2. Hạn chế 55](#_Toc187094987)

[5.3. Hướng phát triển 56](#_Toc187094988)

**DANH MỤC HÌNH ẢNH**

[Hình 2.1 Mô hình Client – Server 5](#_Toc187094989)

[Hình 2.2 HTML 6](#_Toc187094990)

[Hình 2.3 CSS 9](#_Toc187094991)

[Hình 2.4 Javascript 24](#_Toc187094992)

[Hình 2.5 PHP 27](#_Toc187094993)

[Hình 2.6 MySQL 32](#_Toc187094994)

[Hình 3.1 Mô hình ERD 39](#_Toc187094995)

[Hình 3.2 Sơ đồ DFD mức ngữ cảnh 42](#_Toc187094996)

[Hình 3.3 mô hình dfd mức 1 43](#_Toc187094997)

[Hình 3.4 mô hình dfd mức 2 43](#_Toc187094998)

[Hình 3.5 sơ đồ website 44](#_Toc187094999)

[Hình 3.6 Giao diện trang chủ 44](#_Toc187095000)

[Hình 3.7 Giao diện trang quản trị 45](#_Toc187095001)

[Hình 3.8 Giao diện trang tìm kiếm sản phẩm 45](#_Toc187095002)

[Hình 3.9 Giao diện trang đăng nhập / đăng kí 46](#_Toc187095003)

[Hình 3.10 Giao diện trang chi tiết sản phẩm 47](#_Toc187095004)

[Hình 3.11 Giao diện trang giỏ hàng 48](#_Toc187095005)

[Hình 3.12 Giao diện trang thanh toán 48](#_Toc187095006)

[Hình 3.13 Giao diện trang lịch sử đơn hàng 48](#_Toc187095007)

[Hình 4.1 các bảng MySQL chứa dữ liệu thử nghiệm 49](#_Toc187095008)

[Hình 4.2 bảng dữ liệu thử nghiệm chi tiết đơn hàng 49](#_Toc187095009)

[Hình 4.3 bảng dữ liệu thử nghiệm đánh giá sản phẩm 49](#_Toc187095010)

[Hình 4.4 bảng dữ liệu thử nghiệm danh mục sản phẩm 49](#_Toc187095011)

[Hình 4.5 bảng dữ liệu thử nghiệm đơn đặt hàng 50](#_Toc187095012)

[Hình 4.6 bảng dữ liệu thử nghiệm người dùng và vai trò của người dùng 50](#_Toc187095013)

[Hình 4.7 bảng dữ liệu thử nghiệm sản phẩm trên trang web 50](#_Toc187095014)

[Hình 4.8 giao diện chức năng tra cứu hoàn chỉnh 51](#_Toc187095015)

[Hình 4.9 Giao diện chức năng đăng nhập / đăng kí hoàn chỉnh 51](#_Toc187095016)

[Hình 4.10 Giao diện trang giỏ hàng và thanh toán hoàn chỉnh 52](#_Toc187095017)

[Hình 4.11 Giao diện xem lịch sử đơn hàng hoàn chỉnh 52](#_Toc187095018)

[Hình 4.12 Giao diện đánh giá sản phẩm hoàn chỉnh 52](#_Toc187095019)

[Hình 4.13 Giao diện quản lí sản phẩm hoàn chỉnh 53](#_Toc187095020)

[Hình 4.14 Giao diện quản lí danh mục hoàn chỉnh 53](#_Toc187095021)

[Hình 4.15 Giao diện quản lí đơn hàng hoàn chỉnh 54](#_Toc187095022)

[Hình 4.16 Giao diện quản lí người dùng hoàn chỉnh 54](#_Toc187095023)

[Hình 4.17 Giao diện đăng nhập trang quản trị hoàn chỉnh 54](#_Toc187095024)

**DANH MỤC BẢNG BIỂU**

[Bảng 3.1 Danh sách các thực thể của mô hình ERD 39](#_Toc187095104)

[Bảng 3.2 Bảng mô tả chi tiết thực thể NguoiDung 40](#_Toc187095105)

[Bảng 3.3 Bảng mô tả chi tiết thực thể DonHang 40](#_Toc187095106)

[Bảng 3.4 Bảng mô tả chi tiết thực thể ChiTietDonHang 40](#_Toc187095107)

[Bảng 3.5 Bảng mô tả chi tiết thực thể SanPham 41](#_Toc187095108)

[Bảng 3.6 Bảng mô tả chi tiết thực thể DanhGiaSanPham 41](#_Toc187095109)

[Bảng 3.7 Bảng mô tả chi tiết thực thể DanhMuc 42](#_Toc187095110)

# TỔNG QUAN NGHIÊN CỨU

## Giới thiệu đề tài

Trong bối cảnh thị trường công nghệ có nhiều bước tiến triển vượt bậc về kỹ thuật và hiệu năng , đặc biệt là trong lĩnh vực sản xuất điện thoại làm cho nhu cầu sử dụng điện thoại của người dùng càng trở nên đa dạng thì việc xây dựng một ứng dụng web giới thiệu phụ kiện điện thoại là một ý tưởng khá phù hợp. việc này không chỉ giúp người dùng tìm hiểu các loại phụ kiện điện thoại phù hợp mà còn giúp tối ưu và bảo vệ điện thoại khỏi các tác nhân gây hư hỏng.

Ứng dụng web này sẽ tập trung vào việc cung cấp thông tin chi tiết về các sản phẩm phụ kiện điện thoại được phân loại rõ ràng theo các hãng sản xuất, cho phép người dùng dễ dàng tra cứu sản phẩm theo thương hiệu hoặc theo mức giá phù hợp với nhu cầu. Hệ thống sẽ được thiết kế với các tính năng chính như:

- **Quản lý cơ sở dữ liệu**: Xây dựng cấu trúc cơ sở dữ liệu tối ưu để lưu trữ thông tin về phụ kiện điện thoại, bao gồm tên sản phẩm, mô tả, hình ảnh, giá cả và thương hiệu.

- **Giao diện người dùng**: Giao diện đơn giản, thân thiện và trực quan, được xây dựng bằng HTML và CSS để giới thiệu chi tiết từng sản phẩm, đồng thời hỗ trợ tìm kiếm và lọc theo thương hiệu hoặc giá.

- **Chức năng xử lý**: Sử dụng PHP để phát triển các tính năng tra cứu, hiển thị dữ liệu động từ cơ sở dữ liệu MySQL, đảm bảo hiệu suất và độ chính xác cao.

Việc xây dựng ứng dụng web này không chỉ giúp cửa hàng nâng cao hình ảnh chuyên nghiệp mà còn mang lại trải nghiệm tiện lợi cho người dùng trong việc tìm kiếm và lựa chọn phụ kiện điện thoại.

## Mục đích nghiên cứu

Mục đích của đề tài là xây dựng một ứng dụng web giới thiệu sản phẩm phụ kiện điện thoại cho cửa hàng Quốc Bảo, đáp ứng các tiêu chí hiện đại, tiện lợi và phù hợp với nhu cầu tìm kiếm thông tin của người dùng. Ứng dụng không chỉ hỗ trợ việc quản lý thông tin sản phẩm mà còn giúp khách hàng dễ dàng tiếp cận và tra cứu các sản phẩm phụ kiện theo thương hiệu hoặc mức giá.

Cụ thể:

Tạo công cụ quản lý thông tin sản phẩm hiệu quả: Phát triển cơ sở dữ liệu hỗ trợ lưu trữ, truy xuất và quản lý dữ liệu về phụ kiện điện thoại, bao gồm thương hiệu, thông số kỹ thuật, giá cả và hình ảnh minh họa.

Cải thiện trải nghiệm người dùng: Thiết kế giao diện thân thiện, trực quan, giúp người dùng nhanh chóng tìm thấy sản phẩm mong muốn thông qua bộ lọc thương hiệu hoặc giá cả.

Nâng cao tính chuyên nghiệp cho cửa hàng: Ứng dụng web là phương tiện hiện đại giúp cửa hàng Quốc Bảo - Thuấn Khang nâng cao uy tín và khả năng tiếp cận khách hàng trong thời đại số hóa.

Nghiên cứu này cũng hướng tới việc tận dụng các công nghệ web như HTML, CSS, PHP và MySQL, đồng thời đề xuất các giải pháp kỹ thuật phù hợp để xây dựng ứng dụng có hiệu năng cao, dễ bảo trì và mở rộng trong tương lai.

## Đối tượng nghiên cứu

Dự án hướng đến việc xây dựng ứng dụng web giới thiệu sản phẩm cho cửa hàng phụ kiện điện thoại Quốc Bảo, với các tính năng quản lý thông tin sản phẩm theo thương hiệu, tra cứu sản phẩm theo giá và thương hiệu, đồng thời cung cấp một giao diện thân thiện, trực quan cho người dùng, đối tượng nghiên cứu sẽ bao gồm những đối tượng như:

### Thông tin sản phẩm phụ kiện điện thoại

Các loại phụ kiện phổ biến như ốp lưng, tai nghe, cáp sạc, kính cường lực, pin dự phòng.

