

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 BÁN HÀNG  
THỜI TRANG NỮ**

*Giáo viên hướng dẫn*  
ThS. Nguyễn Khắc Quốc

*Sinh viên thực hiện:*  
Họ tên: Châu Thị Mỹ Hương  
Mã số sinh viên: 110122082  
Lớp: DA22TTD

Trà Vinh, Tháng 12 Năm 2024

Trà Vinh, Tháng 12 Năm 2024

KHOA KỸ THUẬT VÀ CÔNG NGHỆ  
BỘ MÔN CÔNG NGHỆ THÔNG TIN



THỰC TẬP ĐO ÁN CƠ SỞ NGÀNH  
HỌC KỲ I, NĂM HỌC 2024-2025

THIẾT KẾ ỨNG DỤNG WEB BÁN HÀNG  
THỜI TRANG NỮ

Giáo viên hướng dẫn  
ThS. Nguyễn Khắc Quốc

Sinh viên thực hiện:  
Họ tên: Châu Thị Mỹ Hương  
Mã số sinh viên: 110122082  
Lớp: DA22TTD

Trà Vinh, Tháng 12 Năm 2024

Thiết kế website ứng dụng web bán hàng thời trang nữ

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

Trà Vinh, ngày ... tháng ... năm 2024

## **Giáo viên hướng dẫn**

(Ký và ghi rõ họ tên)

Châu Thị Mỹ Hương

Thiết kế website ứng dụng web bán hàng thời trang nữ

## **NHẬN XÉT CỦA THÀNH VIÊN HỘI ĐỒNG**

*Trà Vinh, ngày ... tháng ... năm 2024*

## Thành viên hội đồng

(Ký và ghi rõ họ tên)

Châu Thị Mỹ Hương

## 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 Thầy Nguyễn Khắc Quốc – 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 báu 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 cảm ơn cô Nguyễn Khắc Quốc 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
1.1. Giới thiệu đề tài .....	1
1.2 Mục đích nghiên cứu .....	2
1.3 Đối tượng nghiên cứu .....	2
1.3.1 Thông tin sản phẩm thời trang nữ .....	3
1.3.2 Công nghệ thiết kế và phát triển web .....	4
1.3.3 Hành vi và nhu cầu tra cứu sản phẩm của người dùng .....	4
1.3.4 Quy trình quản lý dữ liệu và xử lý thông tin .....	4
1.4 Phạm vi nghiên cứu .....	5
1.4.1 Công nghệ sử dụng .....	5
1.4.2 Chức năng ứng dụng .....	5
1.4.3 Quy mô nghiên cứu .....	5
1.4.4 Khả năng nghiên cứu và áp dụng .....	5
1.5. Phương pháp nghiên cứu .....	6
CHƯƠNG 2 NGHIÊN CỨU LÝ THUYẾT .....	7
2.1 Tổng quan về vấn đề nghiên cứu .....	7
2.2 Các khái niệm cơ bản .....	7
2.3 Các mô hình, lý thuyết và phương pháp nghiên cứu có liên quan .....	7
2.4. Khái niệm về mô hình Client-Server .....	7
CHƯƠNG 3 HIỆN THỰC HÓA NGHIÊN CỨU .....	51
3.1. Mô tả bài toán .....	51
3.2. Phân tích thiết kế hệ thống .....	51
3.2.1. Yêu cầu chức năng .....	51
3.2.2. Yêu cầu phi chức năng .....	52

# Thiết kế website ứng dụng web bán hàng thời trang nữ

---

3.3. Kiến trúc hệ thống .....	53
3.3.1. Thiết kế dữ liệu .....	53
3.3.2. Thiết kế giao diện .....	58
3.3.2.1. Sơ đồ website .....	58
3.3.2.2. Giao diện trang chủ .....	58
CHƯƠNG 4 KẾT QUẢ NGHIÊN CỨU .....	59
4.1. Bảng cơ sở dữ liệu .....	59
4.2. Kết quả thực nghiệm .....	60
4.2.1. Chức năng tra cứu .....	60
4.2.2. Chức năng đăng nhập đăng kí.....	61
4.2.3. Chức năng giỏ hàng .....	61
CHƯƠNG 5 KẾT LUẬN .....	62
5.1. Kết luận .....	62
5.2 Hướng phát triển.....	62
5.3 Kết luận cuối cùng.....	63
TÀI LIỆU THAM KHẢO .....	64

## **DANH MỤC HÌNH ẢNH**

Hình 2.1 Mô hình Client – Server .....	8
Hình 2.2 HTML .....	10
Hình 2.3 CSS .....	22
Hình 2.4 Javascript .....	38
Hình 2.5 PHP .....	41
Hình 2.6 MySQL .....	47
Hình 3.1 Mô hình ERD .....	53
Hình 3.2 Sơ đồ website.....	58

## **DANH MỤC BẢNG BIỂU**

Bảng 3.1 Danh sách các thực thể của mô hình ERD .....	54
Bảng 3.2 Bảng mô tả chi tiết thực thể NguoiDung .....	54
Bảng 3.3 Bảng mô tả chi tiết thực thể DonHang.....	54
Bảng 3.4 Bảng mô tả chi tiết thực thể ChiTietDonHang .....	55
Bảng 3.5 Bảng mô tả chi tiết thực thể SanPham .....	56
Bảng 3.6 Bảng mô tả chi tiết thực thể DanhGiaSanPham.....	56

## CHƯƠNG 1. TỔNG QUAN NGHIÊN CỨU

### 1.1. Giới thiệu đề tài

Trong bối cảnh ngành thời trang nữ đang ngày càng phát triển với sự thay đổi liên tục về xu hướng và nhu cầu của người tiêu dùng, việc xây dựng một ứng dụng web giới thiệu các sản phẩm thời trang nữ là một ý tưởng phù hợp và cần thiết. Mục đích của ứng dụng này là giúp người dùng dễ dàng tìm kiếm các sản phẩm thời trang nữ phù hợp với sở thích và nhu cầu, đồng thời giúp các cửa hàng thời trang nâng cao hình ảnh và tiếp cận khách hàng một cách hiệu quả.

#### Mục tiêu đề tài

Mục tiêu của đồ án này là thiết kế và phát triển một website thương mại điện tử cho ngành thời trang, đặc biệt là các sản phẩm thời trang bền vững. Website sẽ không chỉ cung cấp các sản phẩm thời trang thân thiện với môi trường mà còn tích hợp các tính năng hiện đại như gio hàng, thanh toán trực tuyến, tìm kiếm nâng cao và quản lý người dùng, giúp người tiêu dùng dễ dàng tiếp cận và mua sắm những sản phẩm có ý thức bảo vệ môi trường.

#### Lý do chọn đề tài

Ngành thời trang hiện nay đang đối mặt với nhiều vấn đề về ô nhiễm môi trường và khai thác tài nguyên không bền vững. Trong khi đó, sự phát triển mạnh mẽ của công nghệ thông tin đã giúp ngành thương mại điện tử bùng nổ, tạo cơ hội cho các sản phẩm bền vững tiếp cận gần hơn với người tiêu dùng. Việc xây dựng một website chuyên biệt cho thời trang bền vững không chỉ đáp ứng nhu cầu mua sắm của người tiêu dùng mà còn đóng góp vào xu hướng phát triển bền vững của ngành thời trang, thúc đẩy người tiêu dùng lựa chọn sản phẩm có lợi cho môi trường.

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

**Giao diện người dùng:** Giao diện 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, giá, hoặc phong cách.

**Chức năng xử lý:** Sử dụng PHP để phát triển các tính năng tra cứu và 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 sản phẩm thời trang nữ.

## 1.2 Mục đích nghiên cứu

Mục đích của đề tài là xây dựng một ứng dụng web giới thiệu sản phẩm thời trang nữ cho cửa hàng, 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 thời trang nữ theo thương hiệu, phong cách 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 thời trang nữ, bao gồm thương hiệu, 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, phong cách hoặc giá cả.
- Nâng cao tính chuyên nghiệp cho cửa hàng:** Ứng dụng web là phương tiện hiện đại giúp cửa hàng 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.

