

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



THỰC TẬP ĐỒ ÁN CHUYÊN NGÀNH
HỌC KỲ I, NĂM HỌC 2024-2025

ỨNG DỤNG ASP.NET CORE MVC XÂY DỰNG WEBSITE BÁN GIÀY

Giáo viên hướng dẫn:

Đoàn Phước Miên

Sinh viên thực hiện:

Lâm Ngọc Triệu

110121067

DA21TTA

Trà Vinh, 11 tháng 11 năm 2024

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



THỰC TẬP ĐỒ ÁN CHUYÊN NGÀNH
HỌC KỲ I, NĂM HỌC 2024-2025

ỨNG DỤNG ASP.NET CORE MVC XÂY DỰNG WEBSITE BÁN GIÀY

Giáo viên hướng dẫn:
Đoàn Phước Miền

Sinh viên thực hiện:
Lâm Ngọc Triệu
110121067
DA21TTA

Trà Vinh, 11 tháng 11 năm 2024

This image shows a full page of primary-ruled paper. It features multiple horizontal rows of small, evenly spaced dots, designed to guide handwriting practice. The dots are arranged in straight lines across the width of the page, with significant white space between each row. There are no margins, text, or other markings on the paper.

Giáo viên hướng dẫn
(Ký tên và ghi rõ họ tên)

This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the page.

Thành viên hội đồng
(Ký tên và ghi rõ họ tên)

LỜI CẢM ƠN

Em xin bày tỏ lòng biết ơn sâu sắc đến các quý thầy cô Khoa Kỹ thuật và Công nghệ, Trường Đại học Trà Vinh, đặc biệt là thầy Đoàn Phước Miền vì đã không ngừng hỗ trợ và chỉ dẫn em. Sự hướng dẫn tận tâm của thầy đã giúp em có cơ hội học hỏi, áp dụng những kiến thức vào thực tiễn, và giải quyết những vấn đề gặp phải trong suốt quá trình học.

Em xin cảm ơn thầy Đoàn Phước Miền vì sự hỗ trợ và hướng dẫn tận tình trong suốt quá trình thực hiện đồ án. Em nhận thức rằng trong quá trình thực hiện, bài báo cáo của em còn có những thiếu sót và em hy vọng nhận được góp ý từ quý thầy cô để hoàn thiện hơn. Điều này không chỉ giúp em cải thiện bản thân mà còn chuẩn bị tốt hơn cho các bài báo cáo trong tương lai.

Em xin gửi lời cảm ơn chân thành nhất đến thầy cô và hy vọng sẽ tiếp tục nhận được sự hướng dẫn và hỗ trợ trong hành trình học vấn của mình.

Trà Vinh , ngày tháng năm 2024

Lâm Ngọc Triệu

MỤC LỤC

<i>CHƯƠNG 1: TỔNG QUAN</i>	<i>1</i>
1.1. Đặt vấn đề	1
1.2. Mục đích nghiên cứu.....	1
<i>CHƯƠNG 2: CƠ SỞ LÝ THUYẾT</i>	<i>2</i>
2.1. HTML (Hyper Text Markup Language)	2
2.1.1. Giới thiệu.....	2
2.1.2. Tìm hiểu về "Hypertext" và liên kết (Links)	2
2.1.3. Lịch sử phát triển	2
2.1.4. Cách hoạt động của HTML	3
2.1.5. Cấu trúc của một phần tử HTML.....	3
2.1.6. Kết luận	3
2.2. CSS (Cascading Style Sheets)	4
2.2.1. Tổng quan về CSS?	4
2.2.2. Lịch sử hình thành và phát triển của CSS	4
2.2.3. Các phiên bản của CSS.....	4
2.2.4. Tại sao chúng ta cần CSS?	5
2.2.5. Các phương pháp sử dụng CSS.....	5
2.2.6. Thông tin	5
2.2.7. Kết luận	6
2.3. JavaScript.....	6
2.3.1. Tổng quan về JavaScript.....	6
2.3.2. JavaScript là gì?	6
2.3.3. Ứng dụng JavaScript.....	7
2.3.4. Lợi ích khi sử dụng JavaScript.....	7
2.3.5. Các thư viện và framework phổ biến của JavaScript	8
2.3.6. Ưu điểm và nhược điểm của JavaScript.....	8
2.3.7. JavaScript và các công cụ phát triển (IDE).....	9
2.3.8. Tầm quan trọng của JavaScript trong thời đại số	9
2.4. Bootstrap.....	9
2.4.1. Tổng quan về Bootstrap	9
2.4.2. Lí do nên sử dụng Bootstrap?	10
2.4.3. Lịch sử hình thành và phát triển của Bootstrap.....	10
2.4.4. Các thành phần chính trong Bootstrap	10
2.4.5. Tính năng nổi bật của Bootstrap	11
2.4.6. Tại sao Bootstrap được yêu thích?	11

2.4.7. Các ứng dụng thực tiễn của Bootstrap	11
2.4.8. Ưu điểm và nhược điểm của Bootstrap.....	12
2.4.9. Kết luận	12
2.5. SQL(Structured Query Language).....	12
2.5.1. Tổng quan.....	12
2.5.2. Giới thiệu.....	13
2.5.3. Lịch sử của SQL.....	13
2.5.4. Tại sao SQL lại quan trọng?	13
2.5.5. Các thành phần chính của hệ thống SQL.....	13
2.5.6. Các câu lệnh SQL quan trọng.....	14
2.5.7. Cách SQL hoạt động	14
2.5.8. Kết luận	15
2.6. Asp.net Core.....	15
2.6.1. ASP.NET Core là gì?	15
2.6.2. Lịch sử hình thành và phát triển	15
2.6.3. Các thành phần chính của ASP.NET Core	15
2.6.4. Các tính năng nổi bật của ASP.NET Core.....	16
2.6.5. Tính ứng dụng của ASP.NET Core trong thực tế	16
2.6.6. Kết luận	16
2.7. Mô hình MVC	17
2.7.1. MVC là gì?.....	17
2.7.2. Thành phần chính của mô hình MVC	17
2.7.3. Cách hoạt động của MVC	17
2.7.4. Những hạn chế của MVC.....	18
2.7.5. Kết luận	18
CHƯƠNG 3: THỰC NGHIỆM	19
3.1. Mô tả bài toán.....	19
CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU.....	23
4.1. Trang chủ.....	23
CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....	24
5.1. Kết luận	24
DANH MỤC TÀI LIỆU THAM KHẢO.....	25

DANH MỤC HÌNH ẢNH

Hình 3-1 Mô hình Diagrams	20
Hình 3-2 Bảng Product.....	20
Hình 3-3 Bảng Category.....	21
Hình 3-4 Bảng ProductSezi.....	21

DANH SÁCH BẢNG BIỂU

TÓM TẮT

Bài báo cáo này em tập trung vào quá trình xây dựng một website chuyên nghiệp dành cho việc bán giày, nhằm tối ưu hóa trải nghiệm mua sắm trực tuyến và thu hút một lượng lớn khách hàng.

Em sử dụng các ngôn ngữ lập trình như HTML, CSS, JavaScript,... kết hợp với Bootstrap để xây dựng trang web một cách linh hoạt và hiệu quả. Sử dụng cơ sở dữ liệu để quản lý thông tin sản phẩm và khách hàng.

Về giao diện em chú trọng vào việc phát triển giao diện người dùng thật đẹp mắt, thân thiện và dễ sử dụng. Sử dụng màu sắc hài hòa, hình ảnh chất lượng của sản phẩm để tạo ra trải nghiệm thú vị cho người dùng.

Tối ưu trải nghiệm người dùng thông qua cải thiện quy trình tìm kiếm sản phẩm nhanh chóng. Chú trọng vào khả năng tương tác trực tuyến và thuận tiện cho khách hàng.

Xây dựng hệ thống quản lý hiệu quả để quản lý sản phẩm và thông tin khách hàng.