Các thương hiệu sản xuất phụ kiện như Apple, Samsung, Oppo, Xiaomi, Sony.

Các thuộc tính của sản phẩm bao gồm tên, mô tả, giá cả, hình ảnh, và thông tin thương hiệu.

### Công nghệ thiết kế và phát triển web

HTML và CSS để xây dựng giao diện người dùng trực quan, hiện đại.

PHP để xử lý logic máy chủ, tương tác với cơ sở dữ liệu và hiển thị nội dung động.

MySQL để thiết kế và quản lý cơ sở dữ liệu lưu trữ thông tin sản phẩm.

### Hành vi và nhu cầu tra cứu sản phẩm của người dùng

Cách khách hàng tìm kiếm thông tin sản phẩm phụ kiện điện thoại theo thương hiệu và giá cả.

Các yếu tố giao diện và chức năng cần thiết để đáp ứng trải nghiệm người dùng.

### Quy trình quản lý dữ liệu và xử lý thông tin

Các thao tác thêm, sửa, xóa và tìm kiếm sản phẩm trong cơ sở dữ liệu

Tương tác giữa giao diện người dùng và cơ sở dữ liệu thông qua các truy vấn động

Việc tập trung nghiên cứu các yếu tố trên sẽ đảm bảo ứng dụng đáp ứng được yêu cầu từ phía cửa hàng Quốc Bảo cũng như mang lại trải nghiệm tốt nhất cho người dùng.

## Phạm vi nghiên cứu

### Công nghệ sử dụng

Sử dụng HTML và CSS để thiết kế giao diện người dùng, đảm bảo tính thẩm mỹ và thân thiện.

Sử dụng PHP để xử lý logic phía máy chủ, quản lý các chức năng động của ứng dụng.

Sử dụng MySQL để thiết kế và quản lý cơ sở dữ liệu, lưu trữ thông tin chi tiết về sản phẩm phụ kiện.

### Chức năng ứng dụng

Quản lý danh sách sản phẩm phụ kiện điện thoại, bao gồm các thông tin như tên, mô tả, hình ảnh, giá cả và thương hiệu.

Xây dựng chức năng tra cứu sản phẩm theo thương hiệu hoặc khoảng giá, giúp người dùng dễ dàng tìm kiếm thông tin mong muốn.

Hiển thị thông tin sản phẩm dưới dạng danh sách hoặc chi tiết, đảm bảo cung cấp đầy đủ thông tin cần thiết.

### Quy mô nghiên cứu

Ứng dụng này tập trung vào giới thiệu sản phẩm, không bao gồm chức năng bán hàng hoặc thanh toán trực tuyến.

Hệ thống được thiết kế để phục vụ nhu cầu của cửa hàng phụ kiện điện thoại vừa và nhỏ với cơ sở dữ liệu quy mô trung bình, khu vực nằm trong TP Trà Vinh.

### Khả năng nghiên cứu và áp dụng

Phân tích cách các công nghệ HTML, CSS, PHP và MySQL hoạt động và tương tác để tạo ra một ứng dụng web hoàn chỉnh.

Giới hạn nghiên cứu vào việc phát triển giao diện và chức năng cơ bản, chưa mở rộng đến các công nghệ tiên tiến hoặc tích hợp công cụ bên ngoài.

Phạm vi nghiên cứu này giúp tập trung nguồn lực và thời gian vào các yếu tố cần thiết để xây dựng một ứng dụng web đơn giản nhưng hiệu quả, đáp ứng được yêu cầu của cửa hàng và mang lại giá trị thực tế.

## Phương pháp nghiên cứu

**Phương pháp nghiên cứu lý thuyết**: tìm hiểu ngôn ngữ HTML, CSS để tạo dựng giao diện, PHP và MySQL để tương tác cơ sở dữ liệu.

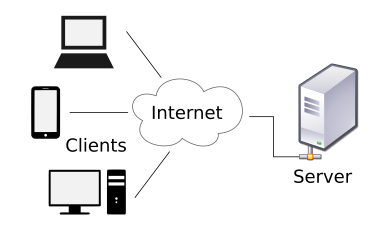
**Phương pháp nghiên cứu thực nghiệm**: Thiết kế, cài đặt và triển khai ứng dụng giới thiệu sản phẩm cửa hàng phụ kiện điện thoại Quốc Bảo.

# NGHIÊN CỨU LÝ THUYẾT

## Tổng quan thiết kế ứng dụng web

### Kiến trúc ứng dụng web

Ứng dụng web sẽ được xây dựng theo kiến trúc mô hình **Client-Server:**



Hình 2.1 Mô hình Client – Server

Trong đó:

**Client** : Là giao diện người dùng , nơi người dùng sẽ tương tác trực tiếp với ứng dụng. Giao diện này sẽ được xây dựng bằng HTML và CSS, có chức năng hiển thị dữ liệu từ phía máy chủ và nhận yêu cầu tìm kiếm từ người dùng.

**Server** : Là máy chủ, nơi xử lý tính năng logic và quản lý dữ liệu. PHP sẽ được sử dụng để xây dựng các script xử lý các yêu cầu từ người dùng, kết nối với cơ sở dữ liệu MySQL và trả kết quả về giao diện người dùng.

Cấu trúc:

* **Client** (Frontend):

**HTML**: Được sử dụng để xây dựng cấu trúc trang web, bao gồm các phần tử như bảng, danh sách sản phẩm, mô tả chi tiết, nút tìm kiếm, bộ lọc.

**CSS**: Dùng để trang trí giao diện, tạo bố cục và thiết kế các phần tử sao cho hợp lý và đẹp mắt. Đảm bảo giao diện thân thiện với người dùng để có thể sử dụng trên các thiết bị khác nhau (máy tính, điện thoại, tablet).

**JavaScript**: Có thể được sử dụng để tạo các tương tác động với người dùng, như tự động hiển thị các kết quả tìm kiếm mà không phải tải lại trang.

* **Server** (Backend):

**PHP**: Xử lý các yêu cầu từ client. Khi người dùng gửi yêu cầu (ví dụ: tìm kiếm sản phẩm theo thương hiệu hoặc giá), PHP sẽ nhận yêu cầu đó, thực hiện logic xử lý và truy xuất dữ liệu từ cơ sở dữ liệu.

**MySQL**: Hệ quản trị cơ sở dữ liệu lưu trữ thông tin các sản phẩm, thương hiệu, giá cả và các thông tin liên quan khác. PHP sẽ kết nối với MySQL thông qua các truy vấn SQL để lấy dữ liệu và gửi trả kết quả về client.

### Ngôn ngữ HTML, CSS, Javascript

#### HTML

HTML (HyperText Markup Language) là ngôn ngữ đánh dấu tiêu chuẩn dùng để tạo và cấu trúc nội dung cho các trang web.



Hình 2.2 HTML

HTML xác định cấu trúc cơ bản của một trang web bằng cách sử dụng các thẻ và thuộc tính để định nghĩa các phần tử như tiêu đề, đoạn văn, liên kết, hình ảnh, và nhiều thành phần khác.

* **Cách hoạt động của HTML**:

HTML hoạt động bằng cách sử dụng các thẻ để bao quanh các phần tử nội dung. Các thẻ này cho trình duyệt biết cách hiển thị nội dung.

* **Cấu trúc cơ bản của HTML**:

Cấu trúc của HTML cơ bản bao gồm các phần tử chính sau đây:

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ

Mô tả được tạo tự động

Giải thích các phần tử:

<!**DOCTYPE** **html**>: Khai báo loại tài liệu và phiên bản HTML đang sử dụng.

<**html**>: Phần tử gốc bao quanh toàn bộ nội dung của tài liệu HTML.

<**head**>: Chứa các thông tin meta, tiêu đề trang, và các liên kết đến tài nguyên bên ngoài như CSS.

<**meta** **charset="UTF**-**8**">: Đặt mã hóa ký tự cho tài liệu.

<**title**>: Tiêu đề của trang web, hiển thị trên tab của trình duyệt.

<**body**>: Chứa nội dung chính của trang web, bao gồm văn bản, hình ảnh, liên kết, v.v

Trong phần body ta thấy được thêm các phần tử nhỏ nằm trong đó:

<**h1**>: Thẻ tiêu đề chính, thường được sử dụng cho tiêu đề lớn nhất trên trang.

<**p**>: Thẻ đoạn văn, chứa văn bản.

<**a**>: Thẻ liên kết, sử dụng thuộc tính href để chỉ định URL đích.

<**img**>: Thẻ hình ảnh, sử dụng thuộc tính src để chỉ định đường dẫn đến hình ảnh và alt để cung cấp mô tả thay thế.

