

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

## LỜI CẢM ƠN

Em xin gửi lời tri ân sâu sắc đến **Thầy TS. Đoàn Phước Miên**, người đã tận tình hướng dẫn và hỗ trợ em trong suốt quá trình thực hiện đồ án. Trong thời gian nghiên cứu và xây dựng hệ thống, Thầy đã định hướng, góp ý và truyền đạt những kiến thức quý giá, giúp em hoàn thành đề tài một cách tốt nhất.

Em cũng xin chân thành cảm ơn các Thầy, Cô trong Khoa đã giảng dạy và trang bị cho em nền tảng kiến thức vững chắc để có thể áp dụng vào thực tiễn. Đồng thời, em xin cảm ơn gia đình và bạn bè đã luôn động viên, khích lệ và tạo điều kiện thuận lợi giúp em hoàn thành đồ án này.

Mặc dù đã cố gắng hoàn thiện đề tài trong khả năng của mình, nhưng do hạn chế về thời gian và kinh nghiệm nên không tránh khỏi những thiếu sót. Kính mong Thầy, Cô thông cảm và chỉ bảo thêm để em có thể hoàn thiện hơn trong tương lai.

**Em xin chân thành cảm ơn!**

[illegible]

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

**MỤC LỤC**

NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN.....	i
LỜI CẢM ƠN.....	ii
NHẬN XÉT CỦA THÀNH VIÊN HỘI ĐỒNG.....	iii
DANH MỤC HÌNH ẢNH.....	vii
DANH MỤC BẢNG BIỂU.....	x
TÓM TẮT ĐỒ ÁN.....	xi
MỞ ĐẦU.....	1
CHƯƠNG 1: TỔNG QUAN.....	3
1.1. Giới thiệu chung.....	3
1.2. Hiện trạng và vấn đề cần giải quyết.....	3
1.3. Hướng tiếp cận.....	3
1.4. Kết quả mong đợi.....	4
CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT.....	5
2.1. Giới thiệu chương.....	5
2.2. Ngôn ngữ lập trình C#.....	5
2.2.1. Đặc trưng của C#.....	5
2.3. Hệ quản trị cơ sở dữ liệu SQL Server.....	5
2.3.1. Ứng dụng của SQL Server.....	5
2.4. Nền tảng .NET Framework 3.5.....	6
2.4.1. Lý do chọn .NET Framework 3.5.....	6
CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU.....	8
3.1. Mô tả bài toán .....	8

3.2. Yêu cầu chức năng.....	8
3.3. Mô hình cơ sở dữ liệu .....	9
3.4. Lược đồ Use Case của hệ thống.....	14
3.5. Kiến trúc hệ thống.....	16
3.5.1. Sơ đồ kiến trúc hệ thống (dạng chữ).....	16
3.5.1.1. Ưu điểm mô hình.....	17
3.6. Sơ đồ luồng xử lý đăng nhập Admin.....	17
3.7. Luồng xử lý chức năng Quản lý người dùng (User.aspx).....	19
3.7.1. Mô tả tổng quan.....	19
3.7.2. Sơ đồ luồng xử lý.....	19
3.7.3. Thêm người dùng (Create_User.aspx).....	21
3.7.4. Cập nhật người dùng (Edit_User.aspx).....	22
3.7.5. Xóa người dùng (Delete_User.aspx).....	23
3.8. Luồng xử lý Quản lý Sản phẩm.....	24
3.8.1. Hiển thị danh sách sản phẩm (Product.aspx).....	24
3.8.1.1. Sơ đồ luồng xử lý.....	24
3.8.1.2. Truy vấn dữ liệu hiển thị.....	24
3.8.2. Thêm sản phẩm (Create_Product.aspx).....	24
3.8.2.1. Luồng xử lý.....	25
3.8.2.2. Câu lệnh SQL sử dụng.....	25
3.8.3. Sửa sản phẩm (Edit_Product.aspx).....	25
3.8.3.1. Luồng xử lý.....	25
3.8.3.2. Câu lệnh SQL update.....	26
3.8.4. Xóa sản phẩm (Delete_Product.aspx).....	26
3.8.4.1. Luồng xử lý.....	26

3.8.4.2. SQL xóa.....	27
3.9. Chức năng Quản lý Danh mục (Category Management).....	27
3.9.1. Hiện thị danh mục.....	27
3.9.1.1. Mô tả chức năng.....	27
3.9.1.2. Luồng xử lý.....	28
3.9.1.3. Câu lệnh SQL sử dụng.....	28
3.9.2. Thêm danh mục (Create_Category.aspx).....	28
3.9.2.1. Mô tả chức năng.....	28
3.9.2.2. Luồng xử lý.....	29
3.9.2.3. SQL thực thi.....	29
3.9.3. Sửa danh mục (Edit_Category.aspx).....	29
3.9.3.1. Mô tả chức năng.....	29
3.9.3.2. Luồng xử lý.....	29
3.9.3.3. SQL thực thi.....	30
3.9.4. Xóa danh mục (Delete_Category.aspx).....	30
3.9.4.1. Luồng xử lý.....	30
3.9.4.2. SQL thực thi.....	31
3.10. Chức năng Quản lý Thương hiệu (Brand Management).....	31
3.10.1. Hiện thị danh sách thương hiệu.....	31
3.10.1.1. Mô tả chức năng.....	31
3.10.1.2. Luồng xử lý dữ liệu.....	32
3.10.1.3. Câu lệnh SQL truy vấn phân trang.....	32
3.10.2. Thêm thương hiệu (Create_Brand.aspx).....	32
3.10.2.1. Mô tả chức năng.....	32
3.10.2.2. Luồng xử lý.....	33

3.10.2.3. SQL thêm dữ liệu.....	33
3.10.3. Sửa thương hiệu (Edit_Brand.aspx).....	33
3.10.3.1. Mô tả chức năng.....	31
3.10.3.2. Luồng xử lý.....	33
3.10.3.3. SQL cập nhật.....	34
3.10.4. Xóa thương hiệu (Delete_Brand.aspx).....	34
3.10.4.1. Luồng xử lý.....	34
3.10.4.2. SQL xóa.....	35
3.11. Chức năng Giỏ hàng và Thanh toán.....	35
3.11.1. Chức năng Giỏ hàng (Cart.aspx).....	35
3.11.1.1. Mô tả chức năng.....	35
3.11.1.2. Luồng xử lý dữ liệu.....	36
3.11.1.3. Ví dụ SQL xử lý giỏ hàng từ CSDL.....	36
3.11.2. Kiểm tra tồn kho trước khi thanh toán.....	37
3.11.2.1. Mô tả.....	37
3.11.2.2. Ví dụ SQL kiểm tra tồn kho thực tế.....	37
3.11.3. Chức năng Thanh toán (Checkout.aspx).....	37
3.11.3.1. Mô tả chức năng.....	37
3.11.3.2. Luồng xử lý dữ liệu.....	38
3.11.3.3. SQL chính trong quá trình thanh toán.....	38
3.11.4. Gửi Email Xác Nhận.....	39
CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU.....	40
CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....	48
DANH MỤC TÀI LIỆU THAM KHẢO.....	49

## DANH MỤC HÌNH ẢNH

Hình 3.5. Sơ đồ cơ sở dữ liệu.....	9
Hình 3.5.1: Sơ đồ kiến trúc hệ thống .....	16
Hình 3.6: Sơ đồ luồng xử lý trang Login.aspx .....	17
Hình 3.7.2: Sơ đồ luồng xử lý trang User.aspx .....	19
Hình 3.7.3: Sơ đồ luồng xử lý trang Create_User.aspx.....	21
Hình 3.7.4: Sơ đồ luồng xử lý trang Edit_User.aspx .....	22
Hình 3.7.5: Sơ đồ luồng xử lý trang Delete_User.aspx .....	23
Hình 3.8.1: Sơ đồ luồng xử lý trang Product.aspx .....	24
Hình 3.8.2: Sơ đồ luồng xử lý trang Create_Product.aspx .....	25
Hình 3.8.3: Sơ đồ luồng xử lý trang Edit_Product.aspx .....	26
Hình 3.8.4: Sơ đồ luồng xử lý trang Delete_Product.aspx .....	27
Hình 3.9.1: Sơ đồ luồng xử lý trang Category.aspx .....	28
Hình 3.9.2: Sơ đồ luồng xử lý trang Create_Category.aspx .....	29
Hình 3.9.3: Sơ đồ luồng xử lý trang Edit_Category.aspx .....	30
Hình 3.9.4: Sơ đồ luồng xử lý trang Delete_Category.aspx .....	31
Hình 3.10.1: Sơ đồ luồng xử lý trang Brand.aspx .....	32
Hình 3.10.2: Sơ đồ luồng xử lý trang Create_Brand.aspx .....	33
Hình 3.10.3: Sơ đồ luồng xử lý trang Edit_Brand.aspx .....	34
Hình 3.10.4: Sơ đồ luồng xử lý trang Delete_Brand.aspx .....	35
Hình 3.11.1: Sơ đồ luồng xử lý trang Cart.aspx .....	36
Hình 3.11.3: Sơ đồ luồng xử lý trang Checkout.aspx .....	38
Hình 4.1. Giao diện đăng nhập .....	40
Hình 4.2. Giao diện trang quản trị.....	40
Hình 4.3. Giao diện quản lý người dùng.....	41
Hình 4.4. Giao diện quản lý sản phẩm.....	41
Hình 4.5. Giao diện quản lý đơn hàng.....	42



Hình 4.6. Giao diện chi tiết đơn hàng.....	42
Hình 4.7. Giao diện quản lý danh mục.....	43
Hình 4.8. Giao diện quản lý thương hiệu.....	43
Hình 4.9. Giao diện đăng ký.....	44
Hình 4.10. Giao diện đăng nhập (trang chủ).....	44
Hình 4.11. Giao diện trang chủ.....	45
Hình 4.12. Giao diện danh sách sản phẩm.....	45
Hình 4.13. Giao diện giỏ hàng.....	46
Hình 4.14. Giao diện trang thanh toán.....	46
Hình 4.15. Giao diện mail xác nhận đơn hàng .....	47

**DANH MỤC BẢNG BIỂU**

Bảng 2.1. So sánh giữa ASP.NET và JSP/Servlet .....	6
Bảng 3.3.1. Bảng user – order.....	10
Bảng 3.3.2. Bảng chi tiết sản phẩm.....	11
Bảng 3.3.3. Bảng user_detail.....	11
Bảng 3.3.4. Bảng nhãn hiệu sản phẩm (brand).....	12
Bảng 3.3.5. Bảng danh mục sản phẩm (category).....	13
Bảng 3.3.6. Bảng chi tiết giỏ hàng (cart_item).....	13
Bảng 3.4.1. Mô tả Use Case của hệ thống.....	15
Bảng 3.5.1. Công nghệ và môi trường sử dụng.....	16
Bảng 3.6.1. Giải thích logic xử lý đăng nhập.....	18
Bảng 3.7.2. Giải thích chi tiết chức năng quản lý User.....	20

## TÓM TẮT ĐỒ ÁN

Đề tài “**Website Bán Laptop**” được thực hiện nhằm xây dựng một hệ thống thương mại điện tử đơn giản, hỗ trợ người dùng có thể xem thông tin sản phẩm, thêm vào giỏ hàng và thực hiện đặt mua trực tuyến. Đồng thời, hệ thống cho phép quản trị viên quản lý danh mục sản phẩm, thương hiệu, đơn hàng và thông tin khách hàng một cách hiệu quả.

Trong quá trình thực hiện, sinh viên sử dụng **ASP.NET Framework 3.5** làm nền tảng phát triển ứng dụng web, kết hợp với **SQL Server** để quản lý cơ sở dữ liệu, cùng **Bootstrap** và **jQuery** để thiết kế giao diện thân thiện, trực quan và có tính tương tác cao. Hệ thống được xây dựng theo **mô hình 3 tầng (Presentation – Business – Data Access)** nhằm đảm bảo tính mở rộng, dễ bảo trì và tái sử dụng mã nguồn.

Kết quả đạt được là một website có đầy đủ các chức năng cơ bản của hệ thống bán hàng trực tuyến: quản lý người dùng, sản phẩm, giỏ hàng và đơn đặt hàng. Giao diện được thiết kế đơn giản, dễ sử dụng, tương thích với nhiều thiết bị. Đồ án đã giúp sinh viên củng cố kiến thức về lập trình web, thiết kế cơ sở dữ liệu, và triển khai hệ thống thực tế bằng công nghệ Microsoft.



## MỞ ĐẦU

### 1. Lý do chọn đề tài

Trong bối cảnh chuyển đổi số ngày càng mạnh mẽ, mua sắm trực tuyến đã trở thành xu hướng phổ biến trong đời sống hiện đại. Việc xây dựng các hệ thống thương mại điện tử không chỉ đáp ứng nhu cầu của người tiêu dùng mà còn giúp doanh nghiệp mở rộng thị trường và nâng cao hiệu quả kinh doanh.

Laptop là mặt hàng có nhu cầu lớn, đa dạng về mẫu mã và thương hiệu, nên việc xây dựng **website bán laptop trực tuyến** giúp người dùng dễ dàng so sánh, lựa chọn sản phẩm phù hợp, đồng thời giúp người quản trị thuận tiện trong việc quản lý dữ liệu và đơn hàng.

Đề tài “**Website Bán Laptop**” được chọn nhằm vận dụng kiến thức đã học về lập trình web, cơ sở dữ liệu và thiết kế giao diện người dùng, qua đó tạo ra một sản phẩm thực tế có khả năng ứng dụng cao.

### 2. Mục tiêu nghiên cứu

Xây dựng một website thương mại điện tử đơn giản, thân thiện, hỗ trợ người dùng tìm kiếm và mua laptop trực tuyến.

Áp dụng các công nghệ **ASP.NET Framework 3.5, SQL Server, Bootstrap** và **jQuery** để thiết kế và triển khai hệ thống.

Thiết kế cơ sở dữ liệu có tính toàn vẹn, quan hệ rõ ràng giữa các thực thể: người dùng, sản phẩm, đơn hàng, giỏ hàng, thương hiệu, v.v.

Tạo giao diện trực quan, tương thích đa thiết bị và có khả năng mở rộng cho các sản phẩm khác trong tương lai.

### 3. Đối tượng và phạm vi nghiên cứu

**Đối tượng nghiên cứu:** Các kỹ thuật lập trình web động sử dụng ASP.NET Framework và cơ sở dữ liệu SQL Server trong việc xây dựng hệ thống bán hàng trực tuyến.

**Phạm vi nghiên cứu:**

Chỉ tập trung phát triển phần **frontend và backend cơ bản** của website bán laptop.

Chức năng thanh toán trực tuyến chỉ mô phỏng ở mức giao diện, chưa tích hợp cổng thanh toán thực tế.

Hệ thống được chạy thử nghiệm trên môi trường **localhost** (máy chủ nội bộ).

**4. Phương pháp nghiên cứu**

**Phương pháp thu thập thông tin:** Tham khảo tài liệu, giáo trình và các website thương mại điện tử hiện có.

**Phương pháp phân tích và thiết kế:** Áp dụng mô hình **phân tích hướng đối tượng (OOP)**, thiết kế **mô hình cơ sở dữ liệu quan hệ (ERD)**, và **mô hình 3 tầng** trong phát triển phần mềm.

**Phương pháp thực nghiệm:** Cài đặt, chạy thử, kiểm thử chức năng và đánh giá kết quả thực tế của hệ thống.

## CHƯƠNG 1: TỔNG QUAN

### 1.1. Giới thiệu chung

Trong thời đại công nghệ số, việc mua sắm trực tuyến đã trở nên phổ biến và quen thuộc đối với người tiêu dùng. Các nền tảng thương mại điện tử như Shopee, Tiki, Lazada hay các website chuyên biệt của từng doanh nghiệp đã góp phần thay đổi mạnh mẽ hành vi mua hàng truyền thống. Laptop là sản phẩm công nghệ có tính cạnh tranh cao, đòi hỏi người mua phải xem xét kỹ về cấu hình, thương hiệu, giá cả và chế độ bảo hành. Vì vậy, việc xây dựng một website bán laptop trực tuyến giúp người dùng có thể tra cứu, so sánh và đặt mua nhanh chóng, đồng thời giúp người bán quản lý hiệu quả thông tin sản phẩm và đơn hàng. Đề tài “Website Bán Laptop” được thực hiện nhằm ứng dụng các kiến thức về lập trình web ASP.NET Framework 3.5, cơ sở dữ liệu SQL Server và thiết kế giao diện hiện đại với Bootstrap và jQuery, qua đó xây dựng một hệ thống web hoàn chỉnh, đáp ứng nhu cầu thực tế.

