

CHƯƠNG 1: TỔNG QUAN ĐỀ TÀI

1.1. Giới thiệu

Trong những năm gần đây, ứng dụng công nghệ thông tin (CNTT) về quản lý web đang phát triển mạnh mẽ, giúp cho các cửa hàng kiếm được lợi nhuận nhiều hơn nhờ vào các trang website bán hàng nhanh. Nên em quyết định chọn đề tài “Xây dựng trang website quản lý lựa chọn cây trồng”. Với mong muốn giúp cho cửa hàng kiếm thêm thu nhập nhiều hơn nhờ vào việc bán online. Ngoài ra, website bán hàng rất thân thiện giúp cho khách hàng mua được thuận lợi hơn, khách hàng sẽ được mua những hàng công nghệ mà mình thích không cần tốn nhiều thời gian để đi ra cửa hàng mua. Đây là điều tuyệt nhất đối với khách hàng.

Một trong các nhà phát triển phần mềm hàng đầu thế giới về lĩnh vực này là Microsoft, họ đã cho ra đời mô hình ASP.NET Core MVC [1] hỗ trợ người phát triển các ứng dụng web trên nền .NET Framework trong đó có tích hợp công nghệ LINQ với Entity Framework hỗ trợ làm việc với các loại cơ sở dữ liệu (CSDL).

Do đó, em chọn đề tài “Xây dựng trang website quản lý lựa chọn cây trồng” để thực hiện đồ án lập trình web đáp ứng theo chương trình đào tạo của Trường, đồng thời mong muốn góp phần đẩy đưa ứng dụng CNTT vào Website để cho việc bán hàng công nghệ được hoàn thiện và thân thiện nhất giúp cho khách hàng hài lòng khi vào Website của em.

1.2. Đối tượng và mô tả đối tượng nghiên cứu

Đối tượng nghiên cứu là các cửa hàng kinh doanh về cây trồng vừa và nhỏ muốn tăng doanh thu thông qua việc bán hàng trực tuyến.

1.3. Mục tiêu của đề tài

- Tìm hiểu kỹ thuật lập trình trên ASP.NET Core MVC.
- Tìm hiểu các công nghệ hỗ trợ như: HTML, CSS, Ajax, JQuery, Web Service...
- Tìm hiểu về ngôn ngữ và kiến thức tương tác CSDL (SQL và Entity Framework).
- Cách điều khiển dữ liệu (Controller).
- Tìm hiểu về thanh toán trực tuyến vào web để cung cấp chức năng thanh toán online qua các thẻ...

- Tìm hiểu về Domain Hosting và triển khai vận hành Website trên môi trường Internet.
- Tìm hiểu GoogleMapAPI để đánh dấu địa điểm của website trên bản đồ, chỉ dẫn đường đi từ vị trí của khách hàng đến cửa hàng của website, giúp thân thiện với người dùng hơn.
- Tìm hiểu Google API để đăng nhập bằng gmail cho trang website.
- Tìm hiểu FB API để đăng nhập bằng FB làm coment, like, share cho web, ...
- Tìm hiểu nhu cầu, nghiệp vụ tìm kiếm sản phẩm phù hợp và giá cả theo mong muốn của người dùng của người dùng
- Hiểu về ngôn ngữ lập trình C# và áp dụng cho xây dựng website
- Hoàn thành website quản lý lựa chọn cây trồng online và tại nhà với các tính năng như:
 - Admin: Thực hiện các chức năng như quản trị hệ thống, thêm, xóa, sửa sản phẩm, quản lý các bài viết, quản lý người dùng, quản lý thành viên, quản lý liên hệ, quản lý bình luận, xem thống kê
 - Khách hàng: Có thể thực hiện các chức năng như tìm kiếm, xem tin tức, xem thông tin chi tiết của các sản phẩm, bình luận, đăng ký, đăng nhập tài khoản trên hệ thống, gửi thông tin liên hệ.

1.4. Phạm vi đề tài

Tìm hiểu gói gọn trong các cửa hàng kinh doanh vừa và nhỏ. Hệ thống không nhắm đến các cửa hàng công ty lớn hoặc các cửa hàng kinh doanh đa dạng các mặt hàng.

Vì là một website bán hàng, nên điều được em chú trọng đó là tính thân thiện với người dùng, giao diện đẹp mắt, cách trình bày sản phẩm cũng như chi tiết sản phẩm, giá cả hợp lý cũng là cách để gây ấn tượng tốt với khách hàng.

Cả khách hàng và nhà quản lý sẽ cảm thấy thoải mái khi sử dụng một trang web đẹp và dễ truy cập.

- **Người dùng (User):**
 - Tham quan trang web.

- Tìm kiếm và lựa chọn sản phẩm cần mua.
- Xem chi tiết 1 sản phẩm.
- Xem quảng cáo.
- Đăng ký, đăng nhập phục vụ cho việc mua bán online.
- Thực hiện việc mua hàng và thanh toán trực tuyến.
- Gửi ý kiến phản hồi, bình luận về sản phẩm qua facebook dưới chi tiết các sản phẩm.
- Gửi liên hệ cho cửa hàng.
- Xem nơi bán của shop thông qua Google Map trên trang liên hệ, giới thiệu.
- **Người quản trị (Admin):**
 - Quản lý trang web của mình thông qua tài khoản Admin rõ ràng, dễ sử dụng, bảo mật cao.
 - Quản lý tất cả các sản phẩm một cách dễ dàng.
 - Xem sản phẩm danh sách sản phẩm mới và danh sách sản phẩm bán chạy.
 - Thêm loại sản phẩm, màu cho loại sản phẩm.
 - Kiểm tra và xử lý đơn đặt hàng.
 - Theo dõi và phản hồi ý kiến khách hàng.

1.5. Cấu trúc đề tài:

- Chương 1: Tổng quan đề tài
- Chương 2: Cơ sở lý thuyết
- Chương 3: Phân tích chương trình
- Chương 4: Thực nghiệm và triển khai
- Kết luận.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1. Giới thiệu về ASP.NET CORE MVC

2.1.1. Giới thiệu về ASP.NET Core

ASP.NET Core là gì?

- ASP.NET Core là một tập hợp các thư viện chuẩn như một framework để xây dựng ứng dụng web. ASP.NET Core không phải là phiên bản tiếp theo của ASP.NET. Nó là một cái tên mới được xây dựng từ đầu. Nó có một sự thay đổi lớn về kiến trúc và kết quả là nó gọn hơn, phân chia module tốt hơn. ASP.NET Core có thể chạy trên cả .NET Core hoặc full .NET Framework.

Vậy .NET Core là gì ?

- .NET Core là môi trường thực thi. Nó được thiết kế lại hoàn toàn của .NET Framework. Mục tiêu chính của .NET Core là hỗ trợ phát triển ứng dụng đa nền tảng cho ứng dụng
- .NET Core nó được hỗ trợ trên Windows, Mac OS và Linux. .NET Core là một framework mã nguồn mở được xây dựng và phát triển bởi Microsoft và cộng đồng .NET trên Github
- .NET Core là một tập con của Full .NET Framework. WebForms, Windows Forms, WPF không phải là một phần của .NET Core. Nó cũng triển khai đặc điểm của .NET Standard.

Vậy .NET Standard là gì ?

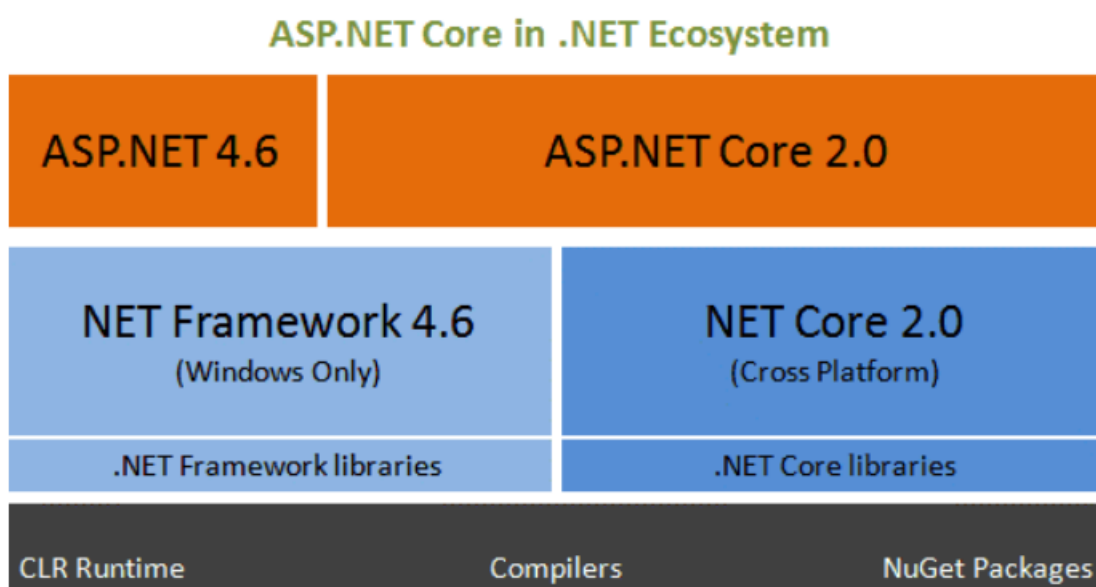
- .NET Standard là một đặc tả chuẩn của .NET API hướng tới hỗ trợ trên tất cả các triển khai của nền tảng .NET. Nó định nghĩa một tập các quy tắc thống nhất cần thiết để hỗ trợ tất cả các ứng dụng trên nền .NET.

Sự khác nhau giữa .NET Core và .NET Framework

- .NET Framework là môi trường cũ hơn và đã tồn tại trên Windows nhiều năm nay. .NET Core hỗ trợ các tính năng trong tập con của .NET Framework. Các tính năng như WebForms, Windows Forms, WPF chưa được đưa lên .NET Core. .NET

Framework chỉ chạy trên Windows trong khi .NET Core có thể chạy trên bất cứ nền tảng nào.

- Từ Core trong ASP.NET Core rất dễ nhầm. Tên đó gợi nhớ việc ứng dụng ASP.NET Core chỉ có thể được build trên .NET Core, điều đó là sai.
- Ứng dụng ASP.NET Core có thể được phát triển sử dụng .NET Core hoặc .NET Framework. Ứng dụng ASP.NET Core được xây dựng sử dụng .NET Core có thể chạy trên bất cứ hệ điều hành nào trong khi nếu được xây dựng trên .NET Framework chỉ có thể chạy trên Windows.



Hình 2. 1. ASP.NET Core in .NET Ecosystem

2.1.2. Tìm hiểu về mô hình lập trình web MVC của ASP.NET Core

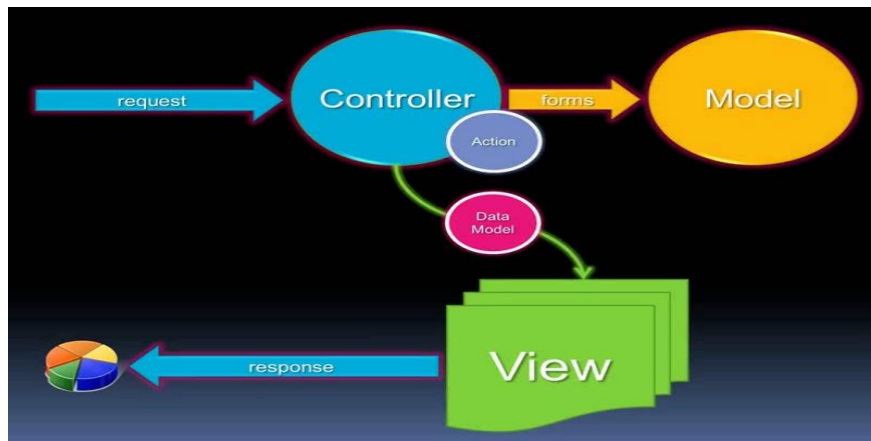
- Mô hình MVC (viết tắt chữ cái đầu của 3 từ Model - View - Controller) là một kiến trúc phần mềm hay mô hình thiết kế được sử dụng trong kỹ thuật phần mềm (đặc biệt đối với phát triển ứng dụng web). Nó giúp cho tổ chức ứng dụng (phân bố source code ứng dụng) thành 3 phần khác nhau *Model*, *View* và *Controller*. Mỗi thành phần có một nhiệm vụ riêng biệt và độc lập với các thành phần khác.
- **Model**: là các thành phần chứa tất cả các nghiệp vụ logic, phương thức xử lý, truy xuất CSDL, đối tượng mô tả dữ liệu như các Class, hàm xử lý... Model được giao nhiệm vụ cung cấp dữ liệu cho CSDL và lưu dữ liệu vào các kho chứa dữ liệu. Tất

cả các nghiệp vụ logic được thực thi ở Model. Dữ liệu vào từ người dùng sẽ thông qua View để kiểm tra ở Model trước khi lưu vào cơ sở dữ liệu. Việc truy xuất, xác nhận và lưu dữ liệu là một phần của Model.

- **View:** View hiển thị các thông tin cho người dùng của ứng dụng và được giao nhiệm vụ cho việc nhận các dữ liệu vào từ người dùng, gọi đi các yêu cầu người dùng đến bộ điều khiển (Controller), sau đó là nhận lại các phản hồi từ bộ điều khiển và hiển thị kết quả cho người dùng. Các trang HTML, JSP, các thư viện thẻ và các file nguồn là một phần của View.
- **Controller:** Controller là tầng trung gian giữa Model và View. Controller được giao nhiệm vụ nhận các yêu cầu từ người dùng (phía máy khách). Một yêu cầu được nhận từ máy khách được thực hiện bởi một chức năng logic thích hợp từ thành phần Model và sau đó sinh ra các kết quả cho người dùng và được thành phần View hiển thị. ActionServlet, Action, ActionForm, struts-config.xml là các thành phần của Controller.

Một ứng dụng web được phát triển theo mô hình MVC, có nguyên lý hoạt động xử lý yêu cầu của người dùng như sau:

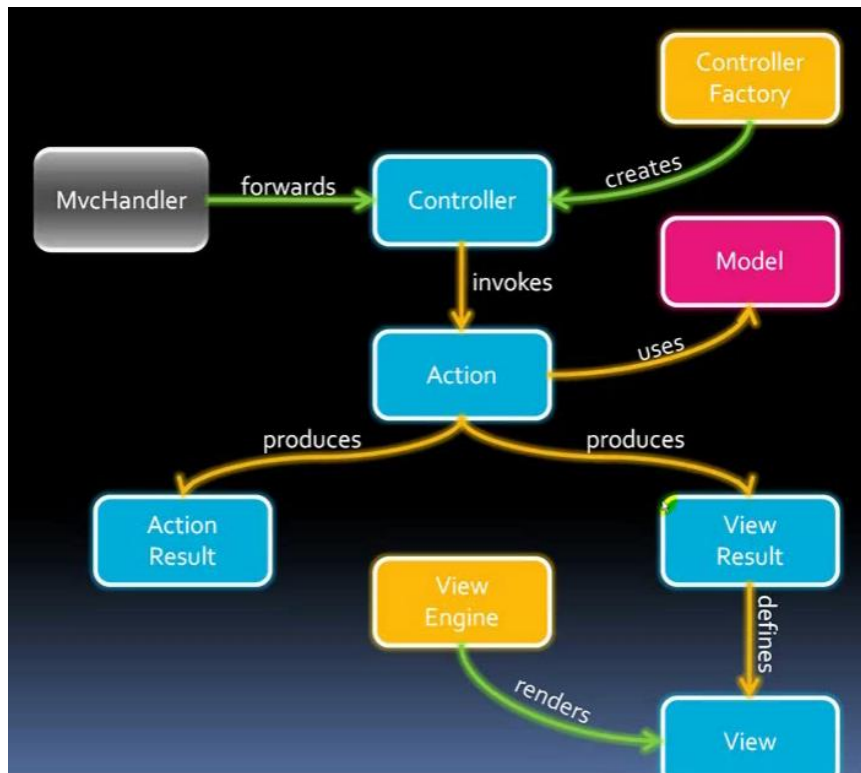
- Khi có yêu cầu phát sinh từ người dùng (phía Client), yêu cầu này sẽ được gửi đến phía Server, tại đây Controller sẽ tiếp nhận yêu cầu để xử lý.
- Bên trong Controller chứa nhiều phương thức (action), mỗi action tương ứng với 1 yêu cầu cụ thể từ người dùng. Controller sẽ lựa chọn action phù hợp với yêu cầu để xử lý. Trong quá trình xử lý, Controller tương tác với Model để có được dữ liệu mà người dùng mong muốn.
- Sau khi Controller làm việc với model để có được dữ liệu theo yêu cầu, Controller sẽ gửi về cho View thành phần Data Model, và nhiệm vụ của View sẽ chuyển Data Model nhận được thành dữ liệu ứng dụng và gửi trả về phía Client để hiển thị kết quả yêu cầu.



Hình 2. 2. Mô hình MVC

2.1.3. Hoạt động của các thành phần chính trong ASP.NET Core MVC

- Khi 1 request phát sinh từ web browser đi đến IIS Web Server thì request đó cuối cùng được đưa đến 1 MVC Handler.
- MVC Handler có nhiệm vụ chọn ra đúng 1 Controller để xử lý request đó. Controller được tạo ra bởi 1 thành phần được gọi là Controller Factory.
- Sau khi được tạo ra, Controller sẽ xác định request này được xử lý bởi Action Method cụ thể nào và sau đó sẽ thực thi Action Method đó. Action method có thể tương tác với những Model Class để có thể truy xuất dữ liệu hoặc thực thi một số business logic.
- Sau khi hoàn tất xử lý Action Method đó sẽ trả về 1 Action Result. ASP.NET MVC cung cấp nhiều Action Result, trong đó có 1 Action Result đặc biệt là View Result. Action Result này có nhiệm vụ làm việc với 1 View nhất định để tạo ra mã HTML để trả về cho web browser và người dùng có thể nhận được kết quả xử lý.
- View Engine là thành phần thực hiện hiển thị 1 View, đi kèm với ASP.NET MVC là Webform View Engine tức là chúng ta có thể viết View bằng ASPX.



Hình 2. 3. Mô tả hoạt động của MVC trong ASP.NET

2.1.4. Tại sao phải sử dụng ASP.NET Core?

- Yêu cầu về xây dựng các ứng dụng thương mại điện tử ngày càng được phát triển và nâng cao. Khi đó ASP không còn đáp ứng được yêu cầu đặt ra. ASP được thiết kế riêng biệt và nằm ở tầng phía trên hệ điều hành Windows và Internet Information Service, do đó các công dụng của nó hết sức rời rạc và giới hạn. ASP.NET đưa ra một phương pháp phát triển hoàn toàn mới khác hẳn so với ASP trước kia và đáp ứng được các yêu cầu đặt ra.

2.1.5. Sự khác biệt giữa ASP.NET và ASP.NET Core

ASP.NET	ASP.NET Core
Xây dựng Asp.Net cho Windows	Asp.Net Core Build cho Windows, Mac và Linux
Asp.Net có hiệu suất tốt	Lỗi ASP.Net có hiệu năng cao hơn so với ASP.Net 4x.

Nó chạy trên .Net Framework hoặc thường được gọi là .Net Framework đầy đủ	Nó chạy trên .Net Core và Full .Net Framework.
Asp.Net hỗ trợ WebForm, Asp.Net MVC và Asp.Net WebAPI.	Asp.Net Core không hỗ trợ WebForm. Nó hỗ trợ các trang web MVC, API Web và Asp.Net ban đầu được thêm vào .Net Core 2.0.
Asp.Net đã sử dụng IIS duy nhất phụ thuộc vào System.web.dll.	Asp.Net Core không phụ thuộc System.web.dll và IIS.
Hỗ trợ C #, VB và nhiều ngôn ngữ khác và cũng hỗ trợ WCF, WPF và WF	Chỉ hỗ trợ ngôn ngữ C #, F #. Hỗ trợ VB để thêm một thời gian ngắn và không hỗ trợ WCF, WPF và WF nhưng hỗ trợ cho các thư viện máy khách WCF có sẵn.
Ứng dụng Asp.Net MVC đã thêm Web.config, Global.asax, Khởi động ứng dụng.	Core không hỗ trợ các tệp Web.config và Global.asax. Nó đang hỗ trợ appsinstall.json.
Hỗ trợ container không tốt hơn ứng dụng ASP.Net Core.	Container hỗ trợ phù hợp nhất cho việc triển khai như Docker.
Tất cả các phiên bản chính được hỗ trợ	Hỗ trợ Core từ Visual Studio 2015 cập nhật 3 và phiên bản hiện tại VS 2017.