Ví dụ về một trang web giới thiệu phụ kiện điện thoại đơn giản bằng HTML:

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Phụ Kiện Điện Thoại Quốc Bảo</title>

</head>

<body>

<h1>Chào Mừng Đến Với Cửa Hàng Phụ Kiện Điện Thoại</h1>

<p>Chúng tôi cung cấp các loại phụ kiện điện thoại chất lượng cao, bao gồm ốp lưng, sạc dự phòng, tai nghe và nhiều hơn nữa.</p>

<h2>Sản Phẩm Nổi Bật</h2>

<ul>

<li><a href="https://vidu.com/op-lung">Ốp Lưng Điện Thoại</a></li>

<li><a href="https://vidu.com/sac-du-phong">Sạc Dự Phòng</a></li>

<li><a href="https://vidu.com/tai-nghe">Tai Nghe</a></li>

</ul>

<h2>Liên Hệ</h2>

<p>Địa chỉ: 123 Đường ABC</p>

<p>Email: <a href="Vidu@gmail.com">contact@vidu.com</a></p>

<img src="Phukien.jpg" alt="Phụ kiện điện thoại">

</body>

</html>

#### CSS

CSS (Cascading Style Sheets) là một ngôn ngữ được sử dụng để định dạng và trình bày nội dung của các tài liệu HTML trên web.



Hình 2.3 CSS

CSS giúp thay đổi cách hiển thị của trang web, bao gồm màu sắc, kích thước, khoảng cách, bố cục, và nhiều yếu tố khác mà không cần thay đổi mã HTML.

* **Cách hoạt động của CSS**:

CSS hoạt động bằng cách áp dụng các thuộc tính (properties) cho các phần tử HTML được chọn.

CSS sử dụng các bộ chọn (selectors) để xác định phần tử nào cần định dạng và các thuộc tính để chỉ định kiểu định dạng đó.

* **Mối liên hệ giữa CSS và HTML**

HTML và CSS là hai phần không thể tách rời trong việc xây dựng trang web: HTML cung cấp cấu trúc nội dung còn CSS giúp trang trí và trình bày nội dung đó.

* **Phương thức kết hợp CSS với HTML**:

**Inline CSS**: Chèn trực tiếp vào thẻ HTML bằng thuộc tính style.

**Internal** **CSS**: Đặt trong phần <style> bên trong phần <head> của trang HTML.

**External** **CSS**: Sử dụng file .css riêng và kết nối với trang HTML qua thẻ <link>. Phương pháp này thường được sử dụng nhất để tái sử dụng và tối ưu hóa mã CSS.

* **Kiến Thức Cơ Bản Về CSS**

1. **Bộ chọn**

Bộ chọn (Selectors) trong CSS là thành phần để chọn và áp dụng các quy tắc định dạng cho các phần tử HTML cụ thể.

**Bộ chọn phần tử (Element Selector):** dùng để chọn và áp dụng định dạng cho tất cả các phần tử HTML cùng loại.

Ví dụ: Để chọn tất cả các thẻ <p> và áp dụng màu chữ là xanh

p {

color: blue;

font-size: 14px;

}

1. **Bộ chọn lớp (Class Selector)**

**Lớp (class)** được dùng để chọn nhiều phần tử với cùng một định dạng, bắt đầu bằng dấu chấm và tên lớp.

Lưu ý: Một phần tử HTML có thể có nhiều lớp bằng cách thêm vào thuộc tính class (ngăn cách bằng dấu cách), ví dụ <div class="button highlight">.

Ví dụ: Để định dạng tất cả các phần tử có class="button" với nền xanh và chữ trắng.

.button {

    background-color: blue;

    color: white;

    padding: 10px 20px;

    border-radius: 5px;

  }

1. **Bộ chọn ID (ID Selector**)

Chọn một phần tử dựa trên id, dùng khi chỉ muốn định dạng một phần tử duy nhất.

Cú pháp**:** #idName { ... }

Lưu ý**:** ID phải là duy nhất trên mỗi trang, không nên sử dụng nhiều lần

Ví dụ:

#header {

    font-size: 24px;

    text-align: center;

    background-color: lightgray; }

1. **Bộ chọn nhóm (Group Selector)**

Giúp nhóm các bộ chọn lại để tiết kiệm mã CSS khi áp dụng cùng một quy tắc cho nhiều phần tử.

Ví dụ: Để áp dụng phông chữ Arial cho tất cả các tiêu đề <h1>, <h2>, và <h3>.

h1, h2, h3 {

    font-family: Arial, sans-serif;

    color: darkblue;

  }

1. **Bộ chọn con (Child Selector)**

Chỉ chọn các phần tử con trực tiếp của phần tử cha.

Cú pháp**:** parent > child

Ví dụ:

ul > li {

    list-style-type: square;

    color: green;

  }

1. **Bộ chọn hậu duệ (Descendant Selector)**

Chọn tất cả các phần tử con (bao gồm cả con trực tiếp và các con lồng sâu) của một phần tử cha.

Cú pháp**:** ancestor descendant

Ví dụ:

div p {

    color: red;

  }

1. **Bộ chọn thuộc tính (Attribute Selector)**

Chọn các phần tử có thuộc tính cụ thể, hữu ích khi cần định dạng phần tử có thuộc tính nhất định như type, href, src, v.v.

Cú pháp:

[**attribute**]: Chọn tất cả phần tử có thuộc tính.

[**attribute**="**value**"]: Chọn phần tử có thuộc tính và giá trị cụ thể.

[**attribute**~="**value**"]: Chọn phần tử có giá trị thuộc tính chứa từ nhất định

/\* Chọn tất cả input có type là text \*/

input[type="text"] {

    border: 1px solid #ccc;

    padding: 5px;

  }

  /\* Chọn tất cả các liên kết có chứa "download" trong href \*/

  a[href\*="download"] {

    color: red;

    text-decoration: underline;

  }

1. **Bộ chọn giả lớp (Pseudo-classes)**

Chọn phần tử trong các trạng thái đặc biệt (khi di chuột, khi được chọn đầu tiên, khi là phần tử con đầu tiên, v.v.).

Cú pháp**:** selector:pseudo-class

Ví dụ:

/\* Đổi màu khi di chuột vào liên kết \*/

a:hover {

    color: green;

  }

  /\* Định dạng dòng đầu tiên của mỗi đoạn văn \*/

  p:first-child {

    font-weight: bold;

  }

  /\* Chọn các phần tử con thứ 2 trong cha \*/

  li:nth-child(2) {

    background-color: yellow;

  }

1. **Box Model (Mô Hình Hộp)**

Box Model là khái niệm trong CSS để hiểu cách phần tử HTML được hiển thị trong trình duyệt.

Mỗi phần tử HTML được xem như một hộp chữ nhật bao gồm các phần: **Content, Padding, Border,** và **Margin**. Những phần này cùng nhau tạo nên cấu trúc hiển thị của phần tử trên trang.

1. **Content**

Định nghĩa: Đây là khu vực chính chứa nội dung của phần tử HTML, có thể là văn bản, hình ảnh, hoặc các phần tử con khác.

Thiết lập kích thước**:** Kích thước của phần content được xác định bằng thuộc tính width và height.

Vai trò**:** Phần tử hiển thị đúng nội dung của nó, và tất cả các thiết lập như padding, border, và margin sẽ bao quanh content

Ví dụ**:**

.content-box {

    width: 300px;

    height: 150px;

    background-color: blue;

  }

1. **Padding (Khoảng cách trong)**

Định nghĩa**:** Là khoảng cách từ rìa ngoài của content đến đường viền (border).

Thiết lập từng hướng**:** Có thể điều chỉnh từng phía của padding bằng cách sử dụng padding-top, padding-right, padding-bottom, và padding-left.

Tác dụng**:** Tạo khoảng trống giữa nội dung và viền, giúp nội dung dễ đọc hơn và tạo cảm giác thoáng hơn.

Cách tính**:** Phần padding sẽ làm tăng kích thước tổng thể của phần tử nếu box-sizing là content-box. Tuy nhiên, với border-box, phần tử sẽ không bị thay đổi kích thước khi thêm padding.

Ví dụ:

.padding-box {

    width: 200px;

    height: 100px;

    padding: 20px;

    background-color: lightblue;

  }

1. **Border (Đường viền)**

Định nghĩa**:** Đường viền bao quanh padding và content, có thể tùy chỉnh về độ dày, màu sắc và kiểu dáng.

Thiết lập riêng biệt cho từng phía**:** border-top, border-right, border-bottom, và border-left.

**Các thuộc tính của border:**

border-width: Độ dày của viền (ví dụ: 1px, 5px).

border-style: Kiểu viền (solid, dotted, dashed, double, groove, none, v.v.).

border-color: Màu sắc của viền

Cách tính kích thước**:** Phần border sẽ làm tăng kích thước tổng thể của phần tử nếu box-sizing là content-box

Ví dụ:

.border-box {

    width: 150px;

    height: 80px;

    padding: 10px;

    border: 5px solid red;

    background-color: lightyellow;

  }

1. **Margin (Khoảng cách ngoài)**

Định nghĩa**:** Khoảng cách từ đường viền của một phần tử đến các phần tử xung quanh.

Thiết lập riêng biệt cho từng phía**:** margin-top, margin-right, margin-bottom, và margin-left.

Tác dụng**:** Giúp tạo không gian giữa các phần tử, tránh chúng chồng lên nhau và tạo layout gọn gàng.

