|  |  |
| --- | --- |
|  |  |
| TẠ ĐỨC MẠNH | **BỘ CÔNG THƯƠNG**  **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **---------------------------------------** |
|  |
| **ĐỒ ÁN TỐT NGHIỆP**  **NGÀNH CÔNG NGHỆ THÔNG TIN** |
|  |
| **XÂY DỰNG WEBSITE BÁN ĐỒ TENNIS BẰNG ASP.NET MVC** |
|  |
|  |
| **CBHD :** **TS. Đặng Trọng Hợp** |
| CÔNG NGHỆ THÔNG TIN | **Sinh viên : Tạ Đức Mạnh** |
| **Mã sinh viên: 2020600026** |
|  |
|  |
|  |
| **Hà Nội - 2024** |
|  |

|  |  |
| --- | --- |
|  |  |
| TẠ ĐỨC MẠNH | **BỘ CÔNG THƯƠNG**  **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **---------------------------------------** |
|  |
| **ĐỒ ÁN TỐT NGHIỆP**  **NGÀNH CÔNG NGHỆ THÔNG TIN** |
|  |
| **XÂY DỰNG WEBSITE BÁN ĐỒ TENNIS BẰNG ASP.NET MVC** |
|  |
|  |
| **CBHD :** **TS. Đặng Trọng Hợp** |
| CÔNG NGHỆ THÔNG TIN | **Sinh viên : Tạ Đức Mạnh** |
| **Mã sinh viên: 2020600026** |
|  |
|  |
|  |
| **Hà Nội - 2024** |
|  |

# LỜI CẢM ƠN

Sau khoảng thời gian học tập tại trường và được tham gia các hoạt động phát triển phần mềm được sự truyền đạt kiến thức, lời khuyên trong công việc và giúp đỡ tận tình của quý thầy cô, em đã có cơ hội ra ngoài thực tập ở công ty, được tiếp xúc với môi trường làm việc thực tế, tiếp xúc với các công cụ quản lý dự án hiện đại,… Em xin chân thành cảm ơn các thầy cô, giảng viên trường Đại học Công Nghiệp Hà Nội nói chung và khoa Công nghệ thông tin nói riêng - những người đã ân cần giảng dạy, truyền đạt kiến thức và hướng dẫn em trong quá trình học tập tại trường.

Bên cạnh đó, em cũng xin chân thành cảm ơn sự hướng dẫn và chỉ bảo tận tình của thầy Đặng Trọng Hợp – người đã luôn quan tâm, chỉ bảo và tạo điều kiện cho em trong suốt quá trình thực tập tốt nghiệp và quá trình thực hiện đề tài. Các định hướng và sự hiểu biết về khoa học của thầy chính là tiền đề để em có thể hoàn thành được đề tài này một cách tốt nhất.

Tuy còn một số hạn chế nhất định về chuyên môn và thời gian nên bài báo cáo và ứng dụng còn nhiều hạn chế. Kính mong quý thầy cô góp ý để đề tài của em hoàn thiện hơn.

Em xin chân thành cảm ơn!

Sinh viên thực hiện

Tạ Đức Mạnh

# MỤC LỤC

[LỜI CẢM ƠN 1](#_Toc166617279)

[MỤC LỤC 2](#_Toc166617280)

[DANH MỤC HÌNH ẢNH 5](#_Toc166617281)

[DANH MỤC BẢNG 7](#_Toc166617282)

[CHƯƠNG 1: TỔNG QUAN 8](#_Toc166617283)

[1.1. Lý do chọn đề tài. 8](#_Toc166617284)

[1.2. Mục đích 9](#_Toc166617285)

[1.3. Đối tượng và phạm vi nghiên cứu. 9](#_Toc166617286)

[1.4. Giới hạn của đề tài. 10](#_Toc166617287)

[1.5. Nội dung thực hiện. 10](#_Toc166617288)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT. 10](#_Toc166617289)

[2.1. Lịch sử ASP .NET. 11](#_Toc166617290)

[2.2. ASP .NET và .NET FRAMEWORK. 11](#_Toc166617291)

[2.3. Kiến Trúc của ASP.NET. 11](#_Toc166617292)

[2.4. Ngôn Ngữ Lập Trình. 11](#_Toc166617293)

[2.5 Công Nghệ và Công Cụ Phát Triển. 11](#_Toc166617294)

[2.6. Lợi Ích của ASP.NET. 12](#_Toc166617295)

[2.7. Tổng quan về SQL Server. 12](#_Toc166617296)

[2.7.1. Khái niệm. 12](#_Toc166617297)

[2.7.2. Các thành phần cơ bản trong SQL Server. 12](#_Toc166617298)

[CHƯƠNG 3: PHÂN TÍCH HỆ THỐNG VÀ THIẾT KẾ CƠ SỞ DỮ LIỆU. 14](#_Toc166617299)

[3.1. Danh sách các Actor. 14](#_Toc166617300)

[3.2. Biểu đồ use case. 15](#_Toc166617301)

[3.2.1. Biểu đồ use case tổng quát. 15](#_Toc166617302)

[3.2.2. Biểu đồ use case người dùng. 16](#_Toc166617303)

[3.2.3. Biểu đồ usecase Admin. 17](#_Toc166617304)

[3.2.4. Đặc tả usecase. 17](#_Toc166617305)

[3.3. Biểu đồ tuần tự. 32](#_Toc166617306)

[3.4. Sơ đồ lớp 38](#_Toc166617307)

[3.6. Các bảng CSDL. 39](#_Toc166617308)

[3.6.1. Bảng category. 39](#_Toc166617309)

[3.6.2. Bảng contact. 40](#_Toc166617310)

[3.6.3. Bảng link. 42](#_Toc166617311)

[3.6.4. Bảng menu. 43](#_Toc166617312)

[3.6.5. Bảng order. 44](#_Toc166617313)

[3.6.6. Bảng ordersdetail. 46](#_Toc166617314)

[3.6.7. Bảng post. 47](#_Toc166617315)

[3.6.8. Bảng product. 48](#_Toc166617316)

[3.6.9. Bảng role. 50](#_Toc166617317)

[3.6.10. Bảng slider. 50](#_Toc166617318)

[3.6.11. Bảng topic. 52](#_Toc166617319)

[3.6.12. Bảng user. 53](#_Toc166617320)

[CHƯƠNG 4: XÂY DỰNG ỨNG DỤNG 55](#_Toc166617321)

[4.1. Giao diện người dùng: 55](#_Toc166617322)

[4.1.1. Giao diện trang chủ: 55](#_Toc166617323)

[4.1.2. Giao diện danh mục sản phẩm. 55](#_Toc166617324)

[4.1.3. Giao diện chi tiết sản phẩm. 56](#_Toc166617325)

[4.1.4. Giao diện đăng kí. 56](#_Toc166617326)

[4.1.5. Giao diện đăng nhập. 57](#_Toc166617327)

[4.1.6. Giao diện giỏ hàng. 57](#_Toc166617328)

[4.1.7. Giao diện thanh toán. 58](#_Toc166617329)

[4.1.8. Giao diện sau khi đặt hàng thành công. 59](#_Toc166617330)

[4.1.9. Giao diện sản phẩm khuyến mãi. 60](#_Toc166617331)

[4.2. Giao diện trang quản lý. 61](#_Toc166617332)

[4.2.1. Giao diện đăng nhập trang quản lý. 61](#_Toc166617333)