Chúng ta cần biên dịch lại sau khi thay đổi mã.	Làm mới trình duyệt lỗi sẽ biên dịch và thực thi mã không cần biên dịch lại.
---	--

2.2. Phân tích công nghệ

Xây dựng web UI và web API với ASP.NET Core MVC

- Người dùng có thể tạo ra những ứng dụng web có thể thực hiện testing theo mô hình MVC.
- Xây dựng HTTP services hỗ trợ nhiều định dạng và đầy đủ những hỗ trợ cho nội dung của dữ liệu trả về.
- Razor cung cấp ngôn ngữ tạo Views hiệu quả.
- Tag Helper cho phép code server side tham gia vào quá trình tạo và render phần tử HTML.
- Model Binding có thể tự động ánh xạ dữ liệu từ HTTP request tới những tham số của method action.
- Model validation thực hiện validate client và server một cách tự động.

Phát triển client-side

- ASP.NET Core được thiết kế tích hợp với nhiều client side frameworks một cách liên tục bao gồm AngularJS, Bootstrap và KnockoutJS.

Ngoài ra, ASP.NET Core còn được đánh giá cao khi mang trong mình nhiều lợi ích khác như:

- Là khung mẫu và gọn gàng hơn kiến trúc nhiều tầng.
- Là một khung công tác với mã nguồn mở. Dễ dàng tạo ứng dụng ASP.NET đa nền tảng trên Windows, Mac và Linux.
- Cấu hình là môi trường sẵn sàng cho đám mây.
- Khả năng lưu trữ trên: IIS, HTTP.sys, Kestrel, Nginx, Apache và Docker.

2.3. Ưu điểm và khuyết điểm của ASP.NET Core MVC

Ưu điểm:

- ASP.NET Core được thay đổi một số kiến trúc nên dẫn đến Modular Framework nhỏ hơn.
- ASP.NET Core không còn phụ thuộc vào nền tảng system.web.dll, Framework này dựa trên một tập hợp nhiều yếu tố của Nuget Packages.
- Bạn có thể tối ưu ứng dụng của mình dễ dàng thông qua những Nuget Packages cần thiết.
- Những ứng dụng web được thiết kế trên nền tảng ASP.NET Core sẽ ít tiêu hao dung lượng bộ nhớ, bảo mật chặt chẽ, tốc độ thực thi nhanh, hiệu năng hoạt động tốt và ổn định.
- ASP.NET Core giúp giảm dịch vụ, dễ bảo trì ứng dụng và tiết kiệm chi phí.
- ASP chỉ sử dụng VBScript và JavaScript mà không sử dụng được các ngôn ngữ mạnh khác: Visual Basic, C++... Trong khi đó ASP.NET cho phép viết nhiều ngôn ngữ: VBScript, JavaScript, C#, Visual Basic.Net, ...
- ASP.Net sử dụng phong cách lập trình mới: Code behide. Tách code riêng, giao diện riêng → dễ đọc, dễ quản lý và bảo trì.
- Trong các trang ASP chúng ta phải viết mã để kiểm tra dữ liệu nhập từ người dùng, ASP.NET hỗ trợ các validation controls để kiểm tra chúng ta không cần viết mã, ...
- Hỗ trợ phát triển Web được truy cập trên các thiết bị di động: PocketPC, Smartphone...
- Hỗ trợ nhiều web server control.
- Hỗ trợ bắt lỗi (debug) JavaScript.
- Cho phép người dùng thiết lập giao diện trang Web theo sở thích cá nhân sử dụng Theme, Profile, WebPart.
- Tăng cường các tính năng bảo mật (security).
- Hỗ trợ kỹ thuật truy cập dữ liệu mới LINQ.
- Hỗ trợ kỹ thuật xây dựng các ứng dụng đa phương tiện SilverLight.
- Hỗ trợ kỹ thuật bất đồng bộ ASP.Net Ajax.

- ASP.Net hỗ trợ mạnh mẽ bộ thư viện phong phú và đa dạng của .Net Framework, làm việc với XML, Web Service, truy cập cơ sở dữ liệu qua ADO.Net, ...
- ASPX và ASP có thể cùng hoạt động trong 1 ứng dụng.
- Kiến trúc lập trình giống ứng dụng trên Windows.
- Hỗ trợ quản lý trạng thái của các control.
- Tự động phát sinh mã HTML cho các Server control tương ứng với từng loại Browser.
- Hỗ trợ nhiều cơ chế Cache.
- Triển khai cài đặt: Không cần lock, không cần đăng ký DLL, cho phép nhiều hình thức cấu hình ứng dụng.
- Hỗ trợ quản lý ứng dụng ở mức toàn cục: Global.aspx có nhiều sự kiện hơn, quản lý session trên nhiều Server, không cần Cookies.
- Trang ASP.Net được biên dịch trước. Thay vì phải đọc và thông dịch mỗi khi trang web được yêu cầu, ASP.Net biên dịch những trang web động thành những tập tin DLL mà Server có thể thi hành nhanh chóng và hiệu quả. Yếu tố này làm gia tăng tốc độ thực thi so với kỹ thuật thông dịch của ASP.

Nhược điểm:

- Bảo mật ứng dụng chưa thực sự được tối ưu hoá vì ASP.NET không cung cấp hoàn toàn quyền truy cập dữ liệu cho các ứng dụng.
- Tốn kém chi phí bởi các yêu cầu giấy phép SQL Server, Visual Studio hay Windows Server.
- Các ứng dụng bị hạn chế hoạt động nếu có thay đổi xảy ra và cần trợ giúp từ GitHub.
- Đối với dự án nhỏ việc áp dụng mô hình MVC gây công kênh, tốn thời gian trong quá trình phát triển.
- Tốn thời gian trung chuyển dữ liệu của các thành phần.

CHƯƠNG 3: PHÂN TÍCH CHƯƠNG TRÌNH

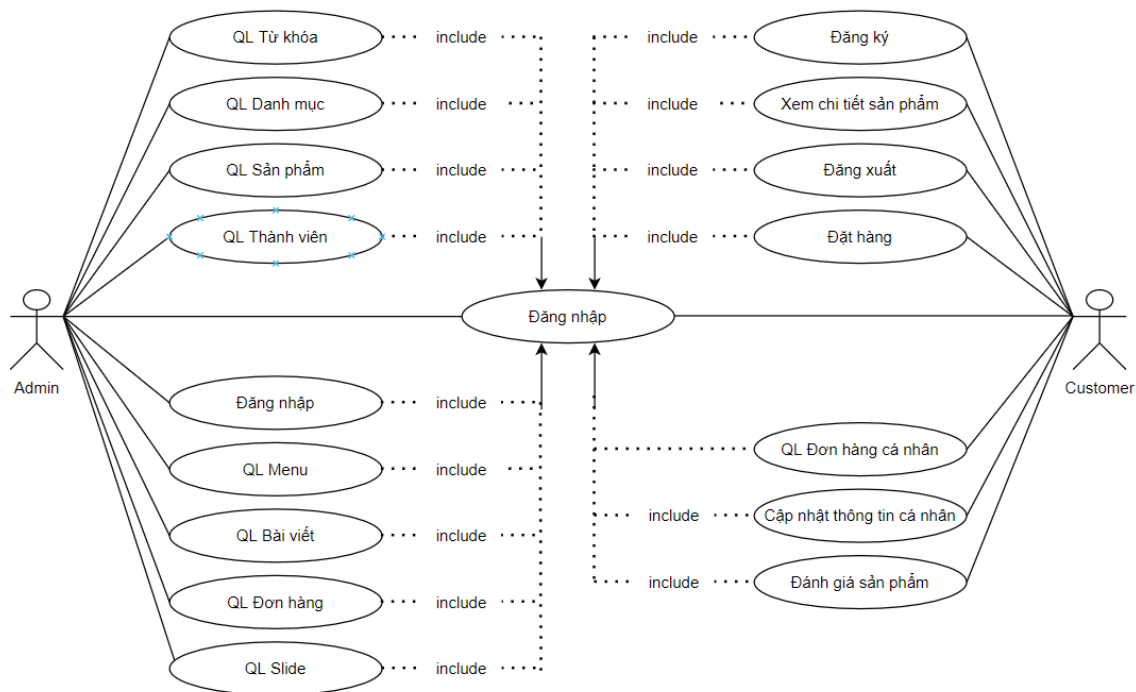
3.1. Phân tích chức năng chương trình

- Website quản lý lựa chọn cây trồng cung cấp loại sản phẩm chất lượng cho người dùng, và đang có chiến lược mở rộng sang hình thức kinh doanh online. Website phục vụ nhiều loại sản phẩm cây trồng, với nhiều người tiêu dùng khác nhau, bên cạnh đó web còn có nhiều hình thức quan trọng là buôn bán, giao dịch... Vì vậy, cần có một hệ thống quản lý chuyên nghiệp để giảm thiểu các rủi ro, lưu trữ lượng lớn dữ liệu và dễ dàng trong việc quản lý.
- Người quản lý cần quản lý các thông tin về các loại sản phẩm của cửa hàng, và các sản phẩm cụ thể của cửa hàng đang bày bán như ,.... Lưu các thông tin về sản phẩm như tên sản phẩm, loại sản phẩm, giá, hình ảnh minh họa... để người dùng lựa chọn theo các tiêu chí của bản thân.
- Quản lý cũng cần lưu lại các thông tin menu và bài viết của cửa hàng để quảng cáo cho các thông tin sản phẩm mà cửa hàng đang đẩy mạnh. Có các trang quảng cáo cho những sản phẩm nổi bật, được khách hàng sử dụng nhiều, mang lại doanh thu lớn cho cửa hàng, và tiếp cận nhiều hơn tới khách hàng mới.
- Khi có thắc mắc hay phản hồi về sản phẩm đã sử dụng, khách hàng liên hệ với cửa hàng. Khi khách hàng mua hàng của cửa hàng, các thông tin cơ bản của khách hàng như họ tên, số điện thoại, địa chỉ... sẽ được lưu lại để tiện cho việc liên lạc, giao hàng cũng như thực hiện tri ân; và lưu các thông tin giao dịch như các sản phẩm khách hàng mua, số lượng, giá tiền, giảm giá và tổng tiền....
- Khách hàng sau khi sử dụng sản phẩm, có thể phản hồi các ý kiến tới cửa hàng, để cửa hàng nắm bắt được chất lượng các sản phẩm và nhu cầu người dùng của mình.
- Từ những ý ở trên, có thể rút ra các yêu cầu mà hệ thống mới xây dựng cần đáp ứng:

- ✓ Hệ thống phân quyền chức năng với 2 nhóm người: người quản trị và khách hàng.
- ✓ Có các chức năng đăng ký, xem sản phẩm, xem chi tiết các sản phẩm, đánh giá sản phẩm đã mua, quản lý đơn hàng và thông tin cá nhân... với khách hàng.
- ✓ Đặc quyền quản lý các chức năng: quản lý thành viên, danh mục, sản phẩm, bài viết, menu, đơn đặt hàng,... với các người quản trị viên của hệ thống.
- ✓ Giao diện hệ thống thân thiện, hài hòa với người dùng.
- ✓ Giao diện dễ sử dụng với quản trị viên, dễ bảo trì.

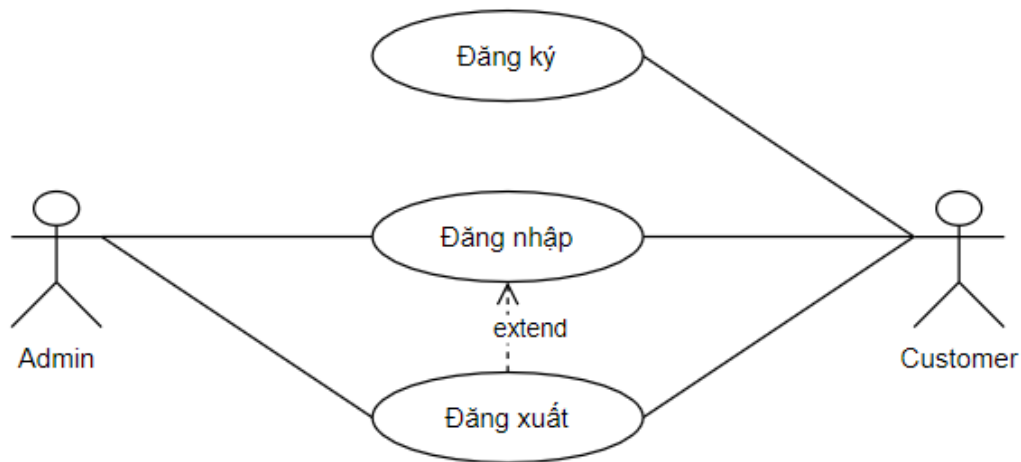
3.2. Biểu đồ Use Case

3.2.1. Use case tổng quát của hệ thống



Hình 3. 1. Use case tổng quát

3.2.2. Use case hệ thống



Hình 3. 2 Use case hệ thống

❖ **Đăng ký:**

- Tác nhân: Khách hàng.
- Mô tả tổng quát: Khách hàng truy cập website có thể đăng ký thành viên, xem các dịch vụ và chi tiết dịch vụ, tìm kiếm các dịch vụ mong muốn hỗ trợ.
- Điều kiện đầu vào: Khi khách hàng truy cập với nhu cầu đặt dịch vụ.

➤ Dòng sự kiện chính:

- Khách hàng truy cập vào trang web của cửa hàng: hệ thống hiển thị danh sách các dịch vụ.
- Khi khách hàng nhấn chọn một dịch vụ: hệ thống hiển thị danh sách chi tiết của dịch vụ.
- Khách hàng nhập từ khóa tìm kiếm và chọn tìm kiếm: hệ thống lấy dữ liệu, và hiển thị các dịch vụ theo từ khóa nhập vào.
- Khách hàng chọn đăng ký: Nhập các thông tin về tài khoản và mật khẩu. Nếu đúng sẽ lưu thông tin khách hàng, nếu sai sẽ thực hiện dòng sự kiện rẽ nhánh A1.

➤ Dòng sự kiện rẽ nhánh A1:

- Hệ thống thông báo việc nhập dữ liệu không hợp lệ.
 - Nhập lại thông tin.
 - Quay lại trang đăng ký.
- Điều kiện đầu ra: Lưu tài khoản khách hàng đã đăng ký.

❖ **Đăng nhập:**

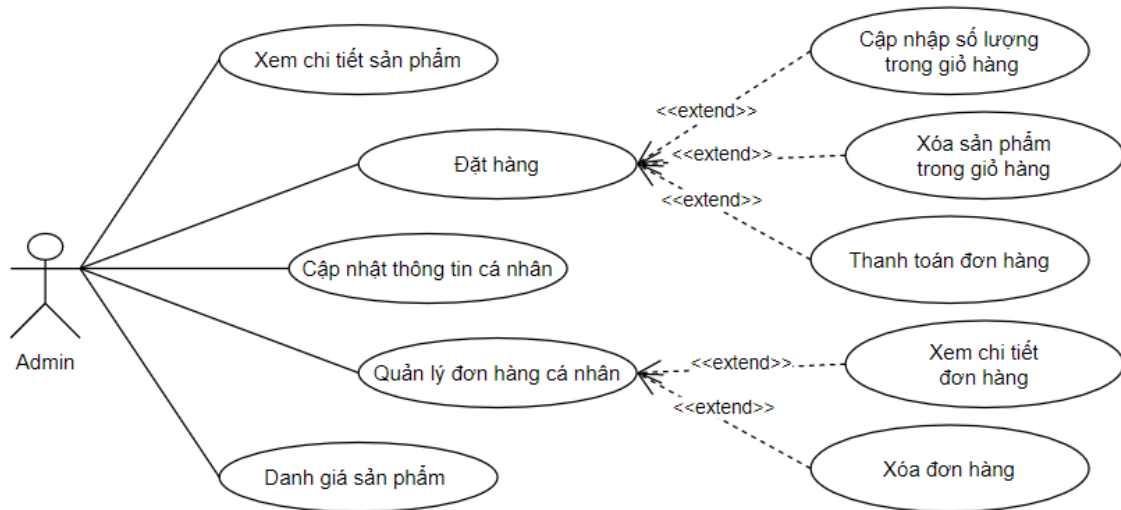
- Tác nhân: Người quản lý và khách hàng.
- Mô tả tổng quát: Người quản lý và khách hàng đăng nhập vào hệ thống.
- Điều kiện đầu vào: UseCase sử dụng khi người dùng muốn đăng nhập vào hệ thống.
 - Dòng sự kiện chính:
 - Hệ thống yêu cầu nhập tài khoản và mật khẩu.
 - Người dùng nhập tài khoản và mật khẩu.
 - Hệ thống kiểm tra tên và mật khẩu vừa nhập:
 - Nếu đúng sẽ kiểm tra quyền, và hiển thị quyền tương ứng của tài khoản.
 - Nếu sai thực hiện luồng sự kiện phụ A1.
 - Dòng sự kiện phụ A1:
 - Hệ thống sẽ thông báo việc nhập dữ liệu không hợp lệ.
 - Người dùng nhập lại tài khoản và mật khẩu.
 - Quay lại bước 3 của luồng sự kiện chính, hoặc hủy bỏ việc đăng nhập, khi đó ca sử dụng kết thúc.

❖ **Đăng xuất:**

- Tác nhân: Người quản lý và khách hàng.
- Mô tả tổng quát: Người quản lý và khách hàng muốn thoát tài khoản khỏi hệ thống.
- Điều kiện đầu vào: UseCase sử dụng khi người quản trị và thành viên chọn đăng xuất.
 - Dòng sự kiện chính:

- Hệ thống hủy việc lưu thông tin đăng nhập.
- Quay về màn hình trang chủ.

3.2.3. Use case của Customer



Hình 3. 3 Use case của Customer

❖ *Xem chi tiết sản phẩm:*

- Tác nhân: Khách hàng.
- Mô tả tổng quát: Khách hàng muốn xem chi tiết một sản phẩm.
- Điều kiện đầu vào: UseCase sử dụng khi khách hàng chọn một sản phẩm.
 - Dòng sự kiện chính:
 - Hệ thống lấy thông tin chi tiết của sản phẩm theo mã.
 - Hiển thị form chi tiết sản phẩm.

❖ **Đặt hàng:**

- Tác nhân: Khách hàng.
- Mô tả tổng quát: Các khách hàng muốn đặt mua sản phẩm của cửa hàng.
- Điều kiện đầu vào: UseCase sử dụng khi bắt đầu đăng nhập thành công hệ thống.
 - Dòng sự kiện chính:
 - Ca sử dụng bắt đầu khi khách hàng đăng nhập vào hệ thống.
 - Chọn giỏ hàng.
 - Hệ thống hiển thị chức năng: Cập nhật số lượng trong giỏ hàng, xóa sản phẩm trong giỏ và thanh toán đơn hàng:
 - Cập nhật số lượng trong giỏ hàng:
 - ✓ Người dùng chọn số lượng sản phẩm mong muốn.
 - ✓ Chọn cập nhật.
 - ✓ Hệ thống kiểm tra số lượng sản phẩm còn lại trong cửa hàng:
 - Nếu còn hàng sẽ cập nhật số lượng trong giỏ hàng.
 - Nếu sai, thực hiện luồng sự kiện rẽ nhánh A1.
 - Xóa sản phẩm trong giỏ hàng:
 - ✓ Người dùng chọn sản phẩm muốn xóa.
 - ✓ Chọn xóa sản phẩm.
 - ✓ Hệ thống xóa sản phẩm khỏi giỏ hàng của khách hàng.
 - Thanh toán đơn hàng:
 - ✓ Khách hàng chọn thanh toán các sản phẩm trong giỏ hàng.
 - ✓ Hệ thống hiển thị form thanh toán.
 - ✓ Khách hàng nhập các thông tin cần thiết, và chọn đặt hàng.
 - ✓ Hệ thống kiểm tra các trường thông tin:
 - Nếu thỏa mãn thực hiện cập nhật thông tin đơn hàng và chi tiết vào cơ sở dữ liệu.

- Nếu không thì yêu cầu nhập lại.

➤ Dòng sự kiện rẽ nhánh A1:

- Hệ thống báo số lượng sản phẩm không đủ.
- Quay lại bước 2 của dòng sự kiện chính.

❖ **Quản lý đơn hàng cá nhân:**

- Tác nhân: Khách hàng.
- Mô tả tổng quát: Khách hàng có thể xem lại các đơn hàng mình đã đặt tại cửa hàng.
- Điều kiện đầu vào: Khách hàng đăng nhập thành công hệ thống, chọn mục các đơn hàng.

➤ Dòng sự kiện chính:

- Người dùng đăng nhập hệ thống.
- Truy cập hệ thống chọn mục các đơn hàng.
- Hệ thống hiển thị danh sách các đơn hàng, cùng với các chức năng người dùng có thể thực hiện: Xóa đơn hàng và xem chi tiết đơn hàng:

▫ Xóa đơn hàng:

✓ Chọn đơn hàng muốn Xóa.

✓ Chọn xóa đơn hàng.

✓ Kiểm tra đơn hàng:

- Nếu đơn hàng chưa giao dịch, thì thực hiện xóa đơn hàng.