Cách hoạt động của auto margin**:** Nếu thiết lập margin: auto; theo chiều ngang (ví dụ margin-left: auto; margin-right: auto;), phần tử sẽ tự động căn giữa theo chiều ngang trong thẻ cha có display: block.

Lưu ý về margin collapse (sụp margin)**:** Nếu hai phần tử liên tiếp có margin-top và margin-bottom, thì khoảng cách thực tế giữa hai phần tử chỉ là giá trị lớn hơn trong hai giá trị margin này.

Ví dụ**:**

.margin-box {

    width: 100px;

    height: 50px;

    border: 2px solid green;

    margin: 20px;

    background-color: lightpink;

  }

1. **Flexbox Và Grid**

**CSS Flexbox**: được thiết kế để tạo bố cục một chiều, linh hoạt trong một hướng: có thể là hàng ngang hoặc cột dọc.

Flexbox giúp căn chỉnh và phân phối không gian giữa các phần tử, đặc biệt hữu ích khi sắp xếp các phần tử có kích thước không cố định hoặc khi cần phân chia khoảng cách đồng đều

**Cách hoạt động:**

Bố cục một chiều: Flexbox chỉ bố trí các phần tử theo hàng ngang hoặc cột dọc.

Thiết lập display: flex cho container để bật chế độ Flexbox.

Các phần tử bên trong (flex items) sẽ được bố trí và điều chỉnh theo trục chính hoặc trục phụ

**Thuộc tính quan trọng**

* **Trên container (cha):**

display**:** flex : Bật chế độ Flexbox cho container.

flex-direction**:** Quy định hướng của trục chính (row, column, row-reverse, column-reverse).

justify**-**content**:** Căn chỉnh các phần tử theo trục chính (ví dụ: center, space-between, space-around).

align-items: Căn chỉnh các phần tử theo trục phụ (ví dụ: center, stretch).

flex**-**wrap**:** Cho phép các phần tử xuống dòng khi container bị hẹp (wrap, nowrap)

* **Trên các phần tử con (flex items):**

flex-grow: Định nghĩa khả năng mở rộng của phần tử để chiếm thêm không gian trống.

flex-shrink: Quy định mức độ co lại khi không gian bị hẹp.

flex-basis: Thiết lập kích thước ban đầu của phần tử.

align-self: Căn chỉnh từng phần tử theo trục phụ độc lập với các phần tử khác (auto, flex-start, center, v.v.)

ví dụ:

.container {

    display: flex;

    flex-direction: row;

    justify-content: space-between;

    align-items: center;

  }

  .item {

    flex-grow: 1;

    padding: 10px;

  }

**CSS Grid**: là hệ thống bố cục hai chiều (2D), được thiết kế để tạo các bố cục phức tạp, có thể điều chỉnh theo cả chiều ngang và chiều dọc.

Grid cho phép chia trang thành các vùng (areas) hoặc xác định vị trí của các phần tử dựa trên hàng và cột

**Cách hoạt động**

Bố cục hai chiều**:** CSS Grid cho phép bố trí phần tử theo cả hàng và cột.

Thiết lập display**:** grid cho container để bật chế độ Grid.

Các phần tử bên trong (grid items) sẽ được đặt vào các ô (cells) trong lưới, dựa trên cấu trúc hàng (rows) và cột (columns) của Grid container.

**Thuộc tính quan trọng**

* **Trên container (cha):**

display: grid; : Bật chế độ CSS Grid cho container.

grid-template-columns và grid-template-rows: Quy định số lượng và kích thước của các cột và hàng trong Grid.

grid-gap: Thiết lập khoảng cách giữa các phần tử (dòng và cột).

grid-template-areas: Định nghĩa các vùng tên để sắp xếp phần tử theo tên vùng (area)

* **Trên các phần tử con (grid items)**

grid-column-start, grid-column-end: Quy định vị trí bắt đầu và kết thúc của phần tử trong cột.

grid-row-start, grid-row-end: Quy định vị trí bắt đầu và kết thúc của phần tử trong hàng.

grid-area: Đặt phần tử vào một khu vực được đặt tên (grid-template-areas)

Ví dụ:

.container {

    display: grid;

    grid-template-columns: 1fr 2fr 1fr;

    grid-template-rows: auto;

    gap: 10px;

  }

  .item1 {

    grid-column: 1 / 3;

  }

  .item2 {

    grid-row: 1 / 2;

  }

1. **Typography (Kiểu Chữ)**

Kiểu chữ trong CSS (Typography) là một phần quan trọng của thiết kế web, ảnh hưởng đến tính thẩm mỹ và trải nghiệm.

Việc điều chỉnh kiểu chữ bao gồm các yếu tố như font chữ, kích thước, khoảng cách và màu sắc.

**Font Family (Phông chữ)**

Thuộc tính font-family được sử dụng để quy định kiểu font cho văn bản.

CSS cho phép chỉ định nhiều font dự phòng (fallback fonts). Khi một font không khả dụng, trình duyệt sẽ dùng font tiếp theo trong danh sách.

Cú pháp:

****

Trong đó:

* Arial: là font chính.
* Helvetica: là font dự phòng nếu font chính không khả dụng
* Sans-serif: font tổng quát.

**Font Size (Kích thước chữ)**

Định nghĩa: Thuộc tính font-size quy định kích thước của font.

Đơn vị:

* **px** (pixel): kích thước cố định.
* **em**: tương đối với font-size của phần tử cha.
* **rem**: tương đối với font-size của phần tử gốc (<html>).
* **%:** phần trăm của kích thước chữ mặc định

Ví dụ:

font-size: 24px;

text-align: center;

background-color: lightgray;

1. **Font Weight (Độ đậm chữ)**

font-weight dùng để điều chỉnh độ đậm của chữ, từ rất mỏng đến rất dày.

* **normal**: độ đậm bình thường.
* **bold**: độ đậm lớn.
* **Số từ 100 đến 900:** biểu thị độ đậm từ rất mỏng (100) đến rất dày (900)

Ví dụ:

p:first-child {

    font-weight: bold;

    font-weight: 700;

  }

1. **Font Style (Kiểu chữ in nghiêng)**

Thuộc tính font-style quy định kiểu in nghiêng của chữ.

* **normal**: không nghiêng.
* **italic**: in nghiêng.
* **oblique**: chữ nghiêng theo một góc

Ví dụ: font-style: italic;

1. **Line Height (Chiều cao dòng)**

line-height điều chỉnh khoảng cách giữa các dòng trong một đoạn văn bản.

* **px:** kích thước cố định.
* **em:** tỷ lệ so với font-size.
* **giá trị không đơn vị:** quy định tỷ lệ so với font-size (ví dụ: 1.5 nghĩa là chiều cao dòng gấp 1.5 lần font-size)

ví dụ: line-height : 1.5;

1. **Letter Spacing** (Khoảng cách giữa các ký tự)

Tạo khoảng cách giữa các ký tự trong từ.

Cú pháp**:** letter-spacing: 2px;

1. **Text Transform (Chuyển đổi kiểu chữ)**

text-transform chuyển đổi kiểu viết hoa hoặc viết thường của văn bản.

* **uppercase**: chuyển đổi toàn bộ chữ thành viết hoa.
* **lowercase**: chuyển đổi toàn bộ chữ thành viết thường.
* **capitalize**: viết hoa chữ cái đầu mỗi từ

ví dụ: text-transform: uppercase;

1. **Text Align (Căn chỉnh văn bản)**

text-align căn chỉnh văn bản trong một phần tử.

* **left**: căn lề trái.
* **right**: căn lề phải.
* **center**: căn giữa.
* **justify**: căn đều hai lề

ví dụ: text-align: center;

1. **Text Decoration (Trang trí chữ)**

text-decoration thêm trang trí đặc biệt như gạch chân, gạch ngang, hoặc không trang trí.

* **underline**: gạch chân văn bản.
* **line-through**: gạch ngang văn bản.
* **none**: bỏ trang trí

Ví dụ: text-decoration: underline;

1. **Color (Màu chữ)**

color xác định màu sắc của văn bản, có thể dùng mã màu (hex, RGB, RGBA) hoặc tên màu (ví dụ: red, blue).

Ví dụ: h1 {:#ff0000 (hex)

Rbg(255, 0, 0)

Hsl(0, 100%, 50%)}

1. **Backgrounds (Nền)**

Trong CSS, thuộc tính nền (background) được sử dụng để thiết lập hình nền cho các phần tử trong tài liệu HTML. Nó có thể bao gồm màu sắc, hình ảnh, gradient, và các thiết lập liên quan khác

**background-color :** xác định màu nền cho phần tử.

Ví dụ: background-color :#ff0000 (hex)

Rbg(255, 0, 0)

Hsl(0, 100%, 50%)

**background-image:** Xác định hình ảnh làm nền cho phần tử.

Ví dụ**:** background-image: url(‘test.png);

**background-position:** xác định vị trí của hình ảnh nền.

Có thể là tọa độ pixel, phần trăm, hoặc các giá trị như top, right, bottom, left, và center.