### 1.2. Hiện trạng và vấn đề cần giải quyết

Hiện nay, hầu hết các cửa hàng bán laptop truyền thống vẫn chỉ dừng ở việc trưng bày và bán hàng trực tiếp. Khách hàng muốn mua sản phẩm phải đến tận nơi, mất thời gian tìm hiểu và so sánh giá cả. Ngoài ra, nhiều website nhỏ của các cửa hàng chưa có hệ thống quản lý giỏ hàng, xử lý đơn hàng, hoặc cơ chế quản trị dữ liệu hiệu quả. Từ thực trạng đó, nhu cầu xây dựng một website quản lý bán laptop trực tuyến là hết sức cần thiết. Hệ thống cần đảm bảo các yêu cầu sau:

- Hiển thị danh sách sản phẩm rõ ràng, dễ tìm kiếm.
- Có chức năng giỏ hàng và đặt hàng trực tuyến.
- Cho phép quản trị viên thêm, sửa, xóa sản phẩm và theo dõi đơn hàng.
- Giao diện thân thiện, tương thích với nhiều loại thiết bị (máy tính, máy tính bảng, điện thoại).

### 1.3. Hướng tiếp cận

Để giải quyết bài toán đặt ra, đề tài áp dụng phương pháp phát triển phần mềm theo mô hình 3 tầng (Three-Tier Architecture) gồm:

- Tầng giao diện (Presentation Layer): hiển thị nội dung và cho phép người dùng tương tác với website.
- Tầng nghiệp vụ (Business Logic Layer): xử lý các quy tắc, chức năng chính như đăng nhập, thêm sản phẩm vào giỏ hàng, thanh toán.
- Tầng dữ liệu (Data Access Layer): kết nối, truy vấn và thao tác với cơ sở dữ liệu SQL Server.

Ngoài ra, đề tài áp dụng phương pháp phân tích hướng đối tượng (OOAD) để thiết kế cơ sở dữ liệu và các lớp xử lý trong hệ thống. Công nghệ Bootstrap và jQuery được sử dụng nhằm đảm bảo giao diện có tính tương tác cao, phản hồi nhanh, thân thiện với người dùng và dễ bảo trì về sau.

#### **1.4. Kết quả mong đợi**

Sau khi hoàn thành, hệ thống Website Bán Laptop dự kiến đạt được các mục tiêu sau:

- Xây dựng thành công website bán hàng trực tuyến với các chức năng cơ bản như quản lý sản phẩm, giỏ hàng, đơn hàng và tài khoản người dùng.
- Thiết kế cơ sở dữ liệu chặt chẽ, đảm bảo mối quan hệ rõ ràng giữa các bảng như user, product, order, cart, brand, v.v.
- Cung cấp giao diện quản trị giúp người quản lý dễ dàng thêm, sửa, xóa và theo dõi dữ liệu sản phẩm, khách hàng, đơn hàng.
- Hỗ trợ người dùng cuối trong việc tra cứu thông tin sản phẩm, thêm sản phẩm vào giỏ hàng và đặt mua trực tuyến.



## CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT

### 2.1. Giới thiệu chương

Chương này trình bày các cơ sở lý thuyết và công nghệ được sử dụng trong quá trình xây dựng đồ án “Website Bán Laptop”. Nội dung bao gồm: ngôn ngữ lập trình C#, hệ quản trị cơ sở dữ liệu SQL Server, nền tảng .NET Framework 3.5, và phần so sánh với JSP/Servlet để thấy rõ lý do lựa chọn công nghệ.

### 2.2. Ngôn ngữ lập trình C#

Ngôn ngữ lập trình C# (C Sharp) được phát triển bởi Microsoft, do Anders Hejlsberg dẫn đầu, ra đời vào năm 2000 cùng với .NET Framework 1.0. C# được thiết kế với mục tiêu kết hợp sức mạnh của C++ và tính đơn giản, an toàn của Java, nhằm tạo ra ngôn ngữ hướng đối tượng mạnh mẽ cho nền tảng Windows.

#### 2.2.1. Đặc trưng của C#

- Hướng đối tượng (OOP): hỗ trợ Encapsulation, Inheritance, Polymorphism, Abstraction.
- An toàn kiểu dữ liệu (Type Safety): giảm lỗi sai kiểu dữ liệu khi biên dịch.
- Tự động quản lý bộ nhớ (Garbage Collection): không cần giải phóng thủ công.
- Hỗ trợ đa luồng (Multithreading) và bất đồng bộ (async/await).
- Tích hợp tốt với .NET Framework và dễ bảo trì.

### 2.3. Hệ quản trị cơ sở dữ liệu SQL Server

Microsoft SQL Server là hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) do Microsoft phát triển. Ra đời năm 1989, SQL Server hỗ trợ ngôn ngữ T-SQL, khả năng mở rộng cao và tích hợp tốt với .NET, Power BI, Azure.

#### 2.3.1. Ứng dụng của SQL Server

SQL Server được ứng dụng trong:

- Website thương mại điện tử, quản lý kho.

- Ứng dụng doanh nghiệp, kế toán, nhân sự.
- Lưu trữ dữ liệu lớn (Data Warehouse) và BI.

## 2.4. Nền tảng .NET Framework 3.5

.NET Framework là nền tảng phần mềm của Microsoft, ra mắt 2002, hỗ trợ lập trình trên Windows. Phiên bản 3.5 (2007) tích hợp LINQ, WCF, AJAX và ASP.NET 3.5 giúp xây dựng ứng dụng web hiệu quả.

### 2.4.1. Lý do chọn .NET Framework 3.5

- Phù hợp với yêu cầu môn học và công cụ đào tạo.
- Tích hợp tốt với SQL Server và Visual Studio.
- Ổn định, dễ triển khai và hiệu năng cao.

**Bảng 2.1. So sánh giữa ASP.NET và JSP/Servlet**

Tiêu chí	ASP.NET (C#)	JSP/Servlet (Java)
Hãng phát triển	Microsoft	Oracle (Sun Microsystems)
Ngôn ngữ chính	C#	Java
Năm ra đời	2002	1999
Môi trường thực thi	.NET CLR	JVM (Java Virtual Machine)
Kết nối CSDL	ADO.NET, LINQ	JDBC
Hỗ trợ IDE	Visual Studio	Eclipse, NetBeans
Công nghệ UI	Web Forms, MVC, Razor	JSP, JSF
Bảo mật	Windows Authentication, dễ cấu hình	Cấu hình thủ công phức tạp hơn
Triển khai	IIS (Windows Server)	Tomcat, GlassFish, v.v.
Hiệu năng & dễ sử dụng	Cao, thân thiện, phù hợp doanh nghiệp	Linh hoạt, đa nền tảng nhưng phức tạp

**Nhận xét:**

ASP.NET thích hợp cho môi trường Windows, triển khai nhanh và tích hợp tốt với SQL Server. JSP/Servlet phù hợp hơn cho môi trường đa nền tảng, linh hoạt về hệ thống. Với đề án này, việc chọn ASP.NET Framework 3.5 là hợp lý vì đồng bộ với công cụ Microsoft, hỗ trợ phát triển và quản lý dữ liệu hiệu quả.

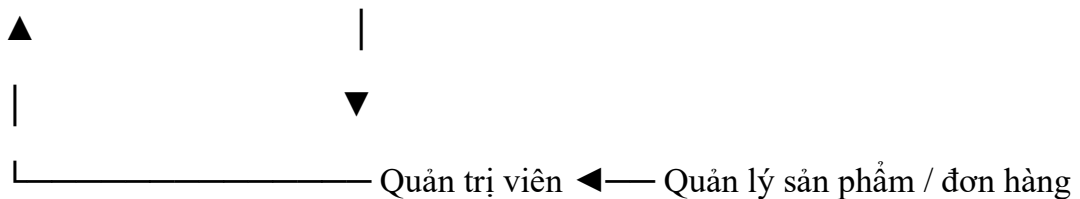
## CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU

### 3.1. Mô tả bài toán

Hệ thống *Website Bán Laptop* được xây dựng nhằm hỗ trợ người dùng dễ dàng tìm kiếm, lựa chọn và mua sản phẩm laptop trực tuyến, đồng thời giúp người quản trị (Admin) có thể quản lý dữ liệu sản phẩm, khách hàng và đơn hàng một cách hiệu quả.

#### Sơ đồ mô tả bài toán (dạng chữ):

Người dùng → Website → Giỏ hàng → Đặt hàng → Thanh toán



#### Diễn giải:

- Người dùng có thể xem sản phẩm, thêm vào giỏ hàng và đặt hàng.
- Quản trị viên có thể quản lý thông tin sản phẩm, thương hiệu, danh mục và đơn hàng.

### 3.2. Yêu cầu chức năng

#### Chức năng của người dùng (User):

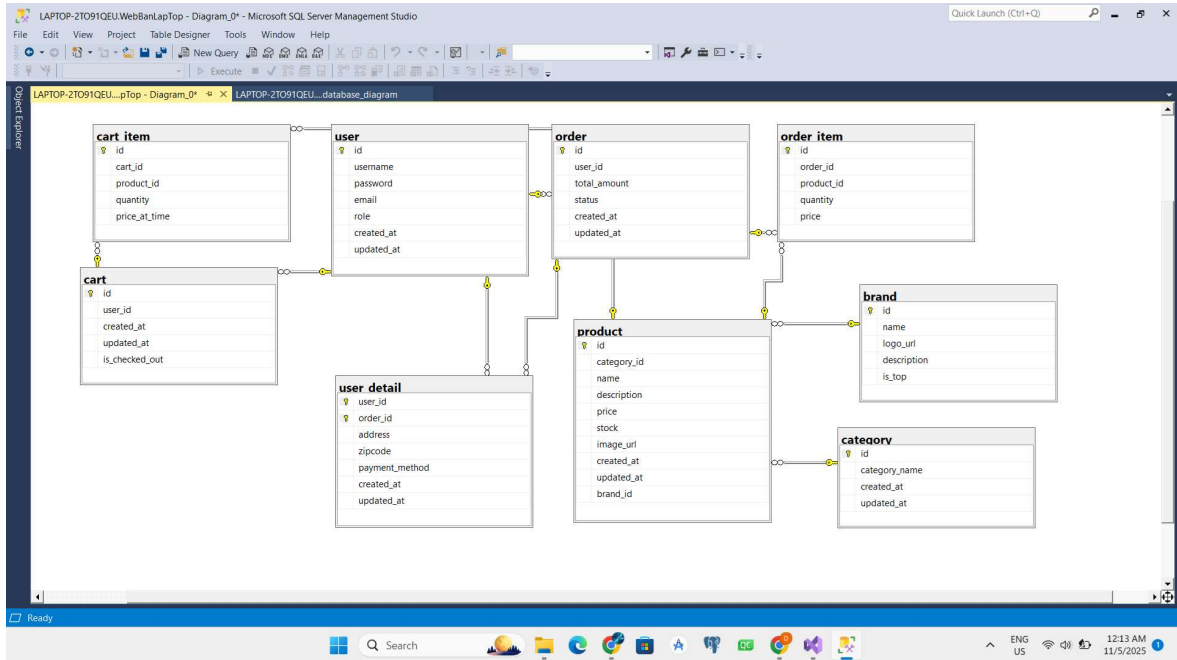
1. Đăng ký / đăng nhập tài khoản.
2. Xem danh sách và chi tiết sản phẩm laptop.
3. Tìm kiếm sản phẩm theo tên, thương hiệu hoặc danh mục.
4. Thêm sản phẩm vào giỏ hàng.
5. Đặt hàng và theo dõi trạng thái đơn hàng.

#### Chức năng của quản trị viên (Admin):

1. Quản lý sản phẩm: thêm, sửa, xóa.
2. Quản lý danh mục và thương hiệu.

3. Quản lý đơn hàng: xem, cập nhật trạng thái.
4. Quản lý người dùng.

### 3.3 Mô hình cơ sở dữ liệu



Hình 3.3: Sơ đồ cơ sở dữ liệu

[user] (id, username, password, email, role)

|

└─< [order] (id, user\_id, total\_amount, status)

|

└─< [order\_item] (order\_id, product\_id, quantity, price)

|

└─< [cart] (id, user\_id, is\_checked\_out)

|

└─< [cart\_item] (cart\_id, product\_id, quantity, price\_at\_time)

|

└─< [user\_detail] (user\_id, order\_id, address, zipcode)

[product] (id, name, description, price, stock, brand\_id, category\_id)

└──< [brand] (id, name, logo\_url)

└──< [category] (id, category\_name)

### Mối quan hệ:

- 1 user có nhiều order, cart, và user\_detail (tùy vào từng order).
- 1 order có nhiều order\_item.
- 1 product thuộc về 1 brand và 1 category.
- 1 cart có nhiều cart\_item.
- 1 order thuộc về 1 user\_detail

### 1) User ↔ Order (1 người dùng có nhiều đơn hàng)

#### Thực tế:

User customer1 (id = 2) đã đặt 2 đơn.

#### SQL kiểm tra:

- `SELECT u.username, o.id AS order_id, o.total_amount, o.status`
- `FROM [user] u`
- `JOIN [order] o ON u.id = o.user_id`
- `WHERE u.id = 2;`

#### Kết quả dự kiến:

**Bảng 3.3.1. Bảng user order**

username	order_id	total_amount	status
customer1	1	4300000.00	paid
customer1	2	1800000.00	pending

### 2) Quan hệ Order → Order\_Item (1 → N)

### 3.3.2 Giải thích

Một đơn hàng có thể chứa nhiều sản phẩm.

#### 3.3.2.1 Ví dụ thực tế

Đơn hàng #1 gồm:

**Bảng 3.3.2. Bảng chi tiết sản phẩm**

Sản phẩm (product.name)	Số lượng	Giá tại thời điểm mua
Dell XPS 13	1	2,200,000.00
HP Spectre x360	1	2,100,000.00

#### Query SQL Server

- *SELECT o.id AS order\_id, p.name AS product\_name, oi.quantity, oi.price*
- *FROM [order] o*
- *JOIN order\_item oi ON o.id = oi.order\_id*
- *JOIN product p ON oi.product\_id = p.id*
- *WHERE o.id = 1;*

### 3) Quan hệ User → User\_Detail (1 → N)

#### Giải thích

Mỗi đơn hàng có thông tin giao hàng riêng → 1 user có nhiều user\_detail.

#### Ví dụ thực tế

User customer1 (id = 2) có:

**Bảng 3.3.3. Bảng user detail**

order_id	address	payment_method
1	123 Đường ABC, Quận 1, TP.HCM(Sài Gòn)	Thẻ tín dụng

<i>order_id</i>	<i>address</i>	<i>payment_method</i>
2	456 Đường XYZ, Quận 3, TP.HCM(Sài Gòn)	Thu COD

**Query SQL Server**

```

SELECT u.username, ud.order_id, ud.address,
ud.payment_method
FROM [user] u
JOIN user_detail ud ON u.id = ud.user_id
WHERE u.id = 2;

```

**4) Quan hệ Product  $\rightarrow$  Brand ( $N \rightarrow 1$ )**

*Giải thích*

Một sản phẩm chỉ thuộc một thương hiệu.

*Ví dụ thực tế*

**Bảng 3.3.4. Bảng nhãn hiệu sản phẩm**

<i>product.name</i>	<i>brand</i>
Asus ROG Strix G15	Asus
MSI Katana GF66	MSI
Dell XPS 13	Dell

**Query SQL Server**

```

SELECT p.name AS product_name, b.name AS brand_name
FROM product p

```



---

```
JOIN brand b ON p.brand_id = b.id;
```

### 5) Quan hệ Product $\rightarrow$ Category ( $N \rightarrow 1$ )

#### 3.3.5 Ví dụ thực tế

**Bảng 3.3.5. Bảng danh mục sản phẩm**

<i>product.name</i>	<i>category_name</i>
<i>Asus ROG Strix G15</i>	<i>Gaming Laptop</i>
<i>Dell XPS 13</i>	<i>Ultrabook</i>

#### Query SQL Server

```
SELECT p.name AS product_name, c.category_name
FROM product p
JOIN category c ON p.category_id = c.id;
```

### 6) Quan hệ Cart $\rightarrow$ Cart\_Item ( $1 \rightarrow N$ )

#### 3.3.6 Giải thích

Một giỏ hàng chứa nhiều sản phẩm.

##### 3.3.6.1 Ví dụ thực tế

Tạo giỏ hàng & item đúng chuẩn:

```
INSERT INTO cart (user_id) VALUES (2); -- cart_id = 1
INSERT INTO cart_item (cart_id, product_id, quantity,
price_at_time)
VALUES
(1, 1, 1, 2500000.00),
(1, 2, 2, 1800000.00);
```

### 3.3.6.2 Kết quả thực tế

**Bảng 3.3.6. Bảng chi tiết giỏ hàng**

<i>product_name</i>	<i>quantity</i>
<i>Asus ROG Strix G15</i>	<i>1</i>
<i>MSI Katana GF66</i>	<i>2</i>

### 3.3.6.4 Query SQL Server

```
SELECT c.id AS cart_id, p.name AS product_name,
ci.quantity, ci.price_at_time
```

```
FROM cart c
```

```
JOIN cart_item ci ON c.id = ci.cart_id
```

```
JOIN product p ON ci.product_id = p.id
```

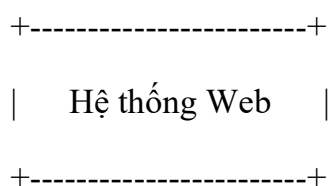
```
WHERE c.id = 1;
```

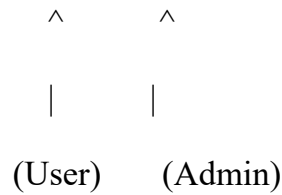
- Một người dùng có thể có nhiều đơn hàng và nhiều giỏ hàng.
- Một đơn hàng có thể chứa nhiều sản phẩm khác nhau.
- Một sản phẩm chỉ thuộc một thương hiệu và một danh mục.
- Một giỏ hàng có thể chứa nhiều sản phẩm, mỗi sản phẩm lưu lại giá tại thời điểm thêm.
- Mỗi đơn hàng có thông tin giao hàng riêng tương ứng trong bảng user\_detail.