- Nếu đơn hàng đã có giao dịch, thực hiện luồng sự kiện rẽ nhánh A1.

▫ Xem chi tiết đơn hàng:

✓ Chọn đơn hàng muốn xem chi tiết.

✓ Chọn xem chi tiết đơn hàng.

✓ Hệ thống chuyển sang form chi tiết đơn hàng.

➤ Dòng sự kiện rẽ nhánh A1:

- Thông báo lỗi.
- Chuyển tới bước 3 của dòng sự kiện chính.
- Điều kiện đầu ra: Các thông tin về đơn hàng được cập nhật vào cơ sở dữ liệu.

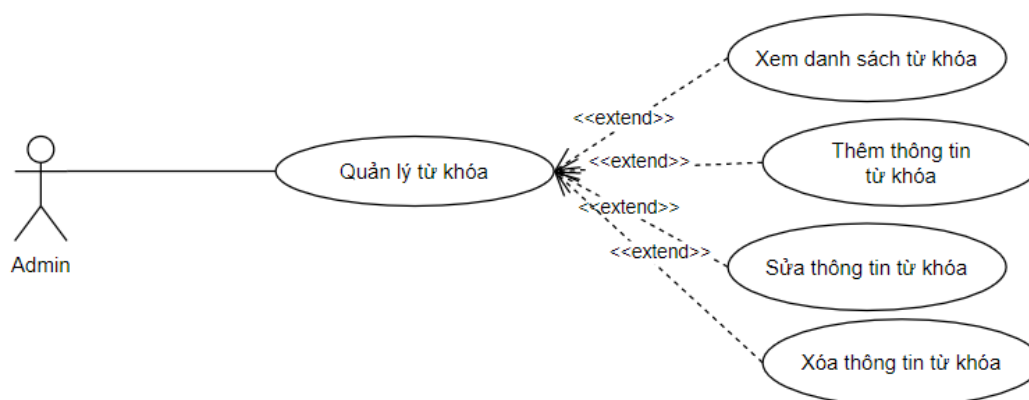
❖ **Cập nhật thông tin cá nhân:**

- Tác nhân: Khách hàng.
- Mô tả tổng quát: Khách hàng cập nhật thông tin cá nhân của mình.
- Điều kiện đầu vào: Khi khách hàng đăng nhập hệ thống thành công và chọn thông tin cá nhân.
 - Khách hàng chọn cập nhật thông tin cá nhân.
 - Hệ thống hiển thị form cập nhật thông tin.
 - Khách hàng nhập các thông tin muốn sửa đổi, và chọn cập nhật.
 - Hệ thống kiểm tra các trường dữ liệu:
 - Nếu đầy đủ và đúng định dạng thì chuyển qua bước tiếp theo.
 - Nếu không thì thực hiện dòng sự kiện rẽ nhánh A1.
 - Hệ thống thông báo cập nhật thông tin cá nhân thành công.
 - Hệ thống cập nhật thông tin cá nhân vào cơ sở dữ liệu.
- Dòng sự kiện rẽ nhánh A1:
 - Hệ thống thông báo việc nhập dữ liệu không hợp lệ.
 - Nhập lại thông tin.
 - Quay lại bước 4 của sự kiện chính.
- Điều kiện đầu ra: Thông tin của khách hàng được cập nhật vào cơ sở dữ liệu.

❖ **Đánh giá sản phẩm:**

- Tác nhân: Khách hàng.
- Mô tả tổng quát: Khách hàng đánh giá sản phẩm sau khi mua hàng.
- Điều kiện đầu vào: Khi khách hàng đăng nhập hệ thống thành công và chọn đánh giá sản phẩm.
 - Khách hàng chọn đơn hàng muốn đánh giá.
 - Khách hàng đánh giá sản phẩm có trong đơn hàng, và chọn lưu.
 - Hệ thống kiểm tra các trường dữ liệu:
 - Nếu đầy đủ và đúng định dạng thì chuyển qua bước tiếp theo.
 - Nếu không thì thực hiện dòng sự kiện rẽ nhánh A1.
 - Hệ thống thông báo cập nhật đánh giá của khách hàng thành công.
 - Hệ thống lưu các đánh giá vào cơ sở dữ liệu.
- Dòng sự kiện rẽ nhánh A1:
 - Hệ thống thông báo việc nhập dữ liệu không hợp lệ.
 - Nhập lại thông tin.
 - Quay lại bước 3 của sự kiện chính.
- Điều kiện đầu ra: Các đánh giá về sản phẩm được cập nhật vào cơ sở dữ liệu.

3.2.4. Use case Quản lý từ khóa



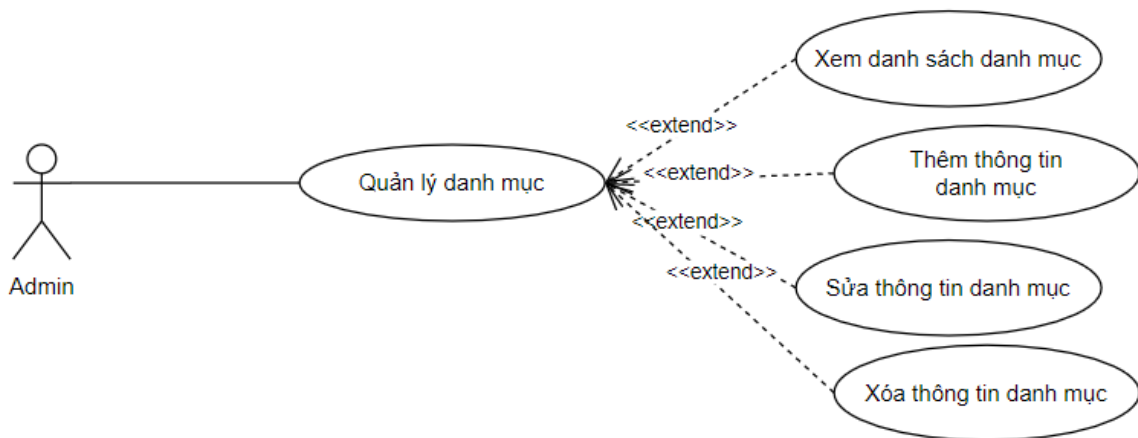
Hình 3. 4 UseCase Quản lý từ khóa

- Tác nhân: Người quản trị.

- Mô tả tổng quát: Người quản trị quản lý các từ khóa.
- Điều kiện đầu vào: Người dùng đăng nhập thành công, với quyền quản trị.
- Dòng sự kiện chính:
 - Quản trị đăng nhập hệ thống.0
 - Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, và chọn quản lý từ khóa.
 - Hệ thống hiển thị danh sách từ khóa, và các chức năng: thêm, sửa và xóa và tìm kiếm từ khóa:
 - Xem danh sách từ khóa:
 - ✓ Quản trị chọn quản lý từ khóa.
 - ✓ Hệ thống hiển thị danh sách từ khóa.
 - Thêm thông tin từ khóa:
 - ✓ Hệ thống hiển thị form nhập thông tin từ khóa.
 - ✓ Người quản lý nhập thông tin từ khóa.
 - ✓ Nhập nút thêm từ khóa.
 - ✓ Hệ thống kiểm tra các trường dữ liệu:
 - Nếu thỏa mãn, thì thực hiện bước tiếp theo.
 - Nếu không, thì thực hiện luồng sự kiện A1.
 - ✓ Hệ thống thông báo thêm từ khóa thành công.
 - ✓ Hệ thống lưu lại thông tin từ khóa.
 - Sửa thông tin từ khóa:
 - ✓ Hệ thống hiển thị form cập nhật thông tin từ khóa.
 - ✓ Người quản lý nhập thông tin cần thay đổi.
 - ✓ Nhập nút cập nhật từ khóa.
 - ✓ Hệ thống kiểm tra các trường dữ liệu:
 - Nếu thỏa mãn, thì thực hiện bước tiếp theo.
 - Nếu không, thì thực hiện luồng sự kiện A1.
 - ✓ Hệ thống thông báo cập nhật từ khóa thành công.

- ✓ Hệ thống cập nhật thông tin từ khóa.
- Xóa từ khóa:
 - ✓ Người quản lý chọn thông tin từ khóa cần xóa.
 - ✓ Hệ thống kiểm tra:
 - Nếu không có ràng buộc, thực hiện bước tiếp theo.
 - Ngược lại, hủy xóa.
- Dòng sự kiện rẽ nhánh A1:
 - Hệ thống báo lỗi.
 - Quản trị thực hiện nhập lại dữ liệu.
 - Quay lại bước 3 của sự kiện phụ.
- Điều kiện đầu ra: Danh sách các từ khóa được cập nhật đầy đủ vào cơ sở dữ liệu.

3.2.5. Use case Quản lý danh mục



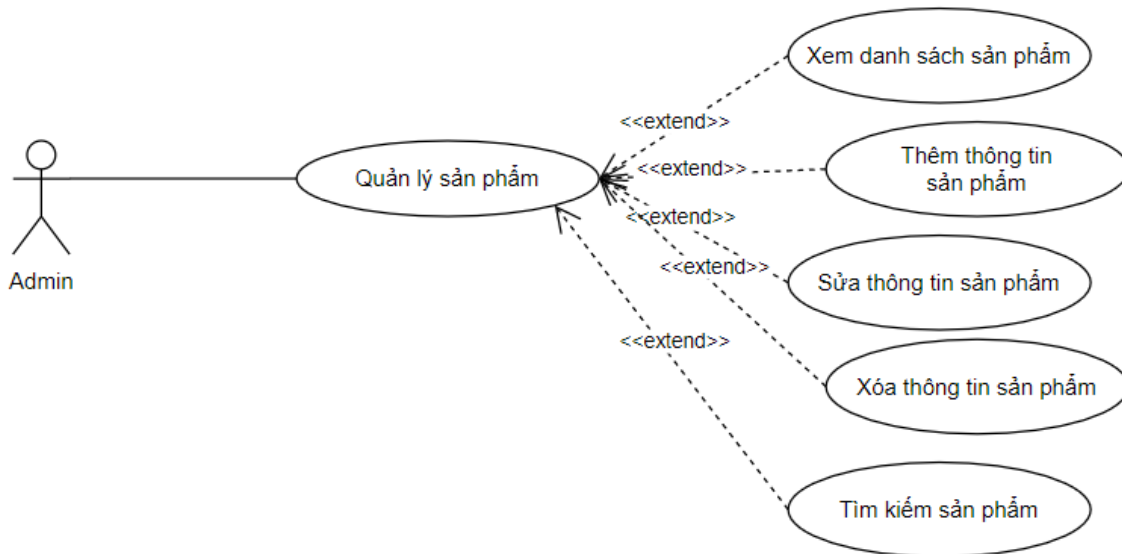
Hình 3. 5 Use case Quản lý danh mục

- Tác nhân: Người quản trị.
- Mô tả tổng quát: Người quản trị quản lý các danh mục, giúp quản lý các sản phẩm dễ dàng.
- Điều kiện đầu vào: Người dùng đăng nhập thành công, với quyền quản trị.
- Dòng sự kiện chính:
 - Quản trị đăng nhập hệ thống.

- Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, và chọn quản lý danh mục.
 - Hệ thống hiển thị danh sách danh mục, và các chức năng: thêm, sửa và xóa và tìm kiếm danh mục:
 - Thêm danh mục:
 - ✓ Hệ thống hiển thị form nhập thông tin danh mục.
 - ✓ Người quản lý nhập thông tin danh mục.
 - ✓ Nhập nút thêm danh mục.
 - ✓ Hệ thống kiểm tra các trường dữ liệu:
 - Nếu thỏa mãn, thì thực hiện bước tiếp theo.
 - Nếu không, thì thực hiện luồng sự kiện A1.
 - ✓ Hệ thống thông báo thêm danh mục thành công.
 - ✓ Hệ thống lưu lại thông tin danh mục.
 - Sửa danh mục:
 - ✓ Hệ thống hiển thị form cập nhật thông tin danh mục.
 - ✓ Người quản lý nhập thông tin cần thay đổi.
 - ✓ Nhập nút cập nhật danh mục.
 - ✓ Hệ thống kiểm tra các trường dữ liệu:
 - Nếu thỏa mãn, thì thực hiện bước tiếp theo.
 - Nếu không, thì thực hiện luồng sự kiện A1.
 - ✓ Hệ thống thông báo cập nhật danh mục thành công.
 - ✓ Hệ thống cập nhật thông tin danh mục.
 - Xóa danh mục:
 - ✓ Người quản lý chọn thông tin danh mục cần xóa.
 - ✓ Hệ thống kiểm tra:
 - Nếu không có ràng buộc, thực hiện bước tiếp theo.
 - Ngược lại, hủy xóa.
- Dòng sự kiện rẽ nhánh A1:

- Hệ thống báo lỗi.
 - Quản trị thực hiện nhập lại dữ liệu.
 - Quay lại bước 3 của sự kiện phụ.
- Điều kiện đầu ra: Danh sách các danh mục được cập nhật đầy đủ vào cơ sở dữ liệu.

3.2.6. Use case Quản lý sản phẩm



Hình 3. 6 Use case Quản lý sản phẩm

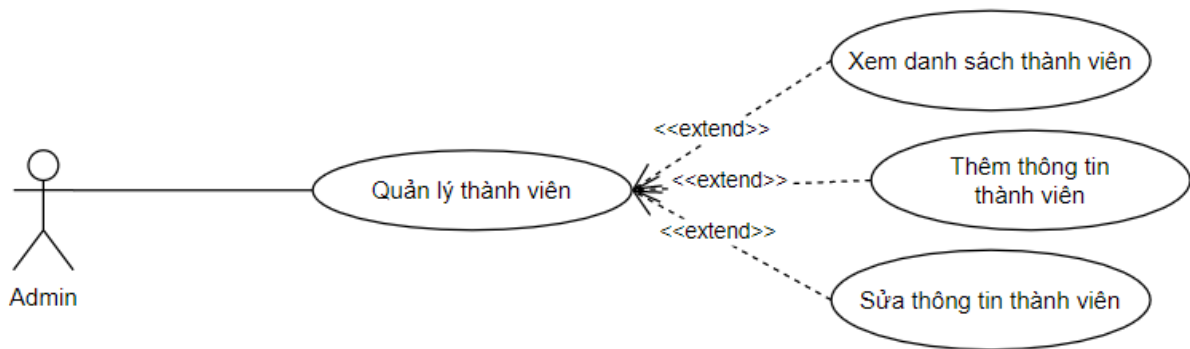
- Tác nhân: Người quản trị.
 - Mô tả tổng quát: Người quản trị quản lý các sản phẩm của cửa hàng.
 - Điều kiện đầu vào: Người dùng đăng nhập thành công, với quyền quản trị.
- Dòng sự kiện chính:
- Quản trị đăng nhập hệ thống.
 - Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, và chọn quản lý sản phẩm.
 - Hệ thống hiển thị danh sách sản phẩm và các chức năng: thêm, sửa và xóa và tìm kiếm sản phẩm:
 - Thêm thông tin sản phẩm:
 - ✓ Hệ thống hiển thị form nhập thông tin sản phẩm.

- ✓ Người quản lý nhập thông tin sản phẩm.
- ✓ Nhập nút thêm sản phẩm.
- ✓ Hệ thống kiểm tra các trường dữ liệu:
 - Nếu thỏa mãn, thì thực hiện bước tiếp theo.
 - Nếu không, thì thực hiện luồng sự kiện A1.
- ✓ Hệ thống thông báo thêm sản phẩm thành công.
- ✓ Hệ thống lưu lại thông tin sản phẩm.
- Sửa thông tin sản phẩm:
 - ✓ Hệ thống hiển thị form cập nhật thông tin sản phẩm.
 - ✓ Người quản lý nhập thông tin cần thay đổi.
 - ✓ Nhập nút cập nhật sản phẩm.
 - ✓ Hệ thống kiểm tra các trường dữ liệu:
 - Nếu thỏa mãn, thì thực hiện bước tiếp theo.
 - Nếu không, thì thực hiện luồng sự kiện A1.
 - ✓ Hệ thống thông báo cập nhật sản phẩm thành công.
 - ✓ Hệ thống cập nhật thông tin sản phẩm.
- Tìm kiếm sản phẩm:
 - ✓ Nhập từ khóa cần tìm.
 - ✓ Nhấn nút tìm kiếm.
 - ✓ Hệ thống kiểm tra thông tin sản phẩm:
 - Nếu có hiển thị danh sách sản phẩm.
 - Nếu không, thực hiện sự kiện rẽ nhánh A1.
- Xóa sản phẩm:
 - ✓ Người quản lý chọn thông tin sản phẩm cần xóa.
 - ✓ Hệ thống kiểm tra:
 - Nếu không có ràng buộc, thực hiện bước tiếp theo.
 - Ngược lại, hủy xóa.

➤ Dòng sự kiện rẽ nhánh A1:

- Hệ thống báo lỗi.
 - Quản trị thực hiện nhập lại dữ liệu.
 - Quay lại bước 3 của sự kiện phụ.
- Điều kiện đầu ra: Danh sách các sản phẩm được cập nhật đầy đủ vào cơ sở dữ liệu.

3.2.7. Use case Quản lý thành viên

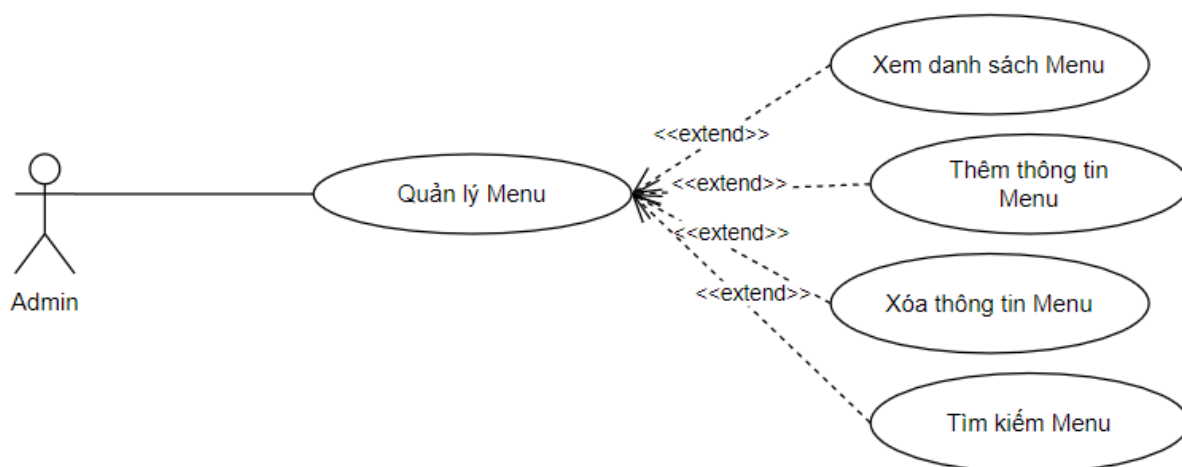


Hình 3. 7 Use case Quản lý thành viên

- Tác nhân: Người quản trị.
 - Mô tả tổng quát: Người quản trị quản lý các thành viên trong hệ thống, giúp theo dõi những thành viên mua hàng thường xuyên, để đưa ra các ưu đãi.
 - Điều kiện đầu vào: Người dùng đăng nhập thành công, với quyền quản trị.
- Dòng sự kiện chính:
- Quản trị đăng nhập hệ thống.
 - Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, và chọn quản lý thành viên.
 - Hệ thống hiển thị danh sách thành viên, và các chức năng: xóa và tìm kiếm thành viên:
 - Tìm kiếm thành viên:
 - ✓ Nhập thành viên cần tìm.
 - ✓ Nhấn nút tìm kiếm.
 - ✓ Hệ thống kiểm tra thông tin thành viên:
 - Nếu có hiển thị danh sách thành viên.

- Nếu không, thực hiện sự kiện rẽ nhánh A1.
- Xóa thành viên:
 - ✓ Người quản lý chọn thông tin thành viên cần xóa.
 - ✓ Hệ thống kiểm tra:
 - Nếu không có ràng buộc, thực hiện bước tiếp theo.
 - Ngược lại, hủy xóa.
- Dòng sự kiện rẽ nhánh A1:
 - Hệ thống báo lỗi.
 - Quản trị thực hiện nhập lại dữ liệu.
 - Quay lại bước 3 của sự kiện phụ.
- Điều kiện đầu ra: Danh sách các thành viên được cập nhật đầy đủ vào cơ sở dữ liệu.

3.2.8. Use case Quản lý menu



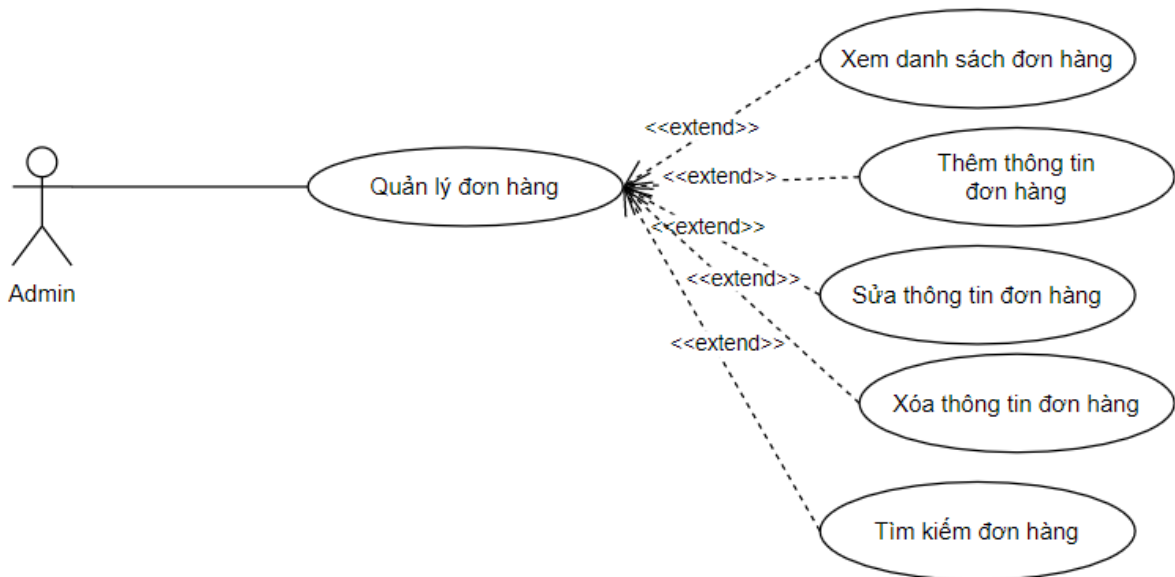
Hình 3. 8 Use case Quản lý menu

- Tác nhân: Người quản trị.
- Mô tả tổng quát: Người quản trị quản lý các menu hiển thị trên giao diện trang chủ.
- Điều kiện đầu vào: Người dùng đăng nhập thành công, với quyền quản trị.
- Dòng sự kiện chính:
 - Quản trị đăng nhập hệ thống.

- Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, và chọn quản lý menu.
- Hệ thống hiển thị danh sách menu, và các chức năng: thêm, xóa và tìm kiếm menu:
 - Thêm menu:
 - ✓ Hệ thống hiển thị form nhập thông tin menu.
 - ✓ Người quản lý nhập thông tin menu.
 - ✓ Nhập nút thêm menu.
 - ✓ Hệ thống kiểm tra các trường dữ liệu:
 - Nếu thỏa mãn, thì thực hiện bước tiếp theo.
 - Nếu không, thì thực hiện luồng sự kiện A1.
 - ✓ Hệ thống thông báo thêm menu thành công.
 - ✓ Hệ thống lưu lại thông tin menu.
 - Sửa menu:
 - ✓ Hệ thống hiển thị form cập nhật thông tin menu.
 - ✓ Người quản lý nhập thông tin cần thay đổi.
 - ✓ Nhập nút cập nhật menu.
 - ✓ Hệ thống kiểm tra các trường dữ liệu:
 - Nếu thỏa mãn, thì thực hiện bước tiếp theo.
 - Nếu không, thì thực hiện luồng sự kiện A1.
 - ✓ Hệ thống thông báo cập nhật menu thành công.
 - ✓ Hệ thống cập nhật thông tin menu.
 - Tìm kiếm menu:
 - ✓ Nhập từ khóa cần tìm.
 - ✓ Nhấn nút tìm kiếm.
 - ✓ Hệ thống kiểm tra thông tin menu:
 - Nếu có hiển thị danh sách menu.
 - Nếu không, thực hiện sự kiện rẽ nhánh A1.

- Xóa menu:
 - ✓ Người quản lý chọn thông tin menu cần xóa.
 - ✓ Hệ thống kiểm tra:
 - Nếu không có ràng buộc, thực hiện bước tiếp theo.
 - Ngược lại, hủy xóa.
- Dòng sự kiện rẽ nhánh A1:
 - Hệ thống báo lỗi.
 - Quản trị thực hiện nhập lại dữ liệu.
 - Quay lại bước 3 của sự kiện phụ.
- Điều kiện đầu ra: Danh sách các menu được cập nhật đầy đủ vào cơ sở dữ liệu.

3.2.9. Use case Quản lý đơn hàng



Hình 3. 9 Use case Quản lý đơn hàng

- Tác nhân: Người quản trị.
- Mô tả tổng quát: Người quản trị quản lý các đơn hàng của thành viên.
- Điều kiện đầu vào: Người dùng đăng nhập thành công, với quyền quản trị.
- Dòng sự kiện chính:
 - Quản trị đăng nhập hệ thống.

- Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, và chọn quản lý đơn hàng.
- Hệ thống hiển thị danh sách đơn hàng, và các chức năng: xem chi tiết, sửa và xóa và tìm kiếm đơn hàng:
 - Xem chi tiết đơn hàng:
 - ✓ Quản trị chọn đơn hàng muốn xem chi tiết.
 - ✓ Chọn xem chi tiết.
 - ✓ Hệ thống hiển thị form chi tiết đơn hàng.
 - Sửa đơn hàng:
 - ✓ Hệ thống hiển thị form cập nhật thông tin đơn hàng.
 - ✓ Người quản lý nhập thông tin cần thay đổi.
 - ✓ Nhập nút cập nhật đơn hàng.
 - ✓ Hệ thống kiểm tra các trường dữ liệu:
 - Nếu thỏa mãn, thì thực hiện bước tiếp theo.
 - Nếu không, thì thực hiện luồng sự kiện A1.
 - ✓ Hệ thống thông báo cập nhật đơn hàng thành công.
 - ✓ Hệ thống cập nhật thông tin đơn hàng.
 - Tìm kiếm đơn hàng:
 - ✓ Nhập từ khóa cần tìm.
 - ✓ Nhấn nút tìm kiếm.
 - ✓ Hệ thống kiểm tra thông tin đơn hàng:
 - Nếu có hiển thị danh sách đơn hàng.
 - Nếu không, thực hiện sự kiện rẽ nhánh A1.
 - Xóa đơn hàng:
 - ✓ Người quản lý chọn thông tin đơn hàng cần xóa.
 - ✓ Hệ thống kiểm tra:
 - Nếu không có ràng buộc, thực hiện bước tiếp theo.
 - Ngược lại, hủy xóa.

➤ Dòng sự kiện rẽ nhánh A1:

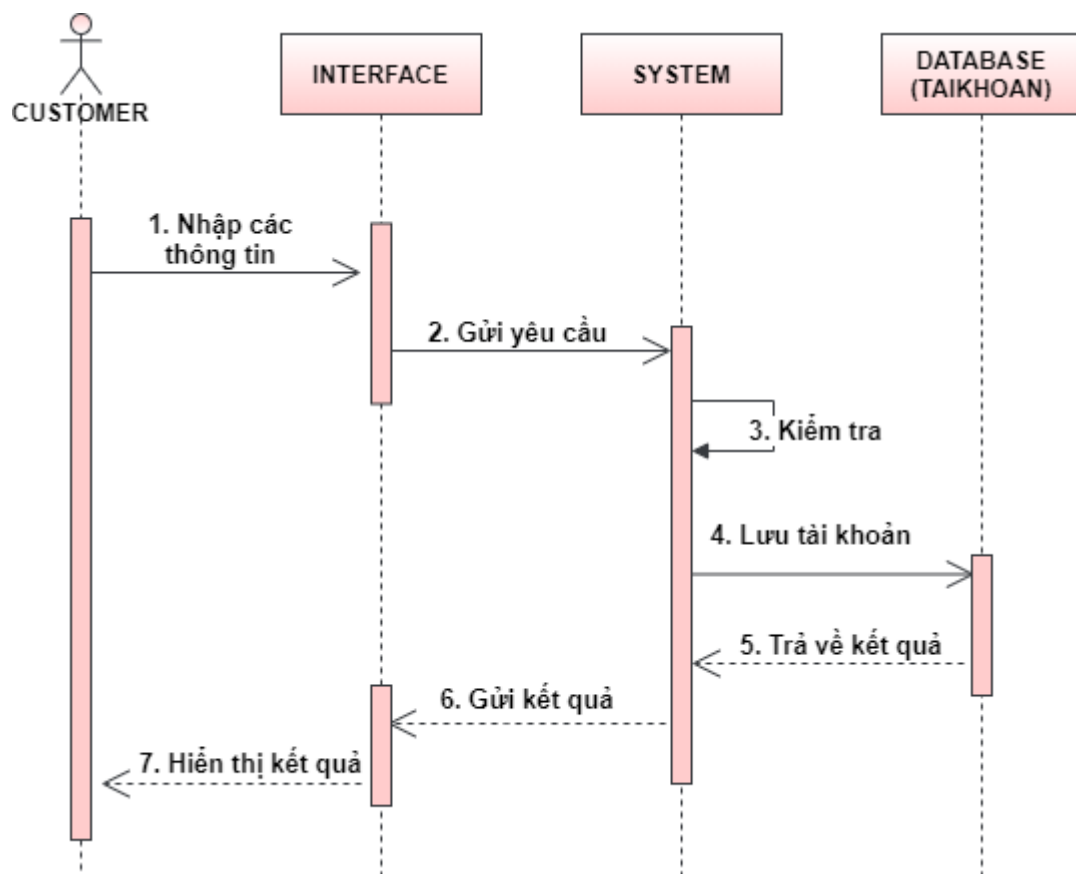
- Hệ thống báo lỗi.
- Quản trị thực hiện nhập lại dữ liệu.
- Quay lại bước 3 của sự kiện phụ.

– Điều kiện đầu ra: Danh sách các đơn hàng được cập nhật đầy đủ vào cơ sở dữ liệu.

3.3. Thiết kế và xây dựng dữ liệu

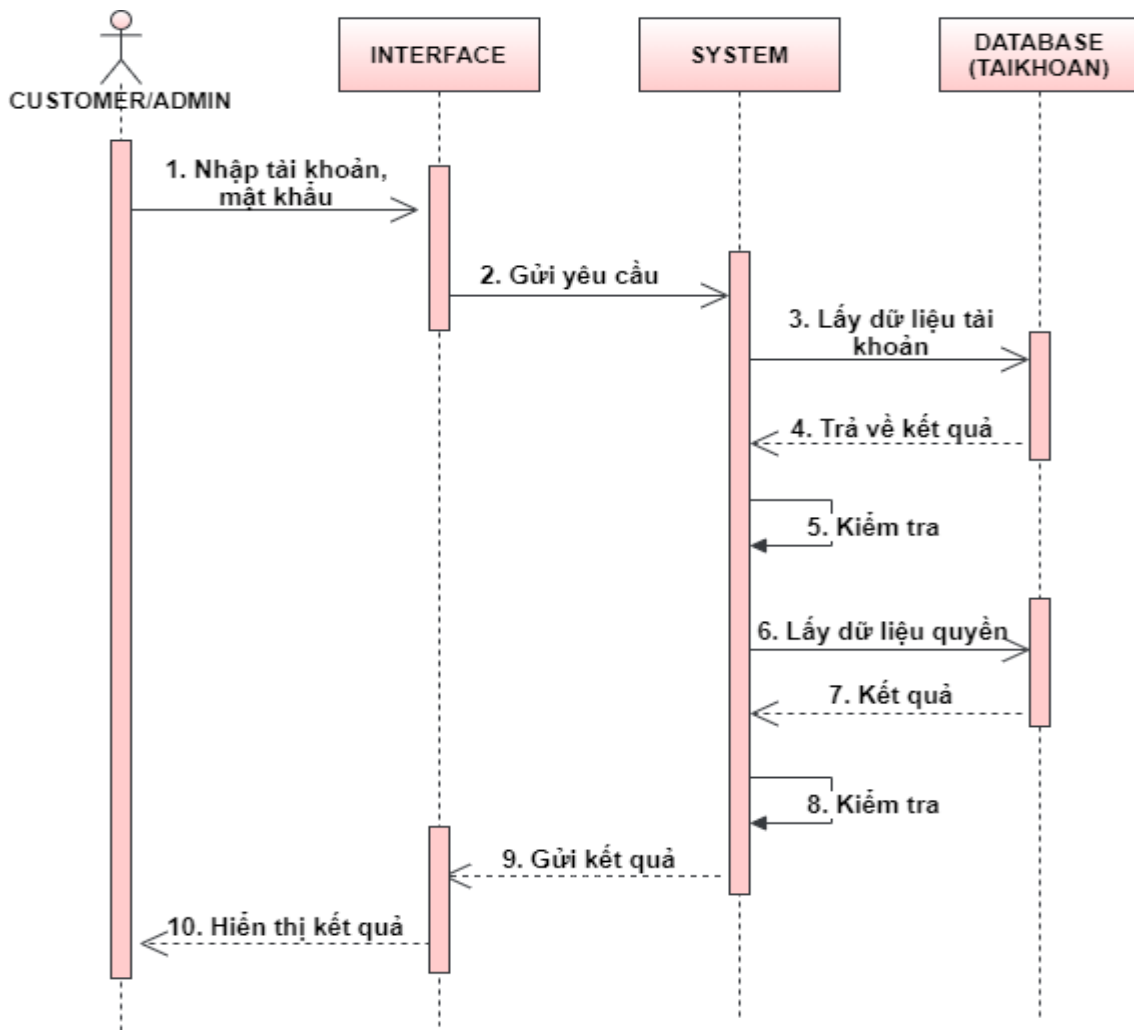
3.3.1. Biểu đồ tuần tự của hệ thống

Biểu đồ tuần tự Đăng ký:



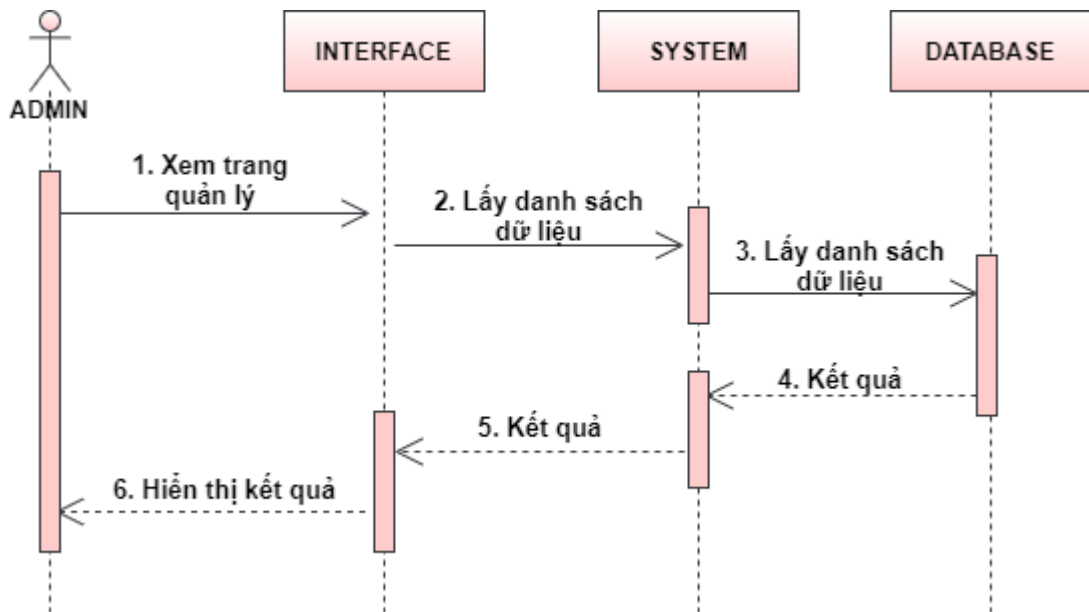
Hình 3. 10 Biểu đồ tuần tự Đăng ký

Biểu đồ tuần tự Đăng nhập:



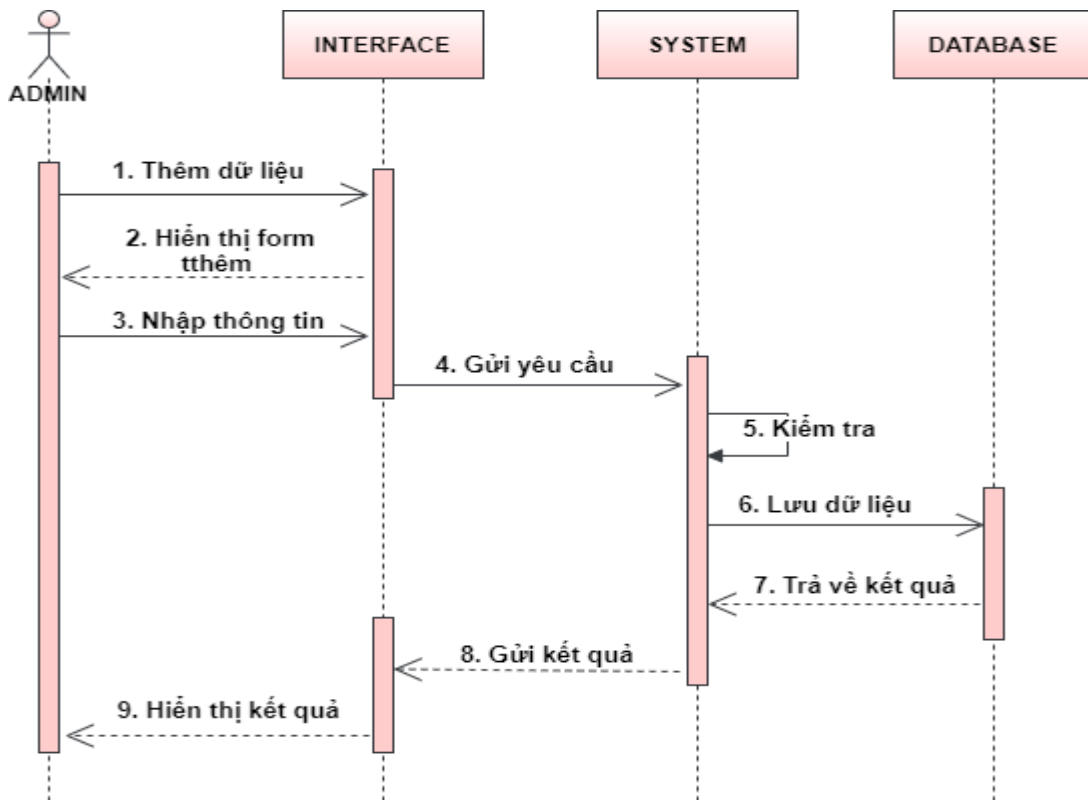
Hình 3. 11 Biểu đồ tuần tự Đăng nhập

Biểu đồ tuần tự Xem dữ liệu trang quản lý:



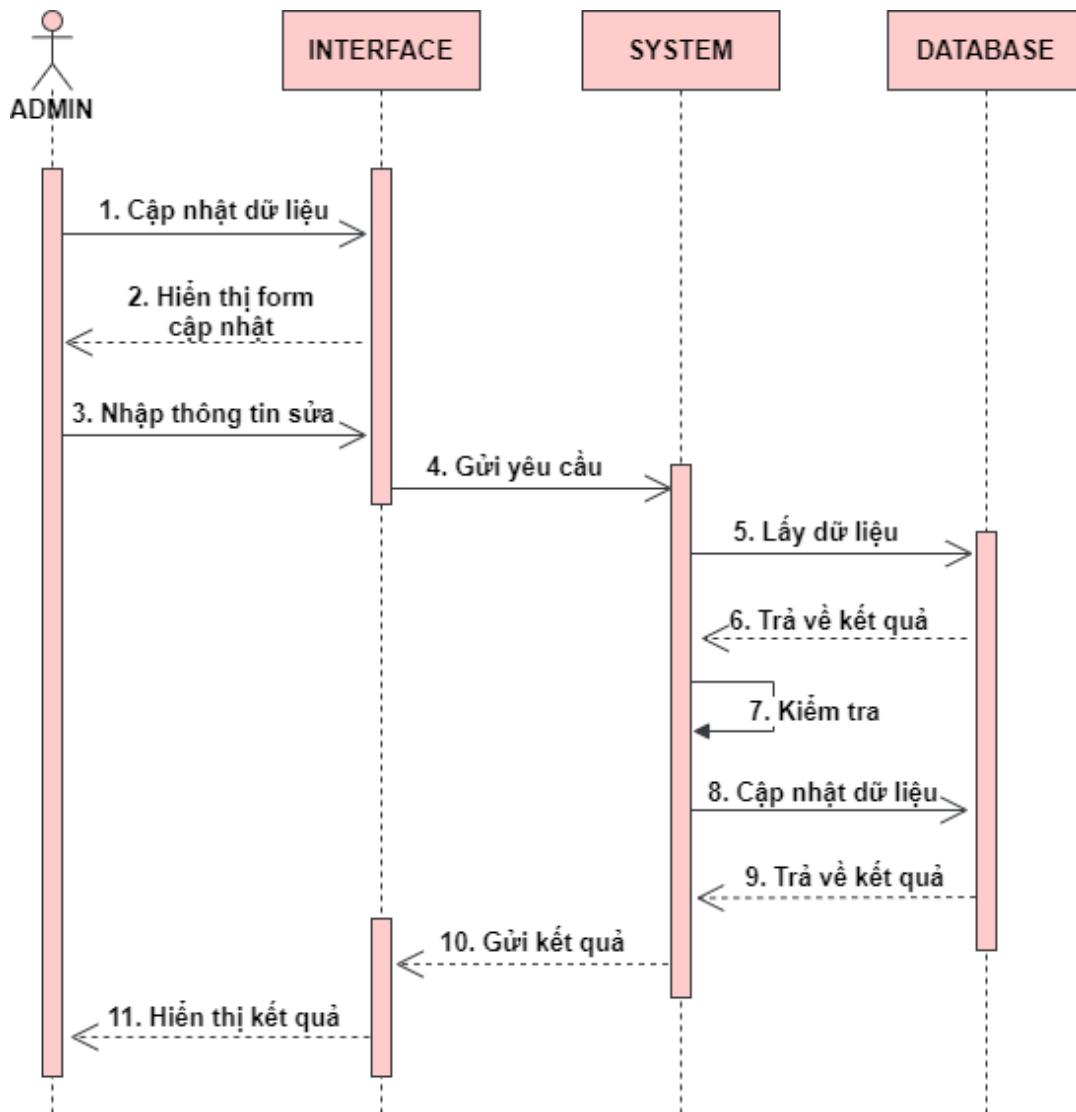
Hình 3. 12 Biểu đồ tuần tự Xem dữ liệu trang quản lý

Biểu đồ tuần tự Thêm dữ liệu trang quản lý:



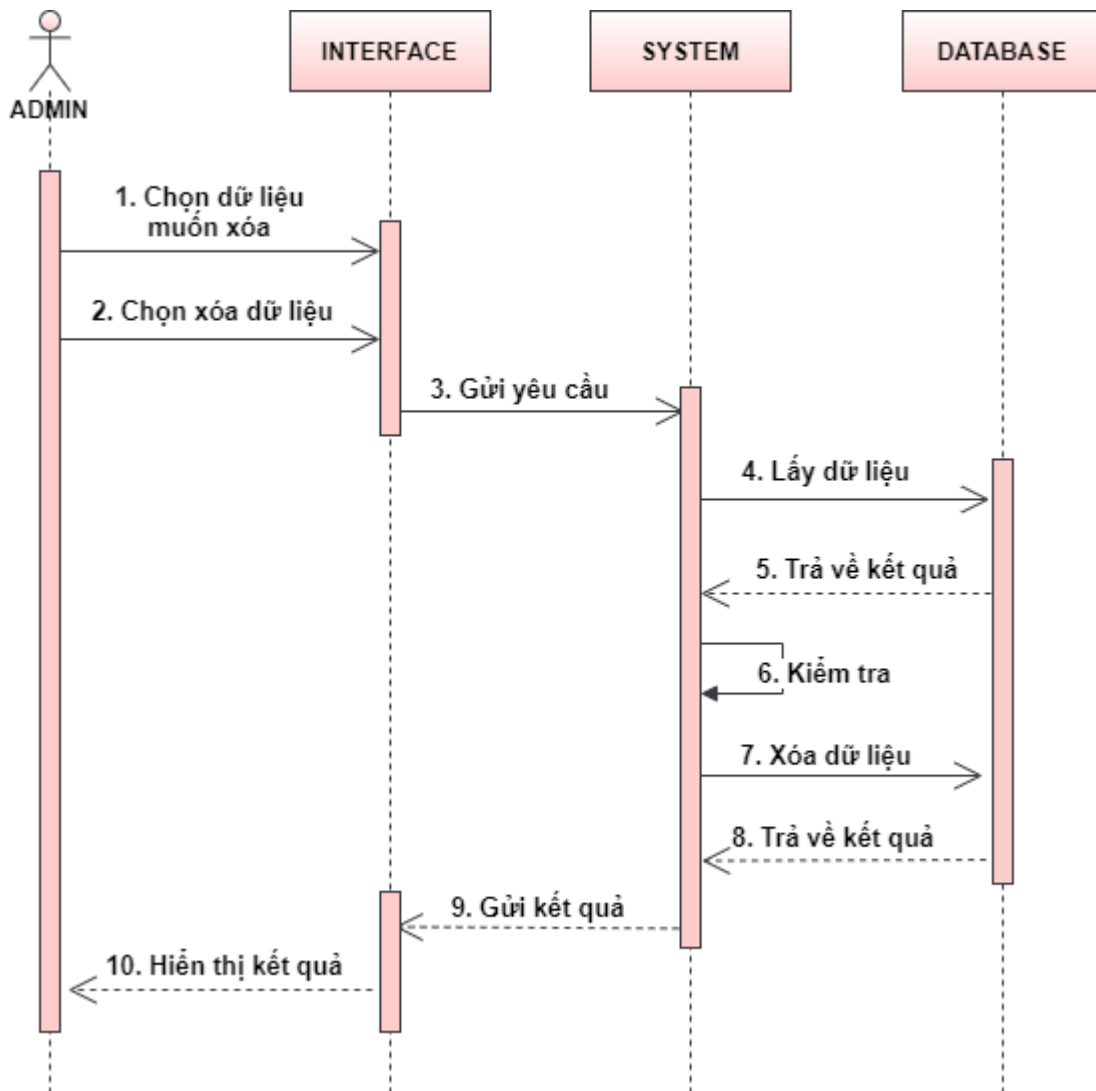
Hình 3. 13 Biểu đồ tuần tự Thêm dữ liệu trang quản lý

Biểu đồ tuần tự Cập nhật dữ liệu trang quản lý:



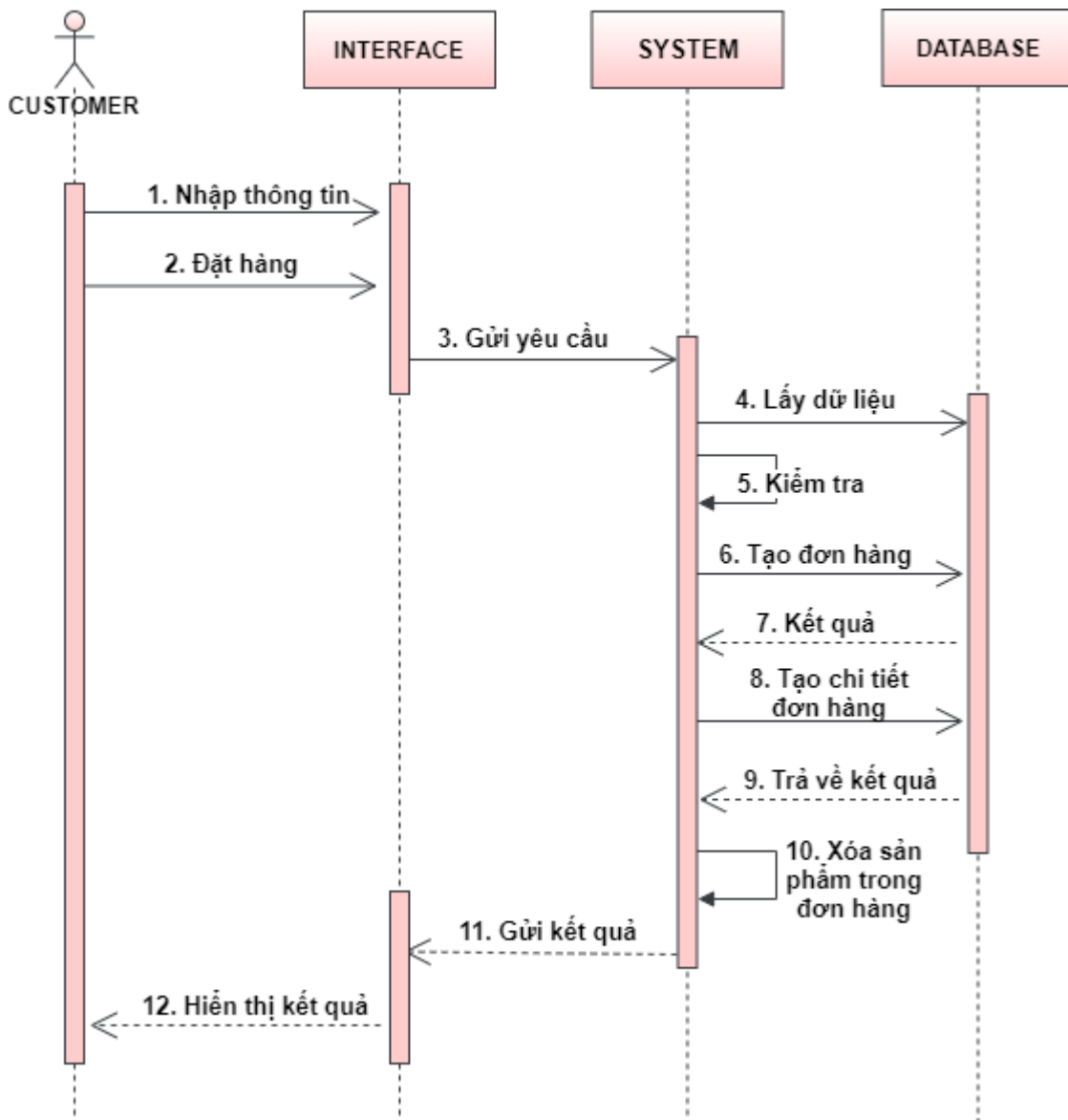
Hình 3. 14 Biểu đồ tuần tự Cập nhật dữ liệu trang quản lý

Biểu đồ tuần tự Xóa dữ liệu trang quản lý:



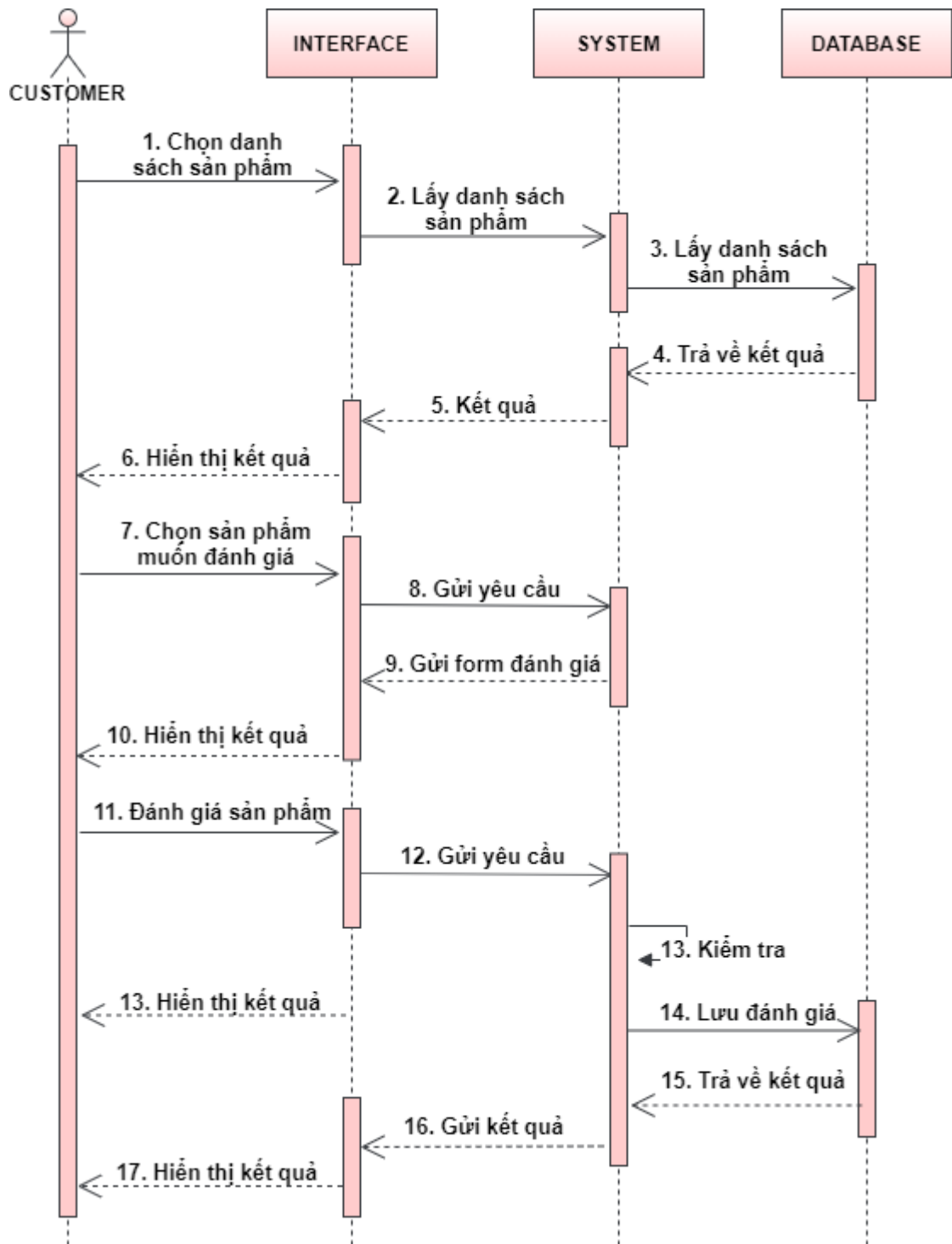
Hình 3. 15 Biểu đồ tuần tự Xóa dữ liệu trang quản lý

Biểu đồ tuần tự Đặt hàng:



Hình 3. 16 Biểu đồ tuần tự Đặt hàng

Biểu đồ tuần tự Đánh giá sản phẩm:



Hình 3. 17 Biểu đồ tuần tự Đánh giá sản phẩm

3.4. Thiết kế cơ sở dữ liệu

3.4.1. Mô tả đối tượng-Lớp

1/Thực thể: **DanhGia**

- Thực thể tượng trưng cho người dùng để đánh giá sản phẩm.
- Các thuộc tính: **Id**, Content, Rating, Ngaycapnhat, Trangthai, Id_sp, Id_kh.

2/ Thực thể: **DanhMuc**

- Thực thể tượng trưng cho từng loại sản phẩm có trong cửa hàng
- Các thuộc tính: **IdDanhmuc**, Tendanhmuc, ParentID

3/ Thực thể: **SanPham**

- Thực thể tượng trưng cho những sản phẩm mà cửa hàng hiện đang bán
- Các thuộc tính: **masp**, idDanhmuc, idthuvien, tensp, hinh, giaban, ngaycapnhat, soluongton, mota, giamgia, giakhuyenmai

4/ Thực thể: **KhachHang**

- Thực thể tượng trưng cho các tài khoản đăng kí tại trang web được quản lý bởi quản trị viên
- Các thuộc tính: **makh**, hoten, tendangnhap, matkhou, email, diachi, dienthoai, ngaysinh, RoleID, statusresetpasswordcode

5/ Thực thể: **DichVu**

- Thực thể tượng trưng cho các dịch vụ tại trang web
- Các thuộc tính: **Id**, dichvu, Hoten, Email, Sdt, Diachi, Trangthai, Tendichvu, Ngaydat, Makh

6/ Thực thể: **DonHang**

- Thực thể tượng trưng cho các dịch vụ tại trang web
- Các thuộc tính: **Madon**, Thanhthoan, Giaohang, Ngaydat, Ngaygiao, Makh

3.4.2. Chuyển đổi lớp sang mô hình cơ sở dữ liệu quan hệ

DanhGia (**Id**, Content, Rating, Ngaycapnhat, Trangthai, Id_sp, Id_kh)

SanPham (masp, idDanhmuc, idthuvien, tensp, hinh, giaban, ngaycapnhat, soluongton, mota, giamgia, giakhuyenmai)

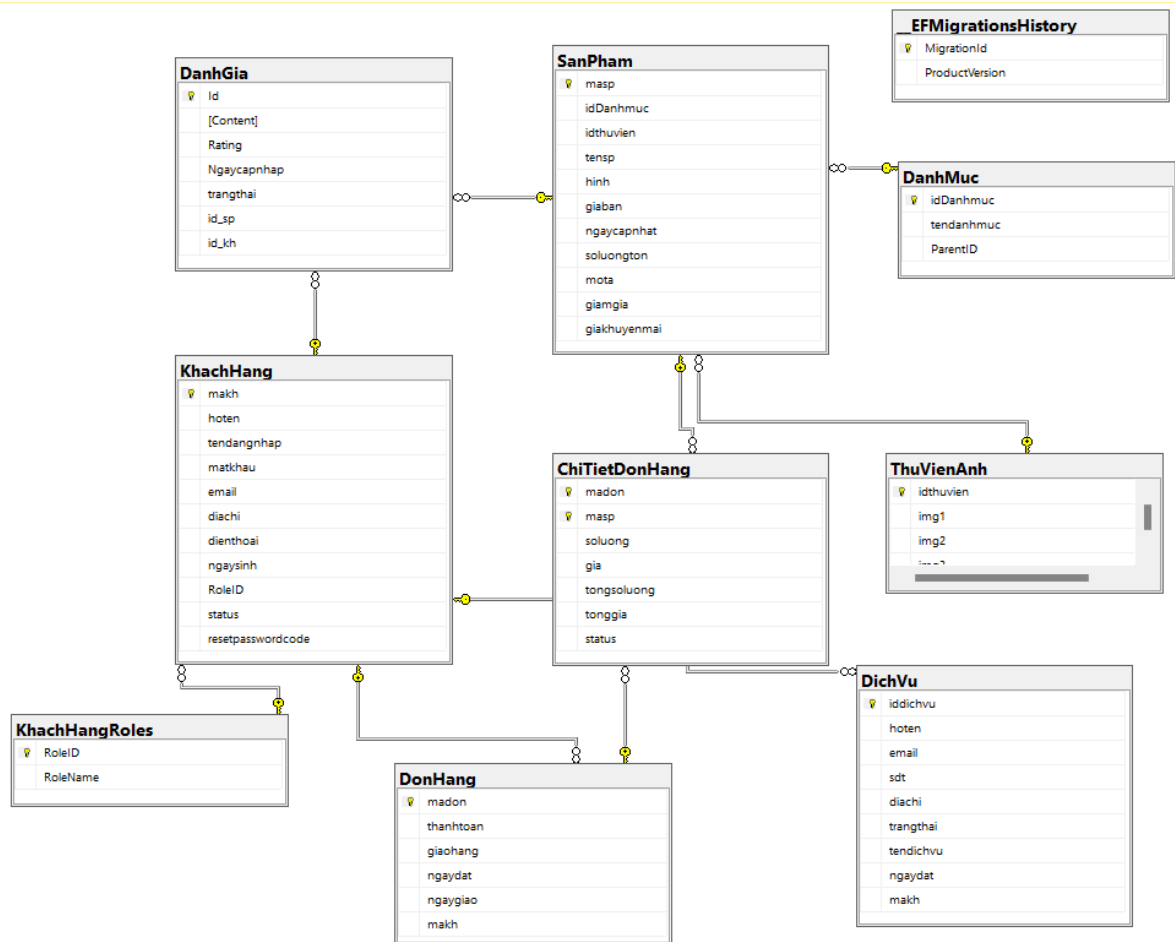
DichVu (Id, dichvu, Hoten, Email, Sdt, Diachi, Trangthai, Tendichvu, Ngaydat, Makh)

KhachHang (makh, hoten, tendangnhap, matkhau, email, diachi, dienthoai, ngaysinh, RoleID, statusresetpasswordcode)

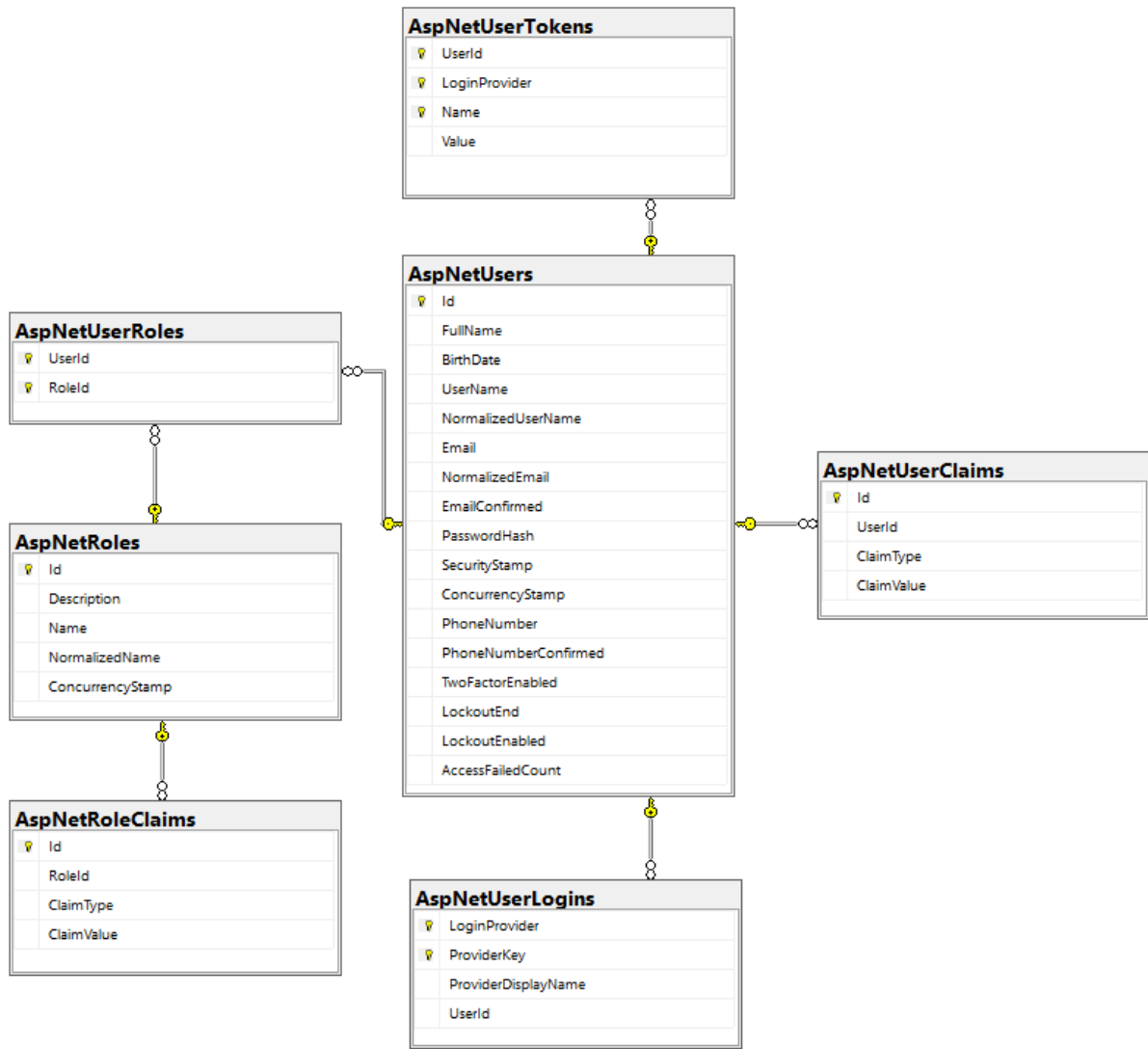
DanhMuc (IdDanhmuc, Tendanhmuc, ParentID)

DonHang (Madon, Thanhthoan, Giaohang, Ngaydat, Ngaygiao, Makh)

3.4.3. Database Diagrams



Hình 3. 18 Database Diagrams



Hình 3. 19 Database Diagrams Login

3.4.4. Các bảng thực thể trong cơ sở dữ liệu

➤ *Bảng Đánh giá:*

STT	Tên thực thể	Kiểu dữ liệu	Ràng buộc	Mô tả
1	Id	Int	Khóa chính	Mã đánh giá
2	Content	Nvarchar	Null	Content
3	Rating	Float	Null	Điểm đánh giá
4	Ngaycapnhat	Datetime	Null	Ngày cập nhật
5	Trangthai	Int	Null	Trạng thái
6	Id_sp	Int	Null	Id sản phẩm
7	Id_kh	Int	Null	Id khách hàng

Bảng 3. 1 Bảng Đánh giá

➤ *Bảng Dịch vụ:*

STT	Tên thực thể	Kiểu dữ liệu	Ràng buộc	Mô tả
1	Iddichvu	Int	Khóa chính	Khóa chính của bảng
2	Hoten	Nvarchar	Not Null	Họ tên
3	Email	Nvarchar	Not Null	Email
4	Sdt	Nvarchar	Not Null	Số điện thoại
5	Diachi	Nvarchar	Not Null	Địa chỉ
6	Trangthai	Nvarchar	Not Null	Trạng thái
7	Tendichvu	Nvarchar	Not Null	Tên dịch vụ
8	Ngaydat	Datetime	Not Null	Ngày đặt
9	Makh	Int	Not null	Mã khách hàng

Bảng 3. 2 Bảng Dịch vụ

➤ *Bảng Khách hàng:*

STT	Tên thực thể	Kiểu dữ liệu	Ràng buộc	Mô tả
1	makh	int	Not Null	Mã khách hàng
2	hoten	nvarchar(50)	Not Null	Tên thương hiệu
3	tendangnhap	varchar(20)	Not Null	Mô tả thương hiệu
4	matkhau	nvarchar(255)	Null	Mật khẩu
5	email	varchar(50)	Null	Email
6	diachi	nvarchar(100)	Null	Địa chỉ
7	dienthoai	varchar(15)	Null	Điện thoại
8	ngaysinh	date	Null	Ngày sinh
9	RoleID	int	Null	Loại tài khoản
10	status	int	Null	Trạng thái
11	resetpasswordcode	nvarchar(255)	Null	Làm mới mật khẩu

Bảng 3. 3 Bảng Khách hàng

➤ *Bảng Danh mục:*

STT	Tên thực thể	Kiểu dữ liệu	Ràng buộc	Mô tả
1	IdDanhmuc	Int	Khóa chính	Khóa chính của bảng
2	Tendanhmuc	Nvarchar	Not null	Tên danh mục
3	ParentID	Nvarchar	Not null	Loại danh mục

Bảng 3. 4 Bảng Danh mục

➤ *Bảng Đơn hàng:*

STT	Tên thực thể	Kiểu dữ liệu	Ràng buộc	Mô tả
1	Madon	Int	Not Null	Mã đơn
2	Thanhthoan	Nvarchar	Not Null	Phương thức thanh toán
3	Giaohang	Nvarchar	Not Null	Phương thức giao hàng
4	Ngaydat	Datetime	Not null	Ngày đặt
5	Ngaygiao	Datetime	Not null	Ngày giao
6	Makh	Int	Not null	Mã khách hàng

Bảng 3. 5 Bảng Đơn hàng

➤ *Bảng Sản phẩm:*

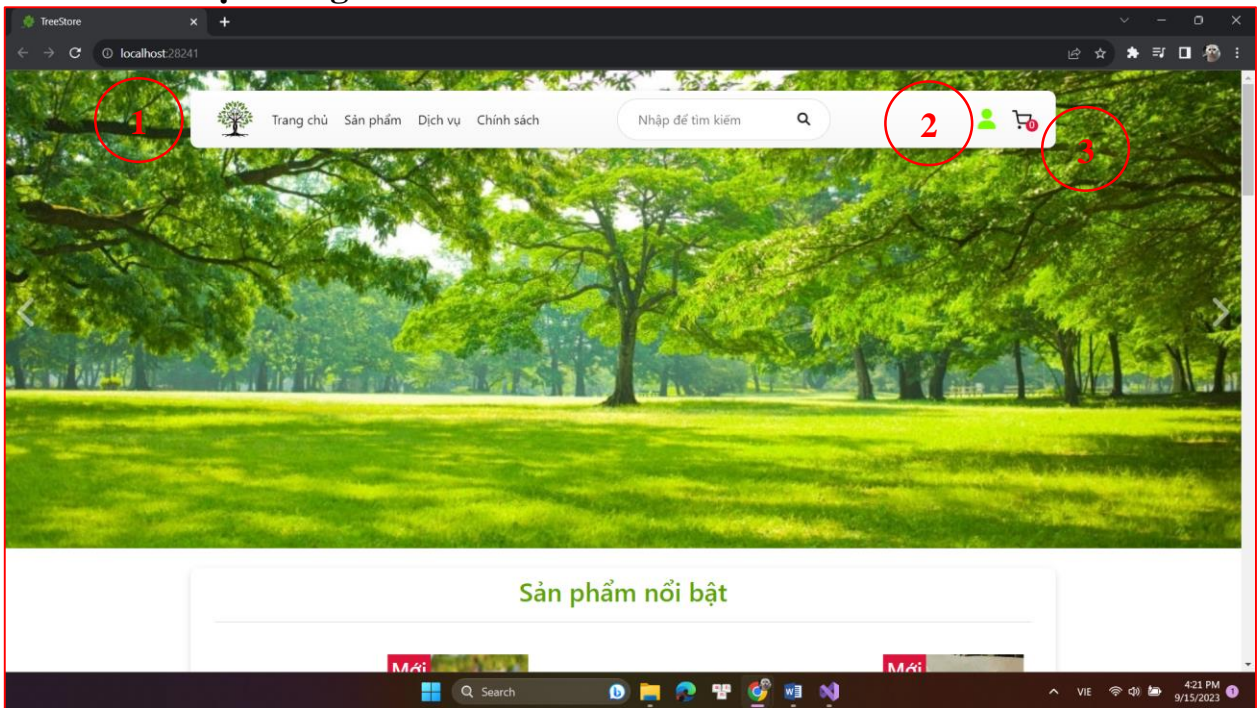
STT	Tên thực thể	Kiểu dữ liệu	Ràng buộc	Mô tả
1	masp	int	Not Null	Mã sản phẩm
2	idDanhmuc	int	Not Null	Mã danh mục
3	idthuvien	int	Not Null	Mã thư viện
4	tensp	nvarchar(MAX)	Not Null	Tên sản phẩm
5	hinh	nvarchar(255)	Not Null	Hình sản phẩm
6	giaban	decimal(18, 0)	Not Null	Giá bán
7	ngaycapnhat	smalldatetime	Not Null	Ngày cập nhật
8	soluongton	int	Not Null	Số lượng tồn
9	mota	nvarchar(500)	Not Null	Mô tả
10	giamgia	int	Not Null	Giảm giá
11	giakhuyenmai	decimal(18, 0)	Not Null	Giá khuyến mãi

Bảng 3. 6 Bảng Sản phẩm

CHƯƠNG 4: THỰC NGHIỆM VÀ TRIỂN KHAI

4.1. Kết quả thực hiện chương trình

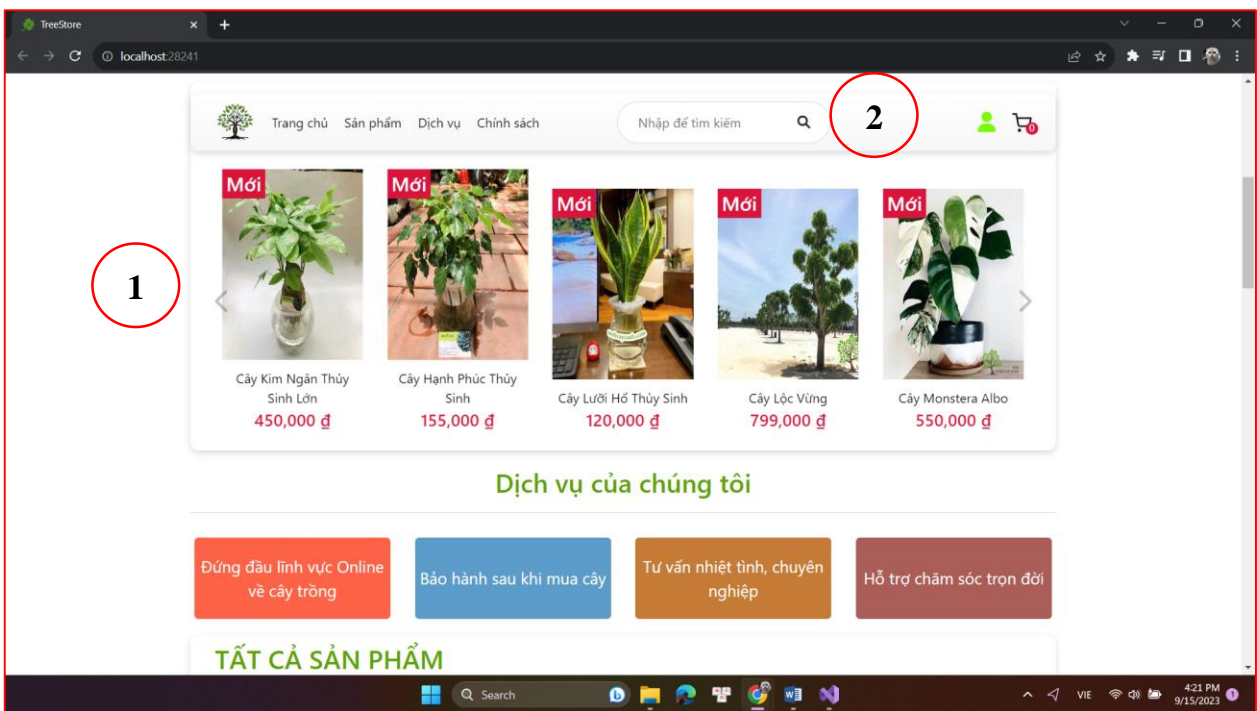
- **Giao diện trang chủ**



Hình 4. 1 Giao diện trang chủ 1

Bảng 4. 1 Mô tả chi tiết hình 4.1

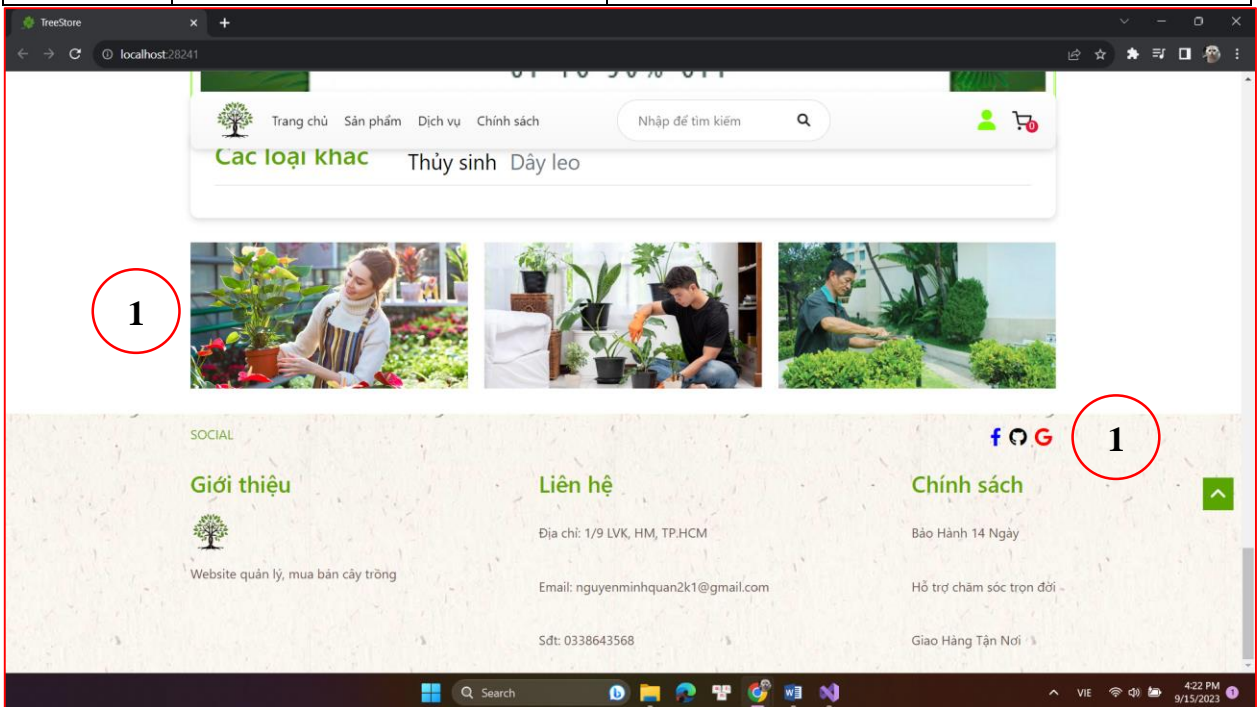
Kí hiệu	Chức năng/Thông tin	Ý nghĩa sử dụng
1	Thanh navigation	Cho phép người dùng dễ dàng điều hướng các trang
2	Đăng nhập	Cho phép người dùng truy cập vào để đăng nhập/đăng ký.
3	Giỏ hàng	Cho phép người dùng truy cập vào giỏ hàng.



Hình 4.2. Giao diện trang chủ 2

Bảng 4. 2 Mô tả chi tiết hình 4.2

Ký hiệu	Chức năng/Thông tin	Ý nghĩa sử dụng
1	Xem sản phẩm	Cho phép người dùng xem chi tiết sản phẩm
2	Tìm kiếm	Cho phép người dùng tìm kiếm sản phẩm

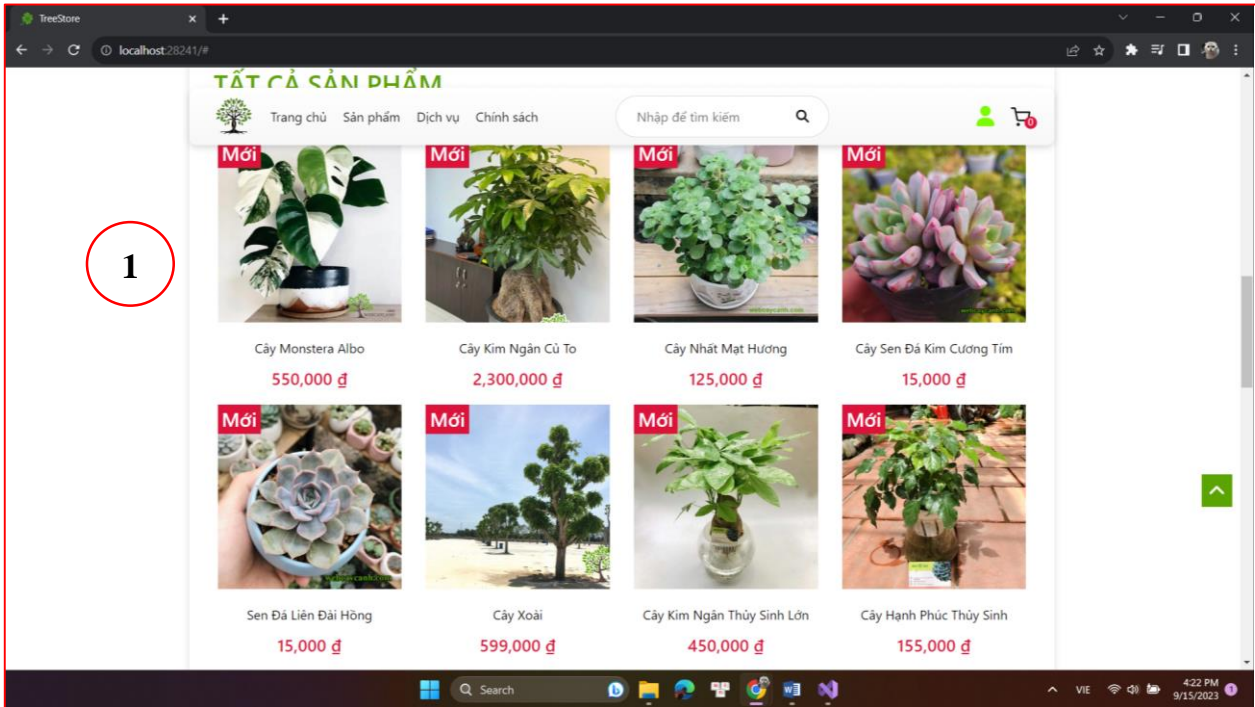


Hình 4. 3 Giao diện trang chủ 3

Bảng 4. 3 Mô tả chi tiết hình 4.3

Ký hiệu	Chức năng/Thông tin	Ý nghĩa sử dụng
1	Xem sản phẩm	Cho phép người dùng xem chi tiết sản phẩm
2	Tìm kiếm	Cho phép người dùng tìm kiếm sản phẩm

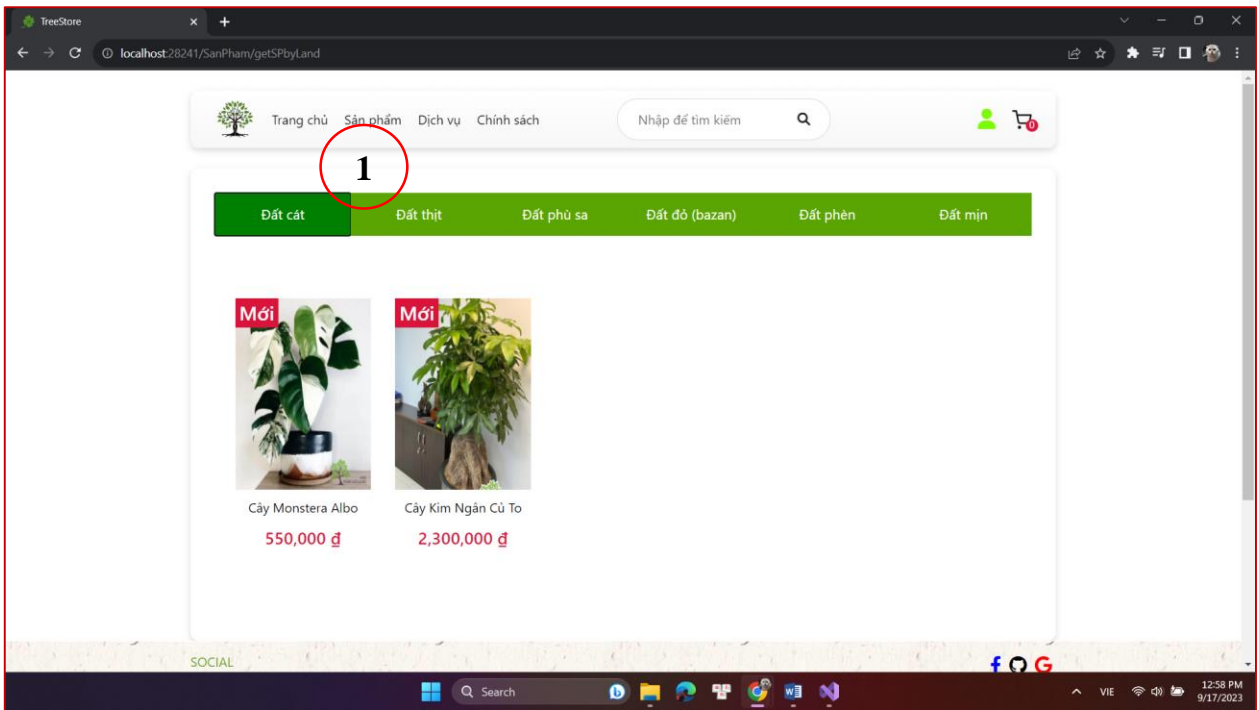
- **Giao diện trang sản phẩm**



Hình 4. 4. Giao diện trang sản phẩm

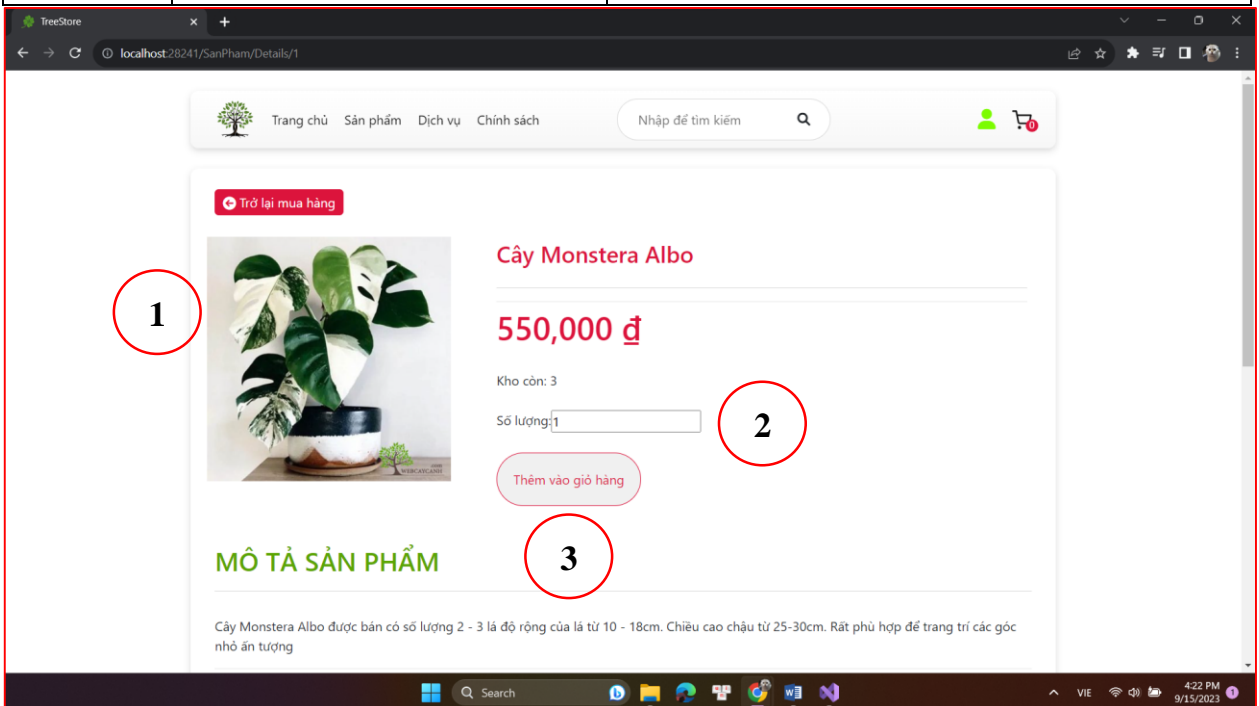
Bảng 4. 4 Mô tả chi tiết hình 4.4

Ký hiệu	Chức năng/Thông tin	Ý nghĩa sử dụng
1	Xem sản phẩm	Cho phép người dùng xem toàn bộ các sản phẩm



Hình 4. 5 Giao diện hiển thị sản phẩm theo Vùng đất

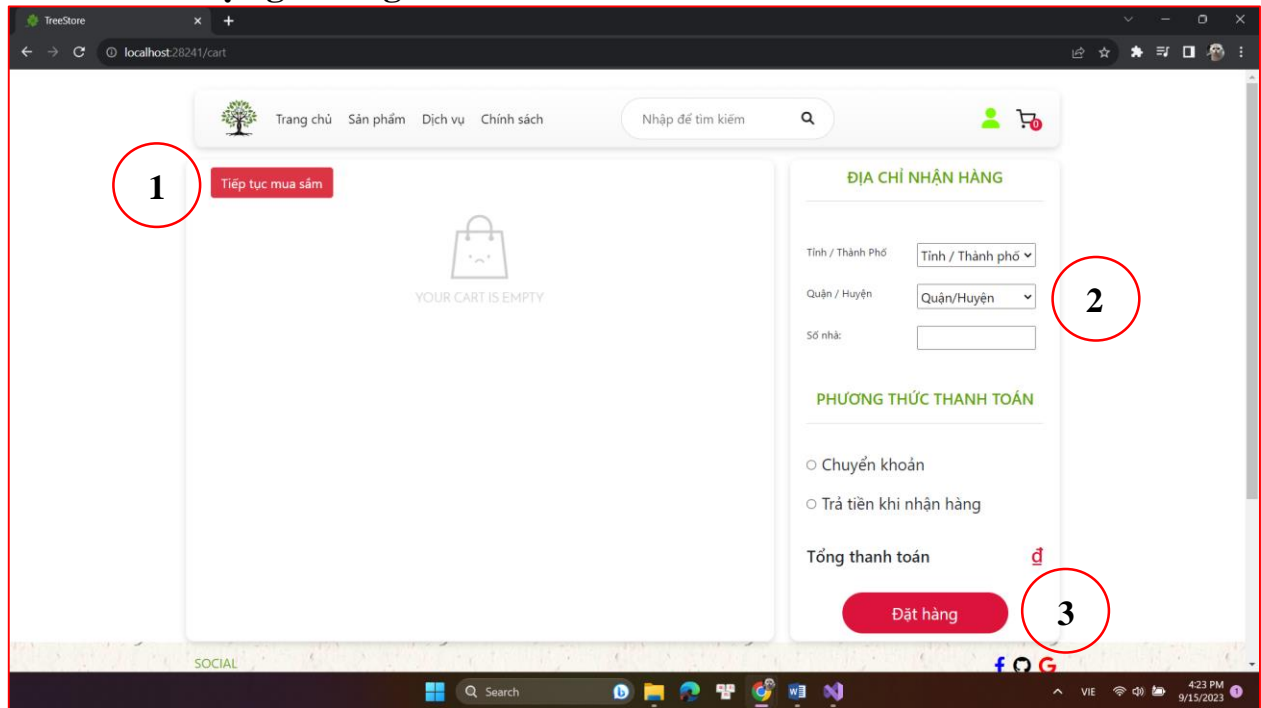
Bảng 4. 5 Mô tả chi tiết hình 4.5		
Ký hiệu	Chức năng/Thông tin	Ý nghĩa sử dụng
1	Xem sản phẩm	Cho phép người dùng xem các sản phẩm theo vùng đất chọn



Hình 4. 6 Giao diện trang chi tiết sản phẩm

Bảng 4. 6 Mô tả chi tiết hình 4.6		
Ký hiệu	Chức năng/Thông tin	Ý nghĩa sử dụng
1	Xem hình sản phẩm	Cho phép người dùng zoom to ra xem các sản phẩm
2	Số lượng	Cho phép người dùng lựa chọn số lượng cần mua
3	Thêm vào giỏ hàng	Cho phép người dùng thêm sản phẩm muốn mua vào giỏ hàng

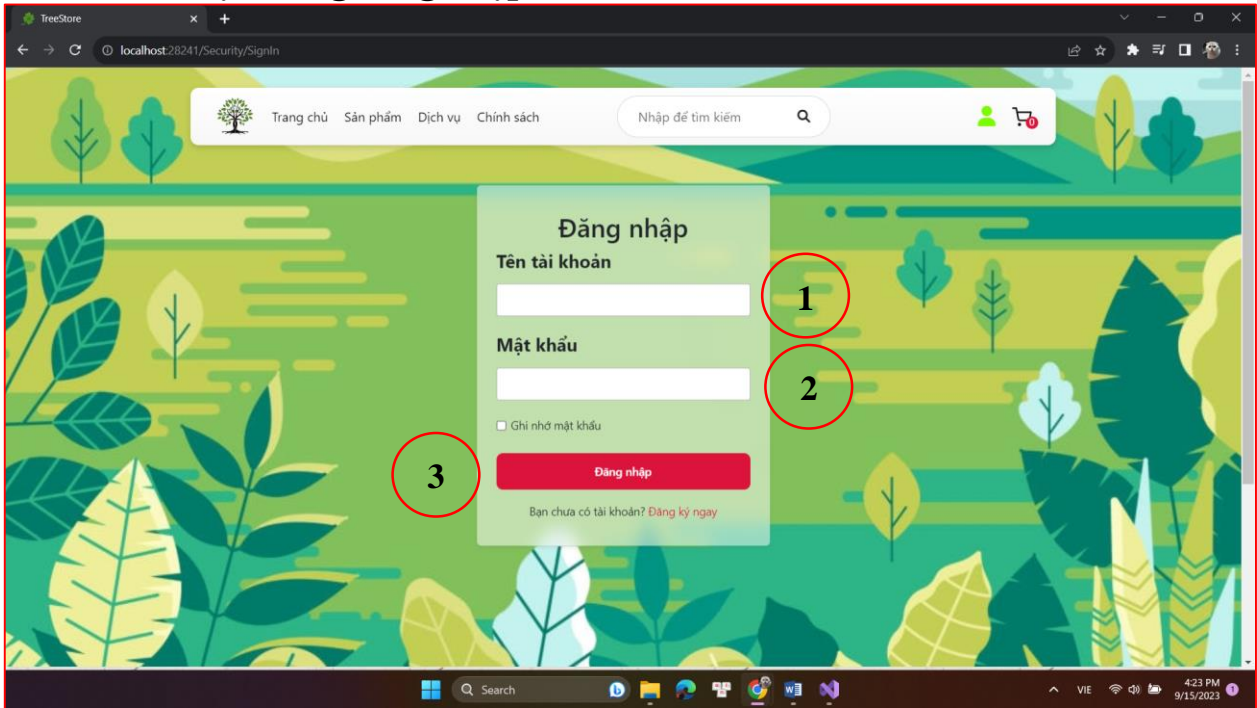
• **Giao diện giỏ hàng**



Hình 4. 7. Giao diện giỏ hàng

Bảng 4. 7 Mô tả chi tiết hình 4.7		
Ký hiệu	Chức năng/Thông tin	Ý nghĩa sử dụng
1	Tiếp tục mua sắm	Cho phép người dùng quay trở lại trang mua sắm
2	Địa chỉ nhận hàng	Cho phép người dùng tạo địa chỉ nhận hàng
3	Đặt hàng	Cho phép người dùng tạo đơn hàng mua sắm

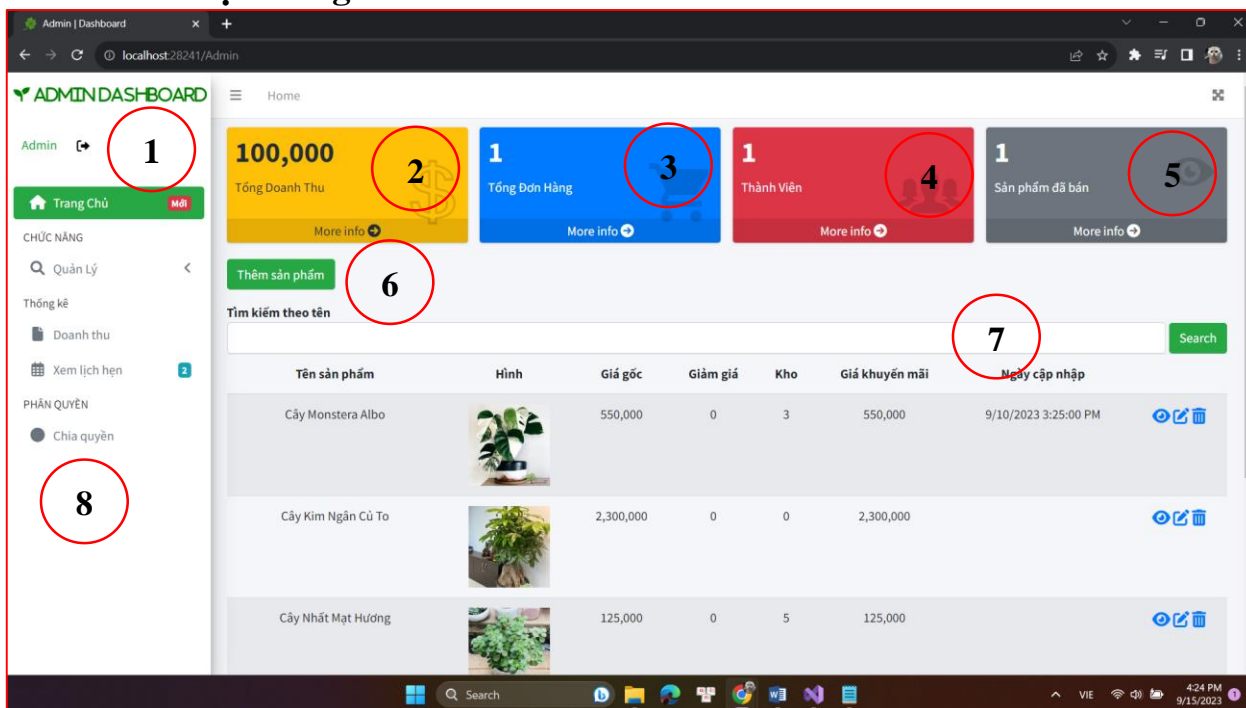
- **Giao diện trang đăng nhập**



Hình 4. 8. Giao diện trang đăng nhập

Bảng 4. 8 Mô tả chi tiết hình 4.8		
Ký hiệu	Chức năng/Thông tin	Ý nghĩa sử dụng
1	Tên tài khoản	Cho phép người dùng nhập tên tài khoản để đăng nhập
2	Mật khẩu	Cho phép người dùng nhập mật khẩu
3	Đăng nhập	Cho phép người dùng đăng nhập vào trang web

- **Giao diện trang Admin**

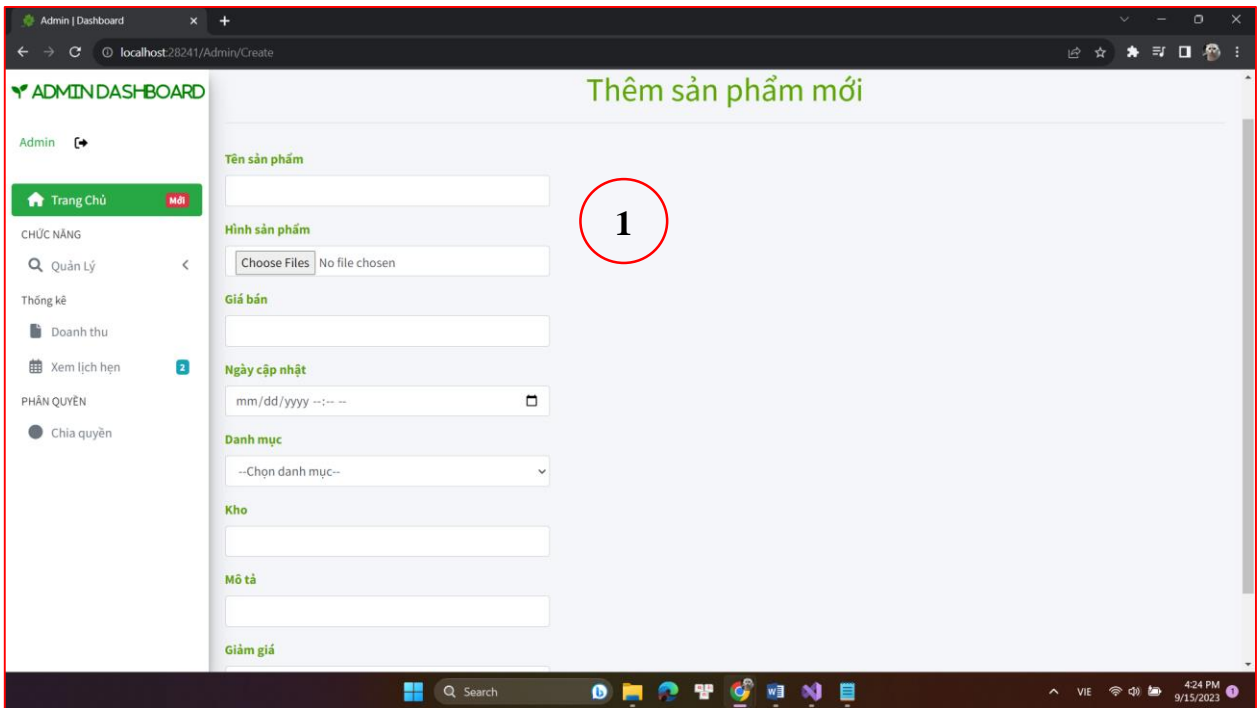


Hình 4. 9. Giao diện trang chủ

Bảng 4. 9 Mô tả chi tiết hình 4.9

Ký hiệu	Chức năng/Thông tin	Ý nghĩa sử dụng
1	Đăng xuất	Cho phép người dùng đăng xuất khỏi trang quản trị
2	Doang thu	Cho phép quản trị viên theo dõi doanh thu
3	Đơn hàng	Cho phép quản trị viên theo dõi có bao nhiêu đơn hàng
4	Thành viên	Cho phép quản trị viên theo dõi các thành viên của trang web
5	Sản phẩm đã bán	Cho phép quản trị viên biết được bao nhiêu sản phẩm đã được bán
6	Thêm sản phẩm	Cho phép quản trị viên thêm sản phẩm mới
7	Tìm kiếm	Cho phép quản trị viên tìm kiếm sản phẩm
8	Menu	Cho phép quản trị viên điều hướng tới các trang quản trị

- Chức năng thêm sản phẩm mới

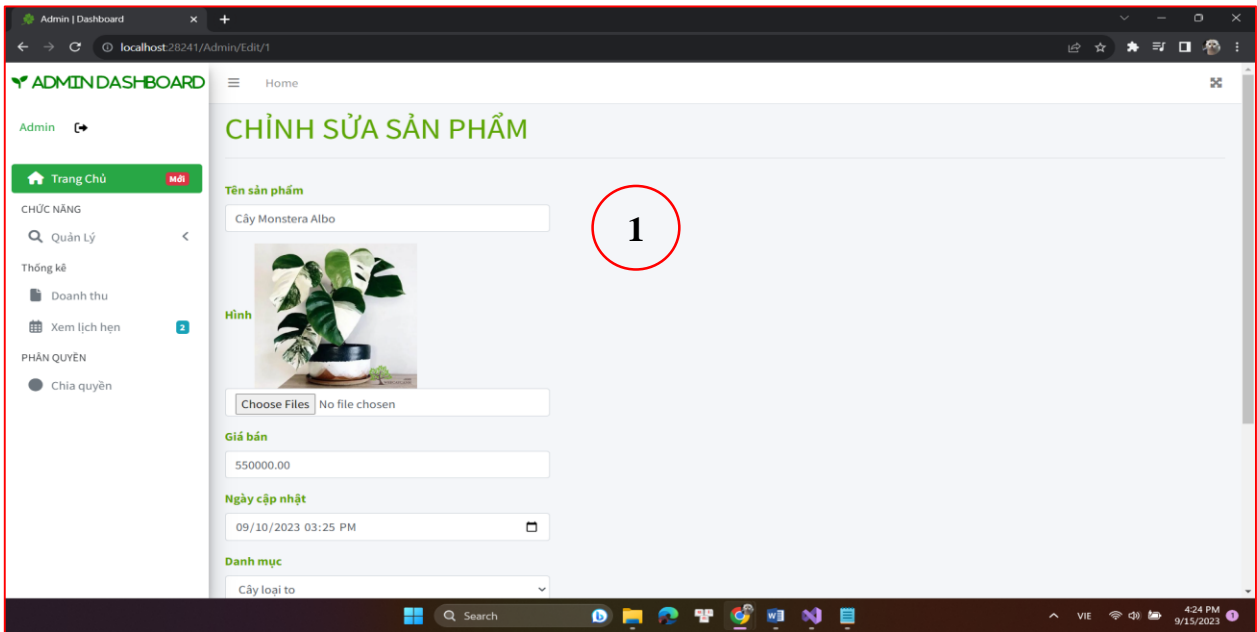


Hình 4. 10. Giao diện chức năng thêm sản phẩm mới

Bảng 4. 10 Mô tả chi tiết hình 4.10

Ký hiệu	Chức năng/Thông tin	Ý nghĩa sử dụng
1	Thêm sản phẩm mới	Cho phép người dùng thêm chi tiết sản phẩm mới