Ví dụ**:** background-position: center (căn giữa)

background-position: 50% 50% (tọa độ phần trăm)

background-position: 10px 20px (tọa độ pixel)

**background-size:** xác định kích thước của hình ảnh nền.

* **cover:** Hình ảnh sẽ được mở rộng để phủ kín toàn bộ phần tử, có thể cắt bớt hình ảnh.
* **contain**: Hình ảnh sẽ được điều chỉnh để nằm vừa vặn trong phần tử mà không bị cắt bớt**.**
* **Kích thước cụ thể:** (ví dụ: 100px 50px)

Ví dụ: background-size: cover;

#### Javascript

JavaScript là một ngôn ngữ lập trình động, thường được sử dụng để tạo ra các trang web tương tác và phong phú. Nó là một phần không thể thiếu của phát triển web hiện đại.



Hình 2.4 Javascript

* **Cách hoạt động của Javascript:**

Trình duyệt: JavaScript chạy trên trình duyệt web, nơi mà nó có thể tương tác với các phần tử HTML và CSS.

Biên dịch và thực thi: Trình duyệt sẽ biên dịch mã JavaScript ngay lập tức khi nó gặp đoạn mã trong tài liệu HTML và thực thi nó

* **Mối liên hệ giữa Javascript, CSS và HTML:**

HTML (HyperText Markup Language): Cung cấp cấu trúc cho trang web. Nó xác định các phần tử như tiêu đề, đoạn văn, hình ảnh, v.v.

CSS (Cascading Style Sheets): Được sử dụng để định kiểu và trang trí các phần tử HTML. Nó quyết định giao diện và bố cục của trang.

JavaScript: Tạo ra sự tương tác cho các phần tử HTML và có thể thay đổi kiểu dáng do CSS xác định. Cả ba công nghệ này hoạt động cùng nhau để xây dựng trang web hoàn chỉnh.

### Tìm hiểu frontend framework

**Bootstrap** là một framework phát triển giao diện người dùng (UI framework) mã nguồn mở phổ biến, với mục đích giúp các lập trình viên xây dựng giao diện web một cách nhanh chóng và hiệu quả. Bootstrap cung cấp một bộ công cụ giao diện người dùng bao gồm các yếu tố giao diện như bảng, biểu mẫu, nút, thanh điều hướng, và các thành phần giao diện khác.

Trong bootstrap có hệ thống lưới (Grid system) giúp bố trí các phần tử một cách linh hoạt trên các kích thước màn hình khác nhau.

* **Cách Hoạt Động**

Hệ thống lưới trong Bootstrap hoạt động dựa trên:

**Hàng** (rows): Sử dụng class .row để nhóm các cột lại với nhau.

**Cột** (columns): Chia bố cục thành tối đa 12 cột mỗi hàng, sử dụng class như .col, .col-4, .col-md-6…

**Breakpoints**: Cung cấp các điểm ngắt cho thiết bị (xs, sm, md, lg, xl, xxl) để điều chỉnh giao diện theo kích thước màn hình.

Mỗi hàng trong hệ thống lưới được bọc trong một container (.container hoặc .container-fluid) để đảm bảo bố cục được căn chỉnh hợp lý.

<div class="container">

<div class="row">

<div class="col-4">Cột 1</div>

<div class="col-4">Cột 2</div>

<div class="col-4">Cột 3</div>

</div>

</div>

* **Điều chỉnh kích thước với breakpoint**

Các breakpoints giúp xác định cách hiển thị trên từng kích thước màn hình:

**.col-sm-\*:** Áp dụng cho màn hình nhỏ (≥576px)

**.col-md-\*:** Áp dụng cho màn hình vừa (≥768px)

**.col-lg-\*:** Áp dụng cho màn hình lớn (≥992px)

**.col-xl-\*:** Áp dụng cho màn hình rất lớn (≥1200px)

**.col-xxl-\*:** Áp dụng cho màn hình cực lớn (≥1400px)

<div class="container">

<div class="row">

<div class="col-sm-6 col-md-4 col-lg-3">Cột 1</div>

<div class="col-sm-6 col-md-4 col-lg-3">Cột 2</div>

<div class="col-sm-6 col-md-4 col-lg-3">Cột 3</div>

<div class="col-sm-6 col-md-4 col-lg-3">Cột 4</div>

</div>

</div>

* **Tùy chỉnh khoảng cách**

Hệ thống lưới hỗ trợ thêm khoảng cách giữa các cột hoặc hàng bằng class:

**Gutter spacing**: Sử dụng .g-1, .g-2, .g-3,… để điều chỉnh khoảng cách

**Padding**/**margin**: Sử dụng class như .**px-\*, .py-\*, .mx-\*, .my-\***

<div class="container">

<div class="row g-3">

<div class="col-6">Cột 1</div>

<div class="col-6">Cột 2</div>

</div>

</div>

### Ngôn ngữ PHP và MySQL

#### PHP

**PHP** là một ngôn ngữ lập trình kịch bản phía server được sử dụng chủ yếu để phát triển ứng dụng web động. Nó có thể được nhúng trực tiếp vào mã HTML và chủ yếu dùng để xử lý các yêu cầu người dùng và tương tác với cơ sở dữ liệu.



Hình 2.5 PHP

* **Cách hoạt động của PHP:**

PHP hoạt động trên mô hình Client-Server. Khi người dùng gửi yêu cầu (request) đến server, PHP xử lý yêu cầu đó trên máy chủ (server-side), thực hiện các thao tác cần thiết (như truy vấn cơ sở dữ liệu, tính toán, xử lý logic nghiệp vụ) và trả về kết quả cho người dùng dưới dạng HTML, XML hoặc các định dạng khác.

Khi trình duyệt của người dùng gửi một yêu cầu HTTP (ví dụ: yêu cầu truy xuất một trang web), yêu cầu này được gửi đến máy chủ web (ví dụ: Apache) mà PHP đang chạy trên đó. PHP sẽ nhận yêu cầu, xử lý các bước cần thiết, và trả về dữ liệu (ví dụ: trang HTML) cho trình duyệt của người dùng.

* **Các cú pháp trong PHP**

1. **Mở và đóng PHP**

Mọi mã PHP được viết theo cấu trúc: <?php ?>.

<?php

echo "Hello world";

?>

Trong đó echo là câu lệnh in ra màn hình.

1. **Biến**

Biến trong PHP được khai báo bằng ký hiệu $, không cần chỉ định kiểu dữ liệu.

<?php

$TenSanPham = "Ốp lưng điện thoại";

$Gia = 150000;

echo "Sản phẩm: $TenSanPham, Giá: $Gia VND";

?>

1. **Hằng số**

Hằng số được khai báo bằng hàm define hoặc từ khóa const.

<?php

define("ThuongHieu", "Phụ kiện Điện Thoại");

const MaSanPham = 123;

echo "Thương hiệu: " . ThuongHieu . ", Mã Sản Phẩm: " . MaSanPham. " Số ";

?>

1. **Câu lệnh điều kiện**

**if, else, else if**: Kiểm tra các điều kiện.

<?php

$SoLuong = 5;

if ($SoLuong > 0) {

echo "Sản phẩm còn hàng!";

} elseif ($SoLuong == 0) {

echo "Sản phẩm đã hết hàng.";

} else {

echo "Không có thông tin.";

}

?>

**Switch-case**: Kiểm tra nhiều giá trị

<?php

$SDanhMuc = "Tai nghe";

switch ($SDanhMuc) {

case "Ốp lưng":

echo "Sản phẩm này là ốp lưng cao cấp.";

break;

case "Tai nghe":

echo "Sản phẩm này là tai nghe không dây.";

break;

default:

echo "Loại phụ kiện chưa được cập nhật.";

}

?>

1. **Vòng lặp**

**For loop:** Lặp với số lần xác định.

<?php

for ($i = 1; $i <= 5; $i++) {

echo "Sản phẩm phụ kiện điện thoại #$i<br>";

}

?>

**While loop**: Lặp khi điều kiện đúng.

<?php

$SoLuong = 3;

while ($SoLuong > 0) {

echo "Còn $SoLuong sản phẩm trong kho.<br>";

$SoLuong--;

}

?>

**Do-while loop:** Lặp ít nhất một lần trước khi kiểm tra điều kiện

<?php

$SoLuong = 0;

do {

echo "Hiển thị sản phẩm ngay cả khi hết hàng.<br>";

} while ($SoLuong > 0);

?>

**Foreach loop**: Duyệt qua mảng

<?php

$SanPham = ["Ốp lưng", "Tai nghe", "Cáp sạc", "Pin dự phòng"];

foreach ($SanPham as $item) {

echo "Sản phẩm: $item<br>";

}

?>

1. **Hàm**

Hàm được định nghĩa bằng từ khóa function.