## 1.3 Đối tượng nghiên cứu

Dự án này hướng đến việc xây dựng ứng dụng web giới thiệu sản phẩm thời trang nữ cho cửa hàng, với các tính năng quản lý thông tin sản phẩm theo thương hiệu, tra cứu sản phẩm theo phong cách, giá cả, và cung cấp giao diện thân thiện, trực quan cho người dùng. Các đối tượng nghiên cứu bao gồm:

### 1.3.1 Thông tin sản phẩm thời trang nữ

Thông tin sản phẩm thời trang nữ là một yếu tố quan trọng trong hệ thống quản lý cửa hàng trực tuyến hoặc cơ sở dữ liệu. Việc lưu trữ và quản lý các thông tin chi tiết về sản phẩm sẽ giúp cửa hàng dễ dàng giới thiệu, tìm kiếm, và bán các mặt hàng thời trang.

#### Tên sản phẩm

**Mô tả:** **Tên sản phẩm giúp khách hàng nhận diện sản phẩm dễ dàng.** Nên đặt tên sản phẩm một cách rõ ràng và chi tiết để người dùng dễ dàng hiểu về sản phẩm đó. Ví dụ: "Áo khoác dạ nữ dáng dài", "Váy xòe công sở màu đen".

#### Mô tả sản phẩm

**Mô tả chi tiết:** Mô tả về sản phẩm sẽ cung cấp thông tin về chất liệu, kiểu dáng, mục đích sử dụng, đặc điểm nổi bật.

**Chất liệu:** Vải cotton, len, dạ, lụa, polyester,...

**Kiểu dáng:** Áo thun, áo sơ mi, váy, đầm, quần tây, quần jean,...

**Chi tiết:** Các yếu tố như độ dài, kiểu cổ, tay áo, chi tiết trang trí như nút, khóa, thêu,...

**Màu sắc:** Các màu sắc có sẵn cho sản phẩm (ví dụ: Đỏ, đen, trắng, xanh dương,...).

**Kích thước:** Các size có sẵn như S, M, L, XL, XXL, hoặc các kích thước khác tùy theo từng sản phẩm.

#### Giá sản phẩm

**Mô tả:** Đây là mức giá của sản phẩm. Có thể có các trường hợp như giá gốc, giá giảm, hay khuyến mãi.

**Giá gốc:** Mức giá ban đầu của sản phẩm.

**Giá khuyến mãi:** Mức giá sản phẩm sau khi áp dụng giảm giá.

**Khuyến mãi:** Thông tin về các chương trình giảm giá, voucher, hoặc các khuyến mãi liên quan.

#### Thương hiệu

## Thiết kế website ứng dụng web bán hàng thời trang nữ

---

**Mô tả:** Thương hiệu của sản phẩm là yếu tố quan trọng giúp xây dựng uy tín cho sản phẩm. Một số sản phẩm thời trang nổi tiếng sẽ được khách hàng nhận diện dựa trên thương hiệu

Ví dụ: Zara, H&M, Mango, Uniqlo, Gucci, Prada,...

### Hình ảnh sản phẩm

**Mô tả:** Hình ảnh là yếu tố quan trọng trong việc thu hút người mua, giúp họ hiểu rõ hơn về sản phẩm. Cần có hình ảnh chất lượng cao, bao gồm ảnh chụp từ nhiều góc độ khác nhau (mặt trước, mặt sau, cận cảnh chi tiết).

**Ảnh chính:** Hình ảnh chính cho sản phẩm.

**Ảnh phụ:** Hình ảnh chi tiết, ví dụ như các góc nhìn khác nhau, hoặc hình ảnh khi sản phẩm được mặc bởi người mẫu.

#### 1.3.2 Công nghệ thiết kế và phát triển web

- HTML và CSS để xây dựng giao diện người dùng hiện đại và thân thiện.
- 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.

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

- Cách khách hàng tìm kiếm thông tin sản phẩm thời trang nữ theo thương hiệu, phong cách và giá cả.
- Các yếu tố giao diện và chức năng cần thiết để đáp ứng trải nghiệm người dùng.

#### 1.3.4 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.

## 1.4 Phạm vi nghiên cứu

### 1.4.1 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 thời trang nữ.

### 1.4.2 Chức năng ứng dụng

- Quản lý danh sách sản phẩm thời trang nữ, 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, phong cách 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.

### 1.4.3 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 thời trang nữ vừa và nhỏ với cơ sở dữ liệu quy mô trung bình, khu vực có thể là thành phố hoặc khu vực cụ thể.

### 1.4.4 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.

### **1.5. Phương pháp nghiên cứu**

Phương pháp nghiên cứu lý thuyết giúp làm rõ các khái niệm, nguyên lý và các mô hình đã được nghiên cứu và ứng dụng trong các hệ thống tương tự. Phương pháp này sẽ giúp định hướng cho việc xây dựng hệ thống.

## CHƯƠNG 2 NGHIÊN CỨU LÝ THUYẾT

### 2.1 Tổng quan về vấn đề nghiên cứu

**Giới thiệu về lĩnh vực nghiên cứu:** Trình bày tổng quan về ngành, lĩnh vực nghiên cứu liên quan đến đồ án của bạn. Ví dụ, nếu bạn thực hiện đồ án về website thời trang, bạn cần giới thiệu về **thương mại điện tử, website bán hàng trực tuyến, lĩnh vực thời trang, và xu hướng tiêu dùng trực tuyến**.

### 2.2 Các khái niệm cơ bản

**Khái niệm về website bán hàng:** Giới thiệu về các khái niệm cơ bản trong lĩnh vực phát triển website thương mại điện tử, chẳng hạn như "Website thương mại điện tử", "Trang web bán lẻ" và "Thanh toán trực tuyến".

**Các công nghệ web:** Giới thiệu về các công nghệ web hiện đại như HTML, CSS, JavaScript, PHP. Cũng có thể đề cập đến các framework và thư viện hỗ trợ phát triển web như Bootstrap,

### 2.3 Các mô hình, lý thuyết và phương pháp nghiên cứu có liên quan

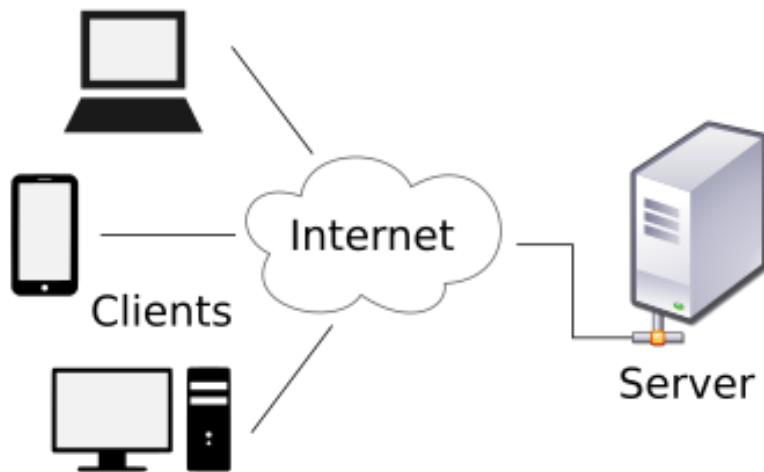
**Mô hình thương mại điện tử B2C (Business-to-Consumer):** Là mô hình doanh nghiệp cung cấp sản phẩm trực tiếp cho người tiêu dùng qua internet. Website thời trang có thể áp dụng mô hình này để bán các sản phẩm như quần áo, giày dép, phụ kiện cho khách hàng trực tuyến.

### 2.4. Khái niệm về mô hình Client-Server

**Client (Khách hàng):** Là các thiết bị hoặc phần mềm gửi yêu cầu (request) đến máy chủ để truy cập tài nguyên hoặc dịch vụ. Client có thể là máy tính cá nhân, điện thoại di động, hoặc một ứng dụng sử dụng các dịch vụ qua mạng (ví dụ: trình duyệt web, ứng dụng di động).

**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.

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



**Hình 0.1 Mô hình Client – Server**

Cấu trúc:

**1. 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.

<h1>Chào mừng đến với trang web của tôi</h1>

<p>Đây là một đoạn văn bản.</p>

<a href="https://www.example.com">Liên kết đến trang khác</a>

**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).

```
body {  
    font-family: Arial, sans-serif;  
    background-color: #f4f4f4;  
}
```

```
h1 {  
    color: #333;  
    text-align: center;  
}
```

**JavaScript** (nếu có): 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.

```
document.getElementById("myButton").addEventListener("click", function() {  
    alert("Bạn đã nhấn nút!");  
});
```

## 1. 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 0.2 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.

❖ **Đặc điểm của HTML:**

Đơn giản và dễ học: HTML có cú pháp đơn giản, dễ hiểu, phù hợp cho người mới bắt đầu.