[4.2.2. Giao diện trang chủ quản lý. 61](#_Toc166617334)

[4.2.3. Giao diện trang quản lý loại sản phẩm. 62](#_Toc166617335)

[4.2.4. Giao diện trang quản lý sản phẩm. 62](#_Toc166617336)

[4.2.5. Giao diện trang quản lý chủ đề bài viết. 63](#_Toc166617337)

[4.2.6. Giao diện trang quản lý bài viết. 63](#_Toc166617338)

[4.2.7. Giao diện trang quản lý đơn hàng. 64](#_Toc166617339)

[4.2.8. Giao diện trang quản lý ảnh bìa. 64](#_Toc166617340)

[4.2.9. Giao diện trang quản lý tài khoản người dùng. 65](#_Toc166617341)

[4.2.10. Giao diện trang quản lý menu. 65](#_Toc166617342)

[4.2.11. Giao diện trang thông tin tài khoản. 65](#_Toc166617343)

[KẾT LUẬN 67](#_Toc166617344)

[TÀI LIỆU THAM KHẢO 68](#_Toc166617345)

# DANH MỤC HÌNH ẢNH

[Hình 3. 1. Biểu đồ usecase tổng quát. 13](#_heading=h.2p2csry)

[Hình 3. 2. Biểu đồ usecase người dùng. 14](#_heading=h.2rb4i01)

[Hình 3. 3. Biểu đồ usecase Admin. 14](#_heading=h.ihv636)

[Hình 3. 4. Biểu đồ tuần tự chức năng đăng kí. 29](#_heading=h.2h20rx3)

[Hình 3. 5. Biểu đồ tuần tự chức năng đăng nhập. 30](#_heading=h.34qadz2)

[Hình 3. 6. Biểu đồ tuần tự chức năng đặt mua. 30](#_heading=h.1jvko6v)

[Hình 3. 7. Biểu đồ tuần tự chức năng thanh toán. 31](#_heading=h.43v86uo)

[Hình 3. 8. Biểu đồ tuần tự chức năng thêm sản phẩm. 31](#_heading=h.2j0ih2h)

[Hình 3. 9. Biểu đồ tuần tự chức năng cập nhật thông tin sản phẩm. 32](#_heading=h.y5sraa)

[Hình 3. 10. Biểu đồ tuần tự chức năng xóa sản phẩm. 33](#_heading=h.3i5g9y3)

[Hình 3. 11. Biểu đồ tuần tự chức năng nhập hàng. 34](#_heading=h.1xaqk5w)

[Hình 3. 12. Biểu đồ tuần tự chức năng tìm kiếm. 34](#_heading=h.4hae2tp)

[Hình 3. 13. Biểu mô hình thực thể. 35](#_heading=h.1bkyn9b)

[Hình 3. 13. Biểu mô hình thực thể.](#_heading=h.1bkyn9b)

[Hình 4. 1. Giao diện trang chủ. 42](#_heading=h.1z989ba)

[Hình 4. 2. Giao diện danh mục sản phẩm. 42](#_heading=h.2ye626w)

[Hình 4. 3. Giao diện chi tiết sản phẩm. 43](#_heading=h.3xj3v2i)

[Hình 4. 4. Giao diện đăng kí. 43](#_heading=h.rtofi4)

[Hình 4. 5. Giao diện đăng nhập. 44](#_heading=h.1qym8dq)

[Hình 4. 6. Giao diện giỏ hàng. 45](#_heading=h.2q3k19c)

[Hình 4. 7. Giao diện thanh toán. 45](#_heading=h.158ubh5)

[Hình 4. 8. Giao diện sau khi đặt hàng thành công. 46](#_heading=h.3gxvt7e)

[Hình 4. 9. Giao diện sản phẩm khuyến mãi. 46](#_heading=h.4g2tm30)

[Hình 4. 10. Giao diện đăng nhập trang quản lý. 47](#_heading=h.3ud1p6f)

[Hình 4. 11. Giao diện trang chủ quản lý. 47](#_heading=h.onm9m1)

[Hình 4. 12. Giao diện trang quản lý loại sản phẩm. 48](#_heading=h.1nsk2hn)

[Hình 4. 13. Giao diên trang quản lý sản phẩm. 48](#_heading=h.2mxhvd9)

[Hình 4. 15. Giao diện trang quản lý chủ đề bài viết. 49](#_heading=h.3m2fo8v)

[Hình 4. 16. Giao diện trang quản lý bài viết. 49](#_heading=h.4l7dh4h)

[Hình 4. 17. Giao diện trang quản lý đơn hàng. 50](#_heading=h.1fhy1k3)

[Hình 4. 18. Giao diện trang quản lý ảnh bìa. 50](#_heading=h.2emvufp)

[Hình 4. 19. Giao diện trang quản lý tài khoản người dùng. 51](#_heading=h.3drtnbb)

[Hình 4. 20. Giao diện trang quản lý menu. 51](#_heading=h.4cwrg6x)

[Hình 4. 21. Giao diện trang thông tin tài khoản. 52](#_heading=h.177c0mj)

# DANH MỤC BẢNG

[Bảng 3. 1. Danh sách các Actor. 12](#_heading=h.3as4poj)

[Bảng 3. 2. Bảng category. 36](#_heading=h.pv6qcq)

[Bảng 3. 3. Bảng contact. 36](#_heading=h.1p04j8c)

[Bảng 3. 4. Bảng link. 37](#_heading=h.2o52c3y)

[Bảng 3. 5. Bảng menu. 37](#_heading=h.3na04zk)

[Bảng 3. 6. Bảng order. 38](#_heading=h.hkkpf6)

[Bảng 3. 7. Bảng ordersdetail. 38](#_heading=h.1gpiias)

[Bảng 3. 8. Bảng post. 39](#_heading=h.2fugb6e)

[Bảng 3. 9. Bảng product. 39](#_heading=h.3eze420)

[Bảng 3. 10. Bảng role. 40](#_heading=h.4e4bwxm)

[Bảng 3. 11. Bảng slider. 40](#_heading=h.27jua8u)

[Bảng 3. 12. Bảng topic. 41](#_heading=h.36os34g)

[Bảng 3. 13. Bảng user. 41](#_heading=h.1lu2dc9)

# CHƯƠNG 1: TỔNG QUAN

## 1.1. Lý do chọn đề tài.

Ngày nay với việc khoa học kỹ thuật ngày càng phát triển mạnh mẽ đặc biệt là công nghệ thông tin, những gì mà công nghệ thông tin tạo ra giúp con người phát triển trong nhiều lĩnh vực.

Đặc biệt trong lĩnh vực thương mại điện tử, việc áp dụng công nghệ thông tin vào thương mại điện từ giúp cho việc mua bán trên mạng ngày càng dễ dàng và thuận tiện hơn. Việc mua bán trên mạng dễ dàng hơn, sẽ kích thích người tiêu dùng mua hàng nhiều hơn qua đó sẽ làm tăng doanh thu cho cửa hàng. Tuy nhiên hiện nay việc mua bán trên mạng cũng còn gặp phải nhiều vấn đề cần phải giải quyết như:

Một là người tiêu dùng lo ngại về chất lượng sản phẩm.

Hai là lo ngại về hàng hóa không đến được tay người tiêu dùng.

Ba là lo ngại về tính bảo mật thông tin của khách hàng trên website thương mại điện tử.