## 3.4. Lược đồ Use Case của hệ thống

Hệ thống gồm 2 tác nhân:

- **Người dùng (User):** sử dụng website để xem sản phẩm và đặt hàng.
- **Quản trị viên (Admin):** quản lý hệ thống và dữ liệu.





User thực hiện:

- └─ Đăng ký tài khoản
- └─ Đăng nhập
- └─ Xem danh sách sản phẩm
- └─ Xem chi tiết sản phẩm
- └─ Tìm kiếm, lọc ra sản phẩm thông qua tên sản phẩm, nhãn hiệu, danh mục
- └─ Thêm sản phẩm vào giỏ hàng
- └─ Xem giỏ hàng
- └─ Đặt hàng
- └─ Xem trạng thái đơn hàng

Admin thực hiện:

- └─ Đăng nhập
- └─ Quản lý sản phẩm (Thêm / Sửa / Xóa)
- └─ Quản lý danh mục và thương hiệu
- └─ Quản lý người dùng (cấp quyền / khóa tài khoản)
- └─ Quản lý đơn hàng (Xác nhận / Giao hàng / Hủy đơn)

**Bảng 3.4.1. mô tả Use Case**

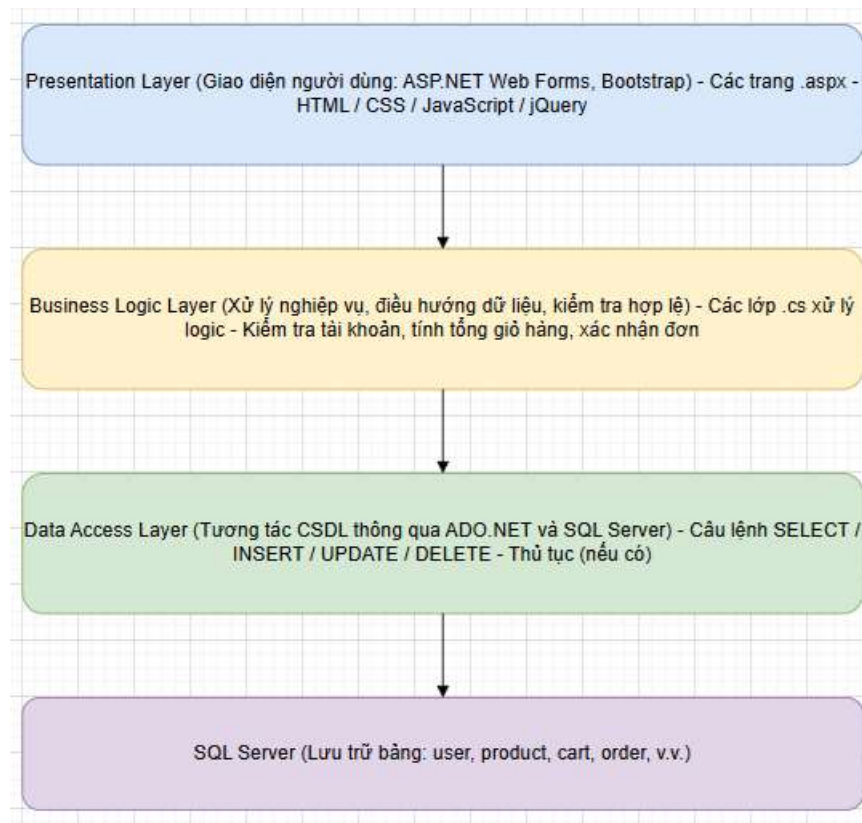
Tác nhân	Use Case	Nội dung chức năng
User	Đăng nhập	Người dùng nhập tài khoản để truy cập hệ thống.

Tác nhân	Use Case	Nội dung chức năng
User	Xem sản phẩm	Hiển thị danh sách laptop theo loại hoặc thương hiệu.
User	Thêm sản phẩm vào giỏ hàng	Lưu sản phẩm để chuẩn bị đặt hàng.
User	Đặt hàng	Tạo đơn hàng và cung cấp thông tin giao nhận.
Admin	Quản lý sản phẩm	Thêm, sửa, xóa thông tin laptop.
Admin	Quản lý đơn hàng	Theo dõi, thay đổi trạng thái đơn hàng.

### 3.5. Kiến trúc hệ thống

Hệ thống được xây dựng theo **mô hình 3 lớp (Three-Tier Architecture)**, đảm bảo dễ bảo trì, mở rộng và quản lý.

#### 3.5.1. Sơ đồ kiến trúc hệ thống



Hình 3.5.1: Sơ đồ kiến trúc hệ thống

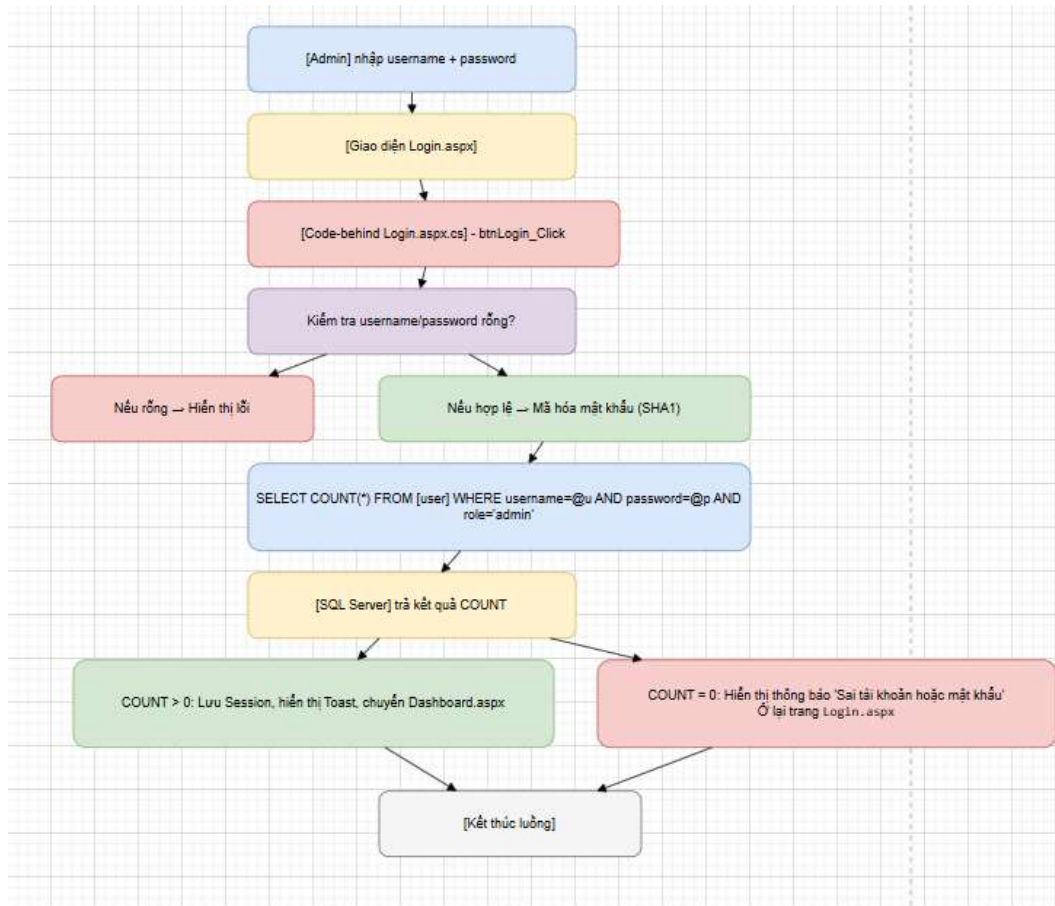
Bảng 3.5.1. Công nghệ và môi trường sử dụng

Thành phần	Công nghệ / Công cụ
Ngôn ngữ lập trình	C#
Nền tảng ứng dụng	ASP.NET Framework 3.5
Giao diện người dùng	HTML, CSS, Bootstrap, jQuery
Lưu trữ dữ liệu	SQL Server
Công cụ phát triển	Visual Studio 2010 / 2012
Môi trường chạy	IIS / IIS Express Windows

### 3.5.1.1 Ưu điểm mô hình

- Dễ bảo trì, dễ mở rộng.
- Mỗi tầng độc lập → thay thế hoặc nâng cấp không ảnh hưởng lẫn nhau.
- **Hiệu quả với project ASP.NET truyền thống.**

### 3.6. Sơ đồ luồng xử lý đăng nhập Admin



Hình 3.6: Sơ đồ luồng xử lý trang Login.aspx

**Bảng 3.6.1. Giải thích logic login**

Bước	Mô tả xử lý
1	Admin nhập tên đăng nhập và mật khẩu vào form Login.aspx.
2	Sự kiện btnLogin_Click được kích hoạt khi nhấn nút Đăng nhập.
3	Kiểm tra đầu vào, nếu thiếu → báo lỗi ngay.
4	Mật khẩu được mã hóa SHA1 (đúng chuẩn bảng user đang lưu).
5	Hệ thống kết nối CSDL và kiểm tra tài khoản có tồn tại và có vai trò admin hay không.

Bước	Mô tả xử lý
6	SQL trả về số lượng bản ghi tìm thấy (0 hoặc 1).
7A	Nếu tài khoản hợp lệ → tạo Session đăng nhập → chuyển đến Dashboard.
7B	Nếu sai thông tin → xuất thông báo lỗi trên giao diện.

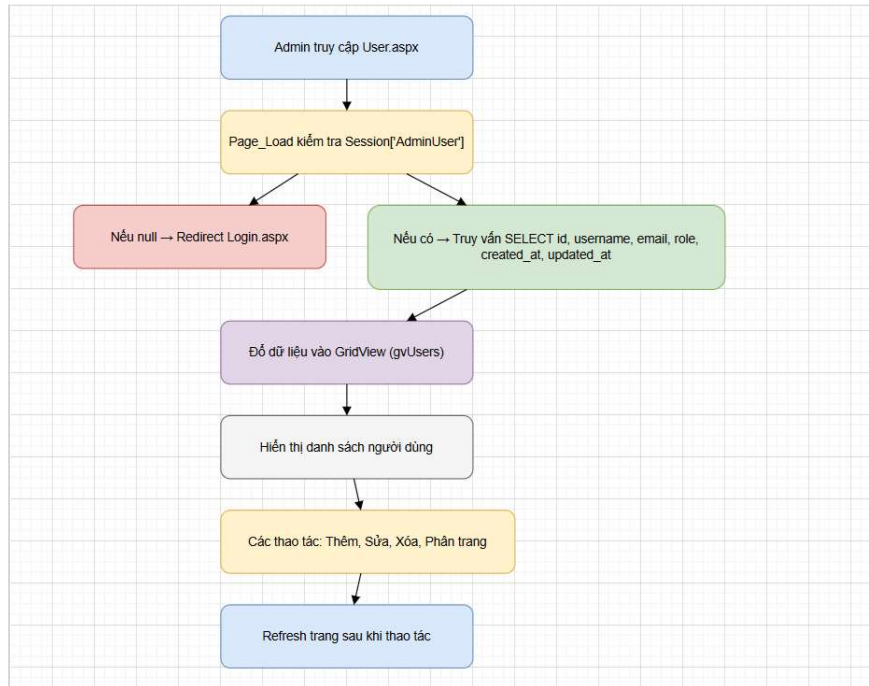
### 3.7. Luồng xử lý chức năng Quản lý người dùng (User.aspx)

#### 3.7.1. Mô tả tổng quan

Trang **User.aspx** có chức năng:

- Hiện thị danh sách tài khoản từ bảng **user**
- Cho phép **thêm** tài khoản mới
- Cho phép **sửa** tài khoản
- Cho phép **xóa** tài khoản
- Có phân trang dữ liệu

#### 3.7.2. Sơ đồ luồng xử lý



Hình 3.7.2: Sơ đồ luồng xử lý trang User.aspx

Bảng 3.7.2. Giải thích chi tiết quản lý user

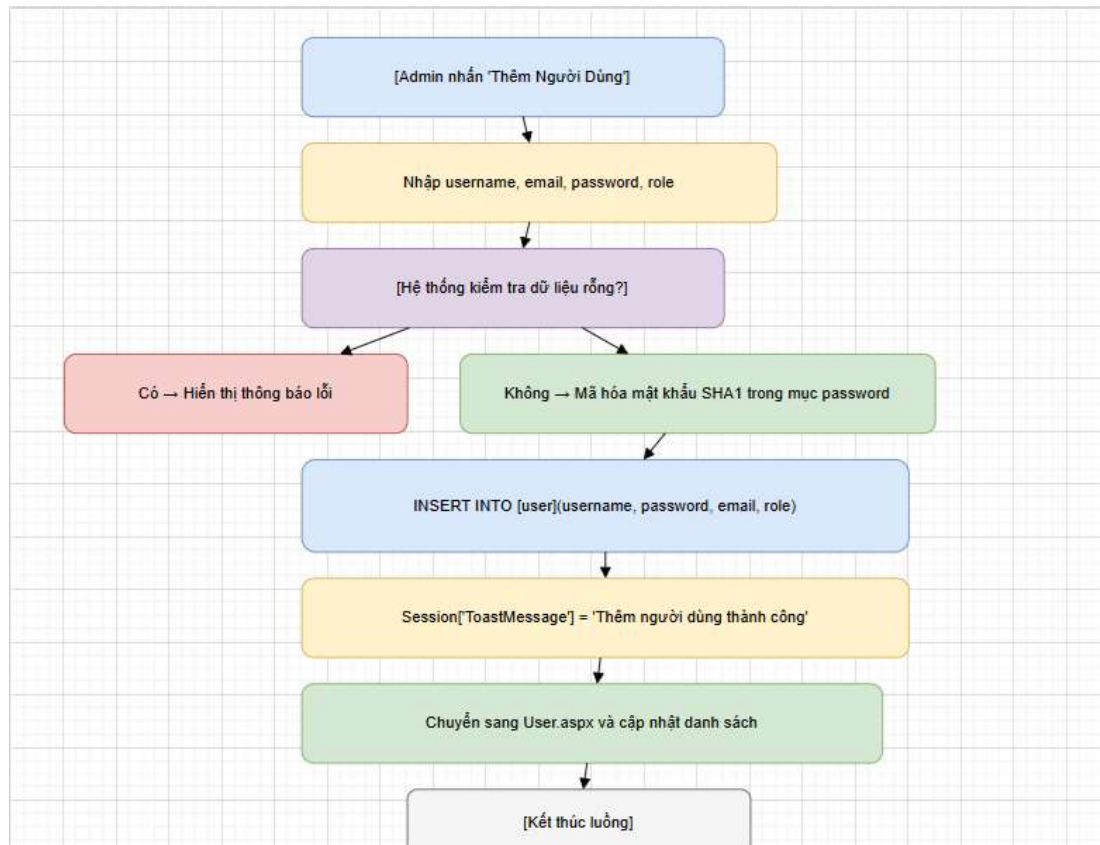
Bước	Hành động	Vai trò	Kết quả
1	Admin mở trang <b>User.aspx</b>	Hệ thống	Kiểm tra Session đăng nhập
2	Nếu chưa đăng nhập → chuyển đến Login.aspx	Hệ thống	Bảo mật
3	Nếu đã đăng nhập → truy vấn danh sách người dùng từ bảng <b>user</b>	SQL Server	Dữ liệu trả về dạng bảng
4	Hệ thống bind dữ liệu vào <b>GridView</b>	ASP.NET	Hiển thị danh sách người dùng
5	Admin nhấn <b>Thêm</b>	Người dùng	Chuyển đến <b>Create_User.aspx</b>
6	Admin nhấn <b>Sửa</b>	Người dùng	Chuyển đến <b>Edit_User.aspx?id=xx</b>
7	Admin nhấn <b>Xóa</b>	Người	Thực thi <b>DELETE FROM</b>



Bước	Hành động	Vai trò	Kết quả
		dùng	[user] WHERE id=@id
8	Admin dùng phân trang	Người dùng	Hệ thống thay đổi offset/trang hiển thị mới

### 3.7.3. Thêm người dùng (Create\_User.aspx)

Diễn giải luồng:



Hình 3.7.3: Sơ đồ luồng xử lý trang Create\_User.aspx

Ví dụ SQL được thực thi:

```

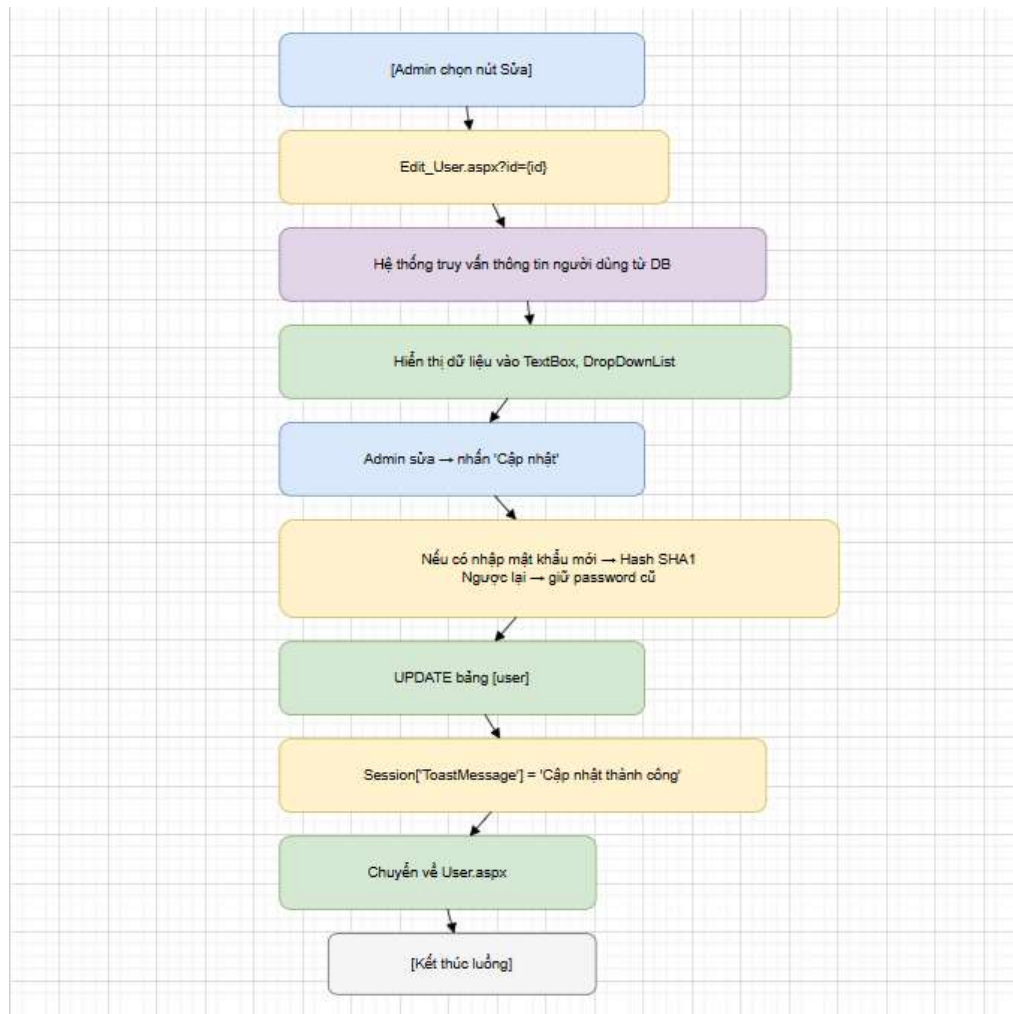
INSERT INTO [user] (username, password, email, role)
VALUES (@username, @password, @email, @role);
    
```

Mô tả thực tế:

- Admin nhập thông tin → Hệ thống mã hóa mật khẩu → Lưu vào DB → Hiển thị thông báo thành công.

### 3.7.4. Cập nhật người dùng (Edit\_User.aspx)

Diễn giải luồng:



Hình 3.7.4: Sơ đồ luồng xử lý trang Edit\_User.aspx

Ví dụ SQL được thực thi:

```
UPDATE [user]
SET username=@username,
    email=@email,
    role=@role,
    updated_at=GETDATE()
WHERE id=@id;
```

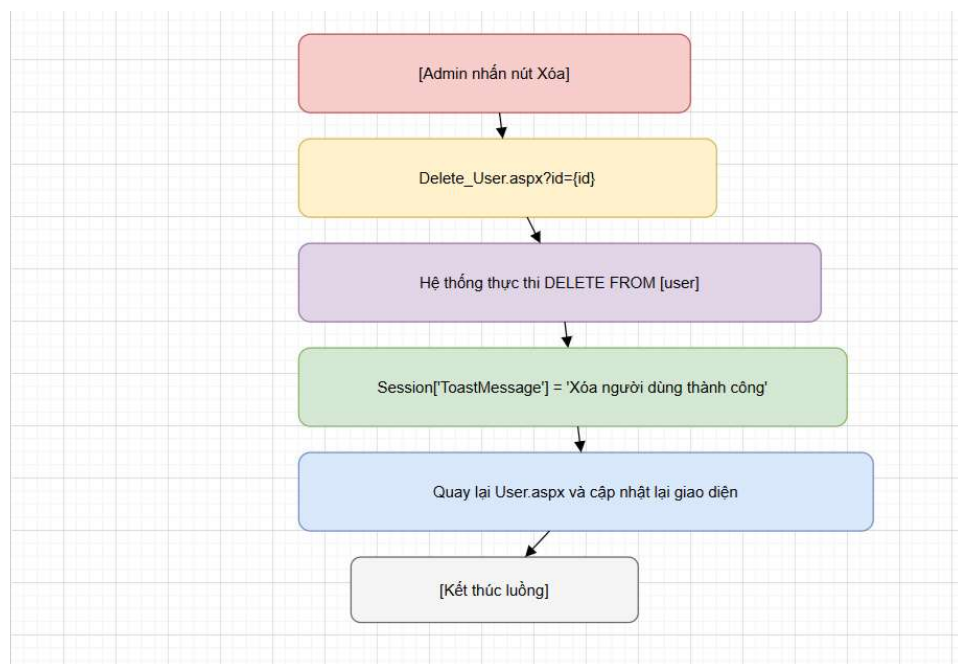
Hoặc nếu có đổi mật khẩu:

```
UPDATE [user]
```

```
SET username=@username,  
    email=@email,  
    password=@password,  
    role=@role,  
    updated_at=GETDATE()  
WHERE id=@id;
```

**Mô tả thực tế:**

- Hệ thống chỉ mã hóa mật khẩu mới nếu Admin nhập, tránh ghi đè rỗng.

**3.7.5. Xóa người dùng (Delete\_User.aspx)****Diễn giải luồng:**

Hình 3.7.5: Sơ đồ luồng xử lý trang Delete\_User.aspx

**Ví dụ SQL được thực thi:**

```
DELETE FROM [user] WHERE id=@id;
```

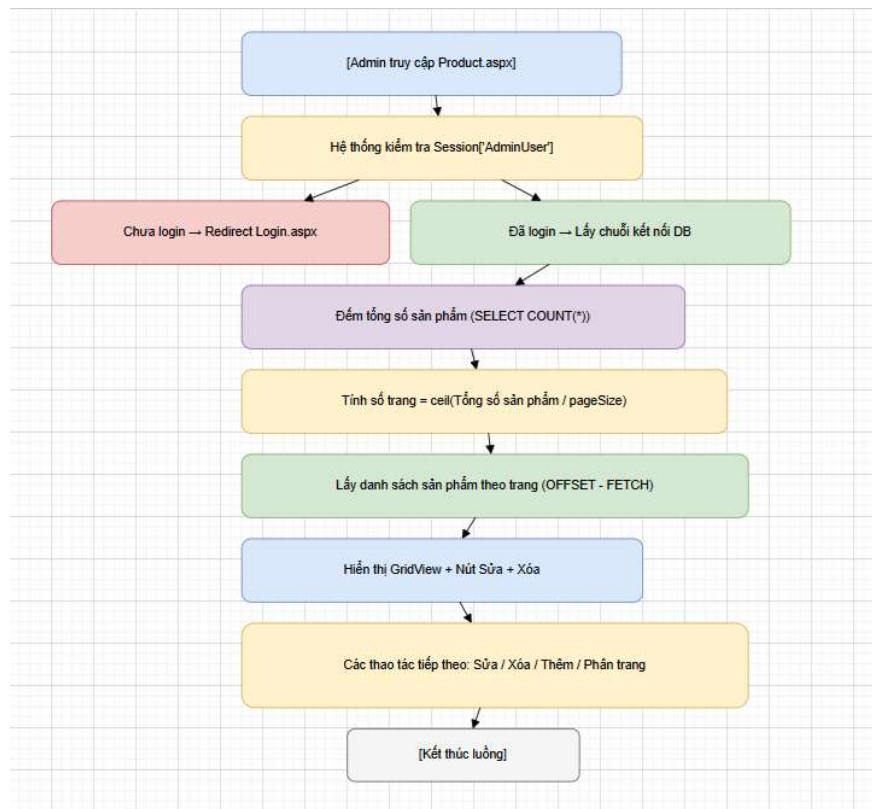
**Mô tả thực tế:**

- Sau khi xóa, hệ thống hiển thị toast "Xóa thành công" và refresh danh sách.

### 3.8. Luồng xử lý Quản lý Sản phẩm (Product.aspx + Create + Edit + Delete)

#### 3.8.1 Hiện thị danh sách sản phẩm (Product.aspx)

##### 3.8.1.1. Sơ đồ luồng xử lý (Flow Diagram)



Hình 3.8.1: Sơ đồ luồng xử lý trang Product.aspx

##### 3.8.1.2. Truy vấn dữ liệu hiển thị

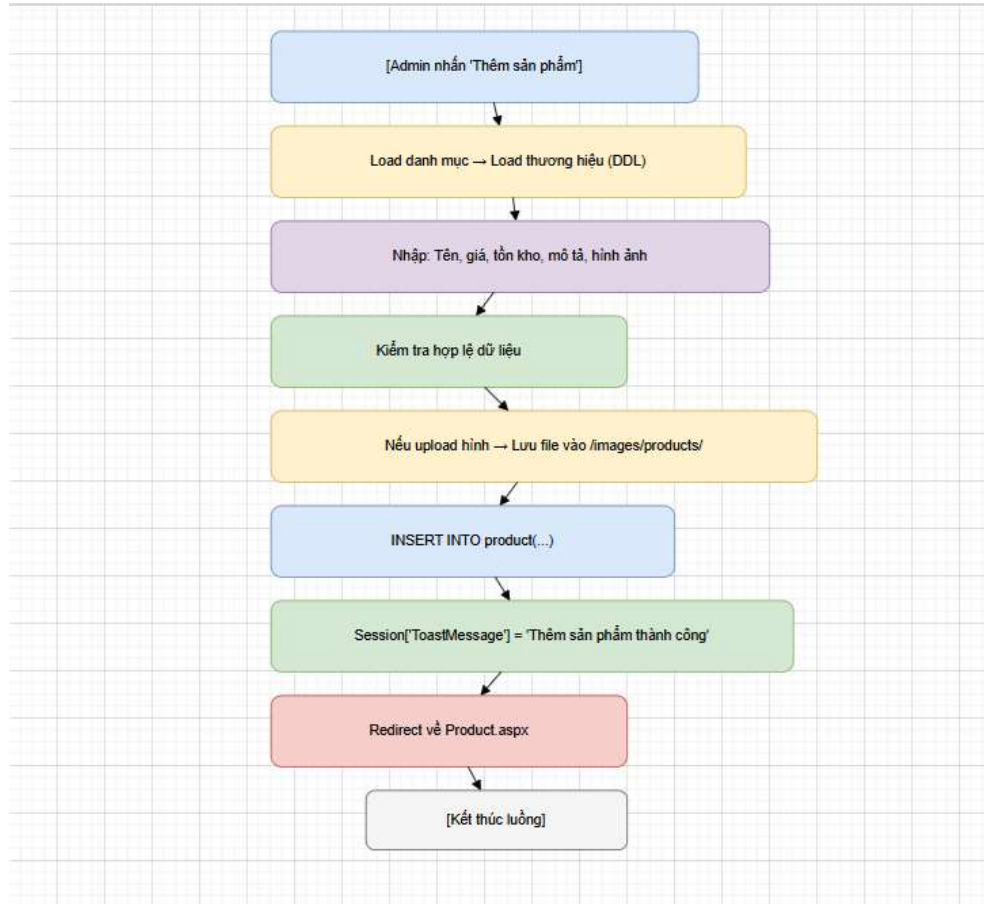
```

SELECT p.*, b.name AS brand_name
FROM product p
LEFT JOIN brand b ON p.brand_id = b.id
ORDER BY p.id DESC
OFFSET @Offset ROWS
FETCH NEXT @PageSize ROWS ONLY;

```

#### 3.8.2. Thêm sản phẩm (Create\_Product.aspx)

### 3.8.2.1. Luồng xử lý



Hình 3.8.2: Sơ đồ luồng xử lý trang *Create\_Product.aspx*

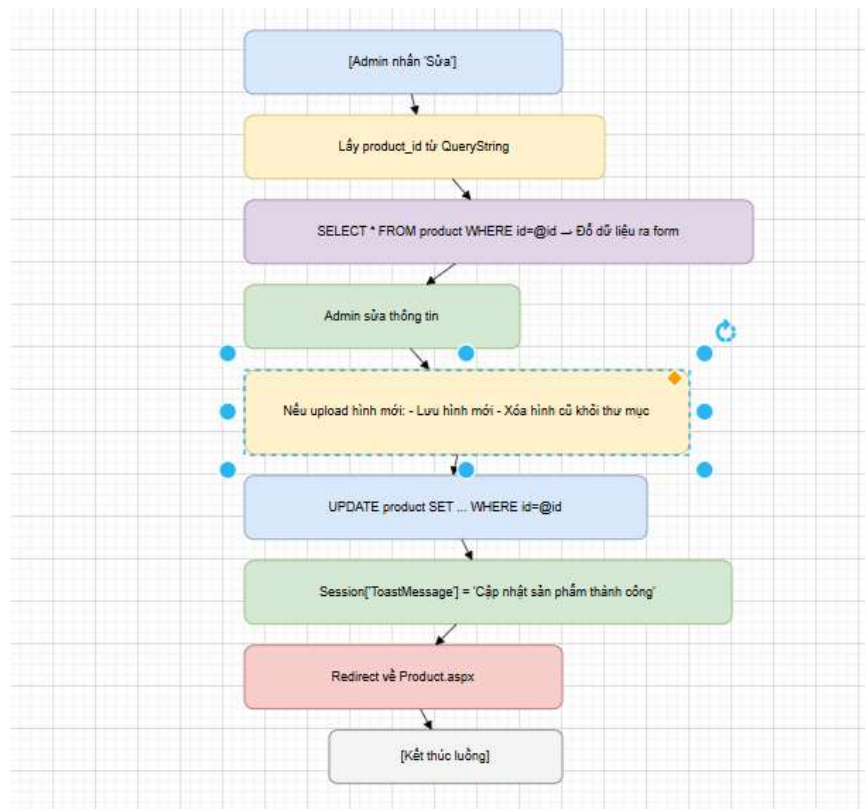
### 3.8.2.2. Câu lệnh SQL sử dụng

```
INSERT INTO product (category_id, brand_id, name, description, price, stock,  
image_url, created_at, updated_at)
```

```
VALUES (@category_id, @brand_id, @name, @description, @price, @stock,  
@image_url, GETDATE(), GETDATE());
```

### 3.8.3. Sửa sản phẩm (Edit\_Product.aspx)

#### 3.8.3.1. Luồng xử lý



Hình 3.8.3: Sơ đồ luồng xử lý trang Edit\_Product.aspx

### 3.8.3.2. Câu lệnh SQL update

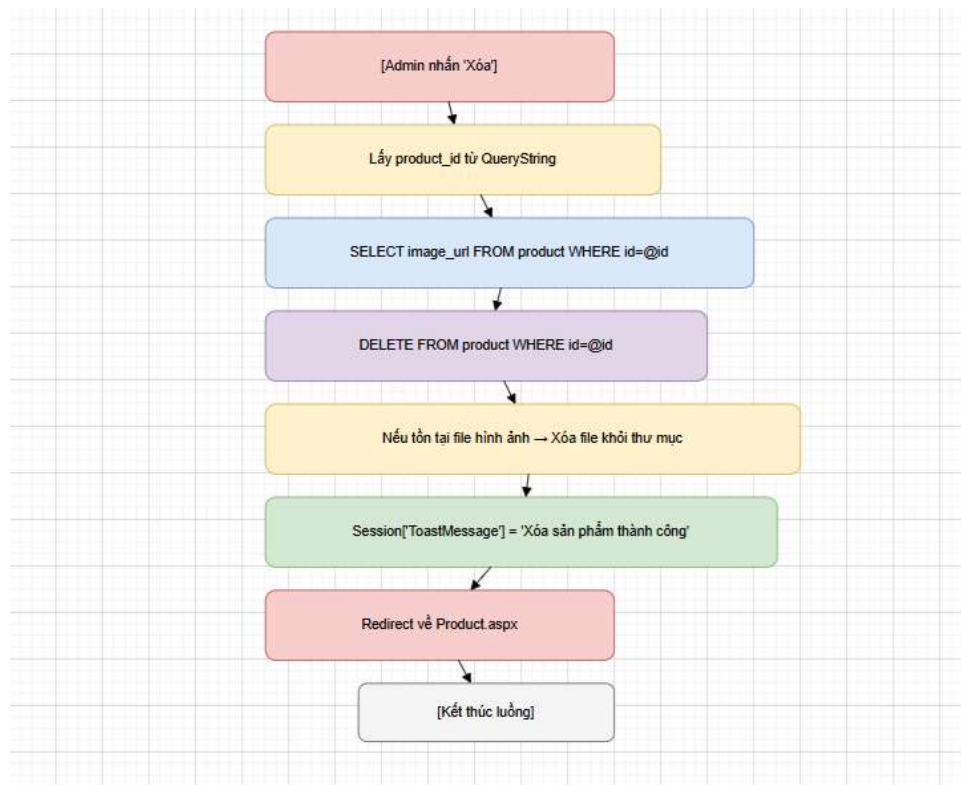
```

UPDATE product SET
  category_id=@category_id,
  brand_id=@brand_id,
  name=@name,
  description=@description,
  price=@price,
  stock=@stock,
  updated_at=GETDATE(),
  image_url = @image_url (nếu có)
WHERE id=@id;
  
```

### 3.8.4. Xóa sản phẩm (Delete\_Product.aspx)

#### 3.8.4.1. Luồng xử lý





Hình 3.8.4: Sơ đồ luồng xử lý trang Delete\_Product.aspx

#### 3.8.4.2. SQL xóa

DELETE FROM product WHERE id=@id;

### 3.9. Chức năng Quản lý Danh mục (Category Management)

Chức năng Quản lý Danh mục cho phép quản trị viên quản lý các loại phân nhóm sản phẩm như: Laptop Gaming, Ultrabook, Laptop Văn phòng, v.v.