Độc lập nền tảng: HTML có thể được sử dụng trên bất kỳ hệ điều hành nào và hiển thị trên mọi trình duyệt web.

Tính linh hoạt: HTML có thể kết hợp với CSS và JavaScript để tạo ra các trang web động và hấp dẫn.

❖ **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:

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Tiêu đề Trang</title>
</head>
<body>
    <h1>Tiêu đề Chính</h1>
    <p>Đây là một đoạn văn bản.</p>
    <a href="https://example.com">Đây là một liên kết</a>
    
</body>
</html>
```

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.

## Thiết kế website ứng dụng web bán hàng thời trang nữ

**<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>Shop Thời Trang Nữ</title>
  </head>
  <body>
    <h1>Chào Mừng Đến Với Shop Thời Trang Của Chúng Tôi</h1>
    <p>. </p>
    <h2>Sản Phẩm Nổi Bật</h2>
    <ul>
      <li><a href="https://vidu.com">
        Thoại</a></li>
      <ul>
        <li><a href="https://vidu.com/vay-dam">Váy Đầm</a></li>
        <li><a href="https://vidu.com/giay-nu">Giày Nữ</a></li>
        <li><a href="https://vidu.com/phu-kien-thoi-trang">Phụ Kiện Thời Trang</a></li>
      </ul> <h2>Liên Hệ</h2> <p>Địa chỉ: 123 Đường XYZ</p>
      <p>Email:<ahref="mailto:contact@vidu.com">contact@vidu.com</a></p>
      
    </ul>
  </body>
</html>
```

```
</body> </html>
```

## ❖ Các Phần Tử Phổ Biến Của Html

### 1. Tiêu đề(<heading>):

Heading được dùng để chỉ ra tiêu đề hoặc đầu mục của các phần nội dung trên trang web.

Có sáu cấp độ tiêu đề, từ <h1> đến <h6>, với <h1> là tiêu đề lớn nhất và <h6> là nhỏ nhất, Mỗi cấp có kích thước và độ quan trọng giảm dần, giúp xác định cấu trúc nội dung.

#### Cách hoạt động:

Trình tự hiển thị: <h1> lớn nhất và quan trọng nhất, <h2> đến <h6> lần lượt nhỏ dần và ít quan trọng hơn. Trình duyệt sẽ hiển thị các tiêu đề với các kích thước mặc định khác nhau, nhưng có thể tùy chỉnh lại bằng CSS.

Tạo cấu trúc nội dung: Heading được sắp xếp theo một cấu trúc phân cấp giúp người đọc dễ dàng hiểu chủ đề và phân chia của tài liệu

#### Giải thích chi tiết:

**<h1>:** Là cấp tiêu đề lớn nhất và quan trọng nhất, nên chỉ có một thẻ <h1> trên mỗi trang, sử dụng cho tiêu đề chính hoặc chủ đề của trang web.

**<h2>:** Được dùng cho các tiêu đề phụ của <h1>, như các mục nội dung lớn trong bài viết, sử dụng <h2> để chia nhỏ và định hướng nội dung , đặc biệt là trong các bài viết dài.

**<h3>:** Thường là tiêu đề phụ của <h2>, dùng để chia nhỏ nội dung trong các phần <h2>, thích hợp cho các tiêu đề nhỏ hơn hoặc tiêu đề của các đoạn giải thích chi tiết.

**<h4>,<h5> và <h6>:** Ít được sử dụng hơn, thường dùng khi cần thêm nhiều lớp phân cấp trong các nội dung rất chi tiết., các thẻ này chủ yếu là tiêu đề phụ của <h3> và thường có kích thước nhỏ, dùng cho các phần bổ sung hoặc tiêu đề nhỏ nhất.

## 2. Đoạn văn (<p>)

Phần tử <p> dùng để tạo một đoạn văn bản trong tài liệu HTML. Mỗi đoạn văn bản mới thường được đặt trong một phần tử <p>.

<p> là một phần tử khối (block element), do đó nó chiếm toàn bộ chiều rộng của khung chứa (container), và các đoạn văn được ngắt dòng tự động sau mỗi <p>.

### Cách hoạt động:

Tạo ngắt dòng tự động: Mỗi phần tử <p> tạo khoảng trống phía trên và phía dưới đoạn văn, giúp phân chia rõ ràng giữa các đoạn

Khả năng lồng ghép: <p> không nên chứa các phần tử khối khác như <div>, <h1>, nhưng có thể chứa các phần tử nội tuyến (inline) như <span>, <a>, hoặc các phần tử biểu tượng (emphasis) như <em>, <strong> hoặc <br>

## 3. Liên kết (<a>)

Phần tử <a> dùng để tạo các liên kết siêu văn bản (hyperlink) đến tài liệu khác hoặc vị trí khác trong cùng tài liệu.

Thuộc tính href: Là thuộc tính bắt buộc của <a> để xác định đường dẫn liên kết. Giá trị của href có thể là một URL đến một trang web bên ngoài, một tệp tài nguyên hoặc một đường dẫn nội bộ.

### Cách hoạt động:

Điều hướng: Khi người dùng nhấp vào liên kết, trình duyệt sẽ điều hướng đến địa chỉ URL được xác định trong thuộc tính href.

Ví dụ:

```
<p><a href="https://vidu.com">Sản phẩm mới</a></p>
```

## 4. Danh sách (<ol>,<ul> và <dl>)

Danh sách có thứ tự (`<ol>`): Được đánh số hoặc có thứ tự tăng dần, dùng khi nội dung có tính tuần tự hoặc yêu cầu thứ tự nhất định.

Danh sách không thứ tự (`<ul>`): Dùng các ký hiệu như dấu chấm hoặc hình tròn để liệt kê các mục không có thứ tự đặc biệt.

Danh sách mô tả (`<dl>`): Được dùng để liệt kê các mục cùng với mô tả của chúng. Đây là dạng danh sách thường dùng cho các từ vựng hoặc khái niệm có giải thích cụ thể

### Cách hoạt động:

Mỗi danh sách đều có một cấu trúc cụ thể, và các mục trong danh sách đều được định dạng sẵn theo cách khác nhau:

#### Danh sách có thứ tự (`<ol>`):

Cấu trúc: Phần tử `<ol>` bao quanh các mục danh sách `<li>`, các mục này sẽ tự động được đánh số hoặc sắp xếp theo thứ tự.

```
<ul>
    <li>Váy đầm</li>
    <li>Áo kiểu nữ</li>
    <li>Chân váy</li>
    <li>Set đồ</li>
    <li>Quần</li>
    <li>Phụ kiện</li>
</ul>
```

**Danh sách không thứ tự (<ul>):**

Cấu trúc: Phần tử <ul> bao quanh các mục danh sách <li>, mặc định sẽ hiển thị với ký hiệu dấu chấm.

```
<ul>
    <li>Váy đầm</li>
    <li>Áo kiểu nữ</li>
    <li>Chân váy</li>
    <li>Set đồ</li>
    <li>Quần</li>
    <li>Phụ kiện</li>
</ul>
```

**Danh sách mô tả (<dl>,<dt> và <dd>)**

<dl>: Phần tử bao quanh danh sách mô tả.

<dt>: Tên của một mục hoặc một thuật ngữ.

<dd>: Mô tả hoặc giải thích cho mục đó.

```
<dl>
    <dt>Phụ kiện khăn choàng</dt>
    <dd>Khăn choàng thời trang giúp giữ ấm và tạo điểm nhấn cho trang phục.</dd>

    <dt>Cao gót</dt>
    <dd>Giày cao gót thanh lịch, phù hợp cho các sự kiện hoặc công sở.</dd>

    <dt>Đây chuyền</dt>
    <dd>Trang sức dây chuyền với thiết kế tinh tế, đa dạng phong cách.</dd>

    <dt>Thắt lưng</dt>
    <dd>Phụ kiện thắt lưng thời trang, làm nổi bật vòng eo và phong cách.</dd>
</dl>
```

## 5.Phân chia (<div>)

Mục đích chính: Dùng để chia trang thành các vùng riêng biệt, chứa các phần tử khác như văn bản, hình ảnh, liên kết, hoặc các phần tử HTML khác.