Bốn là lo ngại về tính bảo mật của hệ thống thanh toán trên website thương mại điện tử.

Mặt khác việc bán hàng qua mạng có nhiều lợi ích hơn so với bán hàng truyền thống như:

Một là quảng bá thông tin và tiếp thị trong thị trường toàn cầu với chi phí thấp.

Hai là cung cấp dịch vụ tốt hơn cho khách hàng.

Ba là tăng doanh thu và giảm chi phí.

Bốn là tạo lợi thế cạnh tranh.

Do đó, để tạo nên một website thương mại điện tử chuyên nghiệp và tạo được lòng tin cho khách hàng là một việc không dễ dàng.

Từ những lý do trên em đã đăng ký đề tài: “Xây dựng website bán đồ tennis sử dụng ASP.Net MVC” làm đồ án tốt nghiệp. Mặc dù đã cố gắng rất nhiều để hoàn thành báo cáo nhưng do thời gian có hạn và kiến thức thực tế chưa nhiều nên còn nhiều thiếu sót cần được bổ xung. Em mong nhận được ý kiến đóng góp của thầy cô và bạn bè.

## 1.2. Mục đích

Thông tin hiện nay là một lĩnh vực rất quan trọng đối với cuộc sống hiện nay, bên cạnh đó các nhu cầu nghe nhìn, giải trí, trao đổi thông tin,... Vì vậy cần các mặt hàng khác nhau để phục vụ nhu cầu trên, nhưng để chọn được những mặt hàng như ý đúng với nhu cầu sử dụng thì không phải dễ, với lý do đó mà em xây dựng nên website bán hàng trực tuyến. Nhằm một phần nào đó giúp khách hàng có thể lựa chọn và mua những mặt hàng mọi lúc mọi nơi mà không cần đi xa mà hàng về tận nhà chỉ cần một cú nhấp chuột.

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

Với mục tiêu cung cấp hệ thống toàn diện trong việc quản lý cửa hàng, đưa thông tin sản phẩm đến với khách hàng nhằm mở rộng khách hàng cũng như tăng thêm doanh số cho cửa hàng. Website phải đáp ứng về mặt giao diện để tăng sự chú ý và sử dụng của người dùng, bên đó là yếu quan trọng nhất là chức năng phải đảm bảo cung cấp đầy đủ và tiện ích để tăng hiệu suất và chất lượng trong quá trình trải nghiệm cũng như sử dụng. Đối tượng nghiên cứu: các sinh viên có nhu cầu mua đồ Tenis,những chủ cửa hàng đồ Tenis trên địa bàn muốn đưa cửa hàng mình chuyển đổi số. Phạm vi nghiên cứu: toàn bộ sinh viên học sinh các trường học quanh khu vực Hà Nội, các cửa hàng đồ Tenis quanh khu vực Hà Nội.

## 1.4. Giới hạn của đề tài.

Khách hàng.

Có kết nối mạng Internet

Các thiết bị sử dụng trên mạng như máy tính, điện thoại ....

## 1.5. Nội dung thực hiện.

Website mà em xây dựng có thể giúp ích cho cửa hàng có thể dễ dàng quảng bá sản phẩm, cập nhật thông tin của các sản phẩm mới, sản phẩm bán chạy của cửa hàng, tư vấn trực tiếp cho khách hàng qua internet, góp phần quảng bá thương hiệu, giảm chi phí kinh doanh và tăng doanh thu cho cửa hàng.

Đối với khách hàng: Khách hàng có thể xem được các thông tin chi tiết về sản phẩm của cửa hàng, tìm kiếm sản phẩm, có thể đăng nhập để mua hàng, gửi các góp ý đến với cửa hàng đến cửa hàng có thể hoàn thiện hơn, làm cho cửa hàng ngày càng trở nên vững mạnh hơn.

# CHƯƠNG 2: CƠ SỞ LÝ THUYẾT.

## 2.1. Lịch sử ASP .NET.

ASP.NET là một framework phát triển bởi Microsoft, xuất hiện lần đầu tiên vào năm 2002. Ban đầu, nó là một phần của .NET Framework truyền thống. Từ phiên bản ASP.NET Core, nó đã trở thành một phần của .NET Core và sau đó là .NET 5/6/7. Các phiên bản cập nhật đưa vào nhiều tính năng mới, cải thiện hiệu suất, và hỗ trợ các nền tảng đa dạng.

## 2.2. ASP .NET và .NET FRAMEWORK.

ASP.NET được xây dựng trên nền tảng .NET Framework, một framework chung cho phát triển ứng dụng trên nền Windows. Với sự xuất hiện của .NET Core và sau đó là .NET 5/6/7, ASP.NET đã được tái thiết kế để tận dụng những cải tiến này, bao gồm tốc độ, khả năng di động, và tích hợp với các dịch vụ đám mây.

## 2.3. Kiến Trúc của ASP.NET.

ASP.NET hỗ trợ cả hai mô hình chính là Web Forms và MVC. Web Forms tập trung vào tích hợp dữ liệu vào trang, trong khi MVC tách biệt logic xử lý và giao diện người dùng. Cả hai mô hình này đều được thiết kế để phát triển ứng dụng web một cách dễ dàng và linh hoạt. Ngoài ra, Web API trong ASP.NET cung cấp khả năng xây dựng các dịch vụ web RESTful.

## 2.4. Ngôn Ngữ Lập Trình.

Ngôn ngữ chủ yếu được sử dụng trong ASP.NET là C#. Có nhiều lợi ích khi sử dụng C#, bao gồm sự tích hợp chặt chẽ với .NET Framework, kiểm soát tốt về bộ nhớ và quản lý lỗi dễ dàng.

## 2.5 Công Nghệ và Công Cụ Phát Triển.

Visual Studio là môi trường phát triển chính cho ASP.NET, cung cấp công cụ mạnh mẽ cho phát triển, gỡ lỗi và triển khai ứng dụng. ASP.NET Core có thể phát triển trên nền tảng không phải Windows và có thể sử dụng Visual Studio Code. Entity Framework và LINQ: Entity Framework là một ORM (Object-Relational Mapping) cho phép tương tác với cơ sở dữ liệu thông qua đối tượng. LINQ là một phần quan trọng trong ASP.NET, cung cấp cú pháp linh hoạt để truy vấn dữ liệu từ các nguồn khác nhau.

## 2.6. Lợi Ích của ASP.NET.

ASP.NET đem lại hiệu suất cao và tốc độ phản hồi nhanh cho người dùng. Nó hỗ trợ đa nền tảng, cho phép triển khai trên nền tảng Windows, Linux và macOS. Bảo mật cao được tích hợp, và khả năng tích hợp dịch vụ đám mây giúp quản lý tài nguyên một cách hiệu quả. Phát Triển Ứng Dụng Web Hiện Đại: ASP.NET tích hợp tốt với các công nghệ front-end như Angular, React, và Vue.js. Sử dụng SignalR cho phát triển ứng dụng web thời gian thực, giúp tạo ra trải nghiệm người dùng tốt hơn.

## 2.7. Tổng quan về SQL Server.

### 2.7.1. Khái niệm.

SQL Server chính là một hệ quản trị dữ liệu quan hệ sử dụng câu lệnh SQL để trao đổi dữ liệu giữa máy cài SQL Server và máy Client. Một Relational Database Management System – RDBMS gồm có: databases, datase engine và các chương trình ứng dụng dùng để quản lý các bộ phận trong RDBMS và những dữ liệu khác.