Việc tổ chức danh mục hợp lý giúp người dùng dễ tìm kiếm sản phẩm và giúp người quản trị dễ kiểm soát dữ liệu hơn.

#### 3.9.1. Hiện thị danh mục

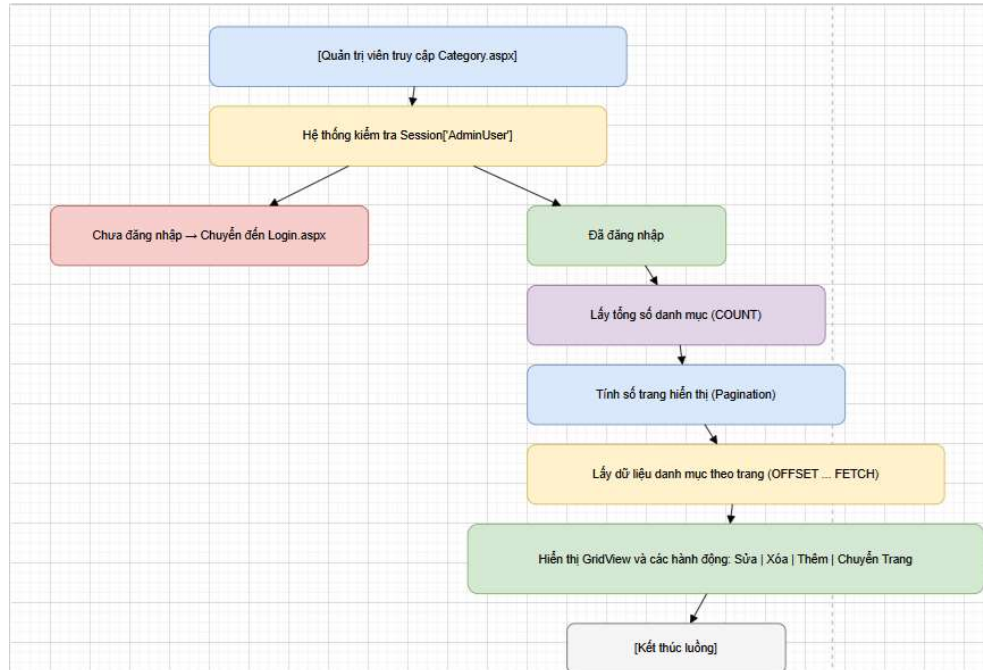
##### 3.9.1.1. Mô tả chức năng

Trang **Category.aspx** có chức năng:

- Hiện thị danh sách danh mục (category) theo dạng bảng (GridView).
- Có phân trang để tránh hiện thị quá nhiều dữ liệu cùng lúc.

- Cho phép **Sửa** hoặc **Xóa** từng danh mục.
- Có nút **Thêm danh mục mới**.

### 3.9.1.2. Luồng xử lý



Hình 3.9.1: Sơ đồ luồng xử lý trang Category.aspx

### 3.9.1.3. Câu lệnh SQL sử dụng

```

SELECT id, category_name, created_at, updated_at
FROM category
ORDER BY id DESC
OFFSET @Offset ROWS FETCH NEXT @PageSize ROWS ONLY;

```

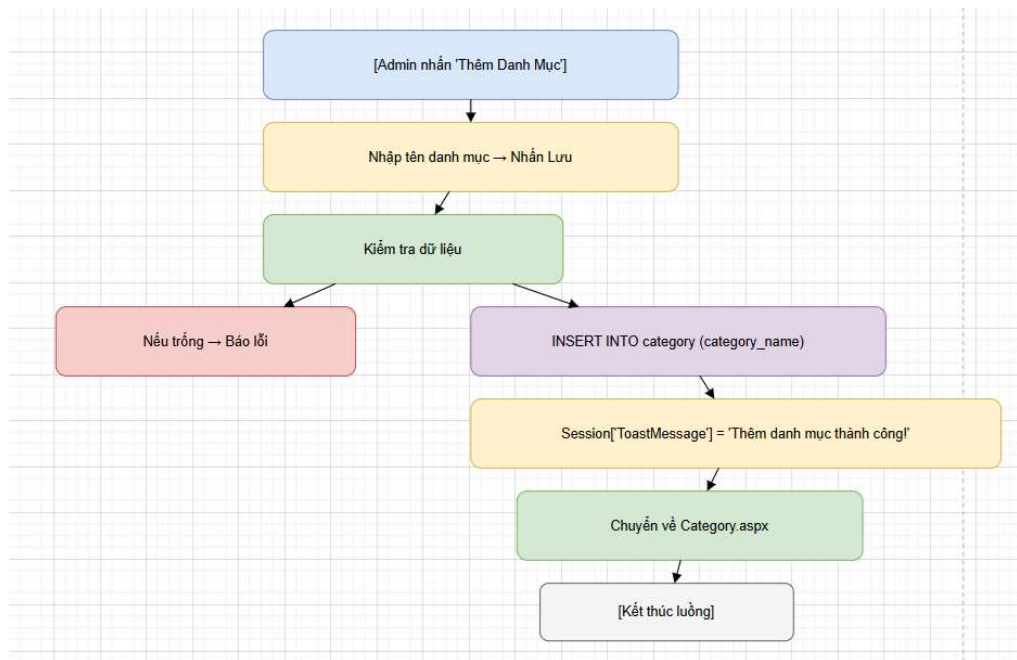
## 3.9.2. Thêm danh mục (Create\_Category.aspx)

### 3.9.2.1. Mô tả chức năng

- Quản trị viên nhập tên danh mục mới
- Hệ thống kiểm tra hợp lệ
- Lưu vào bảng category
- Hiện thị thông báo và quay lại danh sách



### 3.9.2.2. Luồng xử lý



Hình 3.9.2: Sơ đồ luồng xử lý trang Create\_Category.aspx

### 3.9.2.3. SQL thực thi

```

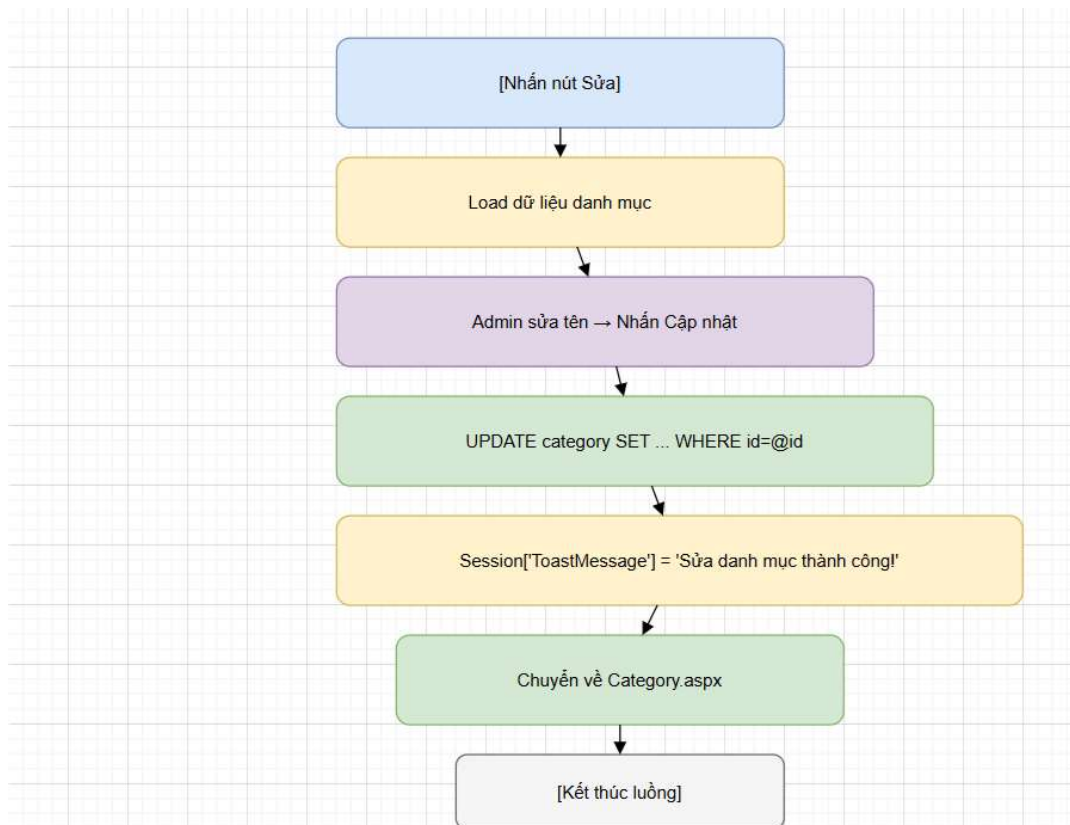
INSERT INTO category (category_name)
VALUES (@category_name);
    
```

## 3.9.3. Sửa danh mục (Edit\_Category.aspx)

### 3.9.3.1. Mô tả chức năng

- Lấy thông tin danh mục theo id từ QueryString.
- Quản trị viên sửa tên danh mục và lưu lại.

### 3.9.3.2. Luồng xử lý



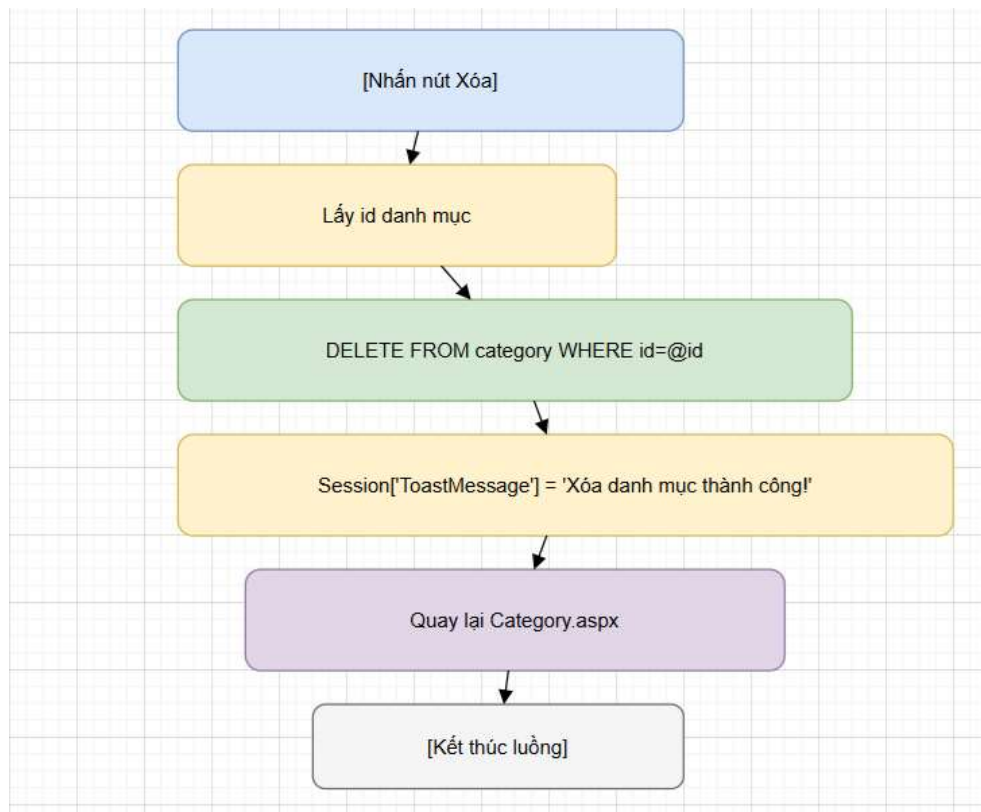
Hình 3.9.3: Sơ đồ luồng xử lý trang *Edit\_Category.aspx*

### 3.9.3.3. SQL thực thi

```
UPDATE category  
SET category_name=@category_name, updated_at=GETDATE()  
WHERE id=@id;
```

### 3.9.4. Xóa danh mục (Delete\_Category.aspx)

#### 3.9.4.1. Luồng xử lý



Hình 3.9.4: Sơ đồ luồng xử lý trang Delete\_Category.aspx

#### 3.9.4.2. SQL thực thi

DELETE FROM category WHERE id=@id;

### 3.10. Chức năng Quản lý Thương hiệu (Brand Management)

Chức năng Quản lý Thương hiệu giúp quản trị viên quản lý các hãng laptop như: Asus, Dell, HP, Lenovo, MSI,...

Việc phân loại sản phẩm theo thương hiệu giúp tăng tính rõ ràng khi hiển thị và lọc sản phẩm.

#### 3.10.1. Hiển thị danh sách thương hiệu

##### 3.10.1.1. Mô tả chức năng

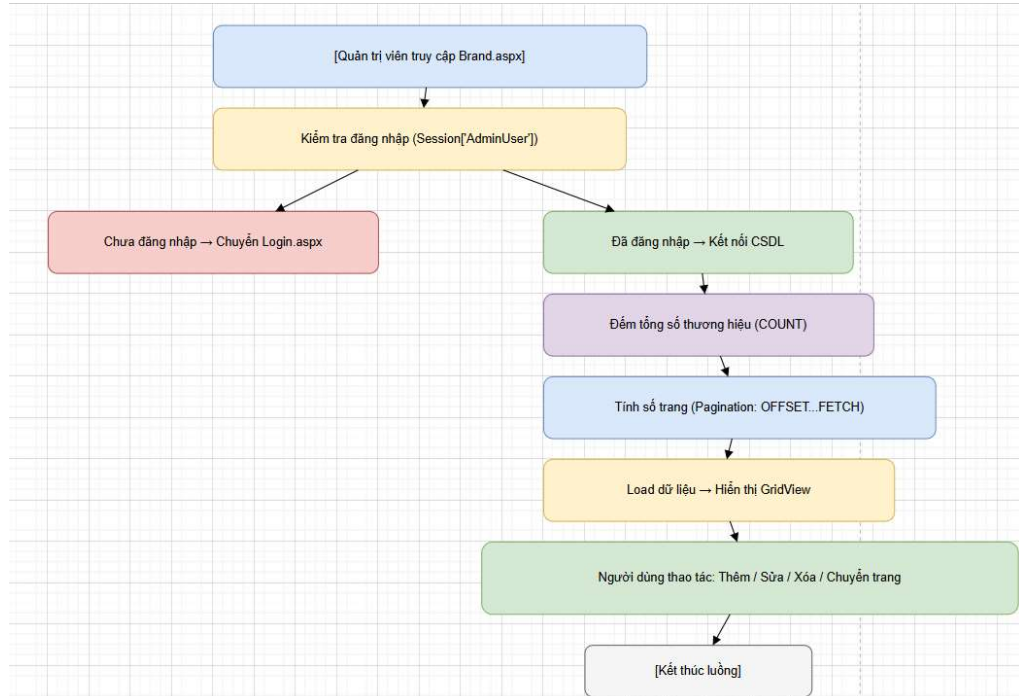
Trang **Brand.aspx** có nhiệm vụ:

- Hiển thị danh sách thương hiệu hiện có trên hệ thống.
- Phân trang danh sách để dễ quan sát.