Không có định dạng sẵn: <div> không có kiểu dáng hoặc định dạng mặc định và thường chỉ có vai trò cấu trúc để có phong cách cụ thể, cần kết hợp với CSS.

### Cách hoạt động:

Khi được áp dụng CSS, <div> sẽ hoạt động như một hộp chứa cho các phần tử bên trong và có thể được định dạng về kích thước, màu sắc, bố cục, và khoảng cách.

**Dạng khôi (block):** <div> là phần tử dạng khôi, chiếm toàn bộ chiều rộng của vùng chứa và thường xuống dòng. Vì thế, <div> thường được dùng làm khôi chính hoặc bao ngoài cho các nội dung khác.

**CSS Box Model:** <div> có thể được kiểm soát thông qua Box Model của CSS với các thuộc tính như margin, padding, border, và width, height.

**Tổ chức nội dung:** <div> thường bao quanh các phần nội dung riêng biệt như header, footer, sidebar, giúp cấu trúc hóa trang HTML và dễ dàng quản lý hơn

### Các thuộc tính:

**Id:** Xác định một mã định danh duy nhất cho một <div> để truy cập và áp dụng CSS hoặc JavaScript.

**Class:** Cho phép sử dụng CSS để định dạng nhiều phần tử <div> có cùng kiểu dáng

Ví dụ:

```
<div id="content">
    <div class="box">
        <h3>Phụ kiện khăn choàng</h3>
        <p>Khăn choàng thời trang, đa dạng phong cách.</p>
    </div>
    <div class="box">
        <h3>Cao gót</h3>
        <p>Giày cao gót thanh lịch, phù hợp cho mọi sự kiện.</p>
    </div>
    <div class="box">
        <h3>Dây chuyền</h3>
        <p>Trang sức dây chuyền sang trọng và tinh tế.</p>
    </div>
    <div class="box">
        <h3>Thắt lưng</h3>
        <p>Thắt lưng thời trang, hoàn thiện mọi bộ trang phục.</p>
    </div>
</div>
```



### 6.Biểu mẫu (form)

**<form>** là phần tử HTML chứa các trường nhập liệu và các nút hành động để người dùng có thể điền thông tin và gửi dữ liệu.

**Phương thức gửi dữ liệu:** Form có thể gửi dữ liệu đến máy chủ bằng các phương thức HTTP như GET và POST.

**Action:** URL mà dữ liệu biểu mẫu sẽ được gửi đến để xử lý khi người dùng nhấn gửi.

#### Cấu trúc:

Phần tử **<form>** có các thuộc tính quan trọng như:

**action:** Xác định URL mà dữ liệu sẽ được gửi đến khi người dùng nhấn gửi.

**method:** Xác định phương thức gửi dữ liệu (GET hoặc POST)

Ví dụ:

```
<form action="/submit" method="post">
  <label for="name">Tên:</label>
  <input type="text" id="name" name="name">
  <button type="submit">Gửi</button>
</form>
```

Ví dụ về form trong HTML

### Các thành phần trong form

**<input>**: Trường nhập liệu đa năng

type="text": Cho phép nhập văn bản đơn giản.

type="password": Che đi các ký tự đã nhập, phù hợp với trường mật khẩu.

type="email": Chỉ chấp nhận định dạng email (có ký hiệu @ và dấu chấm).

type="number": Chỉ chấp nhận số, có thể giới hạn giá trị.

type="checkbox": Tạo ô đánh dấu cho phép lựa chọn nhiều mục.

type="radio": Tạo các lựa chọn đơn.

type="submit": Tạo nút gửi biểu mẫu.

type="file": Cho phép tải lên file

**<label>**: Nhãn cho các trường nhập liệu, gắn với phần tử nhập liệu qua thuộc tính for giúp cải thiện khả năng sử dụng và truy cập

**<textarea>**: Trường nhập văn bản nhiều dòng, cho phép nhập văn bản dài hơn và có thể điều chỉnh kích thước.

**<select>** và **<option>**: Danh sách lựa chọn (dropdown) dùng để tạo danh sách các mục mà người dùng có thể chọn

**<button>**: Nút hành động có thể kích hoạt JavaScript hoặc gửi form

### Các thuộc tính quan trọng của form

required: Yêu cầu người dùng phải điền dữ liệu vào trường trước khi gửi.

placeholder: Văn bản gợi ý cho người dùng nhập thông tin vào trường.

maxlength: Giới hạn số ký tự người dùng có thể nhập vào trường.

pattern: Định dạng mẫu (regex) mà giá trị nhập phải tuân theo.

autocomplete: Cho phép tự động hoàn thành dữ liệu trên các trường nhập

### 7. Hình ảnh (<img>)

Thẻ <img> là một thẻ tự đóng (self-closing tag), nghĩa là không cần có thẻ đóng, ví dụ:

```

```

#### Các thuộc tính của <img>

**src**: đường dẫn đến hình ảnh, có thể là đường dẫn tương đối (relative path) hoặc đường dẫn tuyệt đối (absolute path).

**alt**: văn bản thay thế mô tả nội dung của hình ảnh. Khi hình ảnh không thể hiển thị, thuộc tính alt sẽ cung cấp thông tin về hình ảnh đó

**width** và **height**: đặt chiều rộng và chiều cao cho hình ảnh. Có thể được xác định bằng pixel hoặc phần trăm so với vùng chứa. Điều chỉnh kích thước giúp trang web tải nhanh hơn, nhất là khi hình ảnh lớn được thu nhỏ để phù hợp với giao diện

**title**: thêm tiêu đề hiển thị khi người dùng di chuột lên hình ảnh, giúp cung cấp thêm thông tin cho người dùng

## 1. 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 0.3 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.

❖ **Đặc điểm của CSS:**

Tiết kiệm thời gian và công sức: CSS cho phép áp dụng cùng một kiểu định dạng cho nhiều trang HTML, dễ dàng thay đổi toàn bộ giao diện trang web chỉ bằng một file CSS.

Kiểm soát bố cục tốt hơn: CSS cung cấp các công cụ định dạng linh hoạt như flexbox, grid, và khả năng điều chỉnh khoảng cách giúp tạo bố cục phức tạp. CSS cho phép thiết kế giao diện linh hoạt, phù hợp trên mọi kích thước màn hình.

Tốc độ tải trang: CSS được lưu trong bộ nhớ cache của trình duyệt, giúp trang tải nhanh hơn.

❖ **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;  
}
```

Ứng dụng: Sử dụng khi muốn định dạng đồng nhất cho tất cả các phần tử cùng loại mà không cần tạo thêm lớp.

#### 2. 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;  
}
```

### 3. 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; }
```

### 4. 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;  
}
```

### 5. 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;  
}
```

## 6. 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;  
}
```

## 7. 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;  
}
```

## 8. 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;
```

}

## 9. 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.

## 10. 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;  
}
```

## 11. 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;  
}
```

## 12.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;  
}
```

## 12. 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;  
}
```

## 13. 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 trực chính hoặc trực phụ

## **Thuộc tính quan trọng**

### **Trên container (cha):**

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

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

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

**align-items:** Căn chỉnh các phần tử theo trực 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.

## Thiết kế website ứng dụng web bán hàng thời trang nữ

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;  
}
```

## 14.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:

```
font-family: Arial, Helvetica, sans-serif;
```

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;
```

## 15.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;  
}
```

## 16.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;
```

## 17.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;
```

## 18.Letter Spacing (Khoảng cách giữa các ký tự)

letter-spacing tạo khoảng cách giữa các ký tự trong từ.

Cú pháp:

```
letter-spacing: 2px;
```

## 19.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;
```

## 20.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;
```

## 21. 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;**

## 22. Color (Màu chū)

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

## Cú pháp:

## 23. 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.

## Cú pháp:

```
background-color: #ff0000  
| | | | |  
| | | | |  
| | | | |  
| | | | |
```

#ff0000  
rgb(255, 0, 0)  
hsl(0, 100%, 50%)

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

Cú pháp:

```
background-image: url('Anhnen.jpg');
```

**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.

Cú pháp:

```
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)

Cú pháp:

```
background-size: cover;
```

## 2. Javascript

## Thiết kế website ứng dụng web bán hàng thời trang nữ

---

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 0.4 Javascript**

❖ **Đặc điểm của Javascript**

Ngôn ngữ lập trình hướng đối tượng: Hỗ trợ các khái niệm như đối tượng và kế thừa.