### 2.7.2. Các thành phần cơ bản trong SQL Server.

Các thành cơ bản trong SQL Server gồm có: Reporting Services, Database Engine, Integration Services, Notification Services, Full Text Search Service,… Tất cả kết hợp với nhau tạo thành một giải pháp hoàn chỉnh giúp cho việc phân tích và lưu trữ dữ liệu trở nên dễ dàng hơn.

Database Engine: Đây là một engine có khả năng chứa dữ liệu ở các quy mô dưới dạng support và table. Ngoài ra, nó còn có khả năng tự điều chỉnh ví dụ: trả lại tài nguyên cho hệ điều hành khi một user log off và sử dụng thêm các tài nguyên của máy khi cần.

Integration Services: là tập hợp các đối tượng lập trình và các công cụ đồ họa cho việc sao chép, di chuyển và chuyển đổi dữ liệu. Khi bạn làm việc trong một công ty lớn thì dữ liệu được lưu trữ ở nhiều nơi khác nhau như được chứa trong: Oracle, SQL Server, DB2, Microsoft Access,… và bạn chắc chắn sẽ có nhu cầu di chuyển dữ liệu giữa các server này. Ngoài ra, bạn còn muốn định dạng dữ liệu trước khi lưu vào database. Chắc chắn Integration Services sẽ giúp bạn giải quyết được công việc này dễ dàng.

Analysis Services: Đây là một dịch vụ phân tích dữ liệu rất hay của Microsoft. Dữ liệu khi được lưu trữ vào trong database mà bạn không thể lấy được những thông tin bổ ích thì coi như không có ý nghĩa gì. Chính vì thế, công cụ này ra đời giúp bạn trong việc phân tích dữ liệu một cách hiệu quả và dễ dàng bằng cách dùng kỹ thuật khai thác dữ liệu – datamining và khái niệm hình khối nhiều chiều – multi dimendion cubes.

Notification Services: Dịch vụ thông báo này là nền tảng cho sự phát triển và triển khai các ứng dụng soạn và gửi thông báo. Ngoài ra, dịch vụ này còn có chức năng gửi thông báo theo dịch thời đến hàng ngàn người dăng ký sử dụng trên nhiều loại thiết bị khác nhau.

Reporting Services: là một công cụ tạo, quản lý và triển khai báo cáo bao gồm: server và client. Ngoài ra, nó còn là nền tảng cho việc phát triển và xây dựng các ứng dụng báo cáo.

Full Text Search Service: là một thành phần đặc biệt trong việc truy vấn và đánh chỉ mục dữ liệu văn bản không cấu trúc được lưu trữ trong các cơ sở dữ liệu SQL Server.

Service Broker: là một môi trường lập trình cho việc tạo ra các ứng dụng trong việc nhảy qua các Instance.

# CHƯƠNG 3: PHÂN TÍCH HỆ THỐNG VÀ THIẾT KẾ CƠ SỞ DỮ LIỆU.

## 3.1. Danh sách các Actor.

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên Actor** | **Ý nghĩa** |
| 1 | Người quản lý | Người quản lý là người có toàn quyền tương tác với hệ thống, quyền sử dụng tất cả các chức năng mà website xây dựng, điều khiển và kiểm soát mọi hoạt động của website. |
| 2 | Người dùng | Người dùng ở đây là những người truy cập vào website, có thể đăng kí làm thành viên hoặc không. Người dùng chỉ có một số quyền nhất định đối với website. |

*Bảng 3. 1. Danh sách các Actor.*

## 3.2. Biểu đồ use case.

### 3.2.1. Biểu đồ use case tổng quát.

A diagram of a diagram

Description automatically generated

*Hình 3. 1.Biểu đồ use case tổng quát.*

Biểu đồ tổng quát các chức năng của hệ thống, gồm 2 tác nhân chính là admin và người dùng kèm theo các chức năng của từng đối tượng.

### 3.2.2. Biểu đồ use case người dùng.

A screenshot of a diagram

Description automatically generated

*Hình 3. 2. Biểu đồ usecase người dùng.*

Biểu đồ usecase theo tác nhân người dùng, gồm các chức năng chính như: Đăng ký tài khoản, Đặt mua, Tìm kiếm thông tin sản phẩm, Thanh toán, Đăng nhập.

### 3.2.3. Biểu đồ usecase Admin.

A diagram of a diagram

Description automatically generated

*Hình 3. 3. Biểu đồ usecase Admin.*

Biểu đồ usecase theo tác nhân admin, gồm các chức năng chính như: Quản lý thông tin, Nhập hàng, Thống kê, Đăng nhập.

### 3.2.4. Đặc tả usecase.

#### 3.2.4.1.. Đặc tả Usecase đăng nhập (Người dùng).

*Hình 3. 4. Đặc tả use case đăng nhập của người dùng.*

|  |  |
| --- | --- |
| Tóm tắt | Actor Người dùng là người sử dụng Use Case này. Use Case này được sử dụng để thực hiện chức năng đăng nhập vào website của người dùng. Sau khi đăng nhập vào website, lúc đó Người dùng mới có thể thực hiện được các chức năng khác, như là: Thanh toán tiền, xem sản phẩm… |
| Dòng sự kiện | a. Dòng sự kiện chính:  (1). Người dùng chọn chức năng đăng nhập từ trang chủ của website.  (2). Trang chủ sẽ mở đến trang đăng nhập.  (3). Người dùng nhập tài khoản của mình (bao gồm tên đăng nhập và mật khẩu).  (4). Hệ thống xác nhận thông tin đăng nhập.  (5). Hiển thị thông báo.  (6). Kết thúc Use Case.  b. Các dòng sự kiện khác:  \* Dòng sự kiện thứ nhất.  (1). Người dùng hủy yêu cầu đăng nhập.  (2). Hệ thống bỏ qua trang đăng nhập, trở lại trang hiện tại.  (3). Kết thúc Use Case.  \* Dòng sự kiện thứ hai.  (1). Người dùng nhập sai thông tin đăng nhập.  (2). Hệ thống từ chối đăng nhập, hiển thị thông báo và trang đăng nhập.  (3). Kết thúc Use Case. |
| \* Các yêu cầu đặc biệt | Không có yêu cầu đặc biệt. |
| Trạng thái hệ thống khi bắt đầu thực hiện Use Case | Không đòi hỏi bất cứ điều kiện nào trước đó. |
| Trạng thái hệ thống sau khi thực hiện Use Case | + Trường hợp đăng nhập thành công: hệ thống hiển thị trang chủ của hệ thống. Sau khi đăng nhập thành công, Người dùng có thể thực hiện các chức năng tương ứng với quyền đăng nhập của mình, như là thanh toán tiền, gửi ý kiến phản hồi  + Trường hợp đăng nhập thất bại: Hệ thống đưa ra thông báo đăng nhập không thành công và hiển thị trang chủ. |

*Bảng 3. 2. Đặc tả use case đăng nhập của người dùng.*

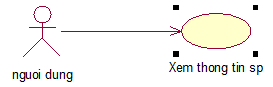
#### 3.2.4.2. Đặc tả Usecase đăng kí tài khoản.