MỞ ĐẦU

Trong thời đại công nghệ ngày nay, sự hiện diện trực tuyến đóng vai trò quan trọng trong việc phát triển kinh doanh và tạo ra trải nghiệm người dùng tốt nhất. Để đáp ứng nhu cầu ngày càng tăng của thị trường và khách hàng, em quyết định lựa chọn đề tài “Ứng dụng Asp.net Core MVC xây dựng website bán giày” cho đề tài nghiên cứu của mình.

1. Lí do chọn đề tài

Thời trang nói chung và giày nói riêng là một trong những loại vật dụng phổ biến nhất trên thế giới. Với nhu cầu về sử dụng giày ngày càng tăng, việc xây dựng website giúp đáp ứng nhu cầu mua sắm trực tuyến, mở rộng khả năng tiếp cận khách hàng.

Thương mại điện tử đang trở thành một phần quan trọng của hệ thống kinh doanh, đặc biệt là trong lĩnh vực dịch vụ như trực tuyến mua sắm về giày. Một trang web bán giày sẽ cung cấp cho khách hàng một phương tiện thuận tiện để lựa và mua sắm.

Xây dựng website đúng cách không chỉ tạo ra một giao diện đẹp mắt, thuận lợi mà còn nâng cao trải nghiệm của khách hàng. Em quan tâm đặc biệt đến cách thiết kế có thể tạo ra không gian trực tuyến tương tác và thân thiện với người dùng.

Một website chuyên nghiệp còn là công cụ quảng bá thương hiệu mạnh mẽ. Nó giúp xây dựng hình ảnh, cung cấp thông tin, và thực hiện các chiến dịch marketing online hiệu quả.

Website hoạt động không ngừng, cho phép khách hàng có thể mua giày bất cứ lúc nào. Là nơi tuyệt vời để thu thập phản hồi và đánh giá từ khách hàng. Khách hàng có thể tùy chỉnh đơn hàng của họ(ví dụ như: thêm hoặc giảm số lượng sản phẩm muốn mua, thay đổi sản phẩm mua).

Bán hàng trực tuyến giúp giảm chi phí vận hành so với cửa hàng truyền thống.

Thu thập và phân tích dữ liệu để hiểu rõ hơn về hành vi mua sắm của khách hàng.

Cơ hội để bán nhiều loại sản phẩm liên quan đến giày, không chỉ là giày mà còn là nhiều sản phẩm đa dạng khác như phụ kiện đi kèm.

Dễ dàng kết nối với các nền tảng mạng xã hội để mở rộng sự hiện diện trực tuyến.

Cung cấp dịch vụ giao hàng nhanh chóng và tiện lợi cho khách hàng.

Cung cấp thông tin chi tiết về sản phẩm, hướng dẫn sử dụng và hỗ trợ khách hàng trực tuyến.

2. Mục tiêu nghiên cứu

Mục tiêu của nghiên cứu này là phân tích và xây dựng một website bán giày hiệu quả, đảm bảo sự thuận tiện cho cả khách hàng và doanh nghiệp.

Em sẽ tập trung vào các yếu tố như trải nghiệm người dùng, quản lý sản phẩm, và tích hợp các chức năng tìm kiếm sản phẩm thật thuận tiện.

3. Đối tượng và phạm vi nghiên cứu

3.1 Đối tượng nghiên cứu

Nghiên cứu này sẽ tập trung vào các cá nhân có nhu cầu sử dụng hoặc quan tâm đến các sản phẩm giày dép, bao gồm cả người tiêu dùng phổ thông và những tín đồ thời trang, đặc biệt là nhóm thanh niên trong độ tuổi từ 18 đến 35.

Đây là những đối tượng thường xuyên cập nhật các xu hướng thời trang mới, có phong cách cá nhân và dành sự quan tâm đặc biệt đến các thương hiệu giày nổi tiếng.

Ngoài ra, nhóm đối tượng gián tiếp của nghiên cứu còn bao gồm các quản trị viên hay chủ các cửa hàng kinh doanh trực tuyến, những người cần sử dụng hệ thống quản lý sản phẩm và đơn hàng nhằm tối ưu hoá quy trình kinh doanh.

3.2 Phạm vi nghiên cứu

Phạm vi nghiên cứu sẽ bao gồm phân tích, xây dựng, và đánh giá hiệu suất của website.

Tập trung vào việc tạo và quản lý nội dung trang web.

Thiết kế và tích hợp các chức năng e-commerce như giỏ hàng, quản lý đơn hàng.

4. Phương pháp nghiên cứu

4.1 Nghiên cứu lý thuyết

Các nguyên tắc kỹ thuật thiết kế website hiệu quả.

Các xu hướng và công nghệ hiện đại trong lĩnh vực thiết kế website.

Nghiên cứu cách thiết kế, xây dựng, chức năng và tính năng của một số website bán giày thành công.

4.2 Nghiên cứu thực nghiệm

Tạo một giao diện người dùng thân thiện và dễ dàng sử dụng, tăng cường tính tương tác và thuận tiện trong quá trình tìm kiếm sản phẩm và đặt hàng.

Cải thiện thời gian tải trang để đảm bảo việc duyệt web mượt mà và không làm mất kiên nhẫn của người dùng.

CHƯƠNG 1: TỔNG QUAN

1.1. Đặt vấn đề

Trong thời đại công nghệ số, việc xây dựng một website bán giày không chỉ là một xu hướng mà còn là một nhu cầu thiết yếu. Em lựa chọn xây dựng website này nhằm đáp ứng nhu cầu ngày càng tăng của việc mua sắm trực tuyến và khao khát của người tiêu dùng với một trải nghiệm mua cả phê thuận tiện và nhanh chóng.

Công cụ mà em muốn áp dụng trong quá trình thiết kế bao gồm các nền tảng phát triển web hiện đại, hệ thống quản lý nội dung linh hoạt.

Kết quả mong muốn từ việc xây dựng này không chỉ dừng lại ở việc tạo ra một website thương mại điện tử chức năng mà còn hướng tới việc xây dựng một thương hiệu website bán giày trực tuyến mạnh mẽ, cung cấp dịch vụ khách hàng xuất sắc và tạo ra một cộng đồng yêu thích giày trực tuyến.

1.2. Mục đích nghiên cứu

Mục đích nghiên cứu không chỉ giới hạn ở việc phát triển một website bán giày mà còn nhằm vào việc khám phá cách thức công nghệ mới có thể tối ưu hóa kinh doanh và tạo ra giá trị gia tăng cho cả người bán và người mua.

Cũng nhằm mục đích quảng bá về giày, thông qua việc cung cấp nội dung về giày dép. Mục tiêu cuối cùng là tạo ra một nền tảng mua sắm trực tuyến thuận tiện, nhanh chóng và đáng tin cậy, mang lại trải nghiệm mua giày tốt nhất cho khách hàng.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

Chương này cung cấp cái nhìn tổng quan về công nghệ và ngôn ngữ lập trình cơ bản mà em sử dụng trong việc thiết kế và phát triển website.

2.1. HTML (Hyper Text Markup Language)

2.1.1. Giới thiệu

HTML (HyperText Markup Language) là ngôn ngữ đánh dấu siêu văn bản cơ bản nhất được sử dụng trong việc xây dựng các trang web. Đây là nền tảng để định nghĩa cấu trúc và nội dung của trang web, giúp trình duyệt hiểu và hiển thị đúng các thông tin mà nhà phát triển muốn truyền tải. Tuy nhiên, để có thể trình bày nội dung đẹp mắt hơn, ngoài HTML còn cần đến các công nghệ khác như **CSS** (dùng để định dạng và thiết kế giao diện) và **JavaScript** (dùng để tạo ra các tương tác và chức năng động).

2.1.2. Tìm hiểu về "Hypertext" và liên kết (Links)

Hypertext có nghĩa là các liên kết kết nối các trang web với nhau, giúp người dùng có thể điều hướng giữa các trang web một cách dễ dàng. Các liên kết này có thể kết nối trong cùng một website hoặc giữa nhiều website khác nhau. Chúng chính là yếu tố nền tảng của World Wide Web. Khi người dùng đăng tải nội dung lên Internet và liên kết nó với các trang web khác, thì họ đã trở thành một phần của mạng lưới thông tin toàn cầu này.