Động và không kiểu tĩnh: Không yêu cầu khai báo kiểu dữ liệu trước khi sử dụng.

Chạy trên trình duyệt: Cho phép phát triển ứng dụng web mà không cần cài đặt phần mềm bổ sung.

Cú pháp đơn giản: Dễ học và dễ hiểu, tương tự như cú pháp của các ngôn ngữ lập trình khác như C hay Java.

❖ **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.

**i. 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 ( $\geq 576\text{px}$ )

.col-md-\*: Áp dụng cho màn hình vừa ( $\geq 768\text{px}$ )

.col-lg-\*: Áp dụng cho màn hình lớn ( $\geq 992\text{px}$ )

.col-xl-\*: Áp dụng cho màn hình rất lớn ( $\geq 1200\text{px}$ )

.col-xxl-\*: Áp dụng cho màn hình cực lớn ( $\geq 1400\text{px}$ )

```
<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

### 3. 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 0.5 PHP**

❖ **Đặc điểm của PHP:**

Dễ học và dễ sử dụng: PHP có cú pháp đơn giản, dễ tiếp cận đối với các lập trình viên mới bắt đầu và mạnh mẽ với các tính năng nâng cao.

Mã nguồn mở (Open-source): PHP là một ngôn ngữ mã nguồn mở, điều này có nghĩa là bạn có thể sử dụng, sửa đổi và phân phối nó miễn phí.

Tương thích với nhiều hệ điều hành: PHP có thể chạy trên nhiều hệ điều hành khác nhau như Linux, Windows.

## Thiết kế website ứng dụng web bán hàng thời trang nữ

Hỗ trợ nhiều cơ sở dữ liệu: PHP hỗ trợ kết nối với nhiều hệ quản trị cơ sở dữ liệu như MySQL, PostgreSQL, SQLite, v.v.

Tính năng mạnh mẽ: PHP cung cấp nhiều thư viện và framework hỗ trợ các tác vụ như gửi email, xử lý file, bảo mật, xác thực người dùng, và tương tác với API.

### ❖ 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.

#### 2. Biến

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

```
<?php  
    $TenSanPham = "Váy đầm ";  
    $Gia = 1500000;  
    echo "Sản phẩm: $TenSanPham, Giá: $Gia VND";
```

```
?>
```

### 3. Hàng số

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

```
<?php  
  
    define("ThuongHieu", "Phụ kiện Quần Áo");  
  
    const MaSanPham = 123;  
  
    echo "Thương hiệu: " . ThuongHieu . ", Mã Sản Phẩm: " .  
MaSanPham. " Số ";  
  
?>
```

### 4. 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 = "váy đầm";
```

```
switch ($SDanhMuc) {  
    case "phụ kiện quần áo":  
        echo "Sản phẩm này là phụ kiện quần áo cao  
cấp .";  
        break;  
    case "váy đầm":  
        echo "Sản phẩm này váy đầm. ";  
        break;  
    default:  
        echo "Loại phụ kiện chưa được cập nhật.";  
    }  
?>
```

## 5. Vòng lặp

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

```
<?php  
for ($i = 1; $i <= 5; $i++) {  
    echo "Sản phẩm phụ kiện quần áo #$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  
$SanPham = ["váy đầm", "quần", "set đồ", "quần", "áo kiểu  
nữ", "phụ kiện";  
foreach ($SanPham as $item) {  
    echo "Sản phẩm: $item<br>";  
}  
?>
```

## 6. Hàm

Hàm được định nghĩa bằng từ khóa function.

```
<?php  
function getAccessoryInfo($TenSanPham, $Gia) {  
    return "Phụ kiện: $ TenSanPham, Giá: $ Gia VND";  
}
```

```
echo getAccessoryInfo("váy đầm", 200000);  
?>
```

## 7. Mảng

Mảng chỉ số:

```
<?php  
  
$SanPham = ["", "", ""];  
  
echo "Sản phẩm phổ biến: " . $SanPham[0];  
  
?>
```

Mảng kết hợp

```
<?php  
  
$SanPham = ["TenSanPham" => "", "" => 500000];  
  
echo "Sản phẩm: " . $SanPham["TenSanPham"] . ", Giá: " .  
$SanPham["Gia"] . " VND";  
  
?>
```

## 4. 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 0.6 MySQL

❖ **Đặc điểm của MySQL:**

Mã nguồn mở: MySQL là một hệ quản trị cơ sở dữ liệu mã nguồn mở và miễn phí, có thể sử dụng, sửa đổi và phân phối lại.

Hiệu suất cao: MySQL nổi bật với khả năng xử lý các truy vấn và thao tác với dữ liệu một cách nhanh chóng và hiệu quả.

Khả năng mở rộng: MySQL có thể hoạt động hiệu quả từ các ứng dụng nhỏ đến các hệ thống web quy mô lớn.

Hỗ trợ đa nền tảng: MySQL có thể chạy trên các hệ điều hành khác nhau như Linux, Windows, macOS, v.v

❖ **Cách hoạt động của MySQL:**

Cấu trúc cơ sở dữ liệu: MySQL lưu trữ dữ liệu trong các bảng (tables), mỗi bảng gồm nhiều cột (columns) và dòng (rows). Các bảng được liên kết với nhau qua khóa ngoại (foreign keys).

Truy vấn SQL: Người dùng tương tác với MySQL thông qua các truy vấn SQL để thực hiện thao tác như thêm, sửa, xóa hoặc lấy dữ liệu từ các bảng.

❖ **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** để tạo một cơ sở dữ liệu mới.

Cú pháp: **CREATE DATABASE database\_name;**

```
CREATE DATABASE ban_hang;
```

**2. 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 ban_hang;
```

**3. 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,  
);
```

#### 4. 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, ...);
```

#### 5. 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;
```

#### 6. 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;
```

#### 7. 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= 'váy đầm';
```

#### 8. Xóa bảng

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

Cú pháp: **DROP TABLE table\_name;**

**DROP TABLE SanPham;**

## ii. 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 có thể sử dụng các câu lệnh SQL để tương tác với cơ sở dữ liệu MySQL (ví dụ: SELECT, INSERT, UPDATE, DELETE) và trả về kết quả cho người dùng.

### **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 sẽ truy vấn MySQL để lấy dữ liệu và hiển thị nó trên giao diện người dùng.

PHP sẽ truy vấn MySQL để lấy dữ liệu và hiển thị nó trên giao diện người dùng.

## CHƯƠNG 3 HIỆN THỰC HÓA NGHIÊN CỨU

### 3.1. Mô tả bài toán

Bài toán này mô tả việc xây dựng một **hệ thống quản lý sản phẩm thời trang** cho một cửa hàng hoặc một nền tảng thương mại điện tử. Mục tiêu chính của hệ thống là quản lý các sản phẩm thời trang (quần áo, giày dép, phụ kiện) từ khi được nhập kho cho đến khi bán cho khách hàng, bao gồm các tính năng như tìm kiếm, lọc sản phẩm, giỏ hàng và thanh toán.

Hệ thống cần có khả năng quản lý thông tin chi tiết về các sản phẩm, cho phép khách hàng tìm kiếm và lựa chọn sản phẩm dễ dàng, cũng như cung cấp một giao diện thuận tiện để thực hiện các giao dịch mua bán trực tuyến.

### 3.2. Phân tích thiết kế hệ thống

#### 3.2.1. Yêu cầu chức năng

##### ❖ Quản lý 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 lý người dùng:

Đăng ký / đăng nhập và đăng xuất 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 lý danh mục sản phẩm:

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

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

❖ **Xử lý đơn hàng:**

Tạo đơn hàng khi người dùng thêm sản phẩm vào giỏ hàng và xác nhận mua.

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.

### 3.2.2. 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:**

Dữ liệu người dùng (mật khẩu, thông tin cá nhân) phải được mã hóa an toàn.