- Cho phép chuyển hướng đến **Thêm, Sửa, và Xóa** thương hiệu.

Có cột hiển thị **logo** và cờ **Thương hiệu nổi bật (is\_top)**.

### 3.10.1.2. Luồng xử lý dữ liệu



Hình 3.10.1: Sơ đồ luồng xử lý trang Brand.aspx

### 3.10.1.3. Câu lệnh SQL truy vấn phân trang

```

SELECT id, name, logo_url, description, is_top
FROM brand
ORDER BY id DESC
OFFSET @Offset ROWS FETCH NEXT @PageSize ROWS ONLY;

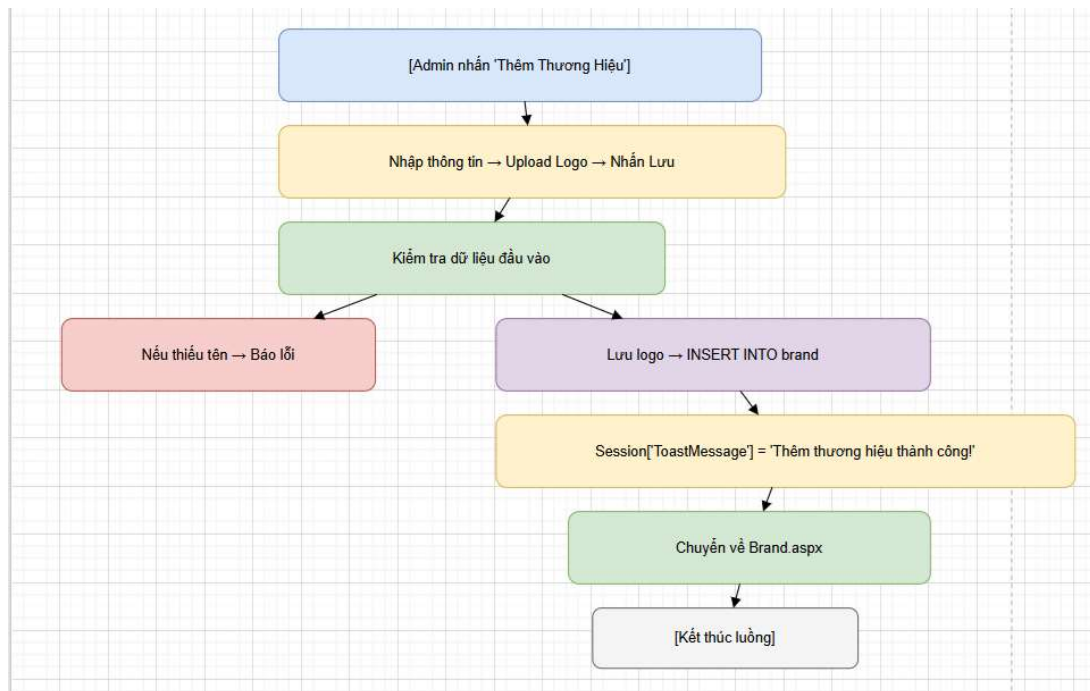
```

## 3.10.2. Thêm thương hiệu (Create\_Brand.aspx)

### 3.10.2.1. Mô tả chức năng

- Quản trị viên nhập tên thương hiệu, mô tả, chọn logo và đánh dấu thương hiệu nổi bật.
- Hệ thống lưu thông tin vào bảng brand.

### 3.10.2.2. Luồng xử lý



Hình 3.10.2: Sơ đồ luồng xử lý trang *Create\_Brand.aspx*

### 3.10.2.3. SQL thêm dữ liệu

```

INSERT INTO brand (name, logo_url, description, is_top)
VALUES (@name, @logo_url, @description, @is_top);

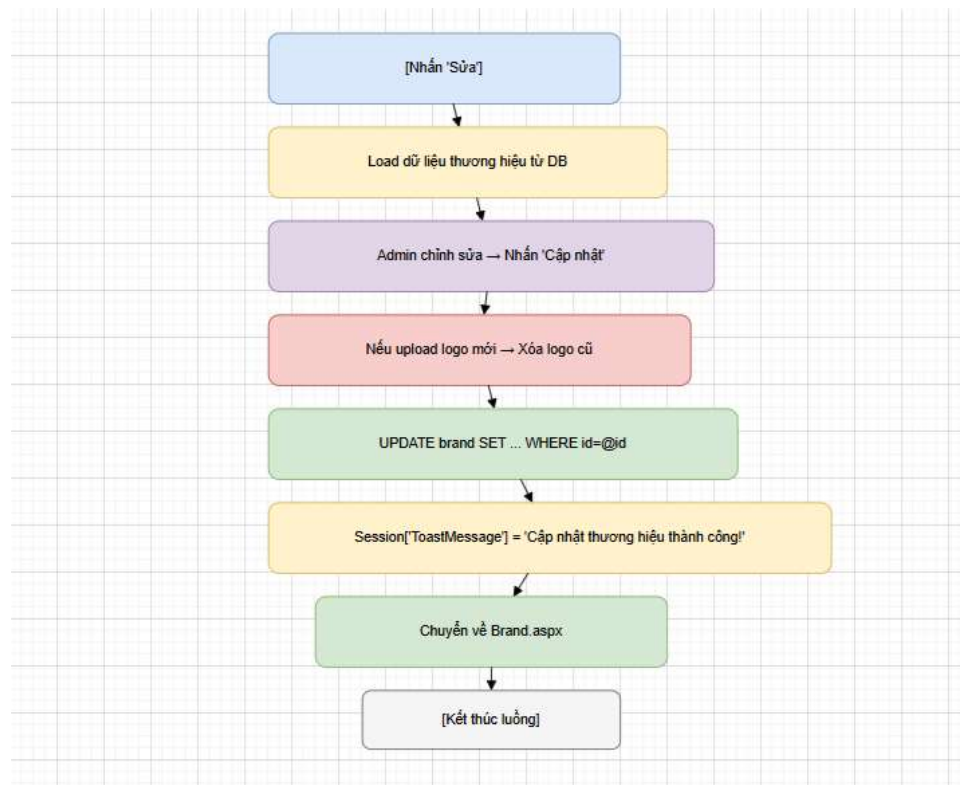
```

### 3.10.3. Sửa thương hiệu (Edit\_Brand.aspx)

#### 3.10.3.1. Mô tả chức năng

- Hiển thị thông tin thương hiệu hiện có.
- Cho phép chỉnh sửa tên, mô tả, logo, trạng thái nổi bật.

#### 3.10.3.2. Luồng xử lý



Hình 3.10.3: Sơ đồ luồng xử lý trang Edit\_Brand.aspx

### 3.10.3.3. SQL cập nhật

UPDATE brand

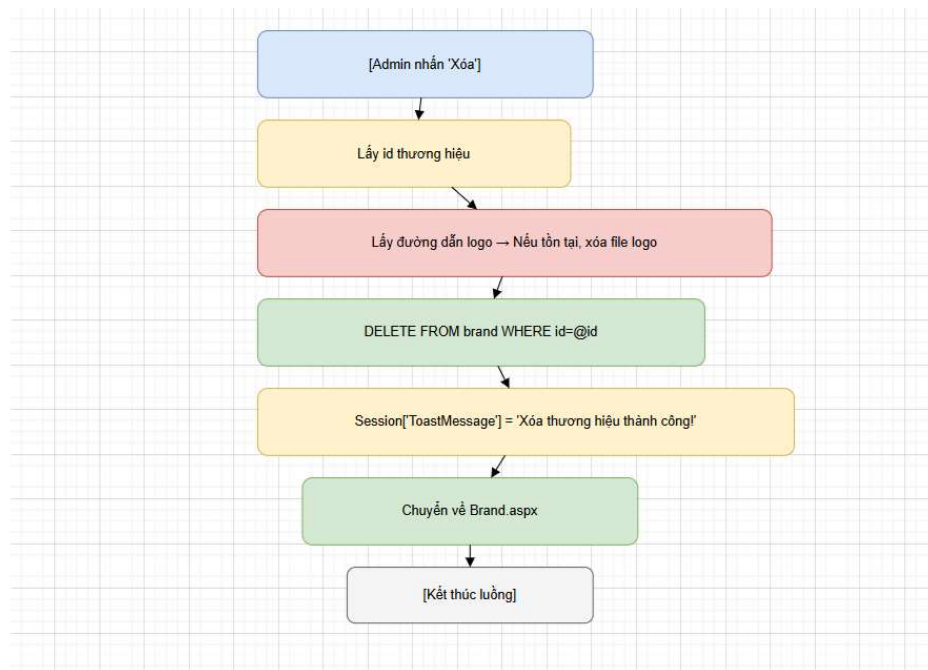
SET name=@name, description=@description, is\_top=@is\_top,

logo\_url=@logo\_url

WHERE id=@id;

### 3.10.4. Xóa thương hiệu (Delete\_Brand.aspx)

#### 3.10.4.1. Luồng xử lý



Hình 3.10.4: Sơ đồ luồng xử lý trang Delete\_Brand.aspx

#### 3.10.4.2. SQL xóa

DELETE FROM brand WHERE id=@id;

### 3.11. Chức năng Giỏ hàng và Thanh toán

Chức năng Giỏ hàng và Thanh toán là một phần quan trọng của hệ thống bán hàng trực tuyến.

Người dùng có thể thêm sản phẩm vào giỏ, cập nhật số lượng, xóa sản phẩm và tiến hành thanh toán để tạo đơn hàng.

#### 3.11.1. Chức năng Giỏ hàng (Cart.aspx)

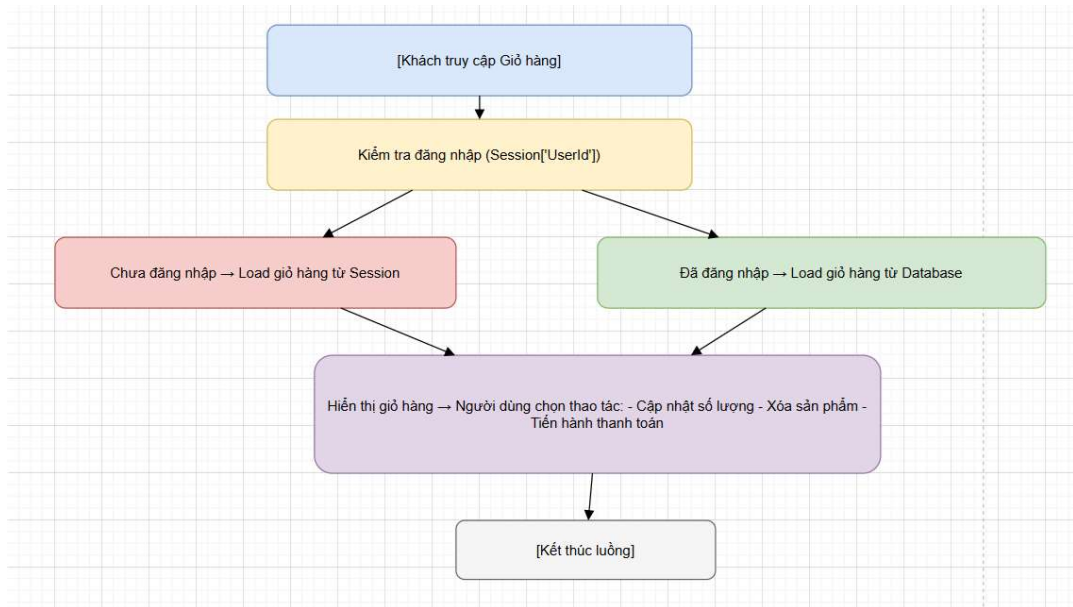
##### 3.11.1.1. Mô tả chức năng

- Hiện thị danh sách sản phẩm mà người dùng đã thêm vào giỏ.
- Nếu người dùng chưa đăng nhập → lưu giỏ tại Session.
- Nếu người dùng đã đăng nhập → giỏ hàng được lưu và đồng bộ trong cơ sở dữ liệu (cart, cart\_item).
- Cho phép:



- Cập nhật số lượng từng sản phẩm.
- Xóa sản phẩm khỏi giỏ.
- Tính tổng tiền, VAT và tổng thanh toán.

### 3.11.1.2. Luồng xử lý dữ liệu



Hình 3.11.1: Sơ đồ luồng xử lý trang Cart.aspx

### 3.11.1.3. Ví dụ SQL xử lý giỏ hàng từ CSDL

#### Lấy danh sách sản phẩm trong giỏ:

```

SELECT p.id, p.name, p.image_url, ci.price_at_time, ci.quantity
FROM cart_item ci
JOIN cart c ON ci.cart_id = c.id
JOIN product p ON ci.product_id = p.id
WHERE c.user_id = @user_id AND c.is_checked_out = 0;
    
```

#### Cập nhật số lượng sản phẩm trong giỏ:

```

UPDATE ci
SET quantity = @qty
FROM cart_item ci
INNER JOIN cart c ON ci.cart_id = c.id
    
```



```
WHERE c.user_id = @user_id AND ci.product_id = @product_id AND  
c.is_checked_out = 0;
```

**Xóa sản phẩm khỏi giỏ:**

```
DELETE ci  
FROM cart_item ci  
INNER JOIN cart c ON ci.cart_id = c.id  
WHERE c.user_id = @user_id AND ci.product_id = @product_id AND  
c.is_checked_out = 0;
```

**3.11.2. Kiểm tra tồn kho trước khi thanh toán****3.11.2.1. Mô tả**

Trước khi chuyển sang trang thanh toán, hệ thống thực hiện kiểm tra số lượng hàng tồn kho khả dụng tại thời điểm mua (tránh trường hợp oversell).

**3.11.2.2. Ví dụ SQL kiểm tra tồn kho thực tế**

```
SELECT  
    p.id,  
    p.stock - ISNULL(SUM(oi.quantity), 0) AS available_stock  
FROM product p  
LEFT JOIN order_item oi ON oi.product_id = p.id  
LEFT JOIN [order] o ON oi.order_id = o.id AND o.status IN ('pending', 'paid')  
WHERE p.id IN (...)  
GROUP BY p.id, p.stock;
```

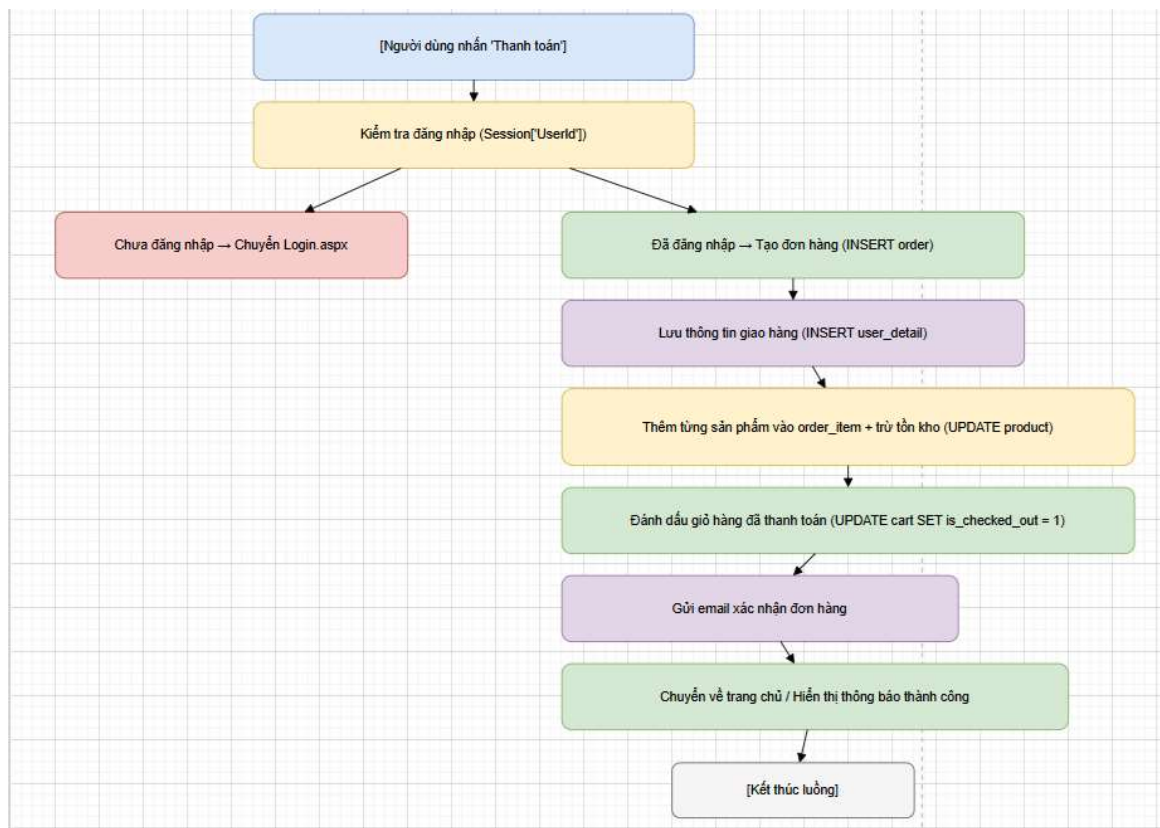
Nếu **available\_stock** < **số lượng trong giỏ** → cảnh báo sản phẩm thiếu.

**3.11.3. Chức năng Thanh toán (Checkout.aspx)****3.11.3.1. Mô tả chức năng**

- Người dùng nhập địa chỉ, mã bưu chính, phương thức thanh toán.
- Hệ thống tạo đơn hàng (**order**).

- Tạo chi tiết người nhận (**user\_detail**).
- Ghi từng sản phẩm tương ứng vào bảng **order\_item**.
- Trừ tồn kho sản phẩm.
- Đánh dấu giỏ hàng đã thanh toán.
- Gửi email xác nhận đơn hàng.

### 3.11.3.2. Luồng xử lý dữ liệu



Hình 3.11.3: Sơ đồ luồng xử lý trang Checkout.aspx

### 3.11.3.3. SQL chính trong quá trình thanh toán

#### Tạo đơn hàng:

```

INSERT INTO [order] (user_id, total_amount)
OUTPUT INSERTED.id
VALUES (@user_id, 0);
    
```

**Thêm vào order\_item và trừ tồn kho:**

```
INSERT INTO order_item (order_id, product_id, quantity, price)
VALUES (@order_id, @product_id, @quantity, @price);
```

```
UPDATE product SET stock = stock - @quantity WHERE id = @product_id;
```

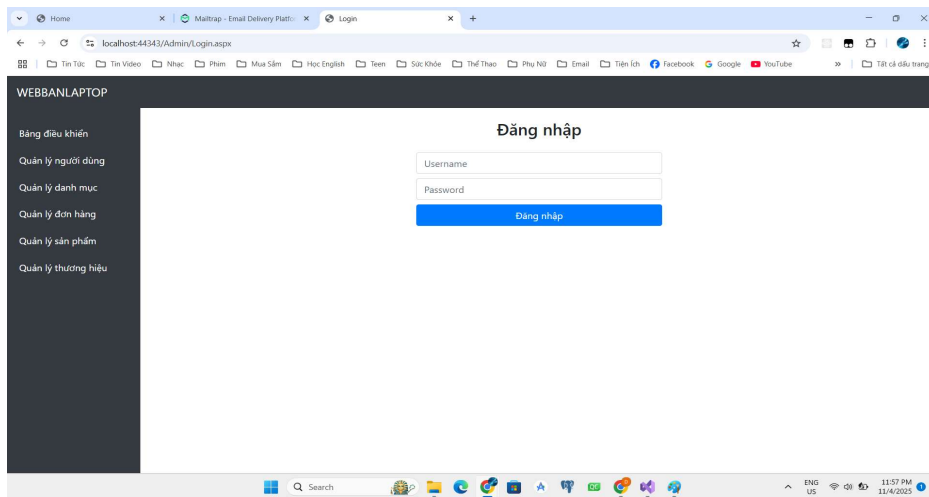
**Đánh dấu giỏ hàng đã thanh toán:**

```
UPDATE cart SET is_checked_out = 1, updated_at = GETDATE()
WHERE user_id = @uid AND is_checked_out = 0;
```

**3.11.4. Gửi Email Xác Nhận**

Sau khi thanh toán thành công, hệ thống gửi email tự động cho khách hàng kèm thông tin đơn hàng (HTML Template).

## CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU

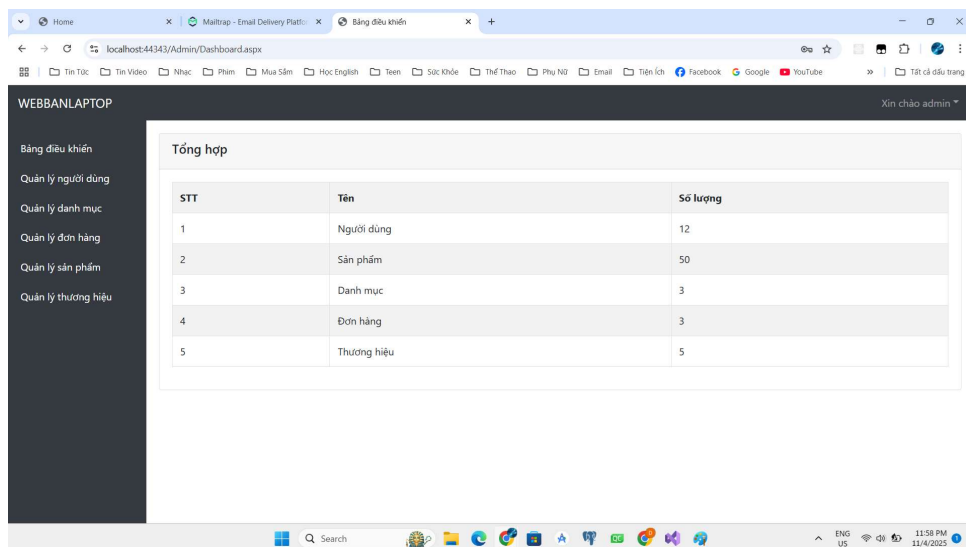


Hình 4.1. Giao diện đăng nhập

### Mô tả:

Trang đăng nhập dành riêng cho quản trị viên. Người dùng cần nhập **Username** và **Password** hợp lệ mới có thể truy cập trang quản trị.

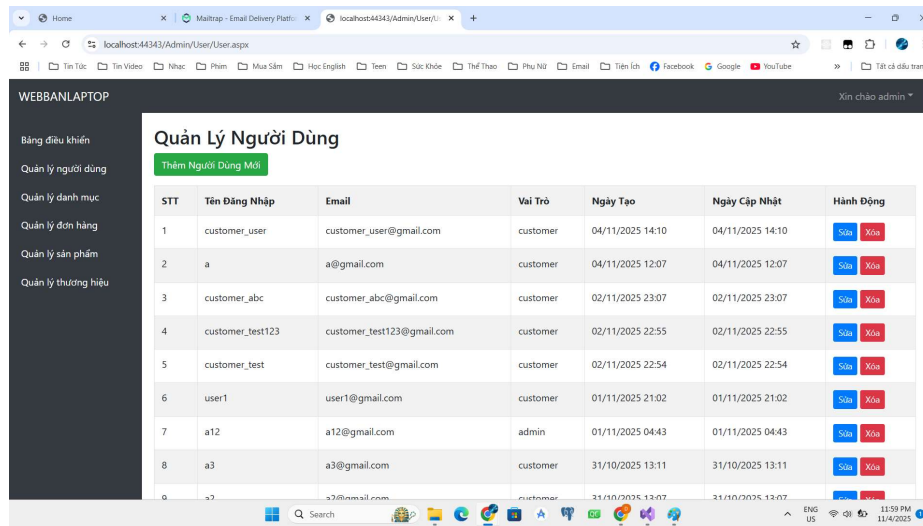
Chức năng giúp bảo vệ hệ thống khỏi truy cập trái phép.



Hình 4.2. Giao diện trang quản trị

### Mô tả:

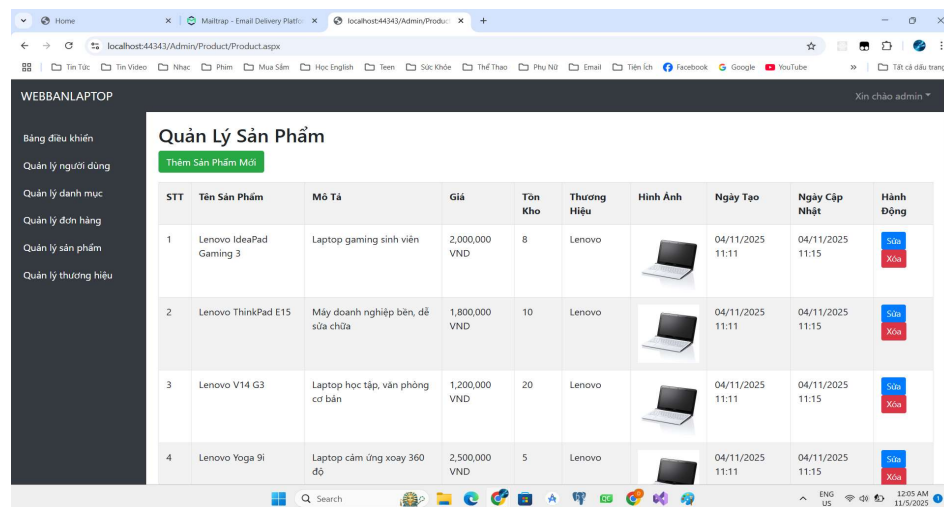
Hiển thị tổng số User, Sản phẩm, Đơn hàng, Danh mục, Thương hiệu → giúp quản trị viên theo dõi hệ thống trực quan.



Hình 4.3. Giao diện quản lý người dùng

### Mô tả:

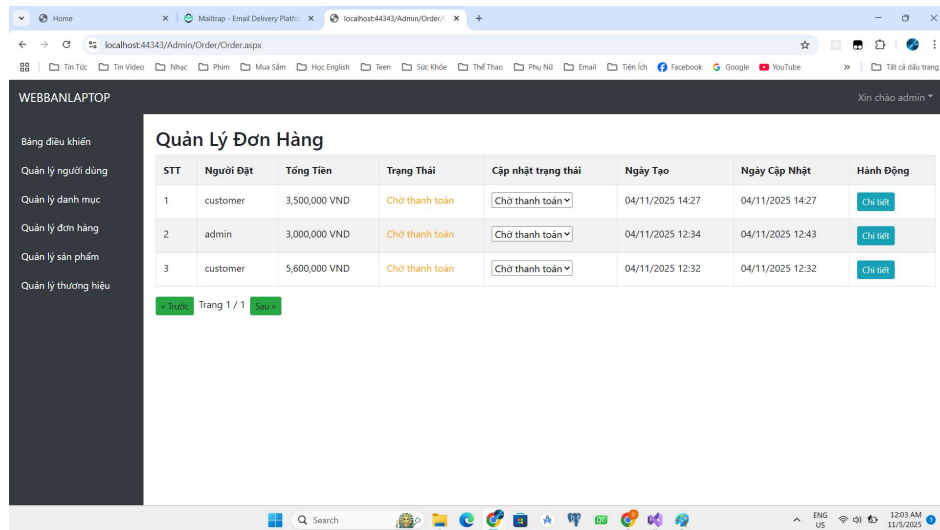
- Quản trị viên có thể phân quyền (admin / customer).
- Hỗ trợ chỉnh sửa và xóa người dùng.



Hình 4.4. Giao diện quản lý sản phẩm

### Mô tả:

- Hiện thị danh sách sản phẩm kèm giá, tồn kho, thương hiệu.
- Cho phép chỉnh sửa chi tiết sản phẩm và xóa nếu cần.

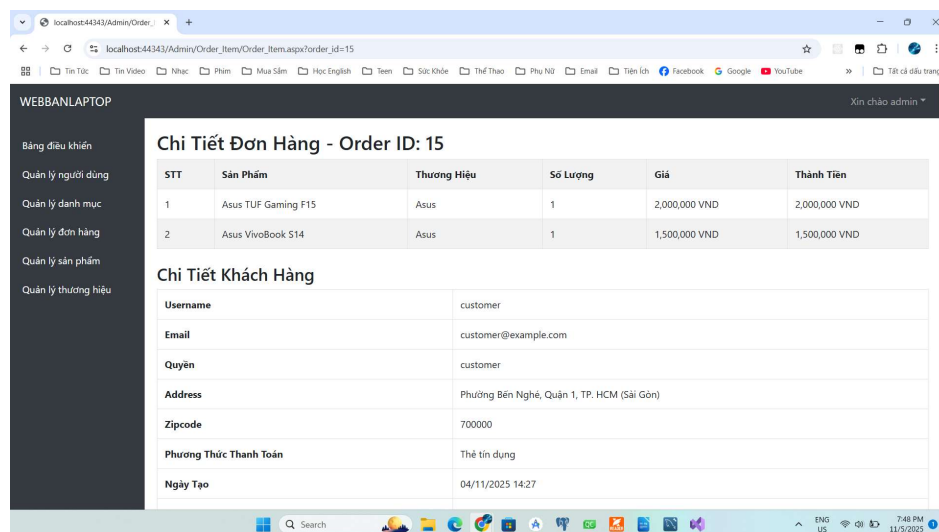


Hình 4.5. Giao diện quản lý đơn hàng

**Mô tả:**

Quản trị viên có thể thay đổi trạng thái đơn hàng (chờ, đã thanh toán, hủy...).

Hỗ trợ xem chi tiết từng đơn.



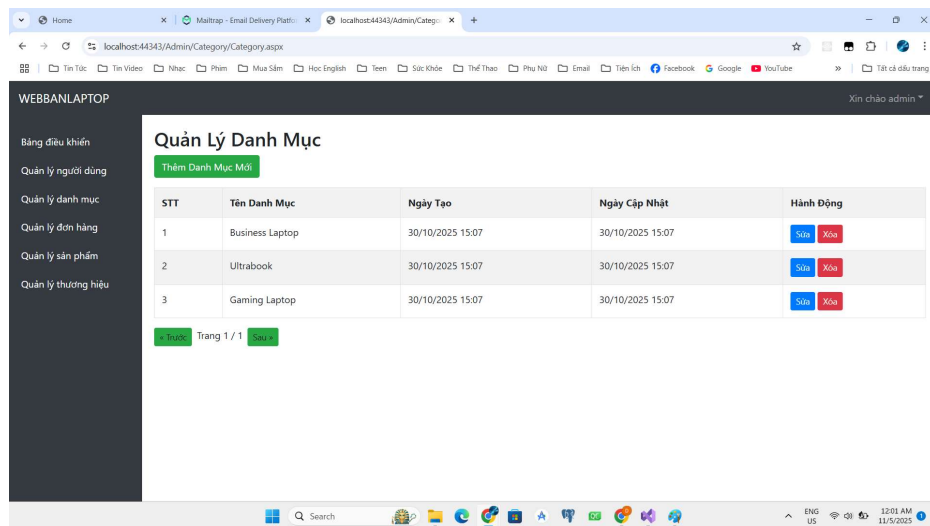
Hình 4.6. Giao diện chi tiết đơn hàng

**Mô tả:**

Hiển thị thông tin sản phẩm trong đơn hàng.

Xem thông tin khách hàng và phương thức thanh toán.

Giúp dễ dàng kiểm soát trạng thái đơn hàng.

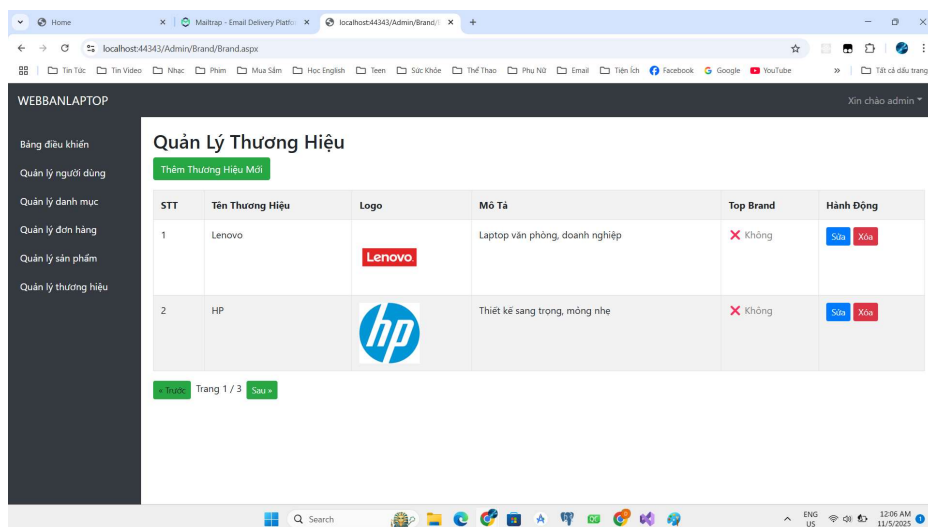


Hình 4.7. Giao diện quản lý danh mục

### Mô tả:

Quản trị viên có thể tạo các danh mục như: Gaming Laptop, Ultrabook, Business Laptop...

Hỗ trợ **phân trang** và cập nhật nhanh.



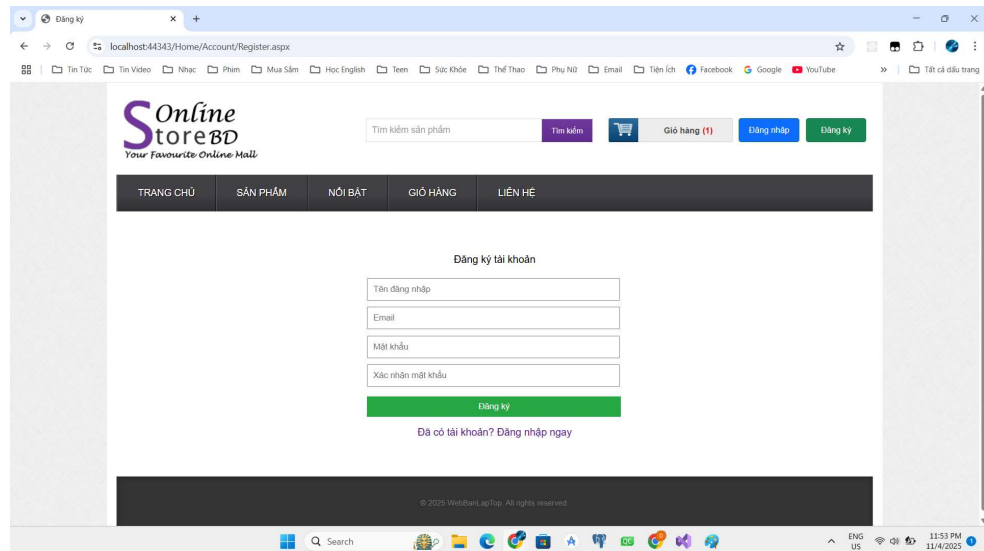
Hình 4.8. Giao diện quản lý thương hiệu

### Mô tả:

Hiển thị danh sách các thương hiệu laptop.

Cho phép **Thêm / Sửa / Xóa** thương hiệu.

Hỗ trợ upload logo và đánh dấu **Top Brand**.

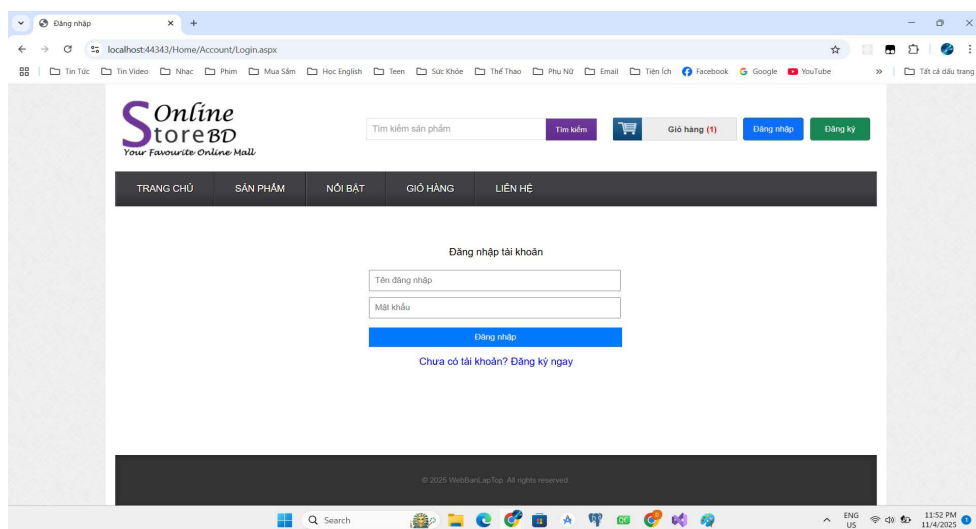


Hình 4.9. Giao diện đăng ký

### Mô tả:

Giao diện cho phép người dùng tạo tài khoản mới bằng cách nhập tên đăng nhập, email, mật khẩu và xác nhận mật khẩu.

Hệ thống kiểm tra tính hợp lệ và lưu thông tin vào cơ sở dữ liệu.



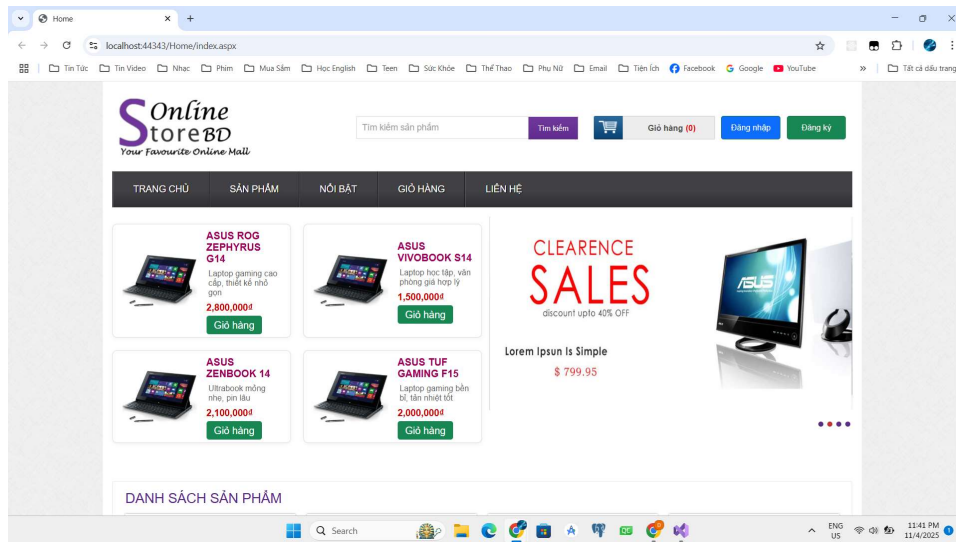
Hình 4.10. Giao diện đăng nhập (trang chủ)

### Mô tả:

Người dùng nhập tài khoản và mật khẩu. Hệ thống xác thực và phân quyền (admin / customer).

Nếu sai thông tin, hiển thị thông báo lỗi.



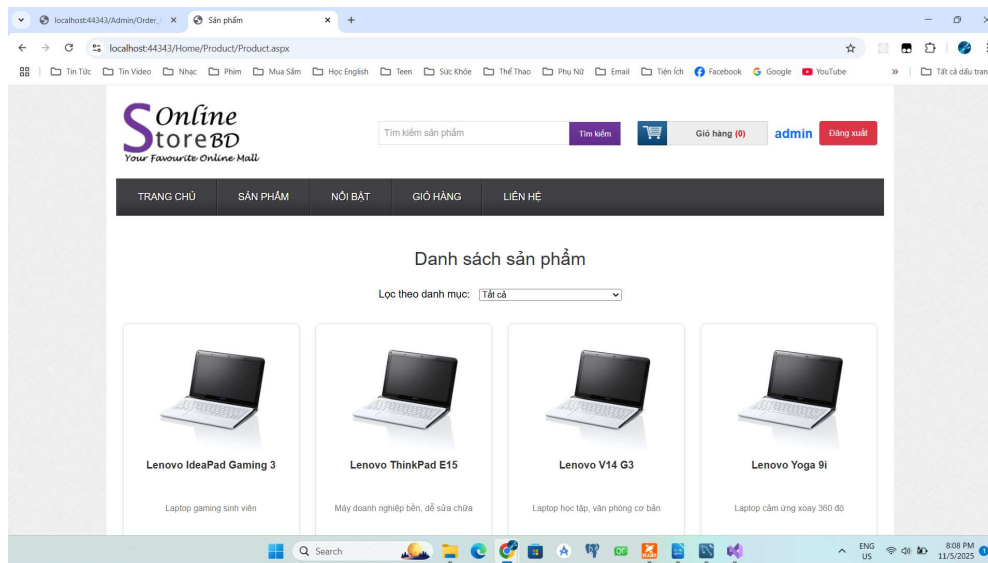


Hình 4.11. Giao diện trang chủ

### Mô tả:

Hiện thị danh sách sản phẩm nổi bật, giá, hình ảnh, nút thêm giỏ hàng.

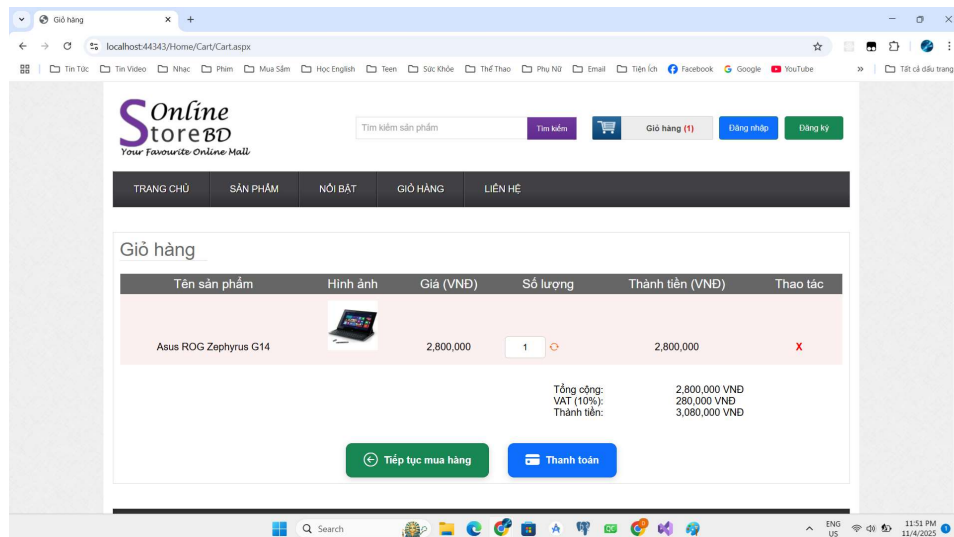
Người dùng có thể tìm kiếm và chuyển sang trang chi tiết sản phẩm.



Hình 4.12. Giao diện danh sách sản phẩm

### Mô tả:

Hiện thị các sản phẩm

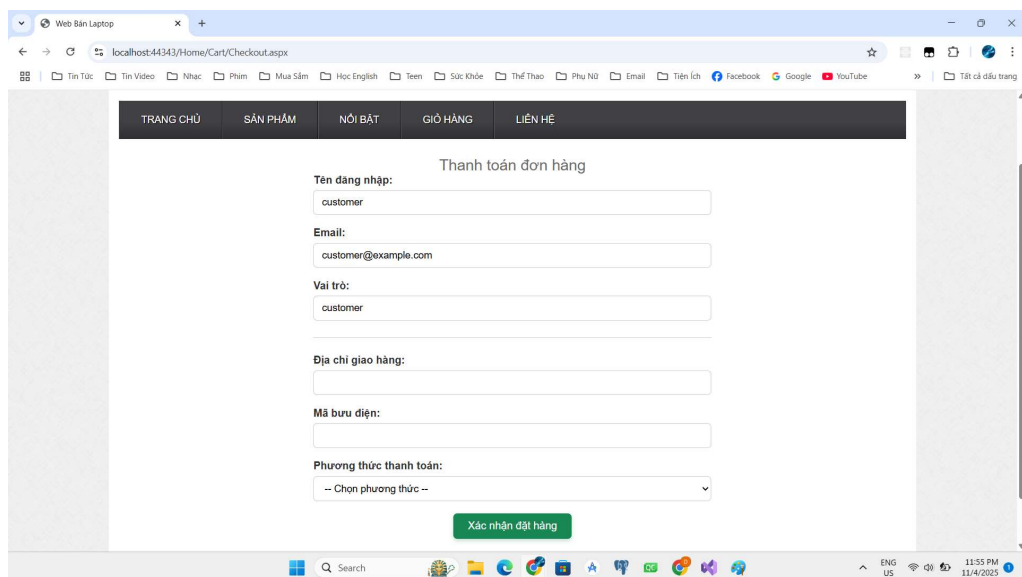


Hình 4.13. Giao diện giỏ hàng

### Mô tả:

Hiển thị các sản phẩm đã thêm: giá, hình ảnh, số lượng, tổng tiền, VAT.

Cho phép cập nhật số lượng hoặc xóa khỏi giỏ.

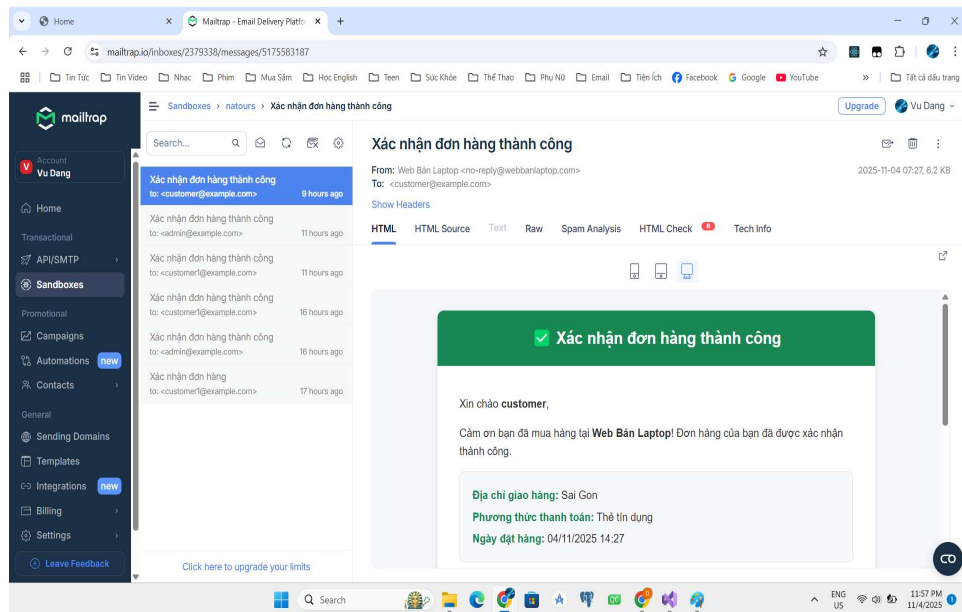


Hình 4.14. Giao diện trang thanh toán

### Mô tả:

Người dùng nhập thông tin nhận hàng và chọn phương thức thanh toán.

Sau khi xác nhận, hệ thống tạo đơn hàng và gửi email xác nhận.



Hình 4.15. Giao diện mail xác nhận đơn hàng

### Mô tả:

Trang email xác nhận đơn hàng hiển thị thông báo **“Xác nhận đơn hàng thành công”** gửi đến khách hàng sau khi thanh toán.

Email bao gồm thông tin: **địa chỉ giao hàng, phương thức thanh toán, ngày đặt hàng**, bảng danh sách sản phẩm.

## **CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

### **KẾT LUẬN**

Sau quá trình nghiên cứu và xây dựng hệ thống Website bán laptop, em đã hoàn thành các mục tiêu đề ra, bao gồm thiết kế và triển khai các chức năng quản lý danh mục, thương hiệu, sản phẩm, người dùng, giỏ hàng và thanh toán đơn hàng. Hệ thống cho phép người dùng dễ dàng đăng ký, đăng nhập, xem thông tin sản phẩm, thêm vào giỏ hàng và thực hiện đặt hàng. Đồng thời, phía quản trị viên có thể kiểm soát và cập nhật dữ liệu trong hệ thống một cách thuận tiện.

Hệ thống được xây dựng với giao diện rõ ràng, dễ sử dụng, cơ sở dữ liệu được tổ chức hợp lý, đảm bảo tính toàn vẹn và khả năng mở rộng. Kết quả thực hiện đáp ứng được yêu cầu ban đầu của đề tài.

### **HƯỚNG PHÁT TRIỂN**

Trong tương lai, hệ thống có thể được mở rộng và nâng cấp theo các hướng sau:

- Tích hợp thanh toán trực tuyến qua các cổng như Momo, VNPay, ZaloPay.
- Xây dựng API để hỗ trợ ứng dụng di động hoặc kết nối với hệ thống khác.
- Tối ưu hiệu năng để phục vụ lượng người dùng lớn.
- Bổ sung tính năng chat hỗ trợ khách hàng trực tiếp.
- Phát triển chức năng gợi ý sản phẩm dựa trên hành vi người dùng (AI/Recommendation System).

## DANH MỤC TÀI LIỆU THAM KHẢO

- [1] A. Troelsen and P. Japikse, *Pro C# and .NET Framework*, 9th ed. Apress, 2018.
- [2] D. Esposito, *ASP.NET Web Forms Modernization*. Microsoft Press, 2014.
- [3] A. Freeman, *Pro ASP.NET Web API*. Apress, 2017.
- [4] J. Kanjilal, *ASP.NET 4.5 in Practice*. Packt Publishing, 2013.
- [5] Microsoft, “ASP.NET Web Forms Overview,” *Microsoft Learn*,  
<https://learn.microsoft.com/en-us/aspnet/web-forms>
- [6] Microsoft, “ADO.NET SQL Server Data Access,” *Microsoft Learn*,  
<https://learn.microsoft.com/en-us/dotnet/framework/data/adonet>
- [7] W3Schools, “HTML, CSS, JavaScript Tutorial,” <https://www.w3schools.com>
- [8] Bootstrap Team, “Bootstrap v4.6 Documentation,” <https://getbootstrap.com/docs/4.6>
- [9] Mailtrap, “Email Testing During Development,” <https://mailtrap.io>
- [10] Microsoft, “Transact-SQL Language Reference,” *Microsoft Learn*,  
<https://learn.microsoft.com/en-us/sql/t-sql>