<?php

function getAccessoryInfo($TenSanPham, $Gia) {

return "Phụ kiện: $ TenSanPham, Giá: $ Gia VND";

}

echo getAccessoryInfo("Ốp lưng trong suốt", 200000);

?>

1. **Mảng**

Mảng chỉ số:

<?php

$SanPham = ["Ốp lưng", "Tai nghe", "Cáp sạc"];

echo "Sản phẩm phổ biến: " . $SanPham[0];

?>

Mảng kết hợp

<?php

$SanPham = ["TenSanPham" => "Pin dự phòng", "Gia" => 500000];

echo "Sản phẩm: " . $SanPham["TenSanPham"] . ", Giá: " . $SanPham["Gia"] . " VND";

?>

1. **Xử lí chuỗi**

Kết hợp chuỗi:

<?php

$ThuongHieu= "Phụ kiện XYZ";

$SanPham= "Ốp lưng silicon";

echo $ ThuongHieu. " giới thiệu sản phẩm: " . $ SanPham;

?>

#### MySQL

**MySQL** là một hệ quản trị cơ sở dữ liệu mã nguồn mở, được sử dụng phổ biến để lưu trữ và quản lý dữ liệu cho các ứng dụng web. MySQL sử dụng **SQL** (Structured Query Language) để thực hiện các thao tác quản lý cơ sở dữ liệu như chèn, cập nhật, xóa và truy vấn dữ liệu.



Hình 2.6 MySQL

* **Các khái niệm cơ bản trong MySQL:**

**Cơ sở dữ liệu (Database)**:Là nơi lưu trữ các bảng dữ liệu. Một ứng dụng có thể có nhiều cơ sở dữ liệu.

**Bảng (Table)**: chứa các dữ liệu được tổ chức thành hàng (records) và cột (fields).

**Cột (Column)**: đại diện cho một loại thông tin cụ thể trong bảng, như tên sản phẩm, giá tiền.

**Hàng (Row)**: trong bảng là một bản ghi (record), chứa dữ liệu chi tiết cho từng đối tượng.

**Các cú pháp cơ bản trong MySQL:**

1. **Tạo cơ sở dữ liệu**

Sử dụng lệnh **CREATE DATABASE** dùng để tạo một cơ sở dữ liệu mới.

Cú pháp: CREATE DATABASE database\_name;

CREATE DATABASE PhuKienDienThoai;

1. **Sử dụng cơ sở dữ liệu**

Sử dụng lệnh **USE** để chọn cơ sở dữ liệu sẽ làm việc.

Cú pháp: USE database\_name;

USE PhuKienDienThoai;

1. **Tạo bảng**

Sử dụng lệnh **CREATE TABLE** để tạo một bảng mới trong cơ sở dữ liệu.

Cú pháp:

CREATE TABLE table\_name (

column1 datatype constraints,

column2 datatype constraints,

...

);

CREATE TABLE DanhMuc ( --Đây là ví dụ

MaDanhMuc VARCHAR(20) AUTO\_INCREMENT PRIMARY KEY,

TenDanhMuc VARCHAR(100) NOT NULL,

);

1. **Thêm dữ liệu**

Sử dụng lệnh **INSERT INTO** để thêm dữ liệu vào bảng.

Cú pháp:

INSERT INTO table\_name (column1, column2, ...)

VALUES (value1, value2, ...);

--Đây là ví dụ

INSERT INTO SanPham(TenSanPham, Gia, SoluongTonKho, Mota)

VALUES

('Ốp lưng', 150000, 20, 'Ốp lưng chất liệu silicon dẻo, bảo vệ tốt.'),

('Tai nghe', 500000, 15, 'Tai nghe không dây chất lượng cao.');

1. **Xem dữ liệu**

Sử dụng lệnh **SELECT** để truy vấn dữ liệu từ bảng

Cú pháp: SELECT column1, column2, ... FROM table\_name;

SELECT TenSanPham, Gia FROM SanPham;

1. **Cập nhật dữ liệu**

Sử dụng lệnh **UPDATE** để thay đổi dữ liệu trong bảng

Cú pháp:

UPDATE table\_name

SET column1 = value1, column2 = value2, ...

WHERE condition;

--Đây là ví dụ

UPDATE SanPham

SET Gia= 140000

WHERE TenSanPham= 'Ốp lưng silicon';

1. **Xóa dữ liệu**

Sử dụng lệnh **DELETE** để xóa dữ liệu từ bảng

Cú pháp: DELETE FROM table\_name WHERE condition;

DELETE FROM SanPham WHERE TenSanPham= 'Ốp lưng silicon';

1. **Xóa bảng**

Lệnh **DROP** **TABLE** để xóa bảng khỏi cơ sở dữ liệu.

Cú pháp: DROP TABLE table\_name;

DROP TABLE SanPham;

### Mối liên hệ giữa PHP và MySQL

PHP và MySQL thường xuyên được sử dụng cùng nhau trong các ứng dụng web động. Mối liên hệ giữa chúng có thể được mô tả như sau:

**PHP kết nối với MySQL:**

PHP sử dụng các API (như MySQLi hoặc PDO) để kết nối và thực hiện các truy vấn trên MySQL.

<?php

$servername = "localhost";

$username = "username";

$password = "password";

$dbname = "myDB";

$conn = new mysqli($servername, $username, $password, $dbname);

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

echo "Connected successfully";

?>

**MySQL lưu trữ dữ liệu:**

MySQL lưu trữ tất cả các dữ liệu của ứng dụng web, chẳng hạn như thông tin người dùng, sản phẩm, đơn hàng, v.v.

<?php

$sql = "SELECT id, firstname, lastname FROM MyGuests";

$result = $conn->query($sql);

if ($result->num\_rows > 0) {

while($row = $result->fetch\_assoc()) {

echo "id: " . $row["id"]. " - Name: " . $row["firstname"]. " " . $row["lastname"]. "<br>";

}

} else {

echo "0 results";

}

?>

# HIỆN THỰC HÓA NGHIÊN CỨU

## Mô tả bài toán

Hệ thống được thiết kế để hỗ trợ việc giới thiệu các sản phẩm phụ kiện điện thoại, giúp người dùng dễ dàng tìm kiếm thông tin sản phẩm, xem đánh giá từ những khách hàng trước và quản lý danh mục, đơn hàng, cũng như thông tin người dùng. Hệ thống phân vai trò người dùng thành **admin** và **user**, mỗi vai trò có quyền hạn riêng.

## Phân tích thiết kế hệ thống

### Đặc tả yêu cầu hệ thống

#### Yêu cầu chức năng

* **Quản lí sản phẩm:**

Tạo mới, chỉnh sửa và xóa sản phẩm (Admin).

Hiển thị danh sách sản phẩm theo danh mục.

Tìm kiếm và lọc sản phẩm theo tên, giá cả, thương hiệu.

Xem chi tiết sản phẩm gồm hình ảnh, mô tả, giá cả, số lượng còn trong kho.

* **Quản lí người dùng**:

Đăng ký / đăng nhập tài khoản.

Phân loại người dùng theo vai trò (Admin hoặc User).

Quản lý thông tin cá nhân (User).

Quản lý danh sách người dùng (Admin).

* **Quản lí danh mục sản phẩm**:

Tạo mới và xóa danh mục sản phẩm (Admin).

Liệt kê tất cả các danh mục với số lượng sản phẩm trong mỗi danh mục.

* **Xử lí đơn hàng**:

Tạo đơn hàng khi người dùng xác nhận mua sản phẩm.

Theo dõi trạng thái đơn hàng (Đang xử lý, Đã giao, Đã hủy).

Cập nhật trạng thái đơn hàng (Admin).

* **Chi tiết đơn hàng**:

Liệt kê các sản phẩm trong đơn hàng cùng số lượng và đơn giá.

Tính tổng giá trị của từng đơn hàng.

* **Đánh giá sản phẩm**:

Người dùng đánh giá sau khi mua hàng. hiển thị các đánh giá từng sản phẩm.

#### Yêu cầu phi chức năng

* **Hiệu năng**:

Thời gian tải trang nhanh chóng.

Hỗ trợ nhiều người dùng truy cập đồng thời mà không bị giảm hiệu suất.

* **Khả năng mở rộng:**

Hệ thống được thiết kế để dễ dàng mở rộng, có thể thêm danh mục, sản phẩm hoặc tính năng mới mà không ảnh hưởng đến hoạt động hiện tại.

* **Tính bảo mật:**

Chỉ người dùng có vai trò Admin mới được phép thực hiện các thao tác như quản lý sản phẩm và đơn hàng.

* **Giao diện trực quan**:

Giao diện thân thiện với người dùng, dễ dàng thao tác .

Hỗ trợ hiển thị đa nền tảng (điện thoại, laptop,…)

* **Khả năng tương thích**:

Hỗ trợ trình duyệt phổ biến như Chrome, Firefox, Safari và Edge.

Tương thích với các hệ thống cơ sở dữ liệu phổ biến như MySQL.