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:**

# Thiết kế website ứng dụng web bán hàng thời trang nữ

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:**

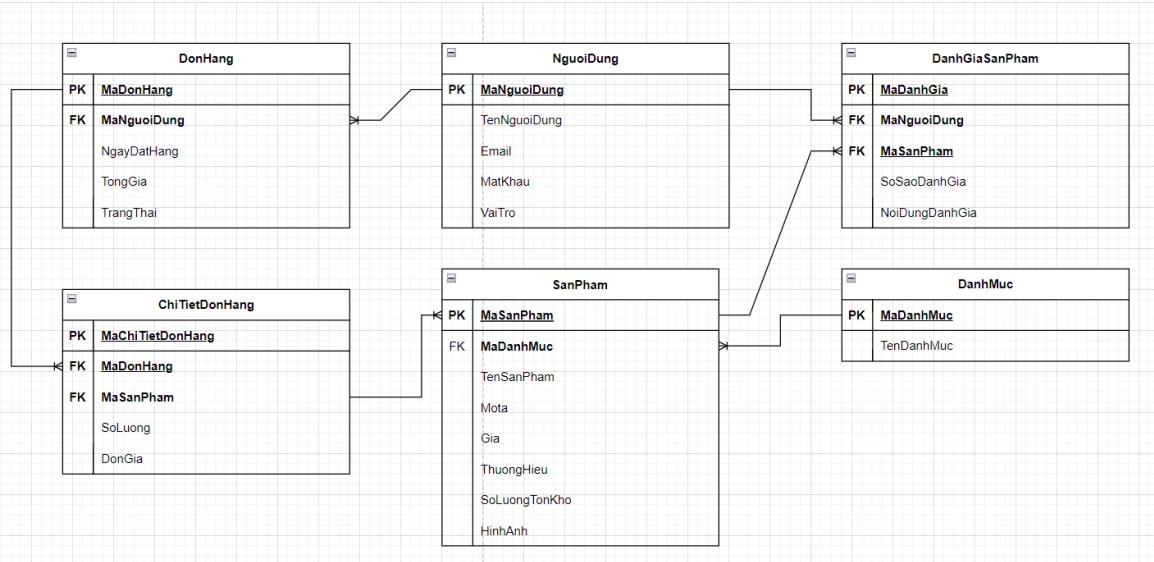
Hỗ trợ trình duyệt phổ biến như Chrome, Firefox, Safari và Edge.

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

## 3.3. Kiến trúc hệ thống

### 3.3.1. Thiết kế dữ liệu

#### 1. Mô hình ERD



**Hình 0.1 Mô hình ERD**

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

STT	Tên thực thể/mối kết hợp	Điễn 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

Bảng 0.1 Danh sách các thực thể của mô hình ERD

3. 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:

STT	Tên thuộc tính	Điễn giải	Kiểu dữ liệu	Ràng buộc toàn vẹn
1	MaNguoiDung	Mã xác định người dùng	int	PK
2	TenNguoiDung	Tên của người dùng	Varchar(100)	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	Varchar(20)	Không

Bảng 0.2 Bảng mô tả chi tiết thực thể NguoiDung

❖ **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:

STT	Tên thuộc tính	Điễn giải	Kiểu dữ liệu	Ràng buộc toàn vẹn
1	MaDonHang	Mã đơn hàng	Varchar(20)	PK
2	MaNguoiDung	Mã xác định người dùng	Int	FK
3	NgayDatHang	Ngày đặt hàng	Date	Không
4	TongGia	Tổng giá đơn hàng	Decimal(10,2)	Không
5	TrangThai	Trạng thái đơn hàng	Varchar(50)	Không

Bảng 0.3 Bảng mô tả chi tiết thực thể DonHang

❖ **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:

<i>STT</i>	<i>Tên thuộc tính</i>	<i>Điễn giải</i>	<i>Kiểu dữ liệu</i>	<i>Ràng buộc toàn vẹn</i>
1	MaChiTietDonHang	Mã chi tiết đơn hàng	Varchar(20)	PK
2	MaDonHang	Mã đơn hàng	Varchar(20)	FK
3	MaSanPham	Mã sản phẩm	Varchar(20)	FK
4	SoLuong	Số lượng sản phẩm	int	Không
5	DonGia	Giá của sản phẩm	Decimal(10,2)	Không

Bảng 0.4 Bảng mô tả chi tiết thực thể ChiTietDonHang

❖ **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:

<i>STT</i>	<i>Tên thuộc tính</i>	<i>Điễn giải</i>	<i>Kiểu dữ liệu</i>	<i>Ràng buộc toàn vẹn</i>
1	MaSanPham	Mã sản phẩm	Varchar(20)	PK
2	MaDanhMuc	Mã danh mục	Varchar(20)	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(100)	Không
7	SoLuongTonKho	Sản phẩm trong kho	Int	Không
8	HinhAnh	Hình ảnh sản phẩm	Varchar(255)	Không

Bảng 0.5 Bảng mô tả chi tiết thực thể SanPham

<i>STT</i>	<i>Tên thuộc tính</i>	<i>Điễn giải</i>	<i>Kiểu dữ liệu</i>	<i>Ràng buộc toàn vẹn</i>
1	MaSanPham	Mã sản phẩm	Varchar(20)	PK
2	MaDanhMuc	Mã danh mục	Varchar(20)	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(100)	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:

<i>STT</i>	<i>Tên thuộc tính</i>	<i>Điễn giải</i>	<i>Kiểu dữ liệu</i>	<i>Ràng buộc toàn vẹn</i>
	MaDanhGia	Mã đánh giá sản phẩm	Varchar(20)	PK
	MaNguoiDung	Mã xác định người dùng	Varchar(20)	FK
	MaSanPham	Mã Sản Phẩm	Varchar(20)	FK
	SoSaoDanhGia	Đánh giá (từ 1-5 *)	Int	Không
	NoiDungDanhGia	Nội dung đánh giá	Text	Không

Bảng 0.6 Bảng mô tả chi tiết thực thể DanhGiaSanPham

❖ **Bảng mô tả thực thể DanhMuc**

## Thiết kế website ứng dụng web bán hàng thời trang nữ

Tên thực thể: DanhMuc

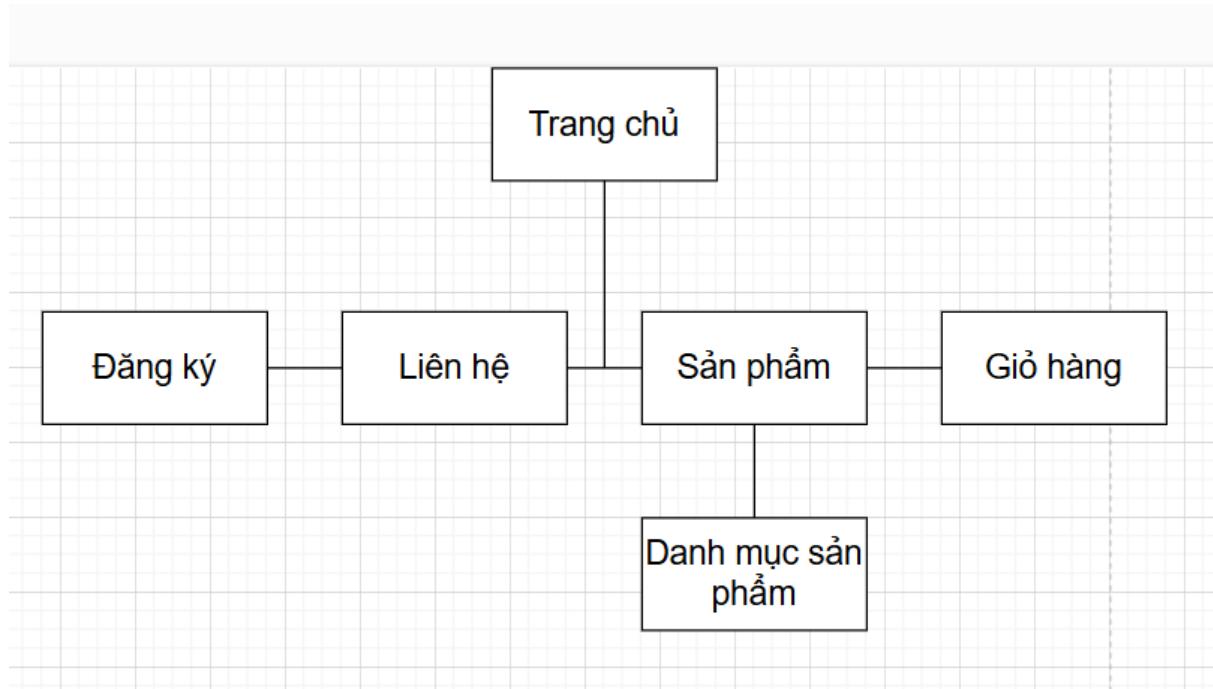
Mô tả: Lưu trữ thông tin về danh mục của sản phẩm có trên web