*Hình 3. 5. Đặc tả use case đăng kí tài khoản.*

|  |  |
| --- | --- |
| Tóm tắt | Actor Người dùng là người sử dụng Use Case này. Use Case này được sử dụng để thực hiện chức năng đăng kí tài khoản của người dùng. Sau khi đăng kí tài khoản, lúc đó Người dùng mới có thể đăng nhập vào website và thực hiện được các chức năng như là: Thanh toán tiền, Gửi ý kiến phản hồi… |
| Dòng sự kiện | a. Dòng sự kiện chính  (1). Người dùng chọn chức năng đăng kí từ trang chủ của website.  (2). Trang chủ sẽ mở đến trang đăng kí.  (3). Người dùng nhập đầy đủ thông tin tài khoản và thông tin cá nhân.  (4). Hệ thống xác nhận thông tin và lưu lại.  (5). Hiển thị thông báo.  (6). Kết thúc Use Case.  b. Các dòng sự kiện khác.  (1). Người dùng hủy yêu cầu đăng kí.  (2). Hệ thống bỏ qua trang đăng kí, trở lại trang hiện tại.  (3). Kết thúc Use Case. |
| Các yêu cầu đặc biệt | Không có yêu cầu đặc biệt. |
| Trạng thái hệ thống khi bắt đầu thực hiện Use Case | Không đòi hỏi bất cứ điều kiện nào trước đó. |
| Trạng thái hệ thống sau khi thực hiện Use Case | + Trường hợp đăng kí thành công: hệ thống hiển thị trang đăng nhập để người dùng có thể đăng nhập vào website và thực hiện các chức năng tương ứng với quyền đăng nhập của mình, như là thanh toán tiền, gửi ý kiến phản hồi.  + Trường hợp đăng kí thất bại: Hệ thống đưa ra thông báo đăng kí không thành công và hiển thị trang chủ. |

*Bảng 3. 3. Đặc tả use case đăng kí tài khoản.*

#### 3.2.4.3. Đặc tả Usecase Xem thông tin sản phẩm.

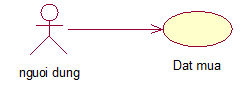


*Hình 3. 6. Đặc tả use case xem thông tin sản phẩm.*

|  |  |
| --- | --- |
| Tóm tắt | Actor Người dùng là người sử dụng Use Case này. Use Case này được sử dụng để thực hiện chức năng xem thông tin sản phẩm cảu người dùng. Sau khi truy cập vào website người dung có thể xem thông tin các sản phẩm có tại website. |
| Dòng sự kiện | a. Dòng sự kiện chính.  (1). Người dùng truy cập vào website và nhấn chọn sản phẩm.  (2). Trang chủ sẽ mở đến trang chi tiết sản phẩm..  (3). Người dùng xem thông tin sản phẩm.  (4). Kết thúc Use Case.  b. Các dòng sự kiện khác.  Không có. |
| Các yêu cầu đặc biệt | Không có yêu cầu đặc biệt. |
| Trạng thái hệ thống khi bắt đầu thực hiện Use Case | Không đòi hỏi bất cứ điều kiện nào trước đó. |
| Trạng thái hệ thống sau khi thực hiện Use Case | Hiển thị trang thông tin sản phẩm. |

*Bảng 3. 4. Đặc tả use case xem thông tin sản phẩm.*

#### 3.2.4.4. Đặc tả Usecase đặt mua.

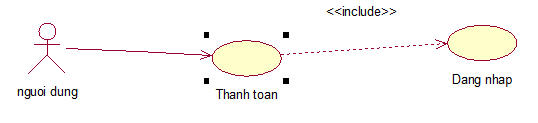
****

*Hình 3. 7. Đặt tả use case đặt mua.*

|  |  |
| --- | --- |
| Tóm tắt | Actor Người dùng là người sử dụng Use Case này. Use Case này được sử dụng để thực hiện chức năng đặt mua sản phẩm của người dùng. Sau khi truy cập vào website và xem thông tin sản phẩm, người dùng có thể đặt mua sản phẩm mà mình cảm thấy vừa ý. |
| Dòng sự kiện | a. Dòng sự kiện chính.  (1). Người dùng chọn chức năng xem thông tin chi tiết của sản phẩm.  (2). Trang chủ sẽ mở đến trang thông tin chi tiết của sản phẩm.  (3). Người dùng xem và nhấn vào nút “đặt mua”.  (4). Hệ thống xác nhận thông tin và lưu lại.  (5). Hiển thị trang thông tin sản phẩm.  (6). Kết thúc Use Case.  b. Các dòng sự kiện khác.  \* Dòng sự kiện thứ nhất.  (1). Người dùng hủy yêu cầu đặt mua.  (2). Hệ thống xóa sản phẩm đó tại giỏ hàng của người dùng trở lại trang hiện tại.  (3). Kết thúc Use Case.  \* Dòng sự kiện thứ hai.  (1). Người dùng thay đổi số lượng sản phẩm vừa đặt mua.  (2). Hệ thống kiểm tra và lưu lại.  (3). Kết thúc Use Case. |
| Các yêu cầu đặc biệt | Không có yêu cầu đặc biệt. |
| Trạng thái hệ thống khi bắt đầu thực hiện Use Case | Không đòi hỏi bất cứ điều kiện nào trước đó. |
| Trạng thái hệ thống sau khi thực hiện Use Case | + Trường hợp đặt mua thành công: hệ thống tiến hành thêm sản phẩm vào giỏ hàng của người đó và hiện ra trang thông tin sản phẩm để người dùng tiếp tục đặt mua.  + Trường hợp đặt mua thất bại: Hệ thống đưa ra thông báo và quay về trang trước đó. |

*Bảng 3. 5. Đặt tả use case đặt mua.*

#### 3.2.4.5. Đặc tả Usecase thanh toán.

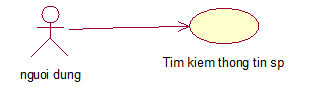
****

*Hình 3. 8. Đặc tả use case thanh toán.*

|  |  |
| --- | --- |
| Tóm tắt | Actor Người dùng là người sử dụng Use Case này. Use Case này được sử dụng để thực hiện chức năng thanh toán tiền cho sản phẩm đã đặt mua. Sau khi đã đặt mua sản phẩm, giỏ hàng của người dùng đã tồn tại một hoặc nhiều sản phẩm, nếu muốn có trong tay các sản phẩm này người dùng cần phải chọn UC này để thanh toán. |
| Dòng sự kiện | a. Dòng sự kiện chính:  (1). Người dùng chọn chức năng đặt mua  (2). Trang chủ sẽ mở đến trang giỏ hàng của người dùng.  (3). Người dùng xem và nhấn vào nút “thanh toán”  (4). Hệ thống xác nhận thông tin và hiển thị trang thanh toán  (5). Người dùng nhập đầy đủ thông tin  (6). Hệ thống đưa ra thông báo và quay trở lại trang giỏ hàng  (7). Kết thúc Use Case  b. Các dòng sự kiện khác:  \* Dòng sự kiện thứ nhất:  (1). Người dùng hủy yêu cầu thanh toán.  (2). Hệ thống hủy việc thanh toán, hiển thị trang trước đó.  (3). Kết thúc Use Case.  \* Dòng sự kiện thứ hai:  (1). Người dùng chưa đăng nhập.  (2). Hệ thống sẽ đưa ra thông báo và yêu cầu đăng nhập.  (3). Kết thúc Use Case. |
| Các yêu cầu đặc biệt | Không có yêu cầu đặc biệt. |
| Trạng thái hệ thống khi bắt đầu thực hiện Use Case | Người dùng phải đăng nhập vào hệt hống mới có thể thực hiện UC này. |
| Trạng thái hệ thống sau khi thực hiện Use Case | + Trường hợp thanh toán thành công: Hệ thống tiến hành tiếp nhận thông tin và chuyển sản phẩm đến tay khách hàng.  + Trường hợp thanh toán thất bại: Hệ thống đưa ra thông báo và quay về trang trước đó. |