### Kiến trúc hệ thống

### Thiết kế dữ liệu

#### Mô hình ERD

A diagram of a program

Description automatically generated with medium confidence

Hình 3.1 Mô hình ERD

#### Danh sách các thực thể và mối kết hợp

Bảng 3.1 Danh sách các thực thể của mô hình ERD

|  |  |  |
| --- | --- | --- |
| *STT* | *Tên thực thể/mối kết hợp* | *Diễn giải* |
| 1 | NguoiDung | Bảng chứa dữ liệu người dùng |
| 2 | DonHang | Bảng chứa dữ liệu đơn hàng |
| 3 | ChiTietDonHang | Bảng chứa dữ liệu chi tiết đơn hàng của người dùng |
| 4 | SanPham | Bảng chứa dữ liệu sản phẩm được bán |
| 5 | DanhGiaSanPham | Bảng chứa dữ liệu các đánh giá về sản phẩm từ người dùng |
| 6 | DanhMuc | Bảng chứa dữ liệu các danh mục của sản phẩm |

#### Chi tiết các thực thể và mối kết hợp

* **Bảng mô tả thực thể NguoiDung**

Tên thực thể: **NguoiDung**

Mô tả: Lưu trữ thông tin của người dùng

Chi tiết thực thể:

Bảng 3.2 Bảng mô tả chi tiết thực thể NguoiDung

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *STT* | *Tên thuộc tính* | *Diễn giải* | *Kiểu dữ liệu* | *Ràng buộc toàn vẹn* |
| 1 | MaNguoiDung | Mã xác định người dùng | Varchar(10) | PK |
| 2 | TenNguoiDung | Tên của người dùng | Varchar(255) | Không |
| 3 | Email | Email của người dùng | Varchar(255) | Không |
| 4 | MatKhau | Mật khẩu người dùng | Varchar(255) | Không |
| 5 | VaiTro | Vai trò của người dùng | Enum | Không |

* **Bảng mô tả thực thể DonHang**

Tên thực thể: **DonHang**

Mô tả: Lưu trữ thông tin về đơn hàng mà người dùng đặt

Chi tiết thực thể:

Bảng 3.3 Bảng mô tả chi tiết thực thể DonHang

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *STT* | *Tên thuộc tính* | *Diễn giải* | *Kiểu dữ liệu* | *Ràng buộc toàn vẹn* |
| 1 | MaDonHang | Mã đơn hàng | Varchar(10) | PK |
| 2 | MaNguoiDung | Mã xác định người dùng | Varchar(10) | FK |
| 3 | NgayDatHang | Ngày đặt hàng | Datetime | Không |
| 4 | TongGia | Tổng giá đơn hàng | Decimal(10,2) | Không |
| 5 | TrangThai | Trạng thái đơn hàng | Enum | Không |

* **Bảng mô tả thực thể ChiTietDonHang**

Tên thực thể: **ChiTietDonHang**

Mô tả: Lưu trữ thông tin về chi tiết đơn hàng mà người dùng đặt

Chi tiết thực thể:

Bảng 3.4 Bảng mô tả chi tiết thực thể ChiTietDonHang

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *STT* | *Tên thuộc tính* | *Diễn giải* | *Kiểu dữ liệu* | *Ràng buộc toàn vẹn* |
| 1 | MaChiTietDonHang | Mã chi tiết đơn hàng | Varchar(10) | PK |
| 2 | MaDonHang | Mã đơn hàng | Varchar(10) | FK |
| 3 | MaSanPham | Mã sản phẩm | Varchar(10) | FK |
| 4 | SoLuong | Số lượng sản phẩm | Int | Không |
| 5 | DonGia | Giá của sản phẩm | Decimal(10,2) | Không |

* **Bảng mô tả thực thể SanPham**

Tên thực thể: **SanPham**

Mô tả: Lưu trữ thông tin về sản phẩm có trên web

Chi tiết thực thể:

Bảng 3.5 Bảng mô tả chi tiết thực thể SanPham

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *STT* | *Tên thuộc tính* | *Diễn giải* | *Kiểu dữ liệu* | *Ràng buộc toàn vẹn* |
| 1 | MaSanPham | Mã sản phẩm | Varchar(10) | PK |
| 2 | MaDanhMuc | Mã danh mục | Varchar(10) | FK |
| 3 | TenSanPham | Tên sản phẩm | Varchar(255) | Không |
| 4 | Mota | Mô tả sản phẩm | Text | Không |
| 5 | Gia | Giá cả | Decimal(10,2) | Không |
| 6 | ThuongHieu | Thương hiệu | Varchar(255) | Không |
| 7 | SoLuongTonKho | Sản phẩm trong kho | Int | Không |
| 8 | HinhAnh | Hình ảnh sản phẩm | Varchar(255) | Không |

* **Bảng mô tả thực thể** **DanhGiaSanPham**

Tên thực thể: **DanhGiaSanPham**

Mô tả: Lưu trữ thông tin các đánh giá về sản phẩm có trên web

Chi tiết thực thể:

Bảng 3.6 Bảng mô tả chi tiết thực thể DanhGiaSanPham

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *STT* | *Tên thuộc tính* | *Diễn giải* | *Kiểu dữ liệu* | *Ràng buộc toàn vẹn* |
|  | MaDanhGia | Mã đánh giá sản phẩm | Varchar(10) | PK |
|  | MaNguoiDung | Mã xác định người dùng | Varchar(10) | FK |
|  | MaSanPham | Mã Sản Phẩm | Varchar(10) | FK |
|  | SoSaoDanhGia | Đánh giá (từ 1-5 \*) | Int | Không |
|  | NoiDungDanhGia | Nội dung đánh giá | Text | Không |

* **Bảng mô tả thực thể DanhMuc**

Tên thực thể: **DanhMuc**

Mô tả: Lưu trữ thông tin về danh mục của sản phẩm có trên web

Chi tiết thực thể:

Bảng 3.7 Bảng mô tả chi tiết thực thể DanhMuc

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *STT* | *Tên thuộc tính* | *Diễn giải* | *Kiểu dữ liệu* | *Ràng buộc toàn vẹn* |
| 1 | MaDanhMuc | Mã danh mục | Int(20) | PK |
| 2 | TenDanhMuc | Tên danh mục | Varchar(255) |  |

### Thiết kế xử lý

#### Mô hình DFD mức ngữ cảnh

A white oval with black text

Description automatically generated

Hình 3.2 Sơ đồ DFD mức ngữ cảnh

#### Mô hình DFD mức 1

A diagram of a flowchart

Description automatically generated

Hình 3.3 mô hình dfd mức 1

#### Mô hình DFD mức 2

A diagram of a diagram

Description automatically generated with medium confidence

Hình 3.4 mô hình dfd mức 2

### Thiết kế giao diện

#### Sơ đồ website

A diagram of a company

Description automatically generated

Hình 3.5 sơ đồ website

#### Giao diện trang chủ

A screenshot of a computer

Description automatically generated

Hình 3.6 Giao diện trang chủ

#### Giao diện trang quản trị

A screenshot of a computer

Description automatically generated

Hình 3.7 Giao diện trang quản trị

#### Giao diện chức năng tìm kiếm sản phẩm

A screenshot of a computer screen

Description automatically generated

Hình 3.8 Giao diện trang tìm kiếm sản phẩm

#### Giao diện chức năng Đăng nhập / Đăng kí

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Hình 3.9 Giao diện trang đăng nhập / đăng kí

#### Giao diện trang chi tiết sản phẩm

A screenshot of a cell phone screen

Description automatically generated

Hình 3.10 Giao diện trang chi tiết sản phẩm

#### Giao diện trang giỏ hàng / thanh toán / Lịch sử đơn hàng

A screenshot of a computer

Description automatically generated

Hình 3.11 Giao diện trang giỏ hàng

A screenshot of a computer

Description automatically generated

Hình 3.12 Giao diện trang thanh toán

A screenshot of a computer

Description automatically generated

Hình 3.13 Giao diện trang lịch sử đơn hàng

# KẾT QUẢ NGHIÊN CỨU

## Dữ liệu thử nghiệm

Dự án thiết kế ứng dụng web giới thiệu sản phẩm phụ kiện điện thoại tại shop Quốc Bảo sẽ bao gồm các dữ liệu thử nghiệm sau:

A screenshot of a computer

Description automatically generated

Hình 4.1 các bảng MySQL chứa dữ liệu thử nghiệm

Trong lần này sẽ có 6 bảng chứa dữ liệu để thử nghiệm trang web nhằm kiểm tra các chức năng để xem trang web có hoạt động hiệu quả hay không.

Bảng **ChiTietDonHang**



Hình 4.2 bảng dữ liệu thử nghiệm chi tiết đơn hàng

Bảng **DanhGiaSanPham**



Hình 4.3 bảng dữ liệu thử nghiệm đánh giá sản phẩm

Bảng **DanhMuc**

A screenshot of a computer

Description automatically generated

Hình 4.4 bảng dữ liệu thử nghiệm danh mục sản phẩm

Bảng **DonHang**



Hình 4.5 bảng dữ liệu thử nghiệm đơn đặt hàng

Bảng **NguoiDung**



Hình 4.6 bảng dữ liệu thử nghiệm người dùng và vai trò của người dùng

Bảng **SanPham**

A screenshot of a computer

Description automatically generated

Hình 4.7 bảng dữ liệu thử nghiệm sản phẩm trên trang web

## Kết quả thực nghiệm

### Chức năng của trang giao diện

#### Chức năng tra cứu

Chức năng tra cứu hiện đang hoạt động khá tốt khi cho phép người dùng tìm kiếm theo giá sản phẩm và theo thương hiệu hoặc danh mục.