2.1.3. Lịch sử phát triển

HTML bắt nguồn từ đề xuất của nhà vật lý Tim Berners-Lee tại CERN vào năm 1980, dẫn đến việc ông phát triển HTML và phần mềm trình duyệt, máy chủ đầu tiên vào cuối những năm 1990. Mô tả công khai đầu tiên về HTML, "HTML Tags", được Berners-Lee công bố vào cuối năm 1991, mô tả 18 phần tử trong thiết kế ban đầu của HTML,. Mười một trong số đó vẫn tồn tại trong HTML4.

HTML được xác định là một ứng dụng của SGML (Standard Generalized Markup Language) và được chính thức hóa bởi Internet Engineering Task Force (IETF) vào giữa năm 1993. Phiên bản đầu tiên, "HTML 2.0", được hoàn thiện bởi IETF vào năm 1995.

Sự phát triển tiếp theo của HTML đã diễn ra dưới sự bảo trợ của W3C, bắt đầu từ năm 1996. Phiên bản quan trọng như HTML 4.01 được xuất bản vào cuối năm 1999,

và HTML5, một phiên bản đột phá, được W3C hoàn thiện vào ngày 28 tháng 10 năm 2014.

HTML5 đã đánh dấu một bước ngoặt trong lịch sử phát triển của HTML, bằng cách giới thiệu nhiều tính năng mới và cải thiện đáng kể so với các phiên bản trước, nhấn mạnh vào khả năng tương tác và đa phương tiện.

2.1.4. Cách hoạt động của HTML

HTML sử dụng các "markup" để chú thích nội dung như văn bản, hình ảnh và các loại nội dung khác để hiển thị trên trình duyệt web. Các **markup** này bao gồm các thành phần (elements) đặc biệt như:

Các thẻ cơ bản: `<head>`, `<title>`, `<body>`, `<header>`, `<footer>`, `<article>`, `<section>`.

Các thẻ định dạng: `<p>` (đoạn văn), `<div>` (phân chia khối), `` (phân chia dòng).

Các thẻ đa phương tiện: `` (hình ảnh), `<audio>` (âm thanh), `<video>` (video), `<canvas>` (vẽ đồ họa).

Các thẻ tương tác: `<form>` (biểu mẫu), `<input>` (trường nhập liệu), `<button>` (nút bấm).

Các thẻ điều hướng: `<nav>` (thanh điều hướng), `<a>` (liên kết).

2.1.5. Cấu trúc của một phần tử HTML

Mỗi phần tử HTML được xác định bởi các "thẻ" (**tags**) bao quanh tên phần tử đó, ví dụ `<p>`, `<h1>`, `<div>`, v.v. Tên của phần tử có thể được viết bằng chữ hoa, chữ thường hoặc kết hợp cả hai, ví dụ như `<Title>`, `<TITLE>`, nhưng chuẩn khuyến nghị hiện nay là nên viết bằng **chữ thường** để đảm bảo tính nhất quán và dễ đọc.

2.1.6. Kết luận

HTML là ngôn ngữ cơ bản nhưng vô cùng quan trọng trong lập trình web. Hiểu và thành thạo HTML là bước đầu tiên giúp người dùng xây dựng các trang web và ứng dụng trực tuyến hiệu quả [1].

2.2. CSS (Cascading Style Sheets)

2.2.1. Tổng quan về CSS?

CSS là ngôn ngữ định dạng trang web, cho phép các nhà phát triển điều chỉnh các yếu tố hiển thị trên trang như màu sắc, kiểu chữ, bố cục, khoảng cách, lề, nền, và nhiều hơn nữa. Nói cách khác, trong khi HTML chịu trách nhiệm về nội dung, CSS giúp cho nội dung đó trở nên hấp dẫn và dễ nhìn hơn đối với người dùng.

- **CSS giúp định dạng văn bản:** Người dùng có thể thay đổi màu sắc, kiểu chữ, kích thước chữ, khoảng cách giữa các dòng và chữ.
- **CSS giúp thiết kế bố cục:** Định vị các thành phần trên trang, căn lề, chia cột, tạo thanh điều hướng, và sắp xếp các phần tử sao cho hợp lý.
- **CSS tạo hiệu ứng đặc biệt:** Sử dụng các hiệu ứng chuyển động, thay đổi hình ảnh khi rê chuột, hiệu ứng mờ, và nhiều hiệu ứng khác để làm trang web trở nên sinh động.

2.2.2. Lịch sử hình thành và phát triển của CSS

Trước khi CSS ra đời, tất cả các yếu tố định dạng trang web đều được thực hiện trực tiếp bằng HTML. Các thẻ như `` để chỉnh màu sắc văn bản hay `<center>` để căn giữa văn bản đã từng rất phổ biến. Tuy nhiên, khi các trang web ngày càng phức tạp hơn, cách tiếp cận này trở nên rối rắm và khó bảo trì. Điều này đã dẫn đến sự ra đời của CSS vào năm 1996.

CSS ban đầu gặp phải sự phản đối từ nhiều nhà phát triển web vì họ đã quen với cách định dạng cũ. Tuy nhiên, CSS nhanh chóng chứng minh được sự hiệu quả của mình khi giúp tách biệt nội dung và phần hiển thị, làm cho mã nguồn trở nên gọn gàng và dễ bảo trì hơn.

2.2.3. Các phiên bản của CSS

Qua nhiều năm, CSS đã trải qua nhiều phiên bản khác nhau để đáp ứng các yêu cầu ngày càng cao của người dùng và sự tiến bộ của công nghệ:

- **CSS1:** Phiên bản đầu tiên, ra mắt năm 1996.
- **CSS2:** Nâng cấp với nhiều tính năng mới vào năm 1998.
- **CSS2.1:** Bản sửa đổi từ CSS2 để khắc phục các vấn đề về tương thích.
- **CSS3:** Phiên bản hiện tại với rất nhiều cải tiến về hiệu ứng, hoạt ảnh, và tính năng đồ họa tiên tiến.

Mặc dù **CSS3** vẫn chưa hoàn toàn được chuẩn hóa, hầu hết các trình duyệt hiện đại đã hỗ trợ phiên bản này. CSS3 mang lại rất nhiều tính năng mới mẻ, cho phép các nhà phát triển tạo ra những trang web bắt mắt và tương tác tốt hơn.

2.2.4. Tại sao chúng ta cần CSS?

Trước đây, các trang web chỉ đơn giản là các đoạn văn bản và hình ảnh nối tiếp nhau. Với sự ra đời của CSS, chúng ta có thể biến những trang web đơn giản thành những giao diện trực quan, hấp dẫn hơn:

- **Tiết kiệm thời gian và công sức:** Khi sử dụng tệp CSS riêng, người dùng chỉ cần định dạng một lần và áp dụng cho toàn bộ trang web. Điều này giúp giảm thiểu việc lặp lại mã và dễ dàng cập nhật khi cần thiết.
- **Tăng tính tương thích:** CSS cho phép trang web của người dùng tương thích với nhiều thiết bị khác nhau như điện thoại di động, máy tính bảng và máy tính để bàn.
- **Tối ưu hóa trải nghiệm người dùng:** Giao diện thân thiện, màu sắc hài hòa, và bố cục hợp lý giúp cải thiện trải nghiệm người dùng, thu hút họ quay lại trang web.

2.2.5. Các phương pháp sử dụng CSS

Có nhiều cách để áp dụng CSS vào trang HTML:

- **Sử dụng tệp CSS riêng biệt:** Đây là cách hiệu quả nhất, giúp người dùng quản lý và bảo trì dễ dàng hơn. Tất cả các định dạng được lưu trữ trong một tệp có phần mở rộng `.css`, và người dùng chỉ cần liên kết tệp này với các trang HTML.
- **Chèn mã CSS trong thẻ `<head>` của HTML:** Người dùng có thể viết mã CSS trực tiếp trong thẻ `<style>` trong phần `<head>` của trang HTML.
- **Sử dụng thuộc tính `style` trong thẻ HTML:** Đây là cách ít được khuyến khích vì làm mã nguồn trở nên lộn xộn và khó bảo trì.