Chi tiết thực thể:

STT	Tên thuộc tính	Điễn giải	Kiểu dữ liệu	Ràng buộc toàn vẹn
1	MaDanhMuc	Mã danh mục	Varchar(20)	PK
2	TenDanhMuc	Tên danh mục	Varchar(100)	

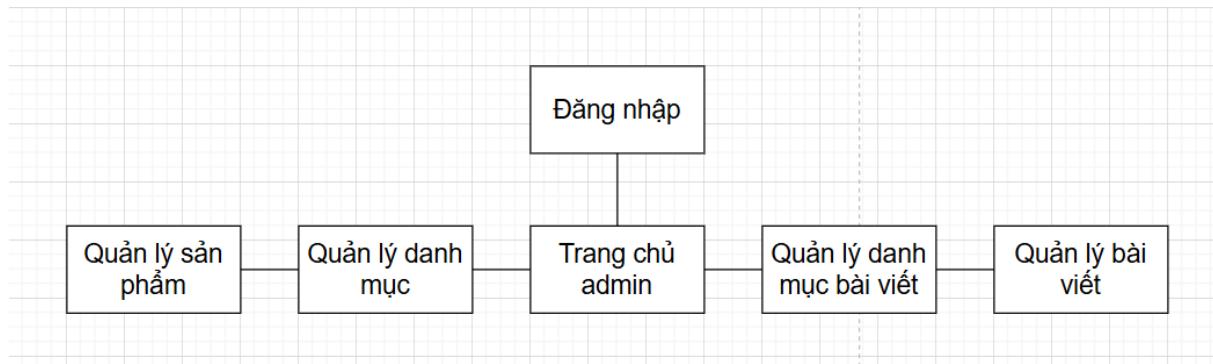
### 3.3.2. Thiết kế giao diện

#### 3.3.2.1. Sơ đồ website



Hình 0.2 Sơ đồ website

#### 3.3.2.2. Giao diện trang chủ



## CHƯƠNG 4 KẾT QUẢ NGHIÊN CỨU

### 4.1. Bảng cơ sở dữ liệu

#### Bảng admin

Extra options						
	← T →	▼ id	ten_dang_nhap	mat_khau	ho_ten	email
<input type="checkbox"/>	Sửa  Chép  Xóa bỏ	1	test	\$2y\$10\$eC2vAmSaU8YnTM2UNrrE2.8pACDo2HK70JtDmT2tKz6...	test	test@gmail.com
<input type="checkbox"/>	Sửa  Chép  Xóa bỏ	2	test2	\$2y\$10\$KST8fQtuz60iHtqO/2d7oeWSffk0mwyKXC6vESL7oiP...	test2	chauthimyhuong27@gmail.com
<input type="checkbox"/>	Sửa  Chép  Xóa bỏ	3	myhuong	\$2y\$10\$.tO1n11TLBLsqnzBrJ.yUeyclDLvx6LMCr.FeiaH0HY...	myhuong	hmy611916@gmail.com
<input type="checkbox"/>	Sửa  Chép  Xóa bỏ	4	12345	\$2y\$10\$7Jl.m1LaNuDB4yUn/5z83.CRJmNe0stE5DCNIMdxZqK...	hien	sefrasf@gmail.com

#### Bảng danh mục

Extra options						
	← T →	▼ id	ten_danh_muc			
<input type="checkbox"/>	Sửa  Chép  Xóa bỏ	1	Váy Đầm			
<input type="checkbox"/>	Sửa  Chép  Xóa bỏ	2	Áo Kiểu Nữ			
<input type="checkbox"/>	Sửa  Chép  Xóa bỏ	3	Chân Váy			
<input type="checkbox"/>	Sửa  Chép  Xóa bỏ	4	Set Đồ			
<input type="checkbox"/>	Sửa  Chép  Xóa bỏ	5	Quần			
<input type="checkbox"/>	Sửa  Chép  Xóa bỏ	6	Phụ Kiện			

#### Bảng sản phẩm

# Thiết kế website ứng dụng web bán hàng thời trang nữ

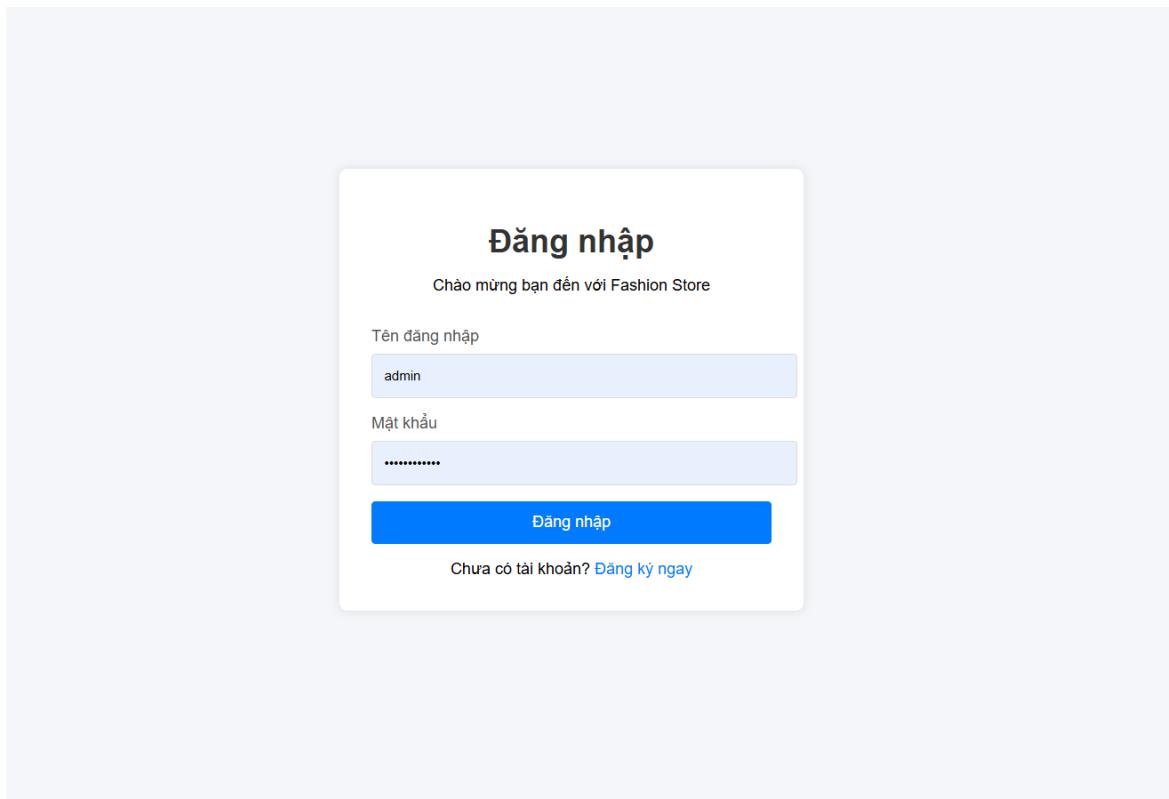
	Sửa	Chép	Xóa bỏ	ma_san_pham	ten_san_pham	id	gia_ban	so_luong	hinh_anh
1	Sửa	Chép	Xóa bỏ	1	Váy Đầm Dự Tiệc Xòe Tay Phòng	1	890000.00	10	images/image/h7.jpg
2	Sửa	Chép	Xóa bỏ	10	Chân Váy Bút Chì Công Sở	3	350000.00	10	images/image/h17.jpg
3	Sửa	Chép	Xóa bỏ	11	Set Đồ Công Sở Cao Cấp	4	890000.00	10	images/image/h18.jpg
4	Sửa	Chép	Xóa bỏ	12	Set Đồ Casual Năng Động	4	750000.00	10	images/image/h22.jpg
5	Sửa	Chép	Xóa bỏ	13	Quần Tây Nữ Công Sở Ông Suông	5	450000.00	10	images/image/h24.jpg
6	Sửa	Chép	Xóa bỏ	14	Quần Culottes Nữ Công Sở	5	420000.00	10	images/image/h25.jpg
7	Sửa	Chép	Xóa bỏ	15	Túi Xách Nữ Công Sở Cao Cấp	6	790000.00	10	images/image/h26.jpg
8	Sửa	Chép	Xóa bỏ	16	Dây Chuyền Bạc Nữ Mắt Trái Tim	6	290000.00	10	images/image/h27.jpg
9	Sửa	Chép	Xóa bỏ	17	Thắt Lưng Nữ Da Cao Cấp	6	350000.00	10	images/image/h28.jpg
10	Sửa	Chép	Xóa bỏ	18	Khăn Choàng Cổ Lụa Cao Cấp	6	250000.00	10	images/image/h30.jpg
11	Sửa	Chép	Xóa bỏ	19	Giày Cao Gót Nữ Mũi Nhọn	6	550000.00	10	images/image/h31.jpg
12	Sửa	Chép	Xóa bỏ	2	Váy Đầm Công Sở Thanh Lịch	1	750000.00	10	images/image/h6.jpg
13	Sửa	Chép	Xóa bỏ	3	Váy Đầm Maxi Đi Biển	1	650000.00	10	images/image/h9.jpg
14	Sửa	Chép	Xóa bỏ	4	Váy Đầm Hoa Nhí Vintage	1	550000.00	10	images/image/h11.jpg
15	Sửa	Chép	Xóa bỏ	5	Áo Sơ Mi Nữ Công Sở	2	420000.00	10	images/image/h10.jpg
16	Sửa	Chép	Xóa bỏ	6	Áo Kiểu Nữ Cổ V Tay Bồng	2	420000.00	10	images/image/h12.jpg
17	Sửa	Chép	Xóa bỏ	7	Áo Blouse Nữ Cổ Nơ	2	450000.00	10	images/image/h13.jpg
18	Sửa	Chép	Xóa bỏ	8	Áo Kiểu Dự Tiệc Sang Trọng	2	520000.00	10	images/image/h21.jpg
19	Sửa	Chép	Xóa bỏ	9	Chân Váy Xếp Ly Tennis	3	320000.00	10	images/image/h16.jpg