- Chức năng chỉnh sửa sản phẩm

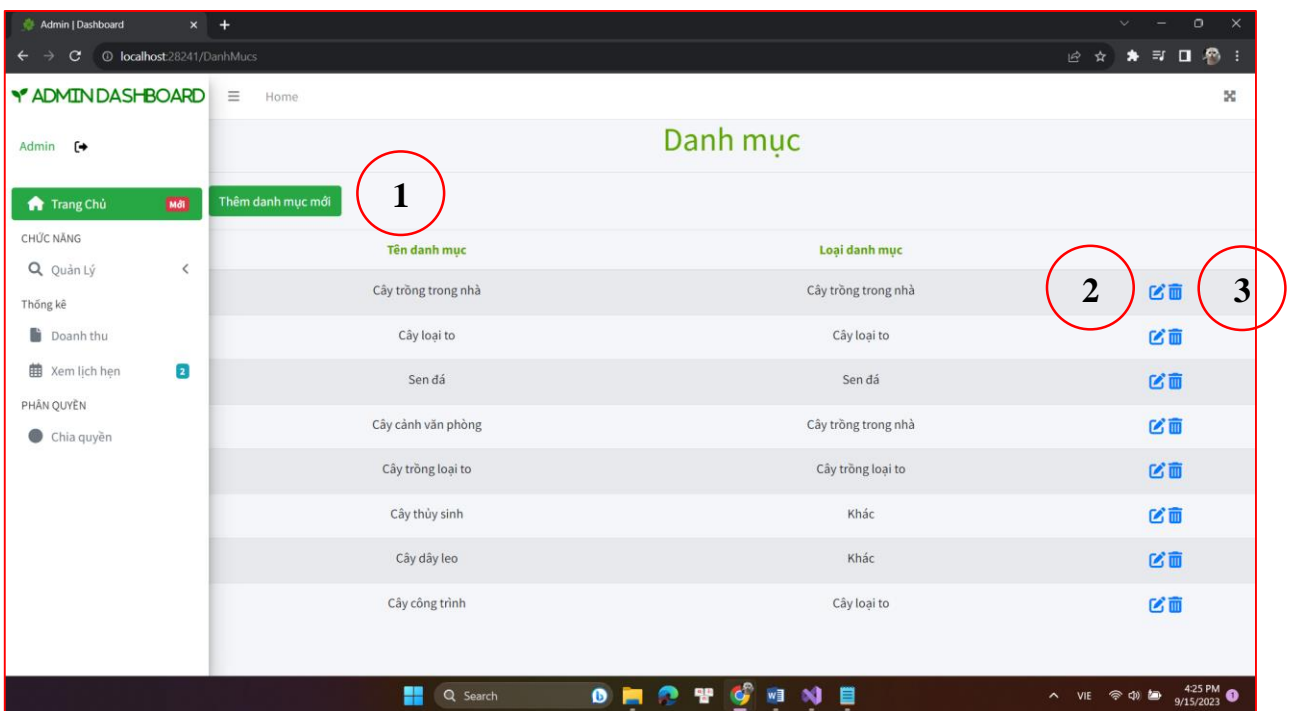


Hình 4. 11. Giao diện chức năng chỉnh sửa sản phẩm

Bảng 4. 11 Mô tả chi tiết hình 4.11

Ký hiệu	Chức năng/Thông tin	Ý nghĩa sử dụng
1	Chỉnh sửa sản phẩm	Cho phép người dùng chỉnh chi tiết sản phẩm

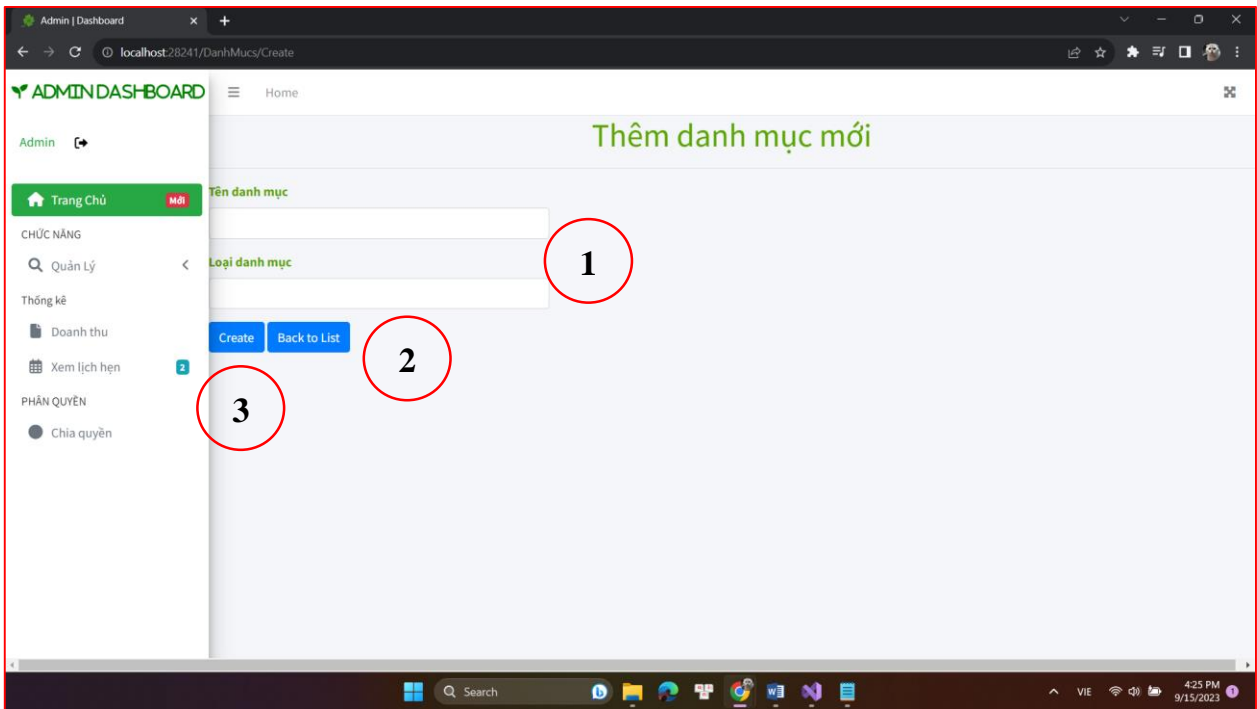
- Chức năng thêm, xóa, sửa danh mục:



Hình 4. 12. Giao diện chức năng danh mục

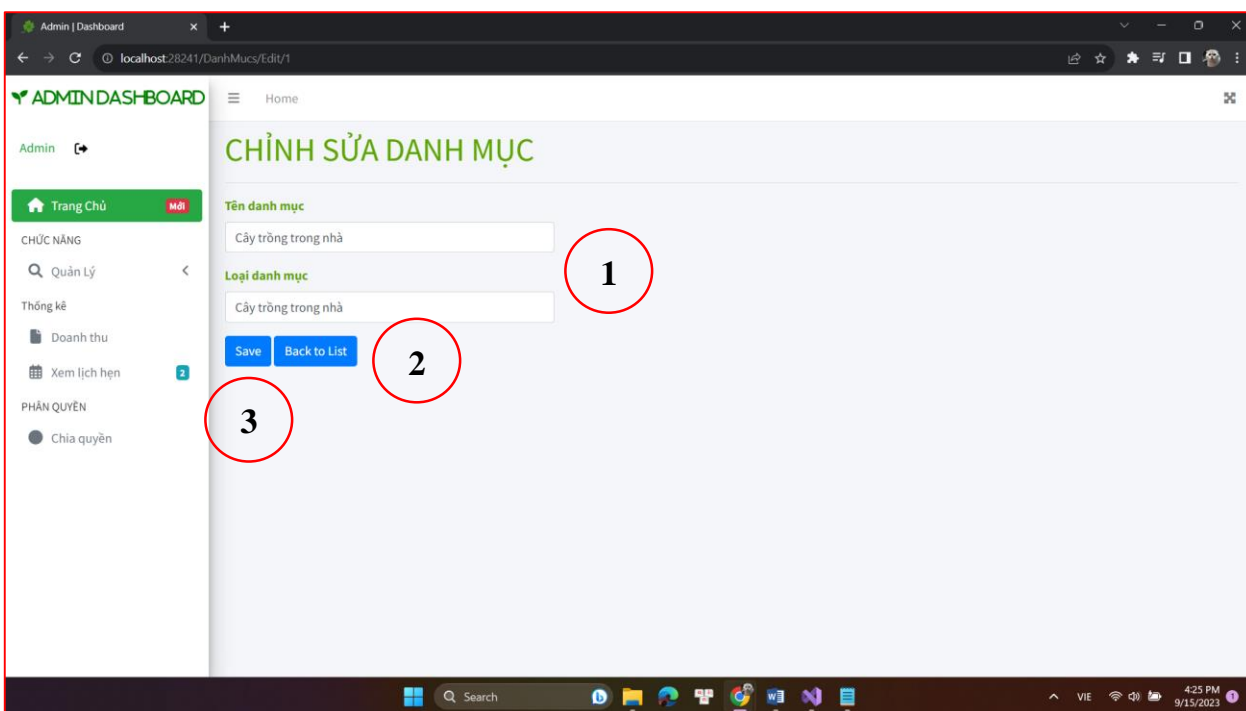
Bảng 4. 12 Mô tả chi tiết hình 4.12		
Ký hiệu	Chức năng/Thông tin	Ý nghĩa sử dụng
1	Thêm danh mục	Cho phép quản trị viên thêm danh mục mới
2	Sửa danh mục	Cho phép quản trị viên truy cập vào trang sửa danh mục
3	Xóa	Cho phép quản trị viên xóa danh mục

- **Giao diện thêm danh mục mới**



Hình 4. 13. Giao diện chi tiết chức năng thêm danh mục

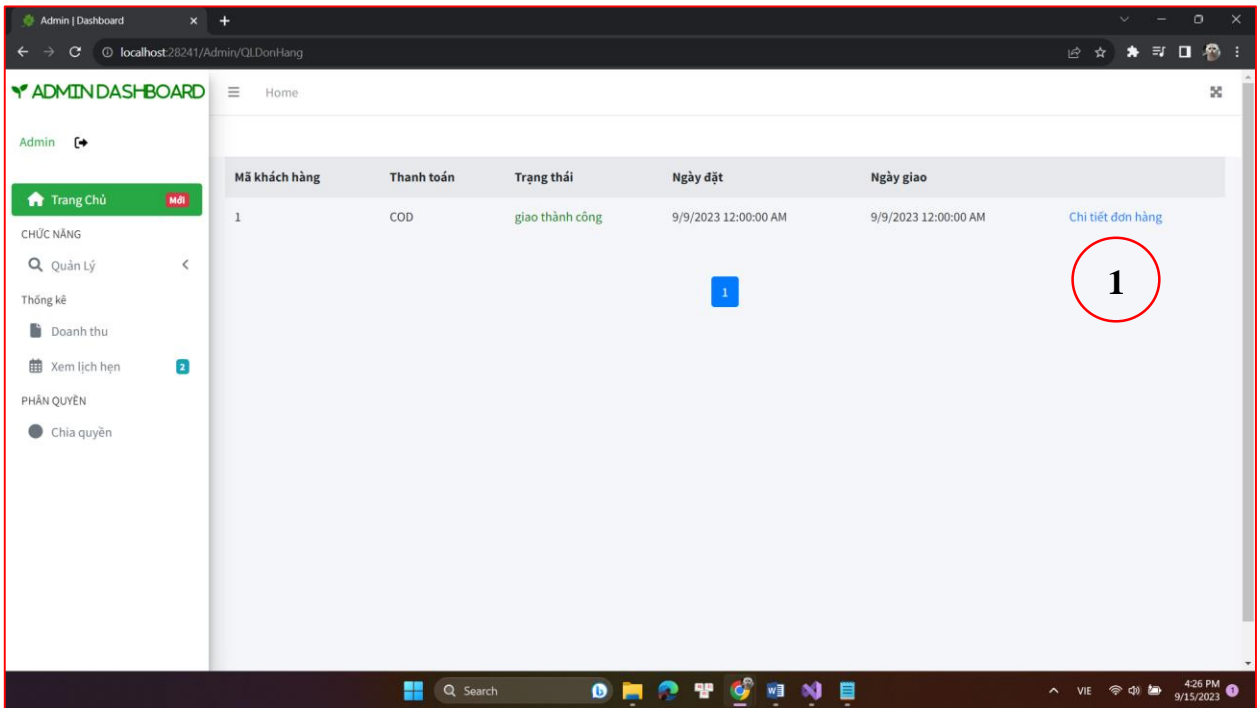
Bảng 4. 13 Mô tả chi tiết hình 4.13		
Ký hiệu	Chức năng/Thông tin	Ý nghĩa sử dụng
1	Thêm danh mục	Cho phép quản trị viên nhập chi tiết danh mục mới
2	Trở lại	Cho phép quản trị viên trở lại trang trước đó
3	Tạo	Cho phép quản trị viên tạo ra danh mục



Hình 4. 14. Giao diện chức năng chỉnh sửa danh mục

Bảng 4. 14 Mô tả chi tiết hình 4.14		
Ký hiệu	Chức năng/Thông tin	Ý nghĩa sử dụng
1	Sửa danh mục	Cho phép quản trị viên chỉnh sửa chi tiết danh mục
2	Trở lại	Cho phép quản trị viên trở lại trang trước đó
3	Lưu	Cho phép quản trị viên lưu danh mục đã chỉnh sửa

- Chức năng quản lý đơn hàng:

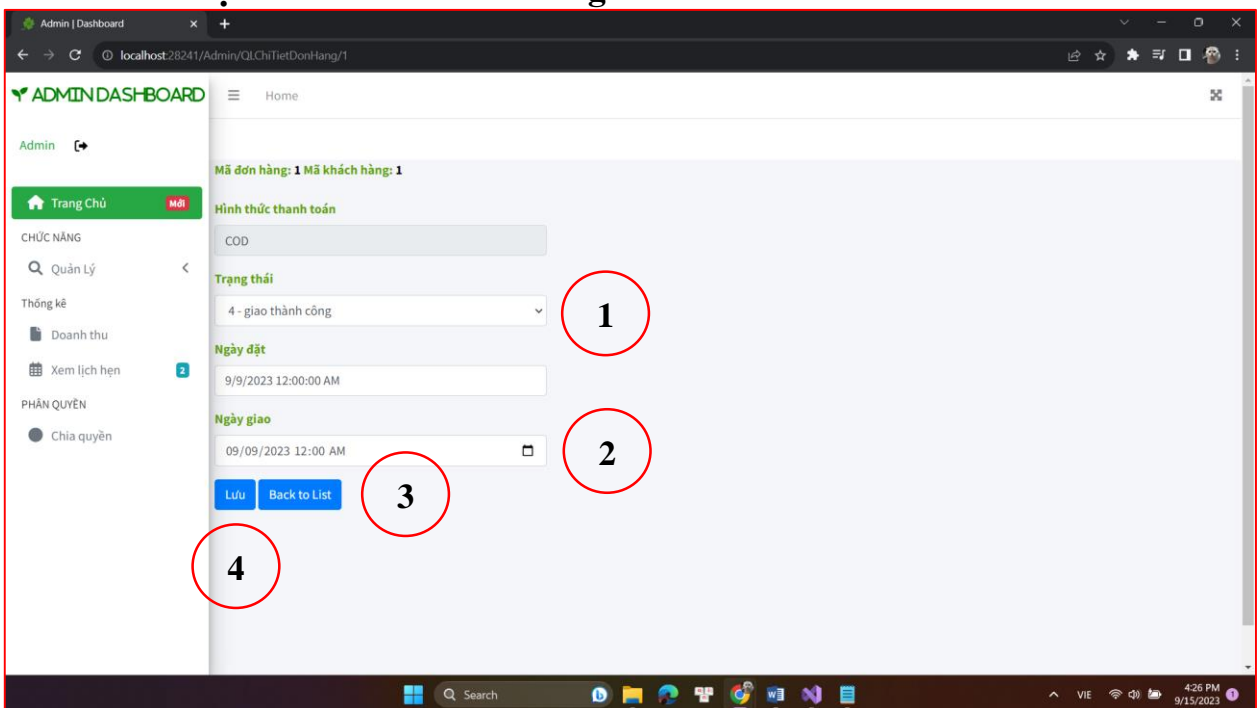


Hình 4. 15 Giao diện chức năng quản lý đơn hàng

Bảng 4. 15 Mô tả chi tiết hình 4.15

Ký hiệu	Chức năng/Thông tin	Ý nghĩa sử dụng
1	Chi tiết đơn hàng	Cho phép quản trị viên xem chi tiết đơn hàng

- Giao diện xem chi tiết đơn hàng



Hình 4. 16 Giao diện chức năng chi tiết đơn hàng

Bảng 4. 16 Mô tả chi tiết hình 4.16		
Ký hiệu	Chức năng/Thông tin	Ý nghĩa sử dụng
1	Trạng thái	Cho phép quản trị viên thay đổi trạng thái đơn hàng
2	Ngày giao	Cho phép quản trị viên thiết lập ngày giao đơn hàng
3	Trở lại	Cho phép quản trị viên trở lại trang trước
4	Lưu	Cho phép quản trị viên lưu lại những thay đổi của đơn hàng

KẾT LUẬN

Kết quả đạt được

Xây dựng website quản lý lựa chọn loại cây trồng cho vùng đất là công việc rất cần thiết cho nhà nông. Dựa trên lý do này, đề tài khóa luận này đã thực hiện được các chức năng của hệ thống Website như sau. Đặc biệt là áp dụng các kiến thức đã học xây dựng website quản lý lựa chọn cây trồng, phân quyền người dùng và người quản trị, với giao diện và chức năng khác nhau. Quản lý các thông tin cần thiết của cửa hàng như các sản phẩm, thông tin khách hàng và các giao dịch mua hàng của khách hàng. Có hầu như đầy đủ chức năng cơ bản để quản lý cửa hàng.

Các chức năng của hệ thống Website gồm:

- **Đăng nhập:** đăng nhập và sử dụng các chức năng của website
- **Lựa chọn cây trồng:** nhiều loại cây trồng để người mua lựa chọn
- **Tìm kiếm:** tìm kiếm cây trồng theo nhiều tiêu chí
- **Đặt hàng:** đặt mua loại cây mà người mua muốn chọn
- **In hóa đơn:** cho phép người mua in hóa đơn đặt hàng
- **Trang quản trị:** thêm, xóa, sửa (sản phẩm, danh mục), xác nhận đơn hàng, theo dõi (doanh thu, hóa đơn, thành viên, sản phẩm đã bán)

Hướng phát triển

Trong thời gian tới, đề tài này sẽ thực hiện các chức năng tiếp theo:

- Hiệu chỉnh giao diện
- Thanh toán Online
- Đăng nhập bằng Facebook, Google,..
- Tích hợp giao diện lên điện thoại

TÀI LIỆU THAM KHẢO

[1]. **Microsoft**, ASP.NET documentation, link: <https://learn.microsoft.com/en-us/aspnet/core/getting-started/?view=aspnetcore-7.0&tabs=windows> (08/2023)

[2]. **Udemy**, Bootstrap 4, link: <https://www.udemy.com/course/bootstrap-4/learn/lecture/4244842#content> (08/2023)

[3]. **AdminLTE**, Dowload, link: <https://github.com/ColorlibHQ/AdminLTE/releases> (08/2023)

[4]. **TMA Solution**, Tất cả những gì bạn cần biết về ASP.NET Core, link: <https://www.tma.vn/Hoi-dap/Cam-nang-nghe-nghiep/Tat-ca-nhung-gi-ban-can-biet-ve-ASP-NET-Core/41324> (08/2023)