2.2.6. Thông tin

Thông tin về CSS có thể đến từ nhiều nguồn khác nhau, chẳng hạn từ trình duyệt, tác giả, hoặc từ chính người dùng. Những thông tin CSS từ tác giả có thể được tách thành tệp tin riêng hoặc chèn trực tiếp vào văn bản HTML. Nhiều tệp tin CSS có thể được sử dụng cùng một lúc. Tác giả có thể quy định những kiểu dáng khác nhau dựa

trên thiết bị hiển thị của người dùng cuối, chẳng hạn cùng một trang web có thể hiển thị bố cục khác nhau khi truy cập trên điện thoại, máy tính bảng, hoặc máy tính cá nhân. Nhờ vậy, tác giả có thể tối ưu hóa văn bản của mình để thích hợp người sử dụng.

2.2.7. Kết luận

CSS là một công cụ mạnh mẽ và linh hoạt giúp các nhà phát triển web tạo ra các trang web đẹp mắt và dễ sử dụng. Hiểu rõ và thành thạo CSS sẽ giúp người dùng xây dựng những giao diện chuyên nghiệp, tối ưu hóa trải nghiệm người dùng, và tiết kiệm thời gian trong quá trình phát triển web.

Việc sử dụng CSS một cách hiệu quả sẽ không chỉ làm cho trang web của người dùng trở nên hấp dẫn hơn mà còn giúp cải thiện hiệu suất và khả năng tương thích của nó trên nhiều thiết bị và trình duyệt khác nhau. Đây là một kỹ năng không thể thiếu cho bất kỳ ai muốn trở thành nhà phát triển web chuyên nghiệp [2].

2.3. JavaScript

2.3.1. Tổng quan về JavaScript

JavaScript là một ngôn ngữ lập trình mạnh mẽ được tích hợp vào các trang web để giúp chúng trở nên sống động và tương tác hơn. Trong thế giới web hiện đại, JavaScript đóng vai trò quan trọng không chỉ trong việc xây dựng giao diện mà còn mở rộng khả năng tương tác với người dùng và tăng trải nghiệm người dùng trên website.

Nếu người dùng đã làm quen với HTML (Hypertext Markup Language) và CSS (Cascading Style Sheets) - hai công nghệ cốt lõi để tạo và định dạng nội dung trang web - thì việc học JavaScript sẽ là bước tiếp theo giúp người dùng đưa website của mình lên một tầm cao mới.

2.3.2. JavaScript là gì?

JavaScript là một ngôn ngữ lập trình **đa nền tảng**, được hỗ trợ bởi hầu hết các trình duyệt hiện đại như Google Chrome, Mozilla Firefox, Safari, Microsoft Edge và thậm chí là các trình duyệt trên thiết bị di động. Điều này có nghĩa là mã JavaScript có thể hoạt động trên hầu hết mọi thiết bị có trình duyệt, từ máy tính để bàn cho đến điện thoại thông minh và máy tính bảng.

JavaScript giúp tăng cường chức năng cho các trang web, cho phép chúng thực hiện những tác vụ mà HTML và CSS không thể tự làm được. Đơn cử, JavaScript có thể được sử dụng để tạo ra các hiệu ứng tương tác như:

Slideshow tự động: Các hình ảnh thay đổi liên tục mà không cần người dùng phải nhấp chuột.

Pop-up quảng cáo: Hiện thị thông báo hoặc khuyến mãi khi người dùng truy cập trang.

Autocomplete: Chức năng gợi ý từ khóa khi người dùng nhập liệu vào ô tìm kiếm, giống như Google.

2.3.3. Ứng dụng JavaScript

JavaScript không chỉ giới hạn trong lập trình website mà còn có thể mở rộng sang nhiều lĩnh vực khác:

Lập trình website: Làm cho trang web trở nên tương tác và thân thiện hơn, từ việc xác thực biểu mẫu, tạo hiệu ứng chuyển động, cho đến xây dựng các ứng dụng web phức tạp.

Lập trình phía máy chủ: JavaScript không chỉ giới hạn ở phía người dùng mà còn có thể được sử dụng để xây dựng các ứng dụng phía máy chủ (server-side) thông qua nền tảng như **Node.js**.

Xây dựng ứng dụng di động và trò chơi: JavaScript có thể được sử dụng để phát triển ứng dụng di động, đặc biệt với các framework như **React Native** hay **Ionic**.

Google AMP (Accelerated Mobile Pages): Dự án của Google nhằm tăng tốc độ tải trang trên thiết bị di động, yêu cầu sử dụng JavaScript để tối ưu hóa trải nghiệm người dùng.

2.3.4. Lợi ích khi sử dụng JavaScript

JavaScript đóng vai trò quan trọng trong việc xây dựng và phát triển website hiện đại. Dưới đây là một số lợi ích nổi bật của JavaScript:

Tăng tính tương tác: JavaScript giúp trang web trở nên sống động và có thể phản hồi ngay lập tức với các thao tác của người dùng như nhấp chuột, di chuyển chuột, hoặc nhập liệu.

Đơn giản hóa quá trình phát triển ứng dụng web: JavaScript cho phép các nhà phát triển tạo ra các ứng dụng web phức tạp nhưng lại dễ quản lý và bảo trì hơn nhờ vào các framework và thư viện như **React**, **Angular**, và **Vue**.

Thiết kế web responsive: JavaScript giúp tối ưu giao diện trang web trên cả máy tính và thiết bị di động, đảm bảo người dùng có trải nghiệm tốt nhất trên mọi loại thiết bị.

Tối ưu hóa hiệu suất: Sử dụng JavaScript để tải trước nội dung, giúp trang web tải nhanh hơn và cải thiện trải nghiệm người dùng.

2.3.5. Các thư viện và framework phổ biến của JavaScript

JavaScript có một hệ sinh thái rất phong phú với nhiều thư viện và framework được phát triển nhằm hỗ trợ các lập trình viên:

jQuery: Thư viện jQuery giúp đơn giản hóa thao tác trên DOM và xử lý sự kiện.

Node.js: Một nền tảng chạy JavaScript trên máy chủ, cho phép xây dựng các ứng dụng web tốc độ cao và hiệu quả.

ReactJS: Framework chuyên dùng để xây dựng giao diện người dùng, đặc biệt là ứng dụng web một trang (Single Page Applications).

AngularJS: Framework giúp xây dựng các ứng dụng web động và phức tạp.

Vue.js: Framework nhẹ và dễ học, lý tưởng cho những dự án nhỏ hoặc vừa.

2.3.6. Ưu điểm và nhược điểm của JavaScript

Ưu điểm:

- Dễ học và sử dụng: JavaScript có cú pháp tương đối dễ hiểu, đặc biệt cho người mới bắt đầu học lập trình.
- Chạy trên mọi trình duyệt: Không cần cài đặt phần mềm bổ sung, JavaScript có thể chạy trực tiếp trên hầu hết các trình duyệt.
- Tăng tính tương tác và trải nghiệm người dùng: JavaScript giúp tạo ra các hiệu ứng chuyển động và tương tác mượt mà, cải thiện trải nghiệm của người dùng trên trang web.
- Nhanh và nhẹ: JavaScript được thực thi trực tiếp trên trình duyệt của người dùng, giúp giảm tải cho máy chủ và tăng tốc độ tải trang.

Nhược điểm:

- Bảo mật: Do JavaScript chạy trực tiếp trên trình duyệt, nó có thể trở thành mục tiêu của các hacker để thực hiện mã độc.

- Tính không đồng nhất: Mặc dù JavaScript được hỗ trợ trên nhiều trình duyệt, đôi khi các tính năng có thể hoạt động khác nhau tùy thuộc vào trình duyệt và phiên bản.

2.3.7. JavaScript và các công cụ phát triển (IDE)

Để viết mã JavaScript, người dùng có thể sử dụng các công cụ như:

- **Visual Studio Code:** Một trong những IDE phổ biến nhất cho lập trình JavaScript.
- **Sublime Text:** Trình chỉnh sửa mã nguồn nhẹ, hỗ trợ nhiều ngôn ngữ lập trình.
- **PHPDesigner:** Một công cụ hỗ trợ lập trình web với nhiều tính năng cho PHP, HTML5, CSS3 và JavaScript.
- **Dreamweaver:** IDE mạnh mẽ cho thiết kế và phát triển các ứng dụng web.