## 4.2. Kết quả thực nghiệm

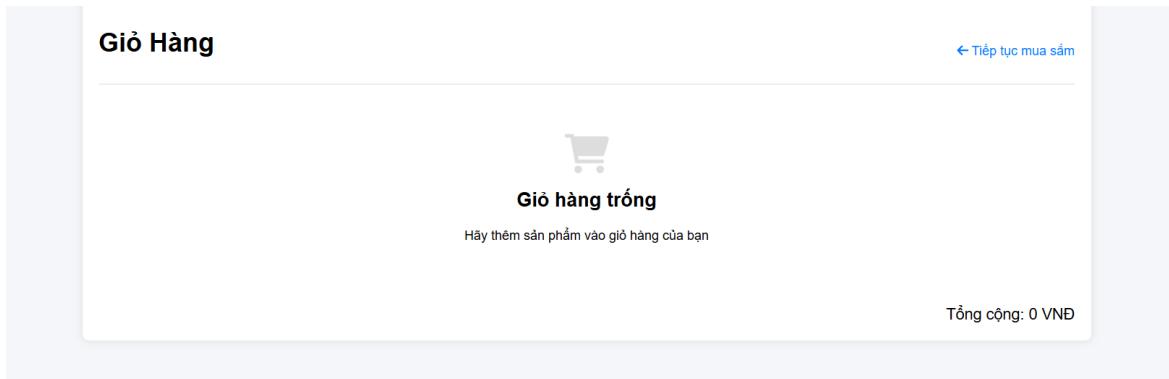
### 4.2.1. Chức năng tra cứu

The screenshot shows a responsive website layout for 'Fashion Store'. At the top, there's a purple header with the store name 'Fashion Store' and a subtext 'Nơi Phong Cách Được Định Nghĩa'. Below the header is a navigation bar with links: Trang Chủ, Sản Phẩm, Giới Thiệu, and Liên Hệ. A search bar is positioned above a main content area. The main content area features a grid of five fashion products: a white lace dress, a black lace dress, a pink lace dress, a blue floral skirt, and a green sequined dress. Below the grid, there are two rows of smaller images showing different outfit combinations.

#### 4.2.2. Chức năng đăng nhập đăng ký



#### 4.2.3. Chức năng giỏ hàng



## CHƯƠNG 5 KẾT LUẬN

### 5.1. Kết luận

Website bán hàng thời trang nữ đã được xây dựng và hoàn thiện với đầy đủ các tính năng cần thiết.

- Giao diện người dùng thân thiện:** Thiết kế hiện đại, phù hợp với thẩm mỹ của đối tượng khách hàng nữ.
- Chức năng quản lý sản phẩm hiệu quả:** Hỗ trợ quản trị viên dễ dàng thêm, sửa, xóa sản phẩm và quản lý danh mục sản phẩm.
- Hỗ trợ mua sắm trực tuyến:** Tích hợp gio hàng, thanh toán trực tuyến, và thông báo đơn hàng giúp khách hàng có trải nghiệm mua sắm tiện lợi.
- Nội dung phong phú:** Cung cấp bài viết liên quan đến xu hướng thời trang, mẹo phối đồ và các chương trình khuyến mãi.
- Bảo mật cao:** Đảm bảo thông tin khách hàng và dữ liệu giao dịch được bảo vệ an toàn.

Với các tính năng hiện có, website không chỉ hỗ trợ bán hàng mà còn nâng cao giá trị thương hiệu, giúp doanh nghiệp tiếp cận đối tượng khách hàng mục tiêu một cách hiệu quả.

### 5.2 Hướng phát triển

Để đáp ứng nhu cầu ngày càng cao của khách hàng và bắt kịp xu hướng công nghệ, website cần được mở rộng và nâng cấp theo các hướng sau:

#### Phát triển ứng dụng di động:

Xây dựng ứng dụng di động trên nền tảng iOS và Android để tăng trải nghiệm người dùng.

Tích hợp các tính năng đặc biệt như thông báo đẩy về khuyến mãi, sản phẩm mới.

#### Tối ưu hóa SEO:

Tăng cường chiến lược SEO để website đạt thứ hạng cao trên các công cụ tìm kiếm.

Đầu tư vào nội dung chất lượng và từ khóa liên quan đến thời trang nữ.

### **Tích hợp trí tuệ nhân tạo (AI):**

Sử dụng AI để gợi ý sản phẩm phù hợp dựa trên lịch sử mua sắm và sở thích của khách hàng.

### **Mở rộng hệ thống thanh toán:**

Hỗ trợ nhiều phương thức thanh toán hơn (ví điện tử, trả góp).

Đảm bảo tích hợp thanh toán quốc tế để phục vụ khách hàng ngoài nước.

### **Tăng cường trải nghiệm thực tế ảo (AR):**

Phát triển tính năng thử đồ ảo, cho phép khách hàng xem trước sản phẩm khi mặc.

Áp dụng AR trong các chiến dịch quảng cáo.

### **Phân tích dữ liệu khách hàng:**

Tích hợp công cụ phân tích dữ liệu để hiểu rõ hành vi người dùng.

Dựa trên dữ liệu để cá nhân hóa chiến dịch marketing.

### **Mở rộng danh mục sản phẩm:**

Bổ sung thêm các mặt hàng thời trang cao cấp, hoặc sản phẩm cho các đối tượng khác như mẹ và bé.

Hợp tác với các nhà thiết kế để ra mắt bộ sưu tập độc quyền.

## **5.3 Kết luận cuối cùng**

Dự án website bán hàng thời trang nữ không chỉ đáp ứng nhu cầu mua sắm trực tuyến mà còn đóng góp quan trọng vào việc phát triển thương hiệu trong thị trường cạnh tranh. Việc tiếp tục đầu tư vào cải tiến và phát triển các tính năng mới sẽ giúp website duy trì lợi thế và thu hút khách hàng lâu dài.

## 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.