A screenshot of a device

Description automatically generated

Hình 4.8 giao diện chức năng tra cứu hoàn chỉnh

#### Chức năng đăng nhập và đăng kí

Chức năng đăng nhập và đăng kí hoạt động hiệu quả khi cho phép người dùng đăng nhập khi đã có tài khoản hoặc đăng kí một cách nhanh chóng.

A white background with a blue line

Description automatically generated

A white background with black lines

Description automatically generated

Hình 4.9 Giao diện chức năng đăng nhập / đăng kí hoàn chỉnh

#### Chức năng Giỏ Hàng và Thanh Toán

Chức năng giỏ hành hoạt động ổn cùng với chức năng thanh toán cho phép người dùng thêm sản phẩm vào giỏ hàng và lựa chọn thanh toán.

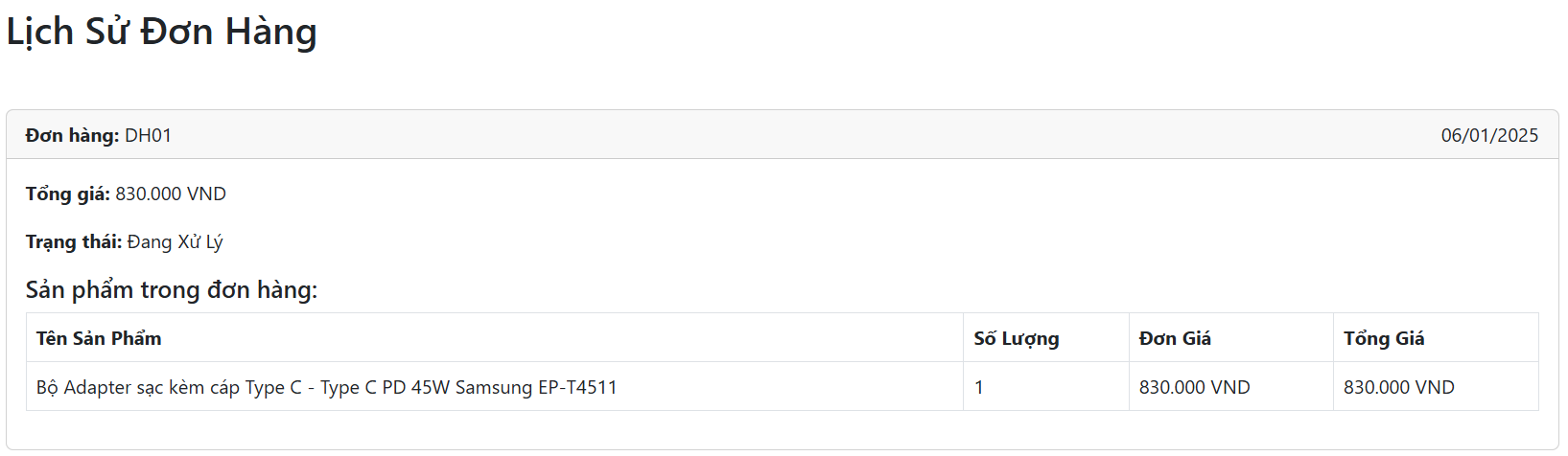
A screenshot of a computer

Description automatically generated

Hình 4.10 Giao diện trang giỏ hàng và thanh toán hoàn chỉnh

#### Chức năng xem lịch sử đơn hàng

Chức năng xem lịch sử đơn hàng cho phép người dùng xem lại những sản phẩm trong đơn hàng mà mình đã đặt.



Hình 4.11 Giao diện xem lịch sử đơn hàng hoàn chỉnh

#### Chức năng đánh giá sản phẩm

Chức năng đánh giá sản phẩm cho phép người dùng tham khảo các ý kiến của những người dùng khác nhằm để quyết định có nên sử dụng sản phẩm hay không.

A screenshot of a phone

Description automatically generated

Hình 4.12 Giao diện đánh giá sản phẩm hoàn chỉnh

### Chức năng của trang quản trị

#### Chức năng quản lí sản phẩm

Đây là chức năng cơ bản cho phép quản trị viên thêm hoặc xóa những sản phẩm nhằm tối ưu hoá cho việc quản lí.

A screenshot of a computer

Description automatically generated

Hình 4.13 Giao diện quản lí sản phẩm hoàn chỉnh

#### Chức năng quản lí danh mục

Chức năng quản lí danh mục chho phép quản trị viên thêm, xóa danh mục để nhằm phân loại danh mục của một sản phẩm nào đó.

A screenshot of a computer

Description automatically generated

Hình 4.14 Giao diện quản lí danh mục hoàn chỉnh

#### Chức năng quản lí đơn hàng

Chức năng quản lí đơn hàng cho phép quản trị viên quản lí và cập nhật trạng thái đơn hàng mà người dùng đặt để dễ dàng quản lí.

A white background with black text

Description automatically generated

A screenshot of a computer

Description automatically generated

Hình 4.15 Giao diện quản lí đơn hàng hoàn chỉnh

#### Chức năng quản lí người dùng

Chức năng quản lí người dùng cho phép quản trị viên xem được có bao nhiêu người dùng đã đăng kí và đăng nhập trên hệ thống và cho biết vai trò của từng người bao gồm cả quản trị viên.

A screenshot of a computer

Description automatically generated

Hình 4.16 Giao diện quản lí người dùng hoàn chỉnh

#### Chức năng đăng nhập/đăng xuất

Chức năng đăng nhập và đăng xuất cho phép quản trị viên truy cập vào trang quản trị đồng thởi cũng cho phép thoát khỏi phiên hoạt động trên trang quản trị

A screenshot of a computer

Description automatically generated

Hình 4.17 Giao diện đăng nhập trang quản trị hoàn chỉnh

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Kết luận

Dự án thiết kế ứng dụng web giới thiệu sản phẩm phụ kiện điện thoại cửa hàng Quốc Bảo đã đạt được những kết quả khá tích cực và mang lại những đóng góp quan trọng trong quá trỉnh triển khai, phát triển cũng như giới thiệu cho cửa hàng phụ kiện điện thoại Quốc Bảo. Tôi đã thành công trong việc tạo ra một nền tảng thân thiện với người dùng giúp họ dễ dàng tìm kiếm các sản phẩm phụ kiện phù hợp với phong cách của họ.

Đối với những đóng góp mới, tôi đã áp dụng các phương pháp, công nghệ mới để cải thiện trải nghiệm của người dùng, giao diện thân thiện, khả năng quản lí hiệu quả có lẽ là những điểm nổi bật trong số đó.

## Hạn chế

Bên cạnh những tính năng đó vẫn còn có những hạn chế:

Phần giao diện của người dùng vẫn quá đơn điệu, chưa được tối ưu và chăm sóc kĩ càng, điều này có thể làm giảm đáng kể trải nghiệm của người dùng cũng như làm cho công tác giới thiệu phụ kiện tại shop Quốc Bảo trở nên khó khăn.

Phần danh mục thì hiện tại chỉ có 4 sản phẩm là: Dây sạc, Ốp lưng, Sạc dự phòng và tai nghe, có thể thấy đây vẫn chưa là một danh mục hoàn chỉnh vì còn thiếu rất nhiều các danh mục sản phẩm phụ kiện điện thoại khác.

Những hạn chế trên cần được khắc phục để tạo nên một ứng dụng web giới thiệu sản phẩm phụ kiện điện thoại hoàn chỉnh. Cải thiện được những hạn chế này thì chất lượng của trang web sẽ được nâng cao cũng như tăng độ thẩm mỹ và ấn tượng của người dùng đối với Shop phụ kiện điện thoại Quốc Bảo.

## Hướng phát triển

Để nâng cao chất lượng trang web và trải nghiệm của người dùng thì tôi nghĩ trang web có thể cần hướng phát triển như sau:

Tối ưu hóa khả năng trải nghiệm của người dùng: tăng cường phát triển các chức năng mới và cải thiện giao diện để tạo ra một trang web thân thiện với mọi đối tượng.

Mở rộng danh mục sản phẩm: bằng cách thêm những sản phẩm mới và cập nhật thông tin mới nhất về lĩnh vực phụ kiện và điện thoại để từ đó tìm kiếm và tạo ra những danh mục chứa những sản phẩm mà người dùng muốn tìm kiếm

Học hỏi và tìm hiểu: tăng cường sự hiểu biết của bản thân bằng cách học thêm những kiến thức cần thiết trong việc thiết kế ứng dụng web để tạo ra các tính năng mới nhằm cải thiện chất lượng trải nghiệm của người dùng.

Tôi cam kết sẽ tiếp tục nỗ lực và cải thiện để mang đến trải nghiệm tốt nhất cho người dùng trong thời gian sắp tới.

**TÀI LIỆU THAM KHẢO**

|  |  |
| --- | --- |
| [1] | W3Schools, "HTML Tutorial," [Online]. Available: https://www.w3schools.com/html/. [Accessed 15 11 2024]. |
| [2] | P. H. Khang, Xây Dựng Ứng Dụng Web Bằng PHP Và Mysql, NXB Phương Đông, 2006. |