*Bảng 3. 6. Đặc tả use case thanh toán.*

#### 3.2.4.6. Đặc tả Use Case Tìm kiếm sản phẩm.

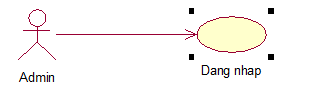
****

*Hình 3. 9. Đặc tả Use Case Tìm kiếm sản phẩm.*

|  |  |
| --- | --- |
| Tóm tắt | Actor Người dùng là người sử dụng Use Case này. Use Case này được sử dụng để thực hiện chức năng tìm kiếm sản phẩm. Với chức năng này, Người dùng có thể tìm kiếm sản phẩm mà mình yêu thích và xem sản phẩm đó có tồn tại trong website hay không? |
| Dòng sự kiện | a. Dòng sự kiện chính:  (1). Người dùng chọn chức năng tìm kiếm tài liệu từ trang chủ .  (2). Người dùng nhập thông tin tìm kiếm.  (4). Hệ thống xác nhận thông tin và thực hiện tìm kiếm.  (5). Hệ thống trả về thông tin tìm kiếm.  (6). Kết thúc Use Case.  b. Các dòng sự kiện khác.  \* Dòng sự kiện thứ nhất:  (1). Người dùng hủy việc việc tìm kiếm.  (2). Hệ thống bỏ qua trang tìm kiếm, trở lại trang chủ.  (3). Kết thúc Use Case.  \* Dòng sự kiện thứ hai:  (1). Hệ thống có lỗi trong quá trình tìm kiếm.  (2). Hệ thống thông báo lỗi.  (3). Kết thúc Use Case. |
| Các yêu cầu đặc biệt | Không có yêu cầu nào đặc biệt. |
| Trạng thái hệ thống khi bắt đầu thực hiện Use Case | Không đòi hỏi bất cứ điều kiện nào trước đó. |
| Trạng thái hệ thống sau khi thực hiện Use Case | + Trường hợp thành công: hệ thống hiển thị thông tin liên quan đến sản phẩm mà người dùng tìm kiếm.  + Trường hợp thất bại: Hệ thống đưa ra thông báo lỗi và trở lại giao diện chính. |

*Bảng 3. 7. Đặc tả Use Case Tìm kiếm sản phẩm.*

#### 3.2.4.7. Đăng nhập (Admin).

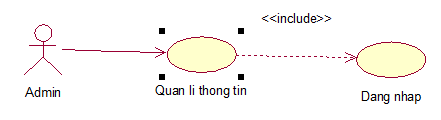


*Hình 3. 10. Đăng nhập ( Admin).*

|  |  |
| --- | --- |
| Tóm tắt | Actor Admin là người sử dụng Use Case này. Use Case này được sử dụng để thực hiện chức năng đăng nhập vào hệ thống của Admin. Sau khi đăng nhập vào hệ thống, trang quản trị sẽ hiển thị và Admin sẽ sử dụng được các chức năng như quản lý, thống kê. |
| Dòng sự kiện | a. Dòng sự kiện chính:  (1). Admin chọn chức năng đăng nhập từ trang chủ.  (2). Hệ thống hiển thị trang đăng nhập.  (3). Admin nhập tài khoản của mình (bao gồm tên đăng nhập và mật khẩu).  (4). Hệ thống xác nhận thông tin đăng nhập.  (5). Hệ thống xử lý quyền đăng nhập.  (6). Hiển thị trang quản trị.  (7). Kết thúc Use Case.  b. Các dòng sự kiện khác.  \* Dòng sự kiện thứ nhất:  (1). Admin hủy yêu cầu đăng nhập.  (2). Hệ thống bỏ qua trang đăng nhập, trở lại trang chủ.  (3). Kết thúc Use Case.  \* Dòng sự kiện thứ hai:  (1). Admin nhập sai thông tin đăng nhập.  (2). Hệ thống từ chối đăng nhập, hiển thị thông báo và trở về trang chủ.  (3). Kết thúc Use Case. |
| Các yêu cầu đặc biệt | Không có yêu cầu đặc biệt. |
| Trạng thái hệ thống khi bắt đầu thực hiện Use Case | Không đòi hỏi bất cứ điều kiện nào trước đó. |
| Trạng thái hệ thống sau khi thực hiện Use Case | + Trường hợp đăng nhập thành công: Hệ thống hiển thị trang chủ quản trị.  + Trường hợp đăng nhập thất bại: Hệ thống đưa ra thông báo đăng nhập không thành công và hiển thị trang chủ. |

*Bảng 3. 8. Đăng nhập ( Admin).*

#### 3.2.4.8. Đặc tả Usecase Quản lý thông tin.

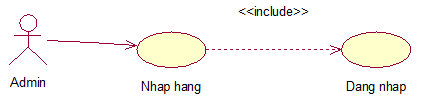


*Hình 3. 11. Đặc tả Usecase Quản lý thông tin.*

|  |  |
| --- | --- |
| Tóm tắt | Actor Admin là người sử dụng Use Case này. Use Case này được sử dụng để thực hiện chức năng quản lý thông tin của Admin. Sau khi đăng nhập vào hệ thống, Admin có thể thêm, cập nhật thông tin, xóa các đối tượng có trong hệ thống như: sản phẩm, … |
| Dòng sự kiện | a. Dòng sự kiện chính:  (1). Admin chọn chức năng quản lý thông tin từ trang chủ quản trị.  (2). Hệ thống hiển thị trang quản lý thông tin.  (3). Admin nhập đầy đủ thông tin của các đối tượng.  (4). Hệ thống kiểm tra và lưu thông tin.  (5). Hiển thị thông báo.  (6). Hiển thị trang quản lý thông tin.  (7). Kết thúc Use Case.  b. Các dòng sự kiện khác:  \* Dòng sự kiện thứ nhất:  (1). Admin hủy yêu cầu quản lý thông tin.  (2). Hệ thống bỏ qua trang quản lý thông tin, trở lại trang chủ quản trị.  (3). Kết thúc Use Case.  \* Dòng sự kiện thứ hai:  (1). Admin nhập sai hoặc thiếu thông tin của các đối tượng.  (2). Hệ thống từ chối thao tác, hiển thị thông báo, trở về trang quản lý thông tin.  (3). Kết thúc Use Case. |
| Các yêu cầu đặc biệt: | Không có yêu cầu đặc biệt. |
| Trạng thái hệ thống khi bắt đầu thực hiện Use Case | Yêu cầu Admin phải đăng nhập vào hệ thống. |
| Trạng thái hệ thống sau khi thực hiện Use Case | + Trường hợp quản lý thông tin thành công: Hệ thống ghi nhận thao tác, tiến hành cập nhật lai dữ liệu và đưa ra thông báo thành công.  + Trường hợp quản lý thông tin thất bại: Hệ thống đưa ra thông báo và hiển thị trang quản lý thông tin. |