2.3.8. Tầm quan trọng của JavaScript trong thời đại số

JavaScript hiện đang được sử dụng trên hơn 92% các trang web trên thế giới, và điều này cho thấy vai trò không thể thiếu của nó trong lĩnh vực lập trình web. Khi người dùng truy cập một trang web có hiệu ứng động, tính năng tự động hoàn tất, hoặc các ứng dụng web một trang, hầu hết những tính năng đó đều được tạo ra bằng JavaScript.

JavaScript không chỉ giúp tăng cường tính năng của website mà còn góp phần quan trọng vào việc phát triển ứng dụng di động và các nền tảng khác, từ đó mở ra vô số cơ hội cho những ai muốn dấn thân vào lĩnh vực lập trình.

Việc thành thạo JavaScript sẽ mang lại cho người dùng một lợi thế lớn, không chỉ giúp họ xây dựng các trang web chuyên nghiệp mà còn mở ra cánh cửa để phát triển các ứng dụng phức tạp và sáng tạo [3].

2.4. Bootstrap

2.4.1. Tổng quan về Bootstrap

Bootstrap là một **framework** (khung làm việc) mạnh mẽ, phổ biến nhất hiện nay dành cho lập trình và thiết kế website. Được xây dựng dựa trên **HTML**, **CSS**, và **JavaScript**, Bootstrap cung cấp một bộ công cụ đầy đủ giúp phát triển trang web chuẩn **responsive** (tương thích với nhiều thiết bị và kích thước màn hình). Với Bootstrap, việc tạo ra giao diện website chuyên nghiệp trở nên nhanh chóng và dễ dàng hơn.

2.4.2. Lí do nên sử dụng Bootstrap?

Bootstrap mang đến rất nhiều lợi ích cho các lập trình viên và nhà thiết kế web, giúp tối ưu hóa quy trình phát triển web:

Tối ưu thời gian phát triển: Nhờ vào các thành phần có sẵn như typography, biểu mẫu, nút, bảng, lưới (grid), thanh điều hướng, và các hiệu ứng động, người dùng có thể nhanh chóng xây dựng giao diện mà không cần viết mã từ đầu.

Thiết kế chuẩn responsive: Bootstrap được xây dựng dựa trên nguyên tắc **mobile-first** (ưu tiên thiết bị di động). Điều này đảm bảo rằng giao diện của người dùng sẽ hiển thị tốt trên mọi thiết bị, từ điện thoại, máy tính bảng, đến máy tính để bàn.

Đễ dàng tùy chỉnh: Người dùng có thể điều chỉnh các thành phần của Bootstrap để phù hợp với dự án của mình mà không cần phải viết lại mã từ đầu. Bootstrap còn cho phép tùy chỉnh giao diện trực tiếp trên **CDN** hoặc tải về và chỉnh sửa các tệp nguồn.

Tương thích trình duyệt: Bootstrap hỗ trợ tất cả các trình duyệt phổ biến như Google Chrome, Firefox, Safari, và Microsoft Edge. Điều này giúp người dùng đảm bảo rằng giao diện của mình sẽ hiển thị nhất quán trên mọi nền tảng.

2.4.3. Lịch sử hình thành và phát triển của Bootstrap

Bootstrap được phát triển bởi **Mark Otto** và **Jacob Thornton** tại Twitter. Ban đầu nó được gọi là **Twitter Blueprint**.

Phiên bản đầu tiên của Bootstrap được công bố vào ngày **19/8/2011** dưới dạng mã nguồn mở trên GitHub.

Phiên bản **Bootstrap 2** ra mắt vào ngày **31/1/2012**, bổ sung lưới **12 cột** giúp tạo bố cục dễ dàng hơn.

Phiên bản **Bootstrap 3** phát hành vào **19/8/2013** với thiết kế thân thiện hơn với **smartphone**.

Phiên bản **Bootstrap 4** xuất hiện vào **2015** với nhiều cải tiến, bao gồm **Sass**, **Flexbox**, và bỏ hỗ trợ cho IE8.

Phiên bản hiện tại **Bootstrap 5** đã loại bỏ jQuery và tập trung tối ưu hiệu suất.

2.4.4. Các thành phần chính trong Bootstrap

Bootstrap bao gồm ba thành phần chính:

- **Bootstrap.CSS:** Chịu trách nhiệm định dạng và bố cục của website.

- **Bootstrap.JS:** Đảm nhận các tương tác và hiệu ứng động, bao gồm thư viện **jQuery** để tối ưu hóa các thao tác với DOM.
- **Glyphicons:** Cung cấp các biểu tượng (icons) giúp tăng tính trực quan cho giao diện.

2.4.5. Tính năng nổi bật của Bootstrap

Dễ Dàng Sử Dụng Và Tùy Chỉnh

- **Dễ dàng thao tác:** Chỉ cần kiến thức cơ bản về HTML, CSS và JavaScript, người dùng có thể bắt đầu với Bootstrap mà không cần nhiều kinh nghiệm.
- **Tùy chỉnh linh hoạt:** Cho phép người dùng điều chỉnh các thuộc tính như kích thước, màu sắc, và kiểu dáng của các thành phần.

Thiết Kế Responsive Mobile-First

- Bootstrap hỗ trợ **Grid System** với **5 lớp responsive** (`col-sm`, `col-md`, `col-lg`, `col-xl`, `col-xxl`) giúp bố cục trang web tự động thay đổi để phù hợp với mọi kích thước màn hình.
- Tích hợp **Flexbox**, giúp các phần tử trong trang có thể điều chỉnh linh hoạt theo bố cục người dùng mong muốn.

Hệ Thống Lưới (Grid System)

- Bootstrap chia trang web thành **12 cột**, giúp người dùng dễ dàng tạo bố cục phức tạp.
- Sử dụng **container** và **container-fluid** để điều chỉnh độ rộng của trang web.

2.4.6. Tại sao Bootstrap được yêu thích?

- **Nhanh chóng và hiệu quả:** Giúp tiết kiệm thời gian khi phát triển giao diện website.
- **Được tối ưu hóa tốt:** Đảm bảo rằng trang web của người dùng tải nhanh hơn và có hiệu suất cao hơn.
- **Hỗ trợ cộng đồng lớn:** Bootstrap có một cộng đồng phát triển mạnh mẽ, luôn cung cấp các bản cập nhật và hỗ trợ miễn phí.

2.4.7. Các ứng dụng thực tiễn của Bootstrap

Bootstrap được ứng dụng rộng rãi trong việc thiết kế và phát triển website chuyên nghiệp:

- **Landing page:** Tạo các trang đích hấp dẫn để thu hút người dùng và tăng tỷ lệ chuyển đổi.
- **Trang thương mại điện tử:** Giúp xây dựng các trang web bán hàng tối ưu với bố cục gọn gàng và hiệu ứng tương tác cao.
- **Blog và diễn đàn:** Bootstrap giúp tạo ra các trang blog và diễn đàn đẹp mắt, dễ sử dụng.
- **Hệ thống quản lý nội dung (CMS):** Tích hợp dễ dàng với các nền tảng như WordPress, Joomla, và Drupal.

2.4.8. Ưu điểm và nhược điểm của Bootstrap

Ưu điểm:

- **Tốc độ phát triển nhanh chóng:** Giúp giảm thiểu thời gian và công sức khi phát triển website.
- **Thiết kế chuẩn responsive:** Đảm bảo giao diện đẹp và nhất quán trên mọi thiết bị.
- **Tài liệu phong phú và cộng đồng mạnh mẽ:** Bootstrap có tài liệu hướng dẫn chi tiết và cộng đồng hỗ trợ rộng lớn.

Nhược điểm:

- **Cấu trúc mã cồng kềnh:** Nếu không tùy chỉnh đúng cách, mã Bootstrap có thể làm tăng kích thước tệp và ảnh hưởng đến tốc độ tải trang.
- **Giao diện dễ bị trùng lặp:** Do nhiều website sử dụng cùng một framework, giao diện có thể trở nên phổ biến và thiếu sáng tạo.

2.4.9. Kết luận

Bootstrap là một công cụ tuyệt vời cho các nhà phát triển web, giúp tối ưu hóa quá trình thiết kế và phát triển website một cách hiệu quả. Với tính năng thiết kế chuẩn responsive, khả năng tùy chỉnh linh hoạt và sự hỗ trợ mạnh mẽ từ cộng đồng, Bootstrap vẫn là một trong những lựa chọn hàng đầu khi xây dựng các dự án web hiện nay [4].

2.5. SQL(Structured Query Language)

2.5.1. Tổng quan

SQL (Structured Query Language) hay **Ngôn ngữ truy vấn có cấu trúc** là một ngôn ngữ lập trình mạnh mẽ được sử dụng để quản lý và thao tác với các cơ sở dữ liệu quan hệ (**Relational Database**). Nó cho phép người dùng **lưu trữ, truy xuất, cập nhật,**

và **xóa dữ liệu** một cách hiệu quả từ các bảng dữ liệu, giúp quản lý thông tin dễ dàng và có hệ thống.

2.5.2. Giới thiệu

SQL là ngôn ngữ chuẩn để tương tác với cơ sở dữ liệu quan hệ (RDBMS). Trong hệ thống cơ sở dữ liệu quan hệ, dữ liệu được lưu trữ dưới dạng **bảng** với các **hàng** và **cột**. Mỗi bảng có thể liên kết với nhau thông qua các mối quan hệ dựa trên các **khóa chính (primary key)** và **khóa ngoại (foreign key)**.

Các hệ thống quản lý cơ sở dữ liệu phổ biến sử dụng SQL bao gồm:

MySQL

Oracle Database

Microsoft SQL Server

PostgreSQL

2.5.3. Lịch sử của SQL

- SQL ra đời vào những năm **1970** tại IBM với tên gọi ban đầu là **SEQUEL**.
- Đến năm **1986**, Viện Tiêu Chuẩn Quốc Gia Hoa Kỳ (**ANSI**) đã thông qua SQL làm tiêu chuẩn chính thức.
- Oracle, công ty phần mềm hàng đầu, đã thương mại hóa SQL và cung cấp hệ thống cơ sở dữ liệu đầu tiên dựa trên ngôn ngữ này.

2.5.4. Tại sao SQL lại quan trọng?

SQL là nền tảng cho hầu hết các ứng dụng cần lưu trữ và xử lý dữ liệu. Một số lý do khiến SQL trở thành một công cụ không thể thiếu:

Tính linh hoạt và mạnh mẽ: SQL có thể dễ dàng tích hợp với nhiều ngôn ngữ lập trình khác như **Java, Python, C#**, giúp xây dựng các ứng dụng phức tạp.

Hiệu quả: Các hệ thống cơ sở dữ liệu sử dụng SQL có thể xử lý **lượng dữ liệu lớn** với **tốc độ cao**.

Dễ học: Cú pháp của SQL khá đơn giản và dễ tiếp cận, ngay cả đối với những người mới bắt đầu.

2.5.5. Các thành phần chính của hệ thống SQL

Một hệ thống SQL bao gồm các thành phần cơ bản sau:

Bảng dữ liệu (Tables): Các bảng là thành phần cơ bản nhất, chứa các hàng và cột để lưu trữ thông tin.

Câu lệnh SQL (SQL Statements): Sử dụng để thao tác dữ liệu, bao gồm truy vấn, thêm, sửa, và xóa dữ liệu.

Quy trình lưu trữ (Stored Procedures): Là tập hợp các câu lệnh SQL được lưu trữ và có thể tái sử dụng để cải thiện hiệu suất.

2.5.6. Các câu lệnh SQL quan trọng

SQL có nhiều loại câu lệnh, nhưng chúng có thể được phân loại thành các nhóm chính như sau:

Ngôn Ngữ Định Nghĩa Dữ Liệu (DDL - Data Definition Language)

- **CREATE:** Tạo ra các bảng, chỉ mục, và các đối tượng khác.
- **ALTER:** Thay đổi cấu trúc của bảng.
- **DROP:** Xóa bảng hoặc đối tượng khỏi cơ sở dữ liệu.

Ngôn Ngữ Truy Vấn Dữ Liệu (DQL - Data Query Language)

- **SELECT:** Truy xuất dữ liệu từ các bảng.

Ngôn Ngữ Thao Tác Dữ Liệu (DML - Data Manipulation Language)

- **INSERT:** Thêm bản ghi mới vào bảng.
- **UPDATE:** Cập nhật thông tin của các bản ghi hiện có.
- **DELETE:** Xóa bản ghi khỏi bảng.

Ngôn Ngữ Kiểm Soát Dữ Liệu (DCL - Data Control Language)

- **GRANT:** Cấp quyền truy cập cho người dùng.
- **REVOKE:** Thu hồi quyền truy cập đã cấp.

Ngôn Ngữ Kiểm Soát Giao Dịch (TCL - Transaction Control Language)

- **COMMIT:** Lưu các thay đổi vào cơ sở dữ liệu.
- **ROLLBACK:** Hoàn tác các thay đổi chưa được lưu.

2.5.7. Cách SQL hoạt động

Quá trình thực thi một câu lệnh SQL bao gồm các bước chính như sau:

Trình phân tích cú pháp (Parser): Kiểm tra cú pháp và xác thực quyền hạn của người dùng.

Công cụ quan hệ (Relational Engine): Tạo ra kế hoạch truy xuất dữ liệu.

Công cụ lưu trữ (Storage Engine): Lưu trữ và truy xuất dữ liệu từ đĩa cứng.

2.5.8. Kết luận

SQL là một trong những công cụ mạnh mẽ và linh hoạt nhất để quản lý và thao tác với dữ liệu. Với khả năng truy xuất dữ liệu nhanh chóng, bảo mật tốt, và dễ tích hợp với các ứng dụng khác, **SQL là nền tảng quan trọng** cho bất kỳ dự án phần mềm nào liên quan đến cơ sở dữ liệu.

Việc nắm vững **SQL** sẽ giúp người dùng trở thành một lập trình viên hoặc chuyên gia dữ liệu **chuyên nghiệp**, mở ra nhiều cơ hội nghề nghiệp hấp dẫn trong lĩnh vực công nghệ thông tin [5].

2.6. Asp.net Core

2.6.1. ASP.NET Core là gì?

ASP.NET Core là một framework phát triển web đa nền tảng mã nguồn mở được Microsoft phát triển. Đây là phiên bản cải tiến của ASP.NET, được thiết kế lại từ nền tảng để mang lại hiệu suất cao hơn và linh hoạt hơn so với phiên bản cũ. ASP.NET Core hỗ trợ nhiều ngôn ngữ lập trình như C#, F#, và VB.NET, đồng thời có khả năng chạy trên nhiều hệ điều hành như Windows, Linux, và macOS. Với việc tối ưu hóa hiệu suất và giảm thiểu chi phí tài nguyên, ASP.NET Core đang trở thành lựa chọn phổ biến cho các dự án phát triển web hiện đại.

2.6.2. Lịch sử hình thành và phát triển

Microsoft lần đầu tiên giới thiệu .NET Framework vào năm 2002 với ASP.NET để xây dựng các ứng dụng web trên nền tảng Windows. Tuy nhiên, với sự gia tăng của các hệ điều hành khác và yêu cầu phát triển ứng dụng đa nền tảng, Microsoft đã ra mắt ASP.NET Core vào năm 2016. Phiên bản mới này không chỉ cải thiện về mặt hiệu năng mà còn mang đến khả năng linh hoạt và mở rộng hơn so với các phiên bản trước đó.

2.6.3. Các thành phần chính của ASP.NET Core

- **Trang .cshtml:** Đây là đơn vị cơ bản của một trang web trong ASP.NET Core (tương tự như .aspx trong ASP.NET). Các trang này chứa HTML, Razor, và mã lập trình để thể hiện nội dung và logic của trang.
- **Code-behind:** File code phía sau (.cs) chứa mã lập trình phục vụ cho trang tương ứng, giúp xử lý sự kiện và thao tác với dữ liệu.

- **appsettings.json:** Đây là file cấu hình chính cho ứng dụng ASP.NET Core, chứa các thiết lập như chuỗi kết nối cơ sở dữ liệu, cấu hình bảo mật, và các tùy chỉnh khác.
- **Startup.cs:** File này khởi tạo ứng dụng ASP.NET Core, định nghĩa các middleware và dịch vụ cần thiết.

2.6.4. Các tính năng nổi bật của ASP.NET Core

Kiến trúc MVC (Model-View-Controller): ASP.NET Core hỗ trợ mô hình MVC giúp tổ chức mã nguồn gọn gàng, dễ quản lý và mở rộng.

Data Access: ASP.NET Core hỗ trợ nhiều phương pháp truy cập dữ liệu như Entity Framework Core, LINQ, và ADO.NET.

Quản lý trạng thái (State Management): ASP.NET Core cung cấp các cơ chế quản lý trạng thái như Session, Cookie, và Cache để duy trì thông tin giữa các yêu cầu HTTP.

Bảo mật (Authentication và Authorization): Hỗ trợ các cơ chế xác thực người dùng như OAuth, JWT, và Identity để bảo vệ ứng dụng.

Caching: ASP.NET Core hỗ trợ caching để tăng hiệu suất ứng dụng bằng cách lưu trữ tạm thời dữ liệu hoặc mã kết quả.

2.6.5. Tính ứng dụng của ASP.NET Core trong thực tế

- **Phát triển website tuyến dụng:** ASP.NET Core hỗ trợ tốt cho việc xây dựng các trang web tuyến dụng với khả năng tích hợp các công cụ tìm kiếm và phân tích dữ liệu.
- **Phát triển ứng dụng e-commerce:** Các thư viện và công cụ mạnh mẽ giúp tạo ra các trang mua sắm trực tuyến, giỏ hàng, và hệ thống thanh toán tích hợp.
- **Phát triển ứng dụng di động:** ASP.NET Core kết hợp với Xamarin giúp xây dựng ứng dụng di động đa nền tảng với mã nguồn chung.
- **Quản lý dữ liệu và dự án:** Các doanh nghiệp có thể tận dụng ASP.NET Core để xây dựng các hệ thống quản lý dự án, lịch trình, và tài nguyên hiệu quả.

2.6.6. Kết luận

ASP.NET Core là một nền tảng mạnh mẽ và linh hoạt để phát triển các ứng dụng web hiện đại, đặc biệt là trong bối cảnh nhu cầu phát triển đa nền tảng ngày càng tăng. Với hiệu suất tối ưu và khả năng tích hợp dễ dàng với các công cụ hiện đại, ASP.NET

Core đang dần trở thành lựa chọn ưu tiên cho các dự án phát triển web, đặc biệt là trong các doanh nghiệp cần bảo mật và hiệu năng cao[6].

2.7. Mô hình MVC

2.7.1. MVC là gì?

MVC (Model-View-Controller) là một mẫu kiến trúc phần mềm được sử dụng rộng rãi trong phát triển ứng dụng web. Mục tiêu chính của mô hình này là tách rời các khía cạnh khác nhau của ứng dụng (giao diện người dùng, logic xử lý, và truy cập dữ liệu) thành các thành phần riêng biệt, giúp tối ưu hóa hiệu suất và dễ dàng bảo trì. ASP.NET Core MVC là một phần của ASP.NET Core framework, được Microsoft phát triển nhằm giúp xây dựng các ứng dụng web hiện đại với hiệu suất cao và bảo mật tốt.

2.7.2. Thành phần chính của mô hình MVC

Model: Model đại diện cho dữ liệu và logic nghiệp vụ của ứng dụng. Nó chịu trách nhiệm lấy dữ liệu từ cơ sở dữ liệu, xử lý logic liên quan và cập nhật trạng thái dữ liệu. Ví dụ, trong một ứng dụng mua sắm, Model sẽ quản lý các sản phẩm, thông tin giỏ hàng, và các giao dịch.

View: là thành phần hiển thị giao diện người dùng (UI). Nó nhận dữ liệu từ Model và trình bày nó cho người dùng. View không chứa logic nghiệp vụ, thay vào đó, nó chỉ chịu trách nhiệm về việc hiển thị và định dạng dữ liệu.

Controller: đóng vai trò trung gian giữa người dùng và hệ thống. Khi người dùng gửi yêu cầu (như nhấn nút hoặc điền vào biểu mẫu), Controller sẽ nhận yêu cầu này, xử lý logic, tương tác với Model và trả về View tương ứng. Ví dụ, khi người dùng thêm sản phẩm vào giỏ hàng, Controller sẽ cập nhật Model và gửi phản hồi cập nhật đến View.

2.7.3. Cách hoạt động của MVC

- **Model** giống như **đầu bếp**: xử lý tất cả các yêu cầu về món ăn (dữ liệu) và chuẩn bị chúng.
- **View** giống như **khách hàng**: hiển thị món ăn đã sẵn sàng (giao diện người dùng).
- **Controller** giống như **nhân viên phục vụ**: nhận yêu cầu của người dùng, truyền đạt đến đầu bếp, và phục vụ món ăn sau khi đã chuẩn bị xong.

2.7.4. Những hạn chế của MVC

- **Độ phức tạp cao:** Đối với các ứng dụng nhỏ hoặc đơn giản, MVC có thể làm tăng sự phức tạp không cần thiết.
- **Đường cong học tập:** Đối với các lập trình viên mới, việc nắm vững MVC và cách tổ chức mã có thể mất nhiều thời gian.
- **Số lượng tệp tăng lên:** MVC yêu cầu tách các thành phần thành nhiều tệp, làm cho dự án trở nên cồng kềnh hơn.

2.7.5. Kết luận

ASP.NET Core MVC là một framework mạnh mẽ và hiệu quả cho việc xây dựng các ứng dụng web hiện đại, đặc biệt là các ứng dụng có yêu cầu cao về hiệu suất và bảo mật. Với mô hình MVC, lập trình viên có thể tối ưu hóa tổ chức code, tăng khả năng bảo trì và dễ dàng mở rộng dự án. Tuy nhiên, nó cũng đòi hỏi sự hiểu biết sâu rộng và kinh nghiệm để sử dụng một cách hiệu quả[7].

CHƯƠNG 3: THỰC NGHIỆM

3.1. Mô tả bài toán

Xây dựng và phát triển một website bán giày chuyên nghiệp, hấp dẫn và thuận tiện cho khách hàng. Mục tiêu là tạo ra một nền tảng trực tuyến giúp khách hàng dễ dàng tìm kiếm và mua sắm các sản phẩm giày dép.

3.1.1. Các tính năng và chức năng

Tích hợp hệ thống quản lý sản phẩm để cập nhật thông tin về giày dép, bao gồm hình ảnh, giá và mô tả.

Danh mục sản phẩm được hiển thị rõ ràng để giúp người dùng dễ dàng tìm kiếm sản phẩm.

Người dùng có thể tạo tài khoản để mua các sản phẩm mình yêu thích.

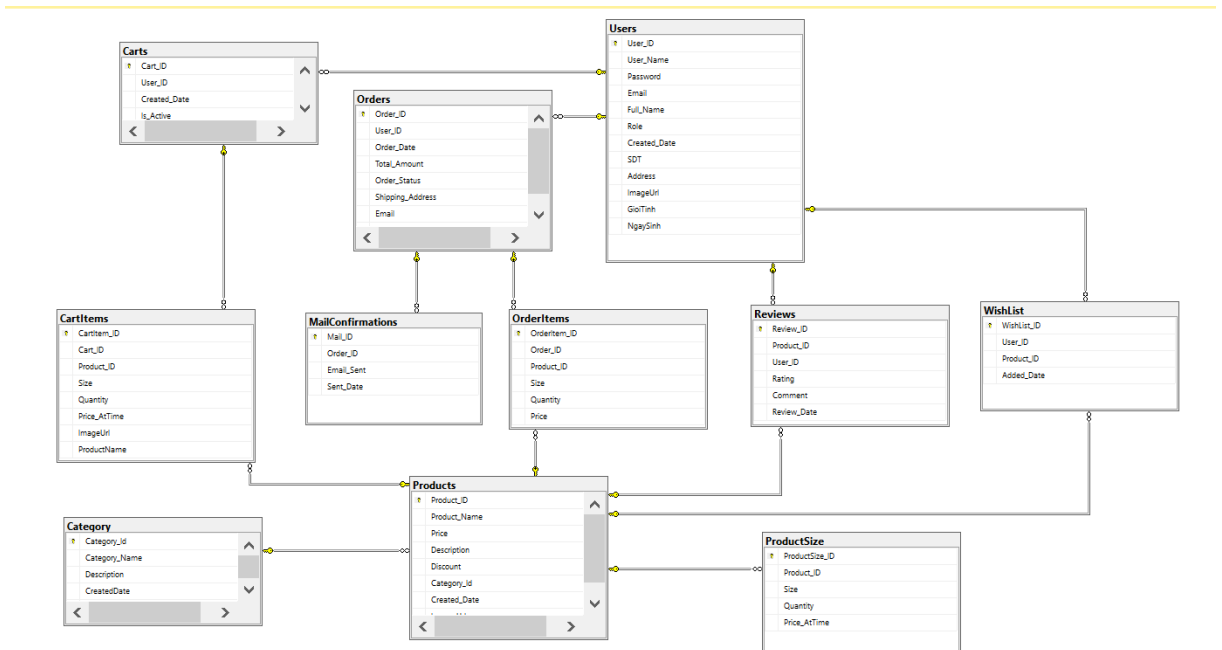
Khách hàng có thể tăng giảm số lượng các sản phẩm trong giỏ hàng một cách dễ dàng.

3.1.2. Giao diện người dùng

Sử dụng màu sắc phù hợp và hình ảnh chất lượng để tạo ra một trang web có thể làm cho người dùng cảm thấy thích thú.

Hiển thị sản phẩm giày dép theo cách thật hấp dẫn và thú vị để người dùng cảm thấy tò mò và bị thu hút.

3.2. Thiết kế dữ liệu



Hình 3-1 Mô hình Diagrams

3.2.1. Product

	Column Name	Data Type	Allow Nulls
🔑	Product_ID	int	<input type="checkbox"/>
	Product_Name	nvarchar(64)	<input type="checkbox"/>
	Price	decimal(18, 2)	<input type="checkbox"/>
	Description	nvarchar(MAX)	<input checked="" type="checkbox"/>
	Discount	int	<input type="checkbox"/>
	Category_Id	int	<input checked="" type="checkbox"/>
	Created_Date	date	<input checked="" type="checkbox"/>
	Image_Url	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Hình 3-2 Bảng Product

Bảng Product dùng để lưu danh sách các sản phẩm gồm có các trường:

- Id: dùng để lưu mã sản phẩm có kiểu dữ liệu int và là khóa chính.
- Name: dùng để lưu tên cho sản phẩm và có kiểu dữ liệu là nvarchar.
- Description: dùng để lưu mô tả sản phẩm và có kiểu dữ liệu là nvarchar.
- ImageUrl: dùng để trữ đường dẫn hình ảnh và có kiểu dữ liệu là nvarchar.
- Price: dùng để lưu giá gốc của sản phẩm và có kiểu dữ liệu decimal.
- Discount: dùng để lưu giá giảm của sản phẩm và có kiểu dữ liệu là int.
- CreateDate: dùng để lưu ngày tạo của sản phẩm và có kiểu dữ liệu là date.

- CategoryId: dùng để lưu mã danh mục sản phẩm tương ứng với sản phẩm, có kiểu dữ liệu là int và là khóa ngoại.

3.2.2. Category

	Column Name	Data Type	Allow Nulls
▼	Category_Id	int	<input type="checkbox"/>
	Category_Name	varchar(64)	<input type="checkbox"/>
	Description	nvarchar(MAX)	<input checked="" type="checkbox"/>
	CreateDate	date	<input checked="" type="checkbox"/>
	Image_Url	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Hình 3-3 Bảng Category

Bảng Category dùng để lưu danh mục sản phẩm và có hai trường:

- Id: dùng để lưu mã của danh mục sản phẩm có kiểu dữ liệu int và là khóa chính.
- Name: dùng để lưu tên cho danh mục sản phẩm và có kiểu dữ liệu là nvarchar.
- Description: dùng để lưu mô tả sản phẩm và có kiểu dữ liệu là nvarchar.
- CreateDate: dùng để lưu ngày tạo của sản phẩm và có kiểu dữ liệu là date.
- ImageUrl: dùng để trữ đường dẫn hình ảnh và có kiểu dữ liệu là nvarchar.

3.2.3. ProductSize

	Column Name	Data Type	Allow Nulls
▼	ProductSize_ID	int	<input type="checkbox"/>
	Product_ID	int	<input checked="" type="checkbox"/>
	Size	nvarchar(15)	<input type="checkbox"/>
	Quantity	int	<input type="checkbox"/>
	Price_AtTime	decimal(18, 2)	<input type="checkbox"/>
			<input type="checkbox"/>

Hình 3-4 Bảng ProductSezi

Bảng ProductSize dùng để lưu trữ kích thước và số lượng của mỗi sản phẩm có các trường sau:

- Id dùng để lưu id của size sản phẩm và có kiểu dữ liệu int.
- ProductId dùng để lưu trữ id sản phẩm tương ứng, có kiểu dữ liệu int và là khóa ngoại.
- Size dùng để lưu kích thước sản phẩm và có kiểu dữ liệu nvarchar.
- PriceAtTime dùng để lưu giá sản phẩm tại thời điểm tính và có kiểu dữ liệu int.decimal.

CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU

Website Bán giày được thiết kế để cung cấp một trải nghiệm người dùng mượt mà và thông tin chi tiết thông qua các màn hình kết quả khác nhau. Dưới đây là mô tả chi tiết về các chức năng chính và màn hình kết quả tương ứng:

4.1. Trang chủ

CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1. Kết luận

.

DANH MỤC TÀI LIỆU THAM KHẢO

- [1] Giới thiệu HTML, <https://developer.mozilla.org/en-US/docs/Web/HTML>, 12/11/2024.
- [2] Giới thiệu CSS, <http://www.laptrinhtanbinh.com/course/33/Xay-dung-trang-web-voi-HTML5-va-CSS3/44/2.1.-Tong-quan-ve-CSS>, [12/11/2024].
- [3] Giới thiệu JavaScript, https://wiki.matbao.net/javascript-la-gi-hoc-lap-trinh-javascript-ngon-ngu-cua-tuong-lai/?gad_source=1&gclid=Cj0KCQiAIsy5BhDeARIsABRc6Zu0-whIxeUdnoEijl4bUT1wDdwXg_m-0EsxUFSV0tt9Ys8G-70oFWoaAsVzEALw_wcB, [12/11/2024].
- [4] Giới thiệu Bootstrap, https://wiki.matbao.net/bootstrap-la-gi-cai-dat-bootstrap-web-chuan-responsive/?gad_source=1&gclid=Cj0KCQiAIsy5BhDeARIsABRc6ZvuP1RHlvzovgOPMt3yMTRgedUYhjRM-MF58_Mji0rQyvUCgk-o0DwaAkVuEALw_wcB, [12/11/2024].
- [5] Giới thiệu SQL Server, <https://aws.amazon.com/vi/what-is/sql/>, [12/11/2024].
- [6] Giới thiệu Asp.net Core, <https://fptshop.com.vn/tin-tuc/danh-gia/asp-net-la-gi-165617>, [16/11/2024].
- [7] Mô hình MVC, <https://itviec.com/blog/mvc-la-gi/#:~:text=MVC%20l%C3%A0%20vi%E1%BA%BFt%20t%E1%BA%AFt%20c%E1%BB%A7a,th%E1%BB%83%20c%E1%BB%A7a%20m%E1%BB%99t%20%E1%BB%A9ng%20d%E1%BB%A5ng.,> [16/11/2024].
- [8] Giáo hàng, <https://learn.microsoft.com/en-us/aspnet/core/fundamentals/app-state?view=aspnetcore-8.0>, [12/11/2024].