  
 **TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP.HCM**  
 **KHOA CÔNG NGHỆ THÔNG TIN**

**ĐỒ ÁN CHUYÊN NGHÀNH**

**Đề tài:**

**WEBSITE BÁN GIÀY**

**Giảng viên hướng dẫn :** **Dương Văn Trà**

**Sinh viên thực hiện:**

**MSSV:** **Họ và tên:** **Lớp:**

**2180606799** **Nguyễn Tấn Phong** **21DTHC2**

**2180603880** **Lê Minh Quân** **21DTHC2**

**2180604393** **Võ Tấn Qui** **21DTHC2**

**Lời cảm ơn**

Chúng em xin bày tỏ lòng kính trọng và biết ơn sâu sắc đến thầy Dương Văn Trà đã tận tình giúp đỡ và hướng dẫn chúng em hoàn thành đồ án chuyên ngành này. Nhờ sự giúp đỡ nhiệt lòng của thầy mà đồ án đã hoàn thành trong 10 tuần, đúng tiến độ của khoa và nhà trường đề ra.

**Mục Lục**

[CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI 3](#_Toc185105381)

[1.1. Lý do chọn đề tài 3](#_Toc185105382)

[1.2. Mục tiêu của đề tài. 4](#_Toc185105383)

[1.3. Khái niệm về website thương mại điện tử. 5](#_Toc185105384)

[CHƯƠNG 2. CƠ SỞ LÝ THUYẾT 6](#_Toc185105385)

[2.1 Tổng quan về .Net Core 6](#_Toc185105386)

[2.2 Đặc điểm nổi bật của .Net Core 6](#_Toc185105387)

[2.3 Thành phần của .Net Core 9](#_Toc185105388)

[2.4 Giới thiệu về Asp.Net Core. 9](#_Toc185105389)

[2.5 Cấu trúc hạ tầng Asp.Net Core 10](#_Toc185105390)

[CHƯƠNG 3. ĐẶC TẢ - THIẾT KẾ - XÂY DỰNG HỆ THỐNG 14](#_Toc185105391)

[3.1. Mô tả chức năng nghiệp vụ của hệ thống 14](#_Toc185105392)

[3.2. Các yêu cầu phi chức năng của website 15](#_Toc185105393)

[3.3. Phân tích các thành phần trong database. 15](#_Toc185105394)

[3.4. Sơ đồ DFD tổng quan. 20](#_Toc185105395)

[3.5. Sơ đồ Uses case tổng quát. 21](#_Toc185105396)

[Chương 4: DEMO 22](#_Toc185105397)

[4.1 Dưới góc nhìn khách hàng Trang chủ cửa hàng Sản phẩm trong cửa hàng 22](#_Toc185105398)

[4.2 Dưới góc nhìn admin 26](#_Toc185105399)

# CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI

* 1. **Lý do chọn đề tài**

A screenshot of a website

Description automatically generated

A collage of different colored shoes

Description automatically generated

Hình 1.1. Tham khảo trang bán giày hàng đầu

Chúng em chọn đề tài "website bán giày" cho báo cáo đồ án chuyên ngành vì nhiều lý do hợp lý và thiết thực. Ngành công nghiệp giày dép đang phát triển mạnh mẽ, với nhu cầu tiêu dùng ngày càng tăng. Việc xây dựng một website bán giày không chỉ giúp chúng em khai thác thị trường tiềm năng này mà còn đáp ứng xu hướng tiêu dùng hiện đại, đặc biệt trong bối cảnh thương mại điện tử đang trở nên phổ biến hơn bao giờ hết.

Website bán giày sẽ cho phép chúng em tiếp cận một đối tượng khách hàng đa dạng, từ trẻ em đến người lớn, từ giày thể thao đến giày thời trang. Điều này không chỉ tạo ra cơ hội kinh doanh mà còn cho phép tùy biến và cá nhân hóa trải nghiệm mua sắm của người tiêu dùng. Bên cạnh đó, việc tích hợp công nghệ hiện đại như trí tuệ nhân tạo và phân tích dữ liệu sẽ cải thiện trải nghiệm người dùng và tối ưu hóa quy trình bán hàng.

Ngoài ra, dự án này sẽ giúp nhóm chúng em phát triển nhiều kỹ năng quan trọng như lập trình web, thiết kế giao diện, quản lý cơ sở dữ liệu và marketing trực tuyến, những kỹ năng rất cần thiết cho sự nghiệp trong tương lai. Cuối cùng, việc xây dựng một website bán giày không chỉ là một nền tảng thương mại mà còn giúp quảng bá các thương hiệu và xu hướng thời trang mới, góp phần vào sự phát triển của ngành thời trang.

## Mục tiêu của đề tài.

Mục tiêu nghiên cứu của đồ án này là thiết kế và phát triển một website bán giày chuyên nghiệp, nhằm đáp ứng nhu cầu ngày càng cao của thị trường thương mại điện tử. Đầu tiên, chúng tôi sẽ xác định yêu cầu và tính năng cần thiết cho website, bao gồm khả năng tìm kiếm sản phẩm theo nhiều tiêu chí như size, màu sắc và loại giày, hiển thị hình ảnh sản phẩm chất lượng cao cùng với thông tin chi tiết, và tích hợp chức năng giỏ hàng cũng như thanh toán an toàn. Đồng thời, chúng tôi sẽ tạo ra một giao diện người dùng hấp dẫn, thân thiện và dễ dàng điều hướng, đảm bảo tính tương thích trên nhiều thiết bị và trình duyệt khác nhau.

Một phần quan trọng khác là cải thiện trải nghiệm người dùng (UX) thông qua việc nghiên cứu hành vi của khách hàng, tối ưu hóa quy trình mua sắm để giảm thiểu tỷ lệ bỏ giỏ hàng, và cung cấp hỗ trợ khách hàng kịp thời qua các chức năng chat trực tuyến hoặc email. Ngoài ra, chúng tôi sẽ triển khai các chiến lược marketing trực tuyến nhằm tăng doanh thu, bao gồm SEO, quảng cáo trên mạng xã hội và tiếp thị qua email, cũng như thiết kế các chương trình khuyến mãi hấp dẫn để khuyến khích khách hàng mua sắm.

Cuối cùng, chúng tôi sẽ thiết lập các chỉ số đo lường hiệu quả (KPI) để theo dõi tỷ lệ chuyển đổi, doanh thu hàng tháng và mức độ hài lòng của khách hàng. Thông qua việc thu thập phản hồi từ người dùng, chúng tôi sẽ liên tục cải tiến website nhằm đáp ứng tốt hơn nhu cầu của khách hàng và tối ưu hóa hoạt động kinh doanh.

## Khái niệm về website thương mại điện tử.

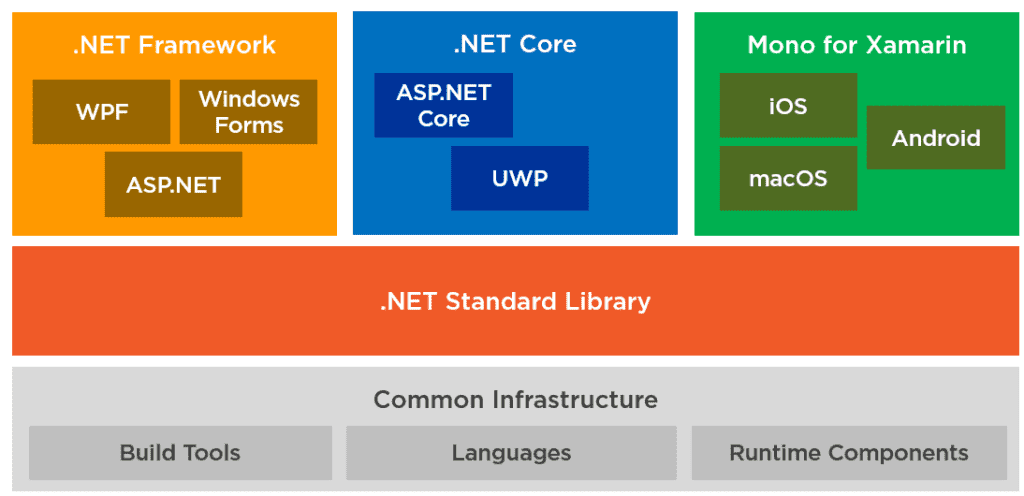
Website thương mại điện tử (e-commerce website) là một nền tảng trực tuyến cho phép người dùng thực hiện các giao dịch mua bán hàng hóa và dịch vụ qua Internet. Khác với các cửa hàng truyền thống, website thương mại điện tử hoạt động như một cửa hàng ảo, nơi người tiêu dùng có thể dễ dàng tìm kiếm, lựa chọn và thanh toán sản phẩm mà không cần phải đến tận nơi.

Các website thương mại điện tử thường được trang bị nhiều tính năng tiện ích như hệ thống quản lý sản phẩm, cổng thanh toán an toàn và giao diện người dùng thân thiện. Điều này giúp cho quá trình mua sắm trở nên thuận tiện hơn bao giờ hết. Chúng cũng cho phép người tiêu dùng theo dõi đơn hàng, từ lúc đặt hàng đến khi nhận được sản phẩm, đồng thời cung cấp các kênh chăm sóc khách hàng để giải đáp thắc mắc và xử lý khiếu nại.

Việc phát triển website thương mại điện tử không chỉ mang lại lợi ích cho người tiêu dùng mà còn mở ra nhiều cơ hội cho doanh nghiệp trong việc tăng trưởng doanh thu và tiếp cận thị trường. Trong thời đại công nghệ số hiện nay, website thương mại điện tử đã trở thành một phần thiết yếu trong chiến lược kinh doanh của nhiều doanh nghiệp, giúp họ nâng cao trải nghiệm mua sắm và đáp ứng nhu cầu ngày càng cao của khách hàng.

# CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

## Tổng quan về .Net Core



*Hình 2-1 Tổng quan về .Net Core*

Nhắc đến.NET sẽ có nhiều người nhầm lẫn giữa các khái niệm bao gồm .NET, .NET Framework, .NET Core. Thực tế thì .NET Core là 1 trong 3 phiên bản khác nhau của .NET bên cạnh .NET Framework và Mono, mỗi phiên bản có Runtime, Libraries và Toolings riêng. Cả 3 phiên bản đều được kế thừa sử dụng hệ thống thư viện cơ bản của .NET (.NET Standard Library).

Nếu như .NET Framework chỉ hoạt động trên Windows thì phiên bản .NET Core có định hướng phát triển đa nền tảng, có thể chạy trên các hệ điều hành Windows, macOS và Linux. .NET Core được thiết kế để cung cấp khả năng tối ưu tốt hơn với ưu điểm nhanh, nhẹ và đa nền tảng; vì thế nó thường được sử dụng để xây dựng các ứng dụng Web, ứng dụng dựa trên kết nối cloud, Internet of Things hay làm Backend cho các ứng dụng mobile.

## Đặc điểm nổi bật của .Net Core

* .NET Core là một nền tảng mã nguồn mở (open-source) được duy trì bởi Microsoft và cộng đồng trên GitHub
* .NET Core chạy được trên các hệ điều hành Windows, macOS và Linux
* .NET Core có khả năng tương thích với .NET Framework và Mono thông qua .NET Standard
* .NET Core đảm bảo tính nhất quán trong việc thực thi mã nguồn trên nhiều kiến trúc hệ thống khác nhau bao gồm x64, x86 và ARM
* .NET Core bao gồm một bộ các công cụ giao tiếp dòng lệnh (Command Line Interface – CLI) giúp cho việc phát triển và tích hợp liên tục
* .NET Core cho phép triển khai sản phẩm một cách linh hoạt bằng cách cài đặt song song trên thiết bị người dùng, hoặc sử dụng toàn hệ thống hay có thể sử dụng với các container Docker



*Hình 2-2 Hình ảnh minh họa đặc điểm .Net Core*

ASP.NET Core được xây dựng trên .NET Core, là một sự học hỏi của Microsoft từ các Framework Module hóa hiện đại phổ biến hiện nay. Nhờ Nuget, nó cho phép chúng ta thiết kế, xây dựng nên các ứng dụng dựa trên các packages có sẵn được tối ưu với dung lượng nhỏ hơn, độ bảo mật tốt hơn và có hiệu năng hoạt động ổn định. ASP.NET Core cũng được tích hợp những client-side framework hiện đại giúp phù hợp cho việc xây dựng Web UI và Web APIs. Nhờ đó mà ASP.NET trở thành một framework mạnh mẽ có khả năng cạnh tranh với nhiều framework của các ngôn ngữ khác trong lĩnh vực lập trình Web hiện nay.

* Các ứng dụng ASP.NET Core có thể chạy trên .NET Core hoặc trên .NET Framework hoàn chỉnh.
* Nó đã được thiết kế để cung cấp một framework tối ưu cho các ứng dụng để triển khai tới cloud hoặc chạy on-premises.
* Nó bao gồm những modular với các thành phần tối thiểu, do đó bạn giữ được tính linh hoạt trong quá trình xây dựng các giải pháp của mình.
* Bạn có thể phát triển và chạy các ứng dụng đa nền tảng từ ASP.NET Core trên Windows, Mac và Linux.

ASP.NET Core đi kèm với những ưu điểm sau:

* ASP.NET Core có một số thay đổi kiến trúc dẫn đến modular framework nhỏ hơn.
* ASP.NET Core không còn dựa trên System.Web.dll. Nó dựa trên một tập hợp nhiều yếu tố của Nuget packages.
* Điều này cho phép bạn tối ưu ứng dụng của mình chỉ cần những NuGet packages cần thiết.
* Lợi ích của diện tích bề mặt ứng dụng nhỏ hơn thì bảo mật chặt chẽ hơn, giảm dịch vụ, cải thiện hiệu suất và giảm chi phí.

Với ASP.NET Core, bạn có thể nhận được các cải tiến sau:

* Xây dựng và chạy các ứng dụng ASP.NET Core đa nền tảng trên Windows, Mac và Linux.
* Được xây dựng trên .NET Core, hỗ trợ side-by-side app versioning.
* Công cụ mới giúp đơn giản hóa việc phát triển web hiện đại.
* Liên kết đơn các web stack như Web UI và API Web.
* Cấu hình dựa trên môi trường đám mây sẵn có.
* Được xây dựng dựa trên cho DI (Dependency Injection).
* Tag Helpers làm cho các Razor makup trở nên tự nhiên hơn với HTML.

## Thành phần của .Net Core

A screen shot of a computer

Description automatically generated

*Hình 2-3 Thành phần .Net Core*

Các thành phần cốt lõi của .NET Core bao gồm:

* .NET Compiler Platform (Roslyn): trình biên dịch ngôn ngữ cho C# và Visual Basic
* CoreFX: tập hợp các Framework Library
* CoreCLR: môi trường thực thi của .NET Core, sử dụng trình biên dịch trung gian dựa trên JIT (Just-In-Time) từ đó dịch sang mã máy trên nền tảng hỗ trợ
* ASP.NET Core: cung cấp Framework để xây dựng các ứng dụng kết nối Internet, ứng dụng Cloud, ứng dụng Web, IoT, …

## Giới thiệu về Asp.Net Core.

ASP.NET Core được xây dựng trên .NET Core, là một sự học hỏi của Microsoft từ các Framework Module hóa hiện đại phổ biến hiện nay. Nhờ Nuget, nó cho phép chúng ta thiết kế, xây dựng nên các ứng dụng dựa trên các packages có sẵn được tối ưu với dung lượng nhỏ hơn, độ bảo mật tốt hơn và có hiệu năng hoạt động ổn định. ASP.NET Core cũng được tích hợp những client-side framework hiện đại giúp phù hợp cho việc xây dựng Web UI và Web APIs. Nhờ đó mà ASP.NET trở thành một framework mạnh mẽ có khả năng cạnh tranh với nhiều framework của các ngôn ngữ khác trong lĩnh vực lập trình Web hiện nay.

## Cấu trúc hạ tầng Asp.Net Core

A diagram of utility framework

Description automatically generated

*Hình 2-2 Hạ tầng .Net Core*

Cấu trúc Asp.net Core bao gồm 3 phần: Application Frameworks, Utility Frameworks, và Platform.

**Application Frameworks** chứa những cái tên có lẽ tương đối quen thuộc như MVC Framework, Razor Pages hay Blazor. Đây là những framework giúp bạn xây dựng các dạng khác nhau của ứng dụng web.

Khối **Utility Frameworks** chứa những thứ cảm tưởng như không liên quan đến Asp.net Core: Identity và Entity Framework. Khối này chứa những framework hỗ trọ ứng dụng, bao gồm bảo mật và cơ sở dữ liệu.

Khối **Platform** là những gì tạo nên nề n tảng chuung nhất mà mọi loại ứng dụng Asp.net Core đều cần sử dụng.

Sau đây là đi vào chi tiết hơn với 3 khối này

* **Application Frameworks:** 
  + ASP.NET Core MVC: Hỗ trợ phát triển ứng dụng web sử dụng mô hình MVC (Model-View-Controller). Đây là framework linh hoạt, cho phép tạo ra các trang web động và dịch vụ API.
  + ASP.NET Core Razor Pages: Một mô hình lập trình trang tập trung vào giao diện, giúp đơn giản hóa việc phát triển các trang web dựa trên HTML, CSS và JavaScript.
  + ASP.NET Core Blazor: Cho phép phát triển ứng dụng web tương tác sử dụng C# thay vì JavaScript. Có hai chế độ chính:

*Blazor Server: Chạy trên máy chủ, truyền thông qua SignalR.*

*Blazor WebAssembly: Chạy trực tiếp trên trình duyệt bằng công nghệ WebAssembly.*

* **Utility Frameworks:**

Utility Frameworks chứa hai framework (không bắt buộc) nhưng lại được sử dụng gần như trong mọi ứng dụng Asp.net Core. Vì vậy, quá trình học Asp.net Core thực tế luôn gắn với hai framework này ở giai đoạn sau.

Entity Framework Core là một ORM (Object-Relational Mapping) giúp ứng dụng tương tác với cơ sở dữ liệu. Entity Framework Core giúp ánh xạ (hai chiều) giữa các bảng cơ sở dữ liệu (ví dụ, SQL Server) với các domain class của ứng dụng. Entity Framework Core không gắn với Asp.net Core mà có thể sử dụng trong bất kỳ ứng dụng .NET nào.

Entity Framework Core có thể xem là phiên bản hiện đại đa nền tảng của Entity Framework ORM chính của .NET Framework cổ điển. Nếu đã biết Entity Framework có thể dễ dàng chuyển đổi sang Entity Framework Core.

Asp.net Core Identity là framework dành cho xác thực (authentication) và xác minh quyền (authorization) người dùng trong ứng dụng. Các ứng dụng đơn giản minh họa thường không sử dụng Identity nhưng đây là lại framework quan trọng hàng đầu với các ứng dụng thực tế

Đây là hai framework phức tạp. Các tài liệu dạy Asp.net Core thường chỉ hướng dẫn sơ lư cách sử dụng các framework này. Thực tế, để giới thiệu đầy đủ mỗi framework cần nguyê một cuốn sách riêng.

* **Platform**

Phần platform chứa nhiều thành phần cấp thấp cần thiết cho việc nhận và xử lý truy vấn HTTP, cũng như tạo ra các phản hồi phù hợp.

Mặc dù phần lớn thời gian trong quá trình phát triển ứng dụng bạn làm việc chủ yếu với applicaton framework, và bản thân các application framework cũng che đi rất nhiều phần của platform, bạn cần hiểu rõ các tính năng quan trọng của platform để có thể sử dụng hiệu quả application framework. Các thành phần của platform khá đa dạng. Chúng ta sẽ điểm qua các thành phần này.

* **HTTP Server**, còn gọi là built-in server với tên gọi Kestrel, có nhiệm vụ tiếp nhận truy vấn HTTP. Kestrel có thể hoạt động độc lập (tích hợp trong một ứng dụng khác) hoặc phối hợp với một web server thông thường (Apache, NGinx, IIS).
* **Middleware** là nhóm thành phần có nhiệm vụ xử lý truy vấn HTTP. Các middleware được xếp theo chuỗi. Khi truy vấn chạy qua mỗi khâu của chuỗi middle sẽ được xem xét xử lý.
* **URL Routing** là cơ chế ánh xạ chuỗi truy vấn HTTP sang thực thi một phương thức nào đó. Do vậy, mỗi URL (địa chỉ gửi về server) sẽ tương ứng với thực thi một phương thức trên server
* **Razor** là cơ chế sinh ra HTML từ dữ liệu và logic của chương trình. Razor được gọi là view engine trong Asp.net Core. Razor sử dụng loại cú pháp đặc biệt kết hợp giữa C# và HTML.
* **Model Binding** là cơ chế ánh xạ dữ liệu chứa trong truy vấn HTTP sang tham số của phương thức cần thực thi. Nhờ cơ chế Model Binding, việc xây dựng các phương thức trong Asp.net Core đơn giản như bất kỳ phương thức c# thông thường nào.
* **Dependency Injection** là cơ chế cho phép tự động sinh và chèn object vào một object khác. Asp.net Core xây dựng sẵn cơ chế này cho bạn mà không cần đến một thư viện thứ ba (như Ninject, Unity).
* **Configuration & Logging** là cơ chế hỗ trợ cấu hình và lưu vết quá trình thực thi ứng dụng.
* **Caching** là cơ chế lưu tạm để tăng hiệu suất cho ứng dụng.

# CHƯƠNG 3. ĐẶC TẢ - THIẾT KẾ - XÂY DỰNG HỆ THỐNG

## Mô tả chức năng nghiệp vụ của hệ thống

***Website shop bán giày gồm có các chức năng:***

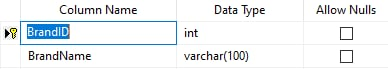
* Đăng ký: Cho phép khách hàng đăng ký thành viên vào hệ thống.
* Đăng nhập: Cho phép khách hàng thành viên, nhân viên chăm sóc khách hàng, và quản trị viên đăng nhập vào hệ thống.
* Đăng xuất: Cho phép quản trị viên, nhân viên chăm sóc khách hàng, và khách hàng thành viên đăng xuất khỏi hệ thống.
* Quản lý tài khoản: Cho phép khách hàng thành viên xem và cập nhật thông tin tài khoản.
* Quản lý giỏ hàng: Cho phép khách hàng xem giỏ hàng và quản lý sản phẩm trong giỏ.
* Đặt hàng: Cho phép khách hàng tiến hành đặt hàng các sản phẩm trong giỏ hàng.
* Quản lý đơn đặt hàng: Cho phép khách hàng thành viên xem tình trạng đơn đặt hàng.
* Xử lý đơn đặt hàng: Cho phép quản trị viên xem và xử lý đơn đặt hàng.
* Quản lý danh mục sản phẩm: Cho phép quản trị viên quản lý danh mục sản phẩm.
* Quản lý sản phẩm: Cho phép quản trị viên quản lý sản phẩm trong hệ thống.
* Quản lý tài khoản hệ thống: Cho phép quản trị viên quản lý tài khoản hệ thống.
* Phân quyền: Cho phép quản trị viên phân quyền cho các loại tài khoản khác nhau.

## Các yêu cầu phi chức năng của website

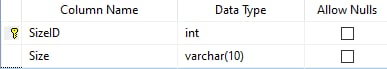
* Giao diện đẹp, đơn giản, thân thiện với người sử dụng.
* Kích thước của cơ sở dữ liệu phải đủ lớn để lưu trữ thông tin tăng lên khi sử dụng.
* Bàn giao website đúng thời gian và địa điểm thích hợp.
* Giao diện của website đơn giản, đủ để người dùng có thể sử dụng nó một cách dễ dàng sau 5 đến 10 phút làm quen.
* Tốc độ phản hồi của trang dưới 10 giây.
* Hệ thống có độ tin cậy cao. Thời gian khắc phục lỗi gặp phải khi hoạt động tối đa là 3 ngày.

## 3.3. Phân tích các thành phần trong database.

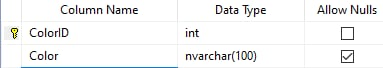
***Bảng 1: Brand***



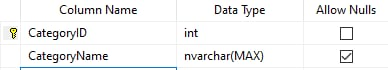
***Bảng 2: Size***



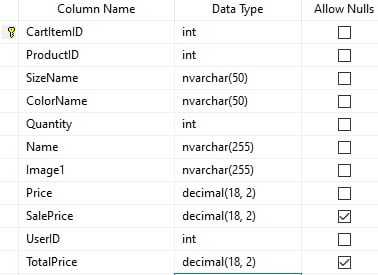
***Bảng 3: Color***



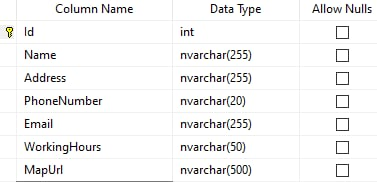
***Bảng 4: Category***



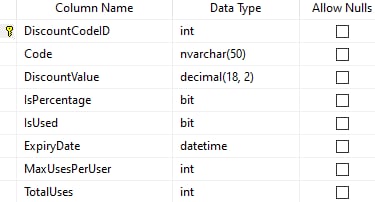
***Bảng 5: Cartitem***



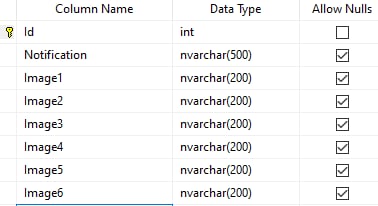
***Bảng 6: ContactInfo***



***Bảng 7: Discountcode***



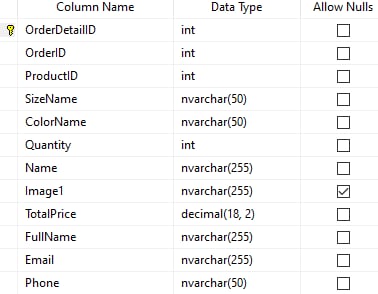
***Bảng 8: HomeContent***



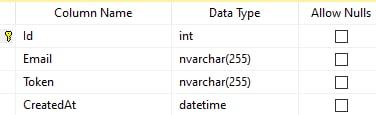
***Bảng 9: Order***



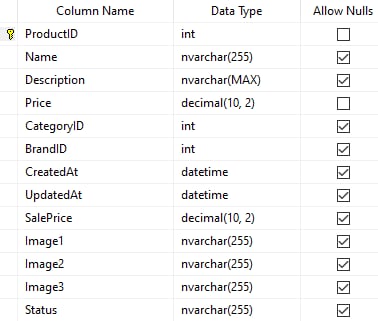
***Bảng 10: Orderdetails***



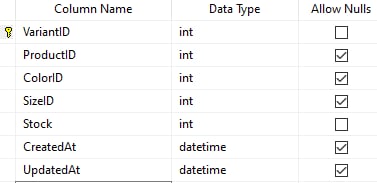
***Bảng 11: PassWordResetToken***



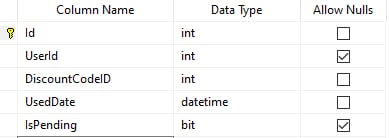
***Bảng 12: Product***



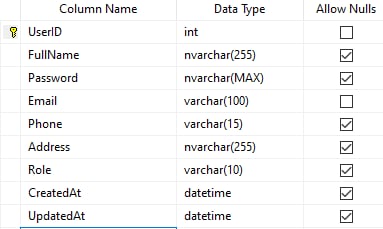
***Bảng 13: ProductVariant***



***Bảng 14: UserDiscount***



***Bảng 16: User***



***Bảng 17: Wishlist***

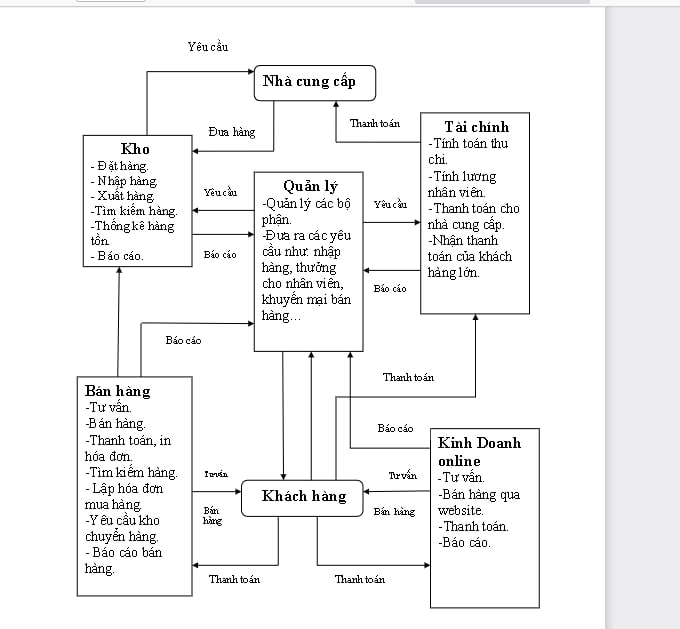


Nhóm đã trao đổi bàn bạc tìm hiểu và tham khảo các shop uy tín và nhận thấy có 1 vài chức năng cần thiết, chung như sau:

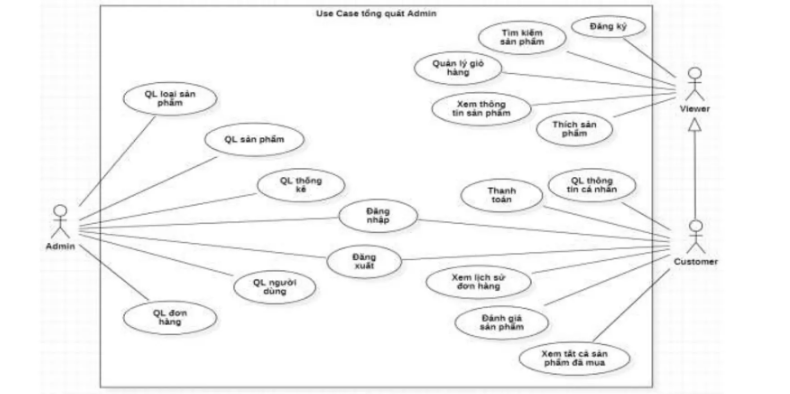
A diagram of a company

Description automatically generated

## 3.4. Sơ đồ DFD tổng quan.



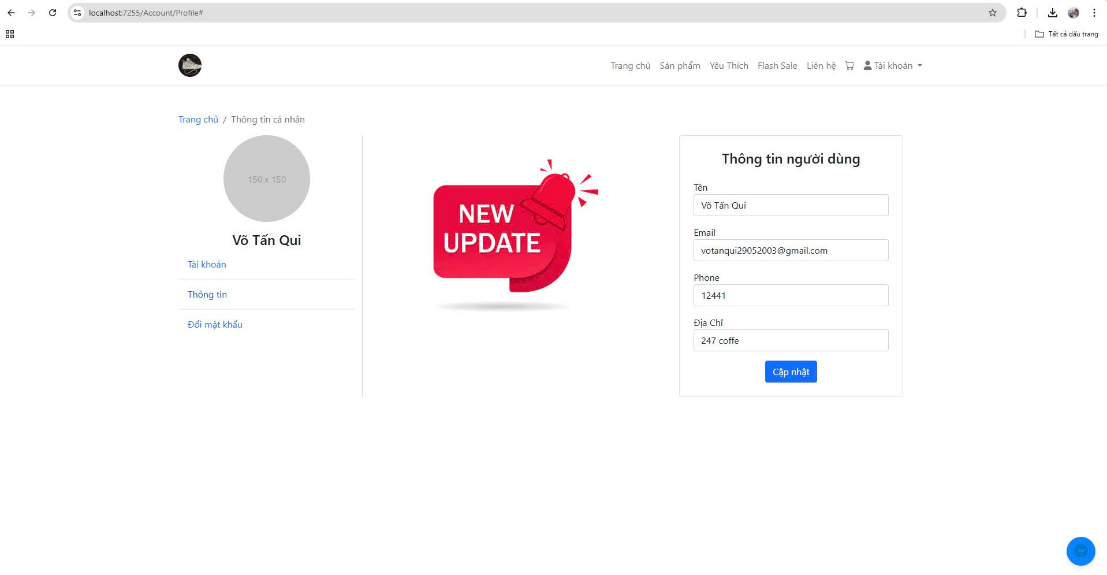
## 3.5. Sơ đồ Uses case tổng quát.

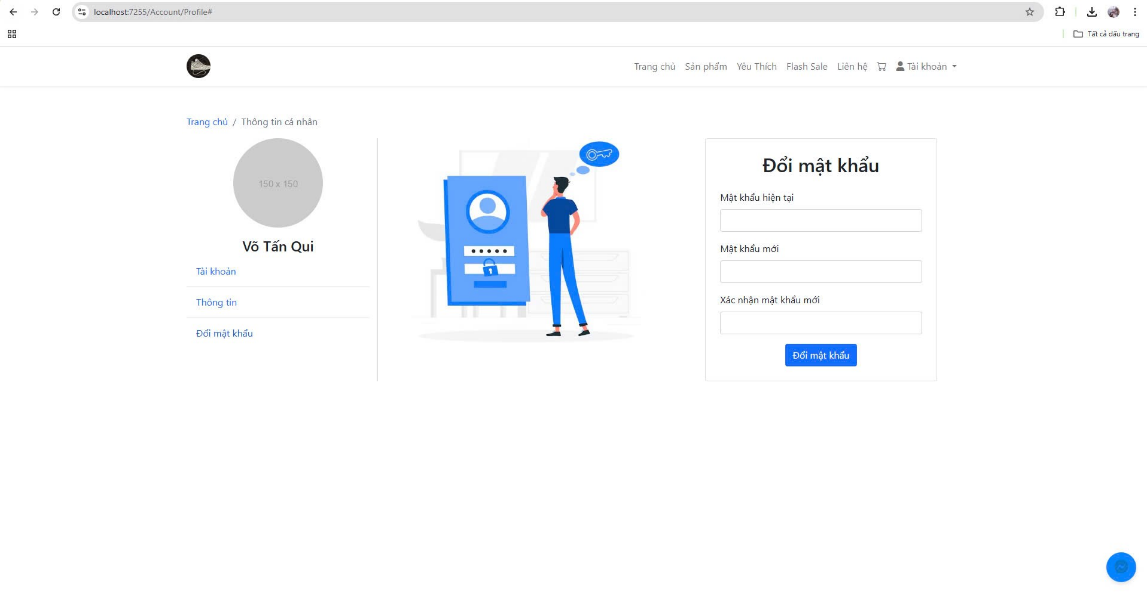


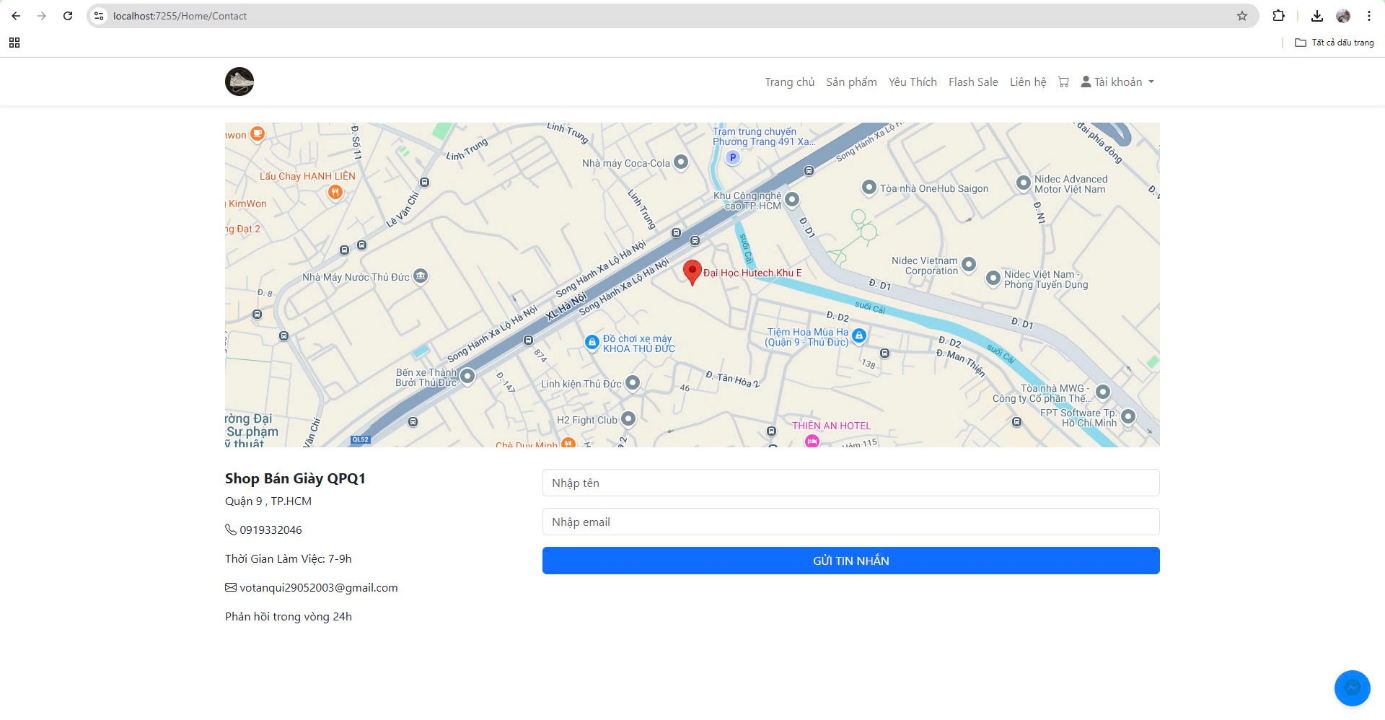
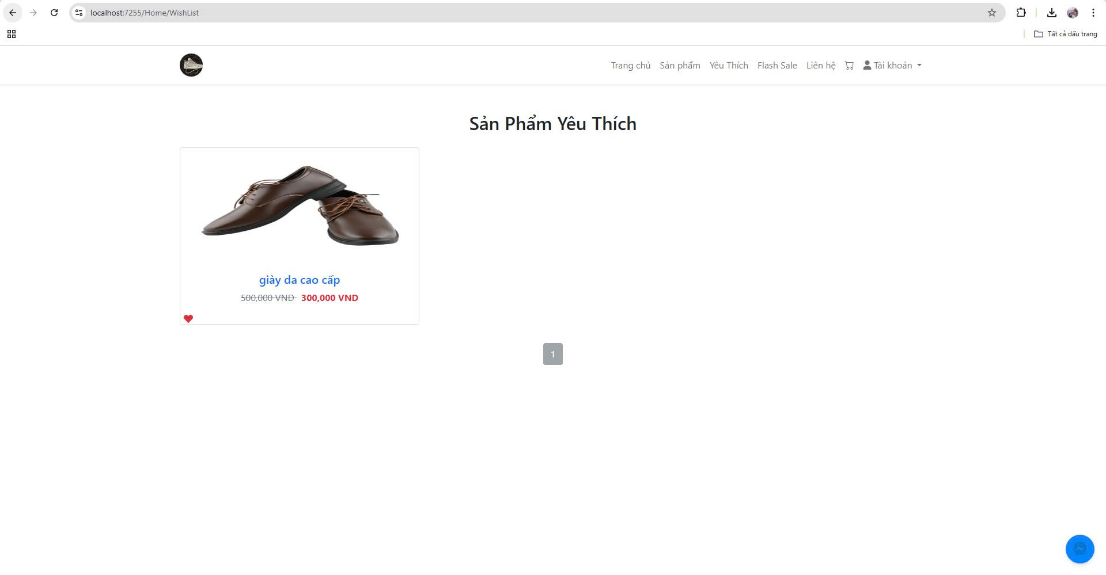
# Chương 4: DEMO

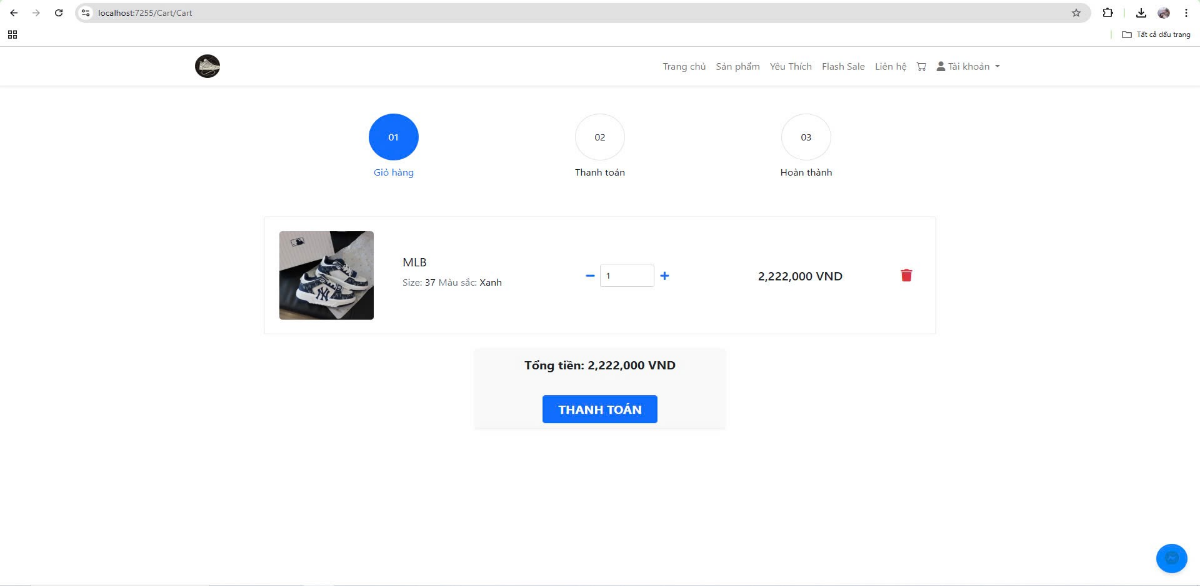
## 4.1 Dưới góc nhìn khách hàng Trang chủ cửa hàng Sản phẩm trong cửa hàng

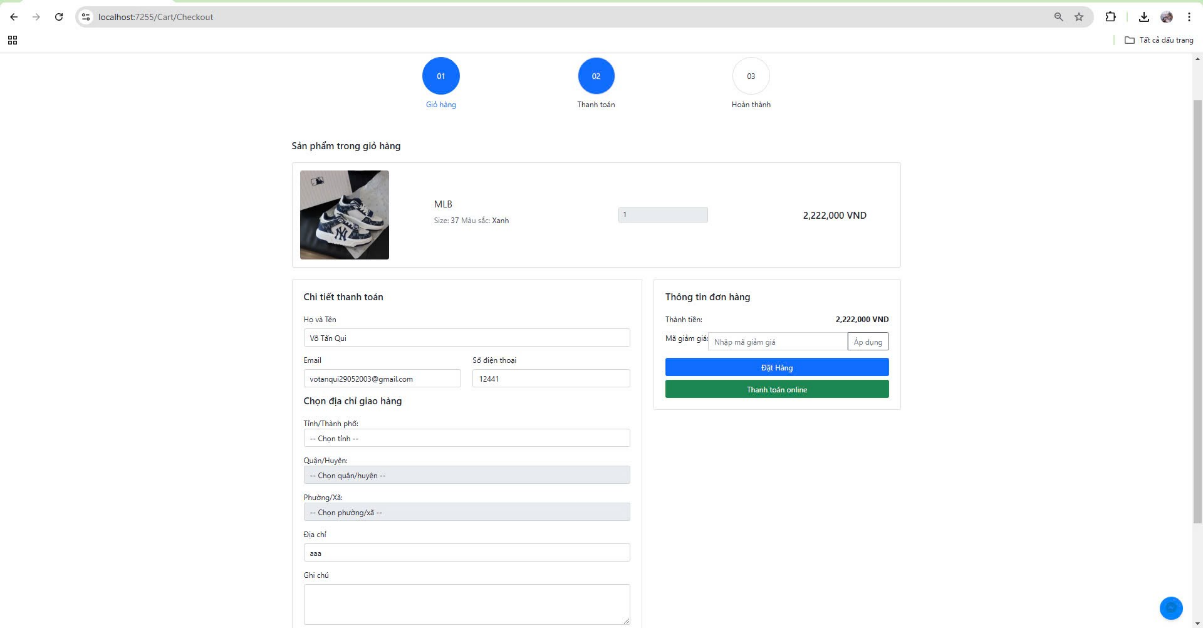
Thông tin người dùng

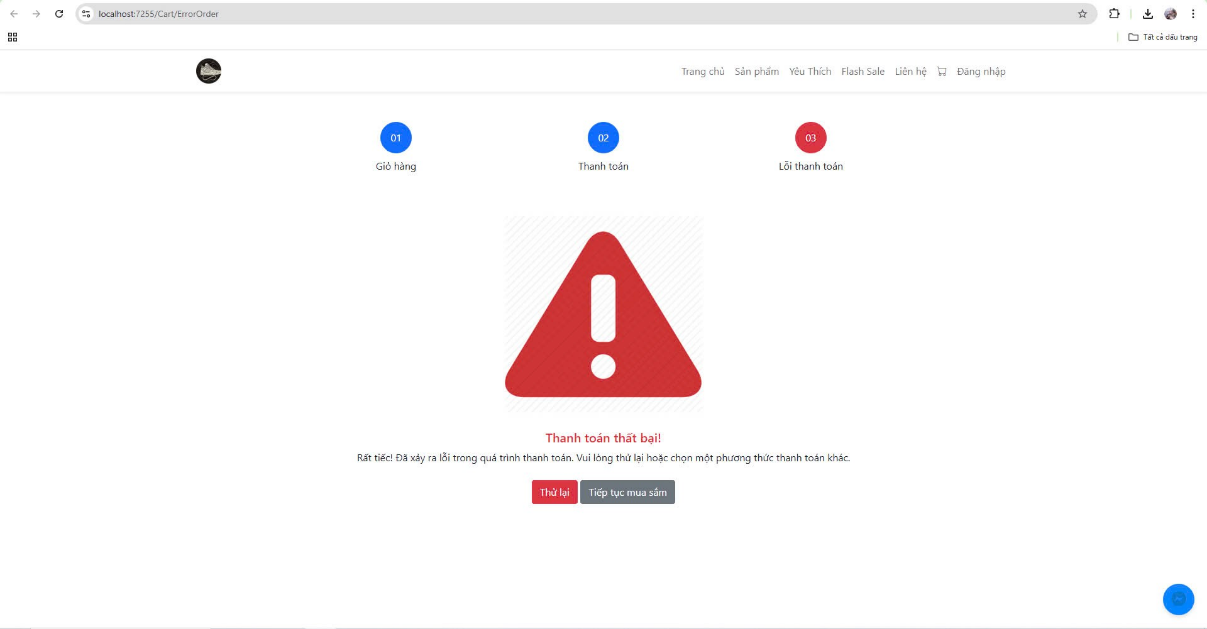
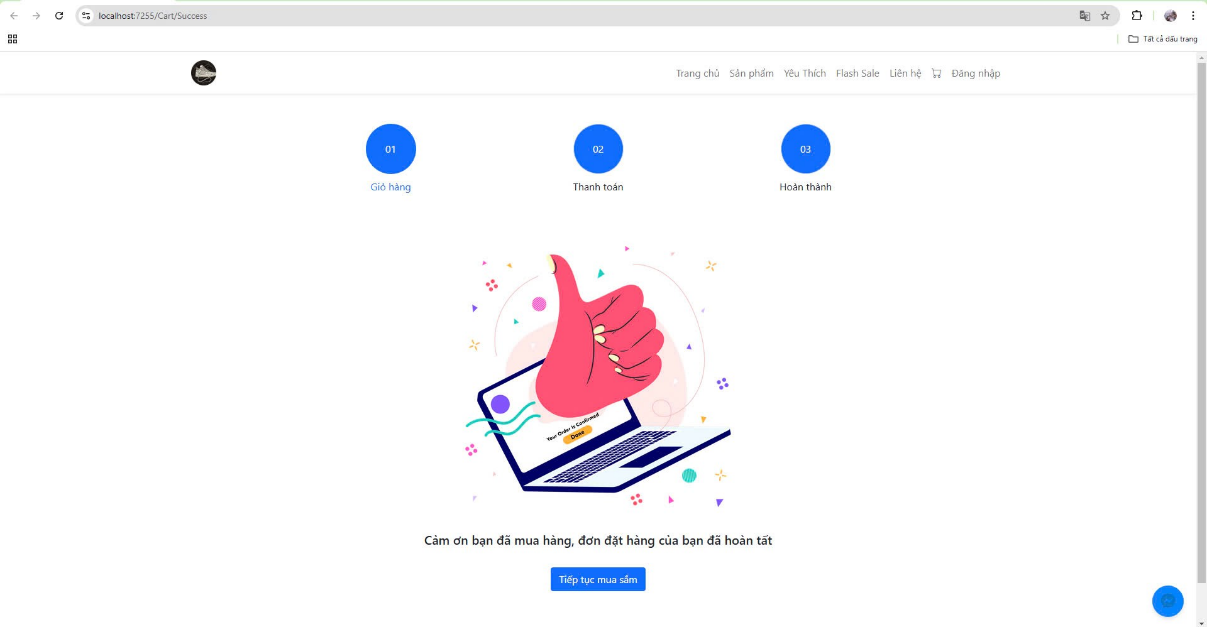


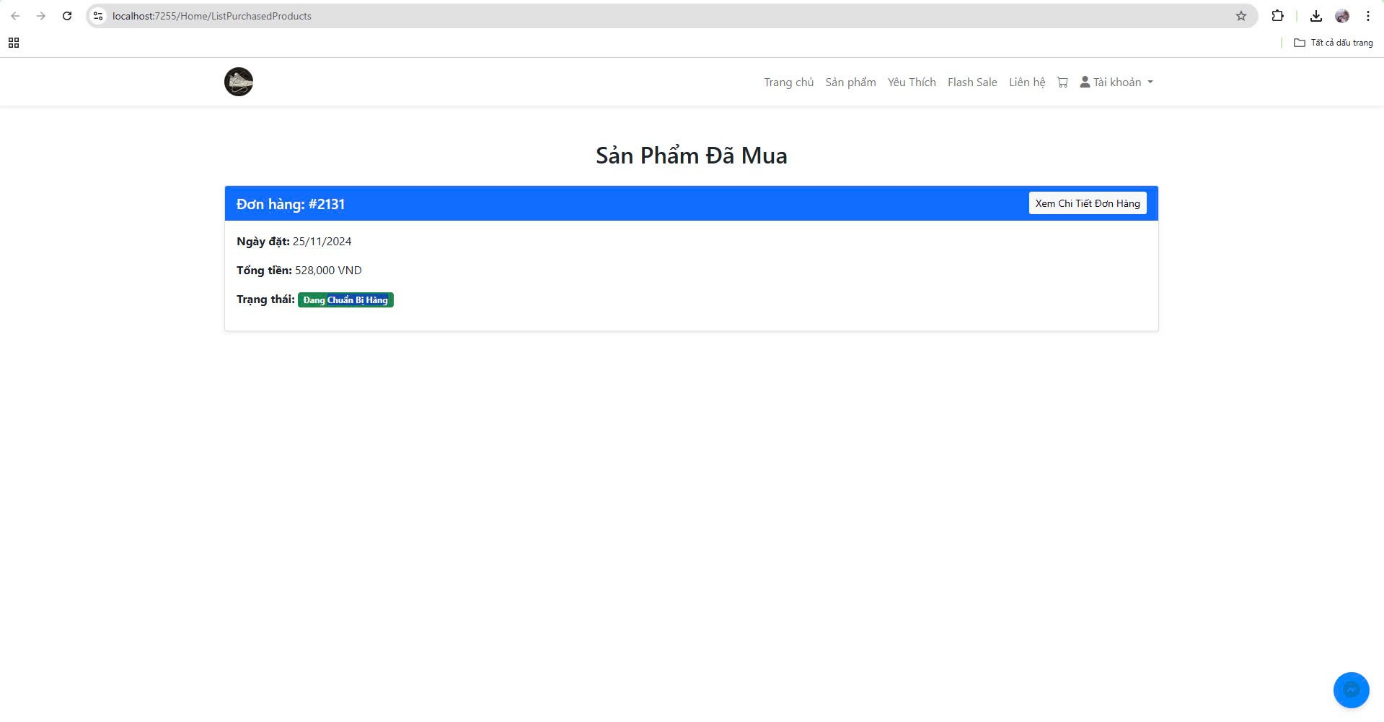
Đổi mặt khẩu người dùng 

Gửi phản hồi cho cửa hàng Chọn món hàng yêu thích 

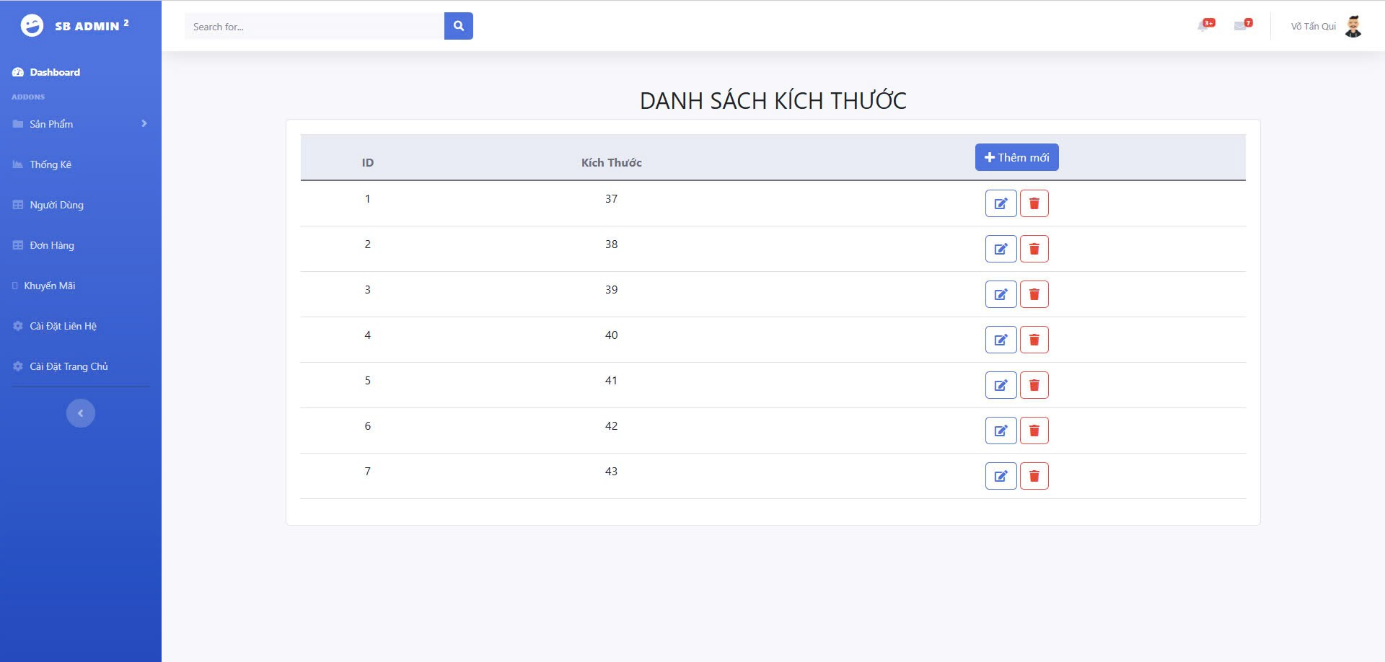
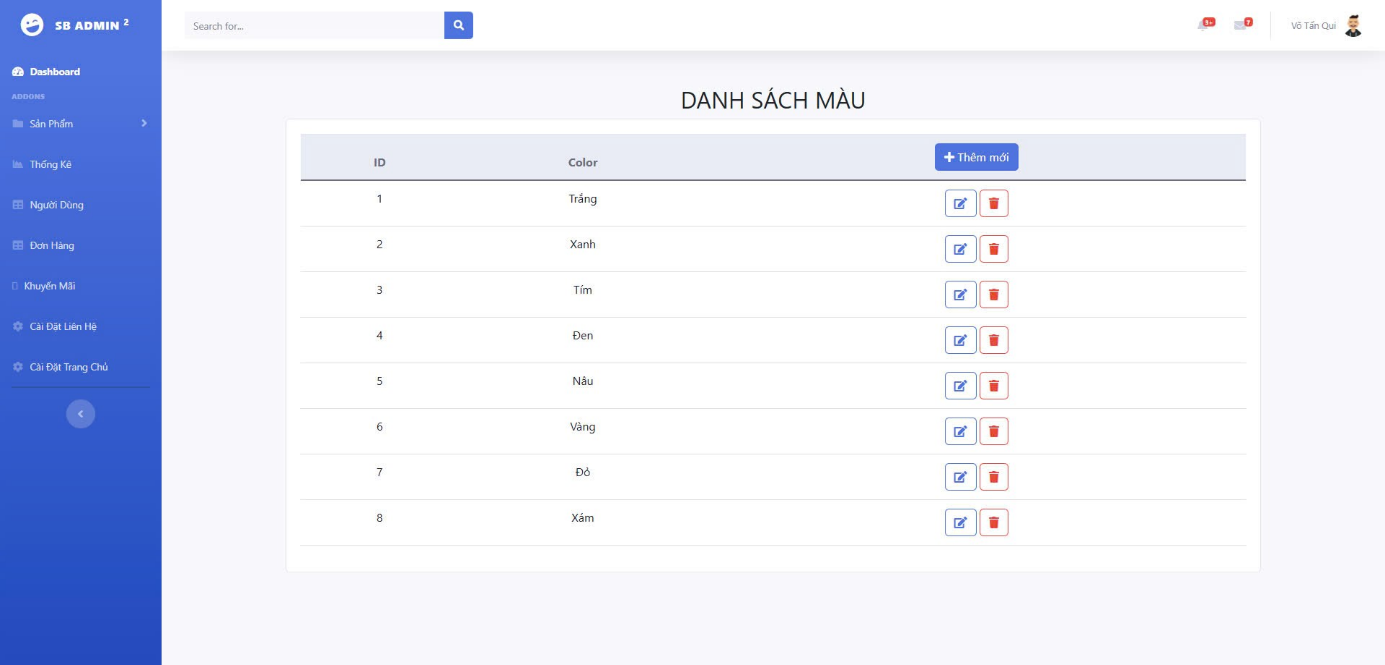
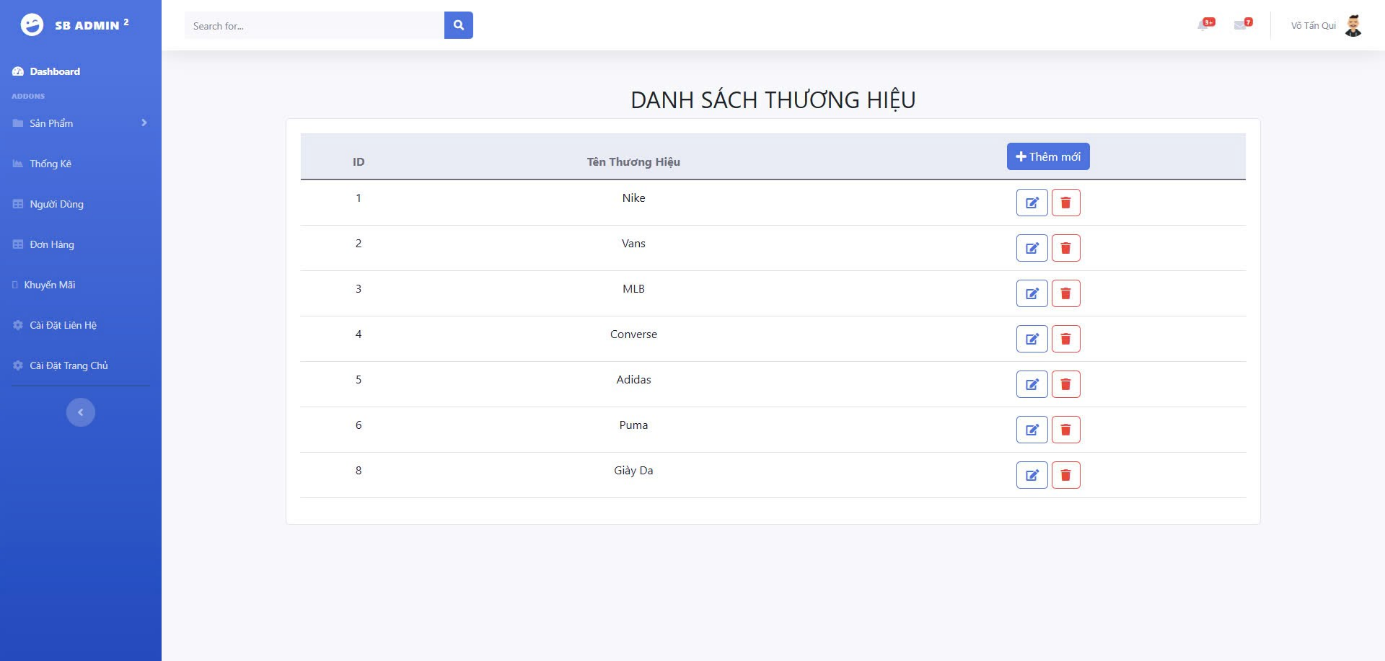
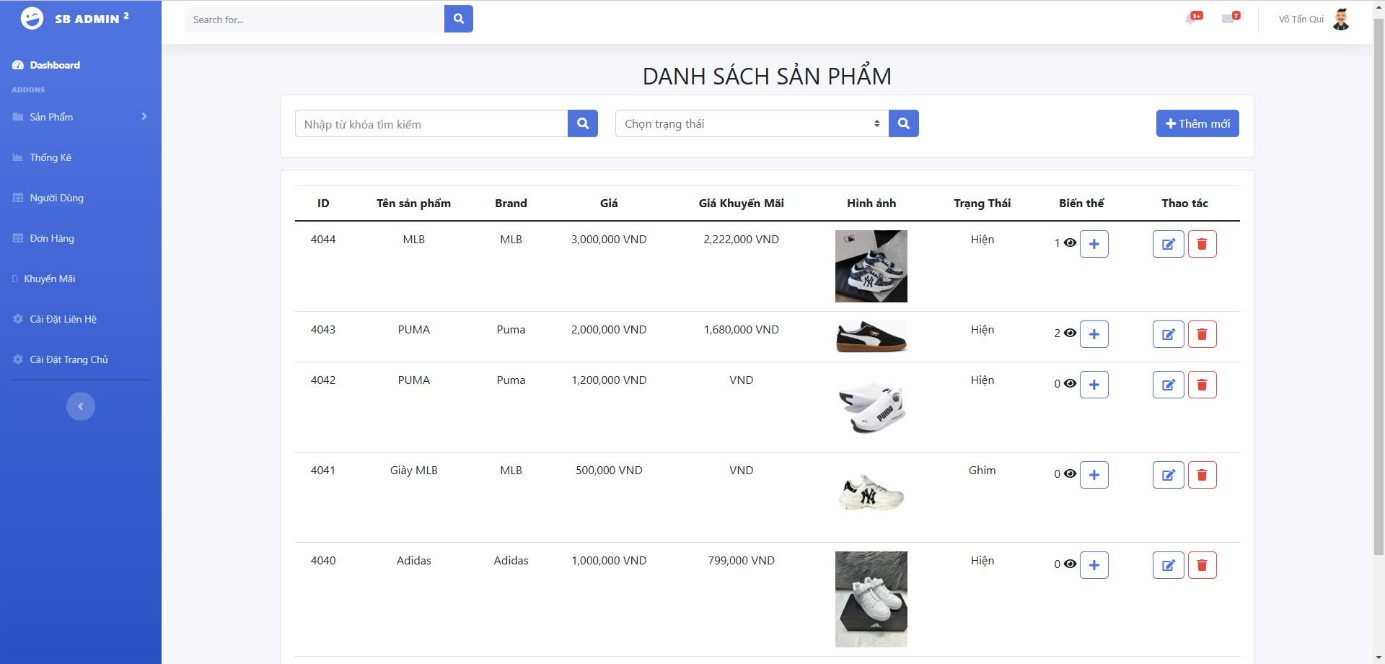
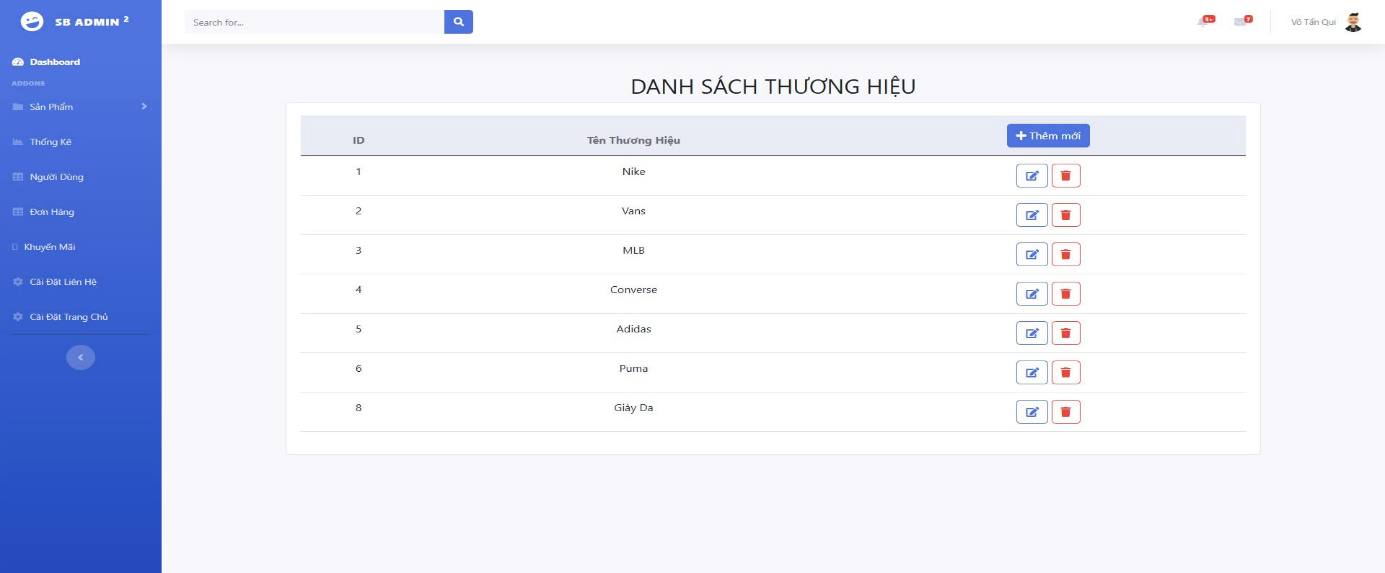
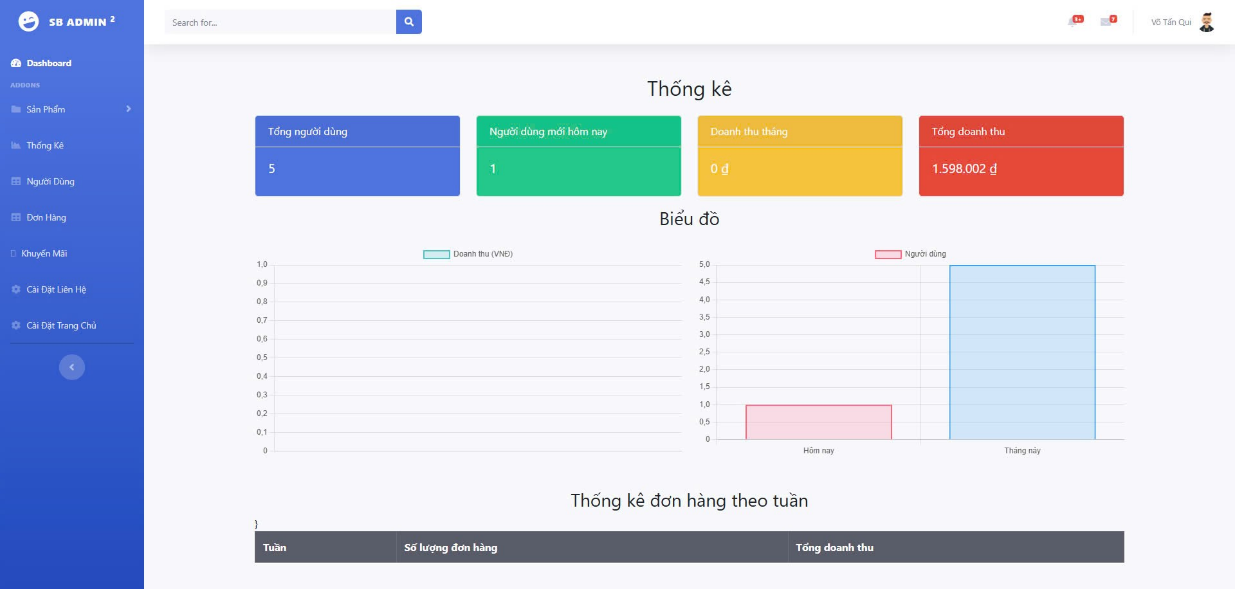
Đặt hàng   


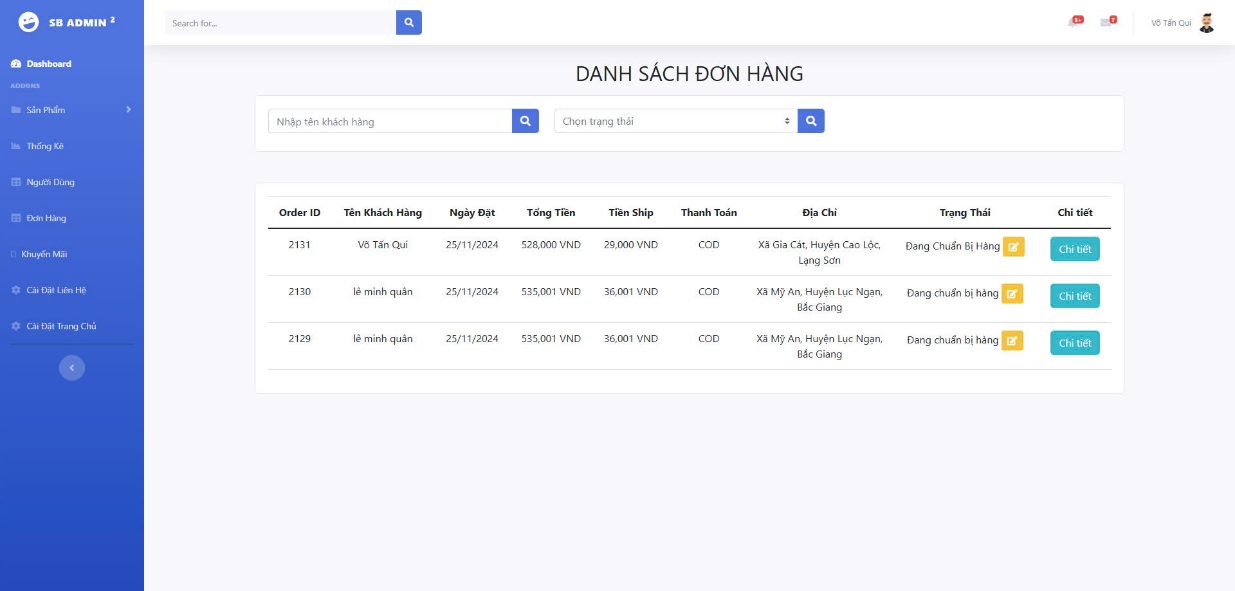
Các phương thức thanh toán 

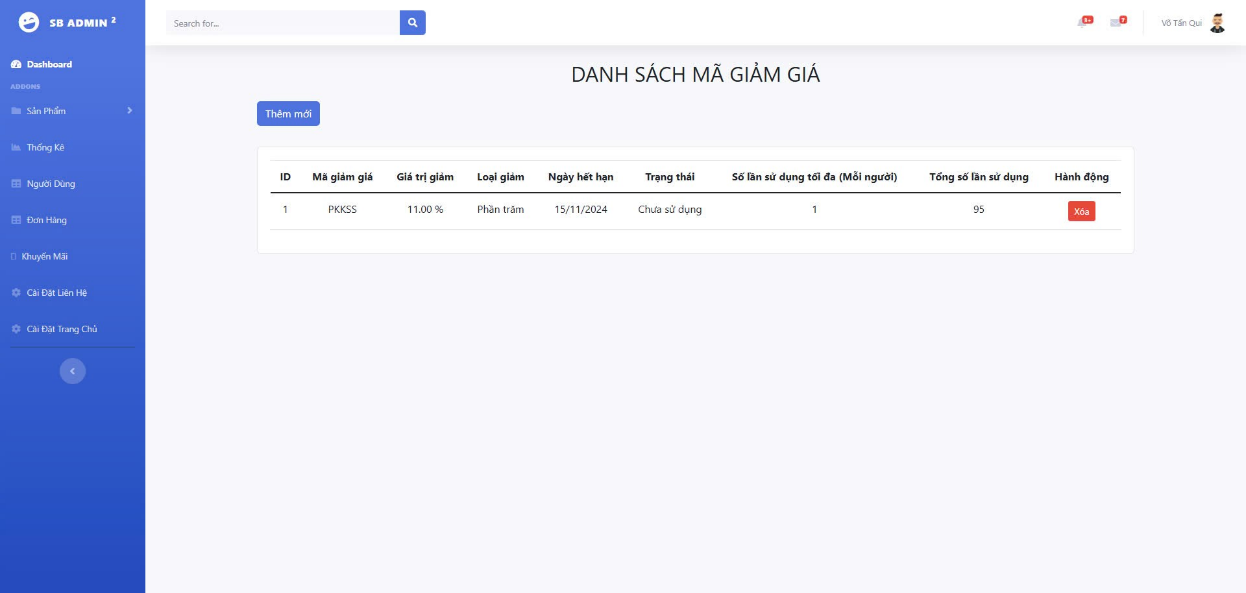
Thành công và ngược lại 

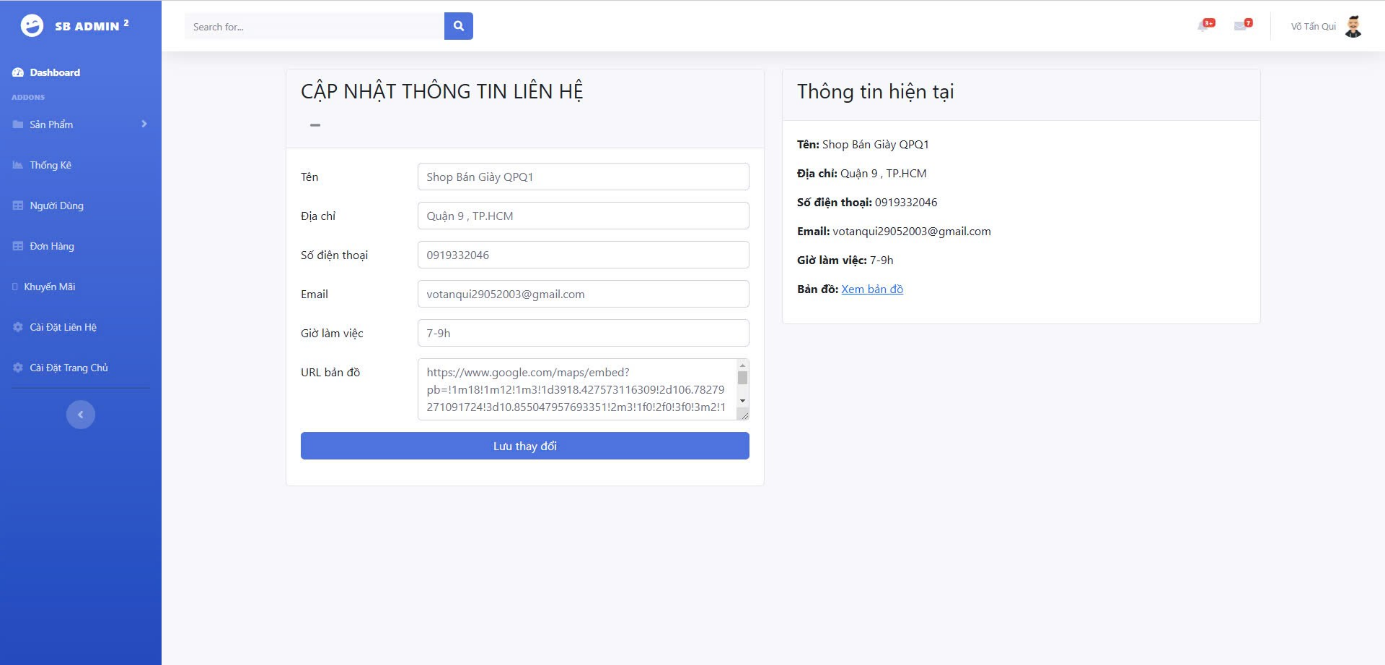
Thông báo xác nhận đơn hàng 

## 4.2 Dưới góc nhìn admin

Thêm xóa sửa thông tin các mặt hàng   
Cập nhật các hạng mục trong cửa hàng Kiểm tra thống kê cửa hàng 

Doanh số cửa hàng Tạo các hạng mục marketting



Cập nhật thông tin cửa hàng Thay đổi giao diện phù hợp 