*Bảng 3. 9. Đặc tả Usecase Quản lý thông tin.*

#### 3.2.4.9. Đặc tả Usecase Nhập hàng.



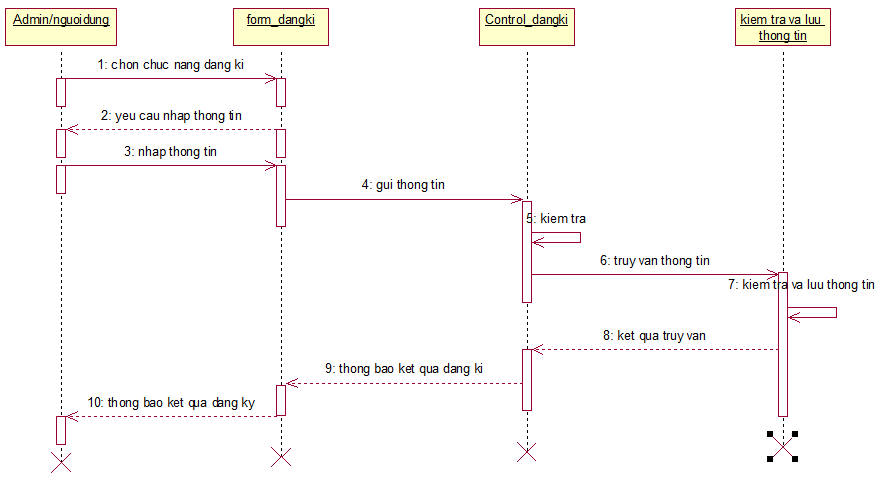
*Hình 3. 12. Đặc tả Usecase Nhập hàng.*

|  |  |
| --- | --- |
| Tóm tắt | Actor Admin là người sử dụng Use Case này. Use Case này được sử dụng để thực hiện chức năng nhập hàng của Admin. Sau khi đăng nhập vào hệ thống, Admin có thể nhập hàng về. |
| Dòng sự kiện | a. Dòng sự kiện chính:  (1). Admin chọn chức năng nhập hàng từ trang chủ quản trị.  (2). Hệ thống hiển thị trang nhập hàng.  (3). Admin nhập đầy đủ thông tin.  (4). Hệ thống kiểm tra và lưu thông tin.  (5). Hiển thị thông báo.  (6). Hiển thị trang nhập hàng.  (7). Kết thúc Use Case.  b. Các dòng sự kiện khác:  \* Dòng sự kiện thứ nhất:  (1). Admin hủy yêu cầu nhập hàng.  (2). Hệ thống bỏ qua trang nhập hàng, trở lại trang chủ quản trị.  (3). Kết thúc Use Case.  \* Dòng sự kiện thứ hai:  (1). Admin nhập sai hoặc thiếu thông tin nhập hàng.  (2). Hệ thống từ chối thao tác, hiển thị thông báo, trở về trang quản lý thông tin.  (3). Kết thúc Use Case. |
| Các yêu cầu đặc biệt | Không có yêu cầu đặc biệt. |
| Trạng thái hệ thống khi bắt đầu thực hiện Use Case | Yêu cầu Admin phải đăng nhập vào hệ thống. |
| Trạng thái hệ thống sau khi thực hiện Use Case | + Trường hợp nhập hàng thành công: Hệ thống ghi nhận thao tác, tiến hành cập nhật lai dữ liệu và đưa ra thông báo thành công.  + Trường hợp nhập hàng thất bại: Hệ thống đưa ra thông báo và hiển thị trang nhập hàng. |

*Bảng 3. 10. Đặc tả Usecase Nhập hàng.*

### 3.3. Biểu đồ tuần tự.

#### 3.3.1. Chức năng đăng kí.

****

*Hình 3. 13. Biểu đồ tuần tự chức năng đăng kí.*

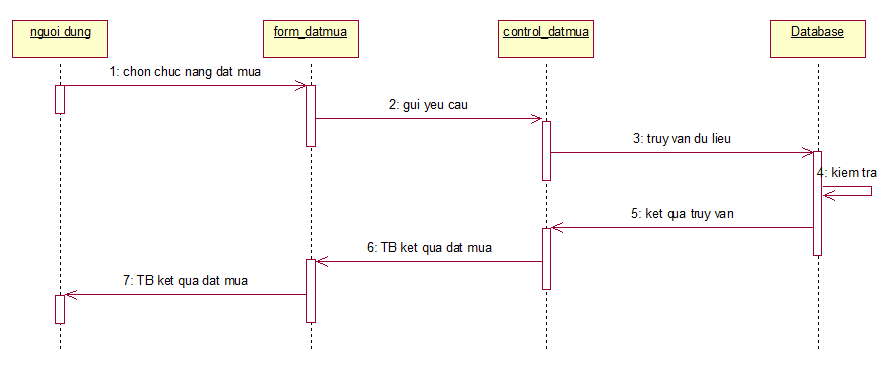
**Mô tả:** Để nhập đăng kí, Admin/ người dùng cần phải chọn chức năng đăng kí, nhập thông tin đầy đủ và gửi thông tin. Hệ thống sẽ kiểm tra và tiến hành truy vấn thông tin, kiểm tra và lưu thông tin. Hệ thống sẽ trả kết quả truy vấn và thông báo kết quả đăng kí.

#### 3.3.2. Chức năng đăng nhập.

*Hình 3. 14. Biểu đồ tuần tự chức năng đăng nhập.*

**Mô tả:** Sau khi khách hàng đã đăng ký tài khoản thành công. Khách hàng muốn đăng nhập vào hệ thống, khách hàng nhấp vào đăng nhập, giao diện đăng nhập hiển thị ra, điền thông tin đăng nhập vào. Nếu khách hàng điền đúng thông tin đã đăng ký trước đó thì hệ thống sẽ gửi thông tinđi và kiểm tra. Nếu đúng sẽ đăng nhập vào hệ thống thành công. Nếu không đúng thì khách hàng vui lòng kiểm tra lại thông tin và nhập lại.

#### 3.3.3. Chức năng đặt mua.

****

*Hình 3. 15. Biểu đồ tuần tự chức năng đặt mua*.

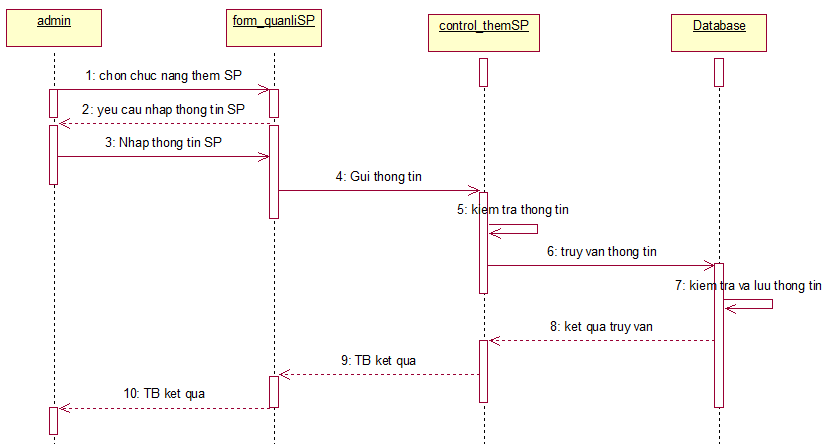
**Mô tả:** Khi người dùng bấm vào nút “ mua ngay”, hệ thống sẽ gửi yêu cầu và tiến hành truy vấn và kiểm tra. Sau đó, hệ thống trả kết quả và thông báo đã đặt mua thành công.

#### 3.3.4. Chức năng thanh toán.

*Hình 3. 16. Biểu đồ tuần tự chức năng thanh toán*.

**Mô tả:** Khi người dùng bấm vào nút “ thanh toán” hệ thống sẽ hiển thị ra các hình thức thanh toán để người dùng lựa chọn. Sau đó, người dùng điền đầy đủ các thông tin để giao hàng, hệ thống sẽ truy vấn thông tin, kiểm tra và thông báo kết quả cho người dùng.

#### 3.3.5. Chức năng thêm sản phẩm.

****

*Hình 3. 17. Biểu đồ tuần tự chức năng thêm sản phẩm.*

**Mô tả:** Quản trị viên có thể thêm sản phẩm, quản trị viên nhấp “thêm sản phẩm”, giao diện sẽ gửi về thông tin để thêm sản phẩm. Quản trị viên nhập thông tin sản phẩm vào. Sau đó giao diện sẽ gửi thông tin đó về hệ thống. Hệ thống sẽ kiểm tra, lưu thông tin và đưa ra kết quả “thêm sản phẩm thành công”.

#### 3.3.6. Chức năng sửa sản phẩm.

*Hình 3. 18. Biểu đồ tuần tự chức năng cập nhật thông tin sản phẩm.*

**Mô tả:** Quản trị viên có thể sửa thông tin sản phẩm bằng cách quản trị viên chọn sản phẩm muốn sửa. Giao diện sẽ đưa quản trị viên đến trang danh sách các sản phẩm. Quản trị viên sửa thông tin sản phẩm mà quản trị viên muốn sửa. Giao diện sẽ gửi thông tin đã sửa của khách hàng về hệ thống, hệ thống sẽ kiểm tra, truy vấn thông tin và lưu thông tin, sau đó sẽ trả về kết quả kiểm tra.

#### 3.3.7. Chức năng xóa sản phẩm.

*Hình 3. 19. Biểu đồ tuần tự chức năng xóa sản phẩm*.

**Mô tả:** Quản trị viên vào chọn chức năng quản lý sản phẩm. Hệ thống hiển thị ra danh sách các sản phẩm. Quản trị viên lựa chọn sản phẩm cần xóa sau đó nhấp vào nút “xóa”. Hệ thống sẽ gửi yêu cầu đi, kiểm tra và xử lý yêu cầu. Hệ thống sẽ kiểm tra và trả về kết quả.

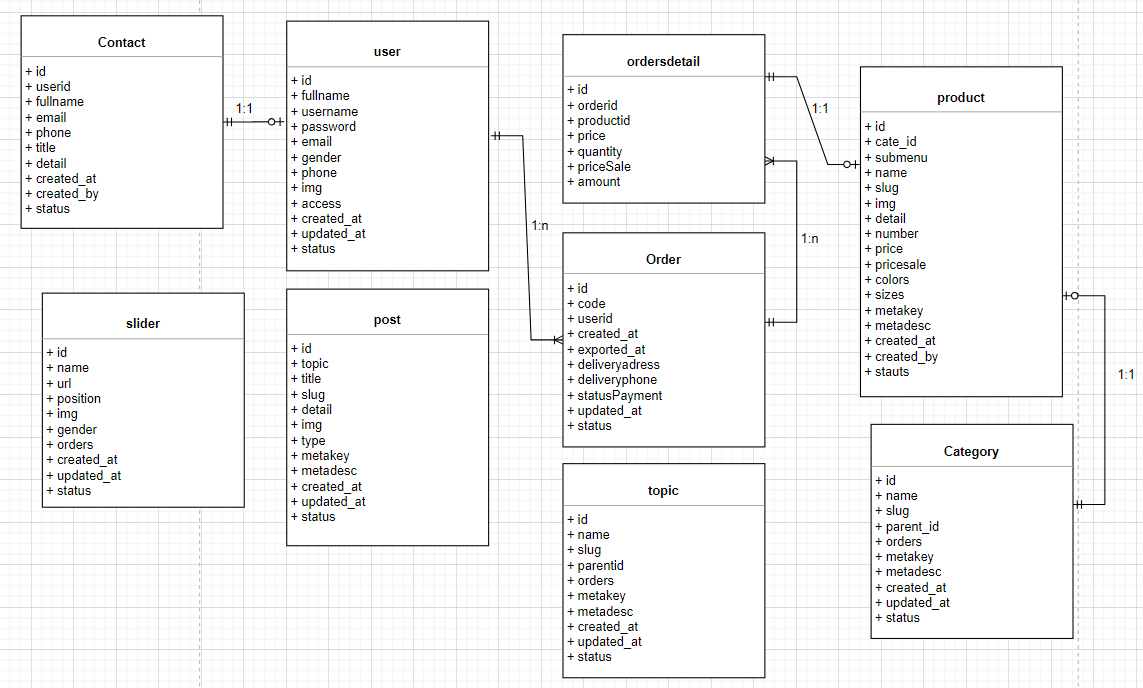
#### 3.3.8. Chức năng nhập hàng.

*Hình 3. 20. Biểu đồ tuần tự chức năng nhập hàng.*

#### 3.3.9. Chức năng tìm kiếm.

*Hình 3. 21. Biểu đồ tuần tự chức năng tìm kiếm.*

## 3.4. Sơ đồ lớp

**

*Hình 3. 22. Sơ đồ lớp*

## 3.6. Các bảng CSDL.

### 3.6.1. Bảng category.

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Ý nghĩa |
| 1 | ID | int | khóa chính |
| 2 | name | nvarchar(255) | tên danh mục |
| 3 | slug | varchar(255) | slug của danh mục |
| 4 | parentid | int | id danh mục cha |
| 5 | orders | int | số thứ tự xuất hiện |
| 6 | metakey | nvarchar(150) | key meta |
| 7 | metadesc | nvarchar(150) | thứ tự meta |
| 8 | created\_at | smalldatetime | thời gian tạo |
| 9 | created\_by | int | người tạo |
| 10 | updated\_at | smalldatetime | thời gian cập nhật |
| 11 | updated\_by | int | người cập nhật |
| 12 | status | int | trạng thái |

*Hình 3. 23. Bảng category.*

**Mô tả:** Bảng được thiết kế để lưu trữ thông tin về các danh mục trong hệ thống. Mỗi dòng trong bảng đại diện cho một danh mục cụ thể và chứa các thuộc tính sau: ID (int): Đây là khóa chính của bảng, là một số nguyên dương được sử dụng để định danh duy nhất cho mỗi danh mục trong hệ thống. Name (nvarchar(255)): Là tên của danh mục, được lưu trữ dưới dạng chuỗi với độ dài tối đa là 255 ký tự. Slug (varchar(255)): Thể hiện một chuỗi ngắn được sử dụng trong các URL hoặc đường dẫn thân thiện với SEO để đại diện cho danh mục. ParentID (int): Chứa ID của danh mục cha, nếu danh mục hiện tại là một phần của danh mục lớn hơn. Orders (int): Đây là một số nguyên thể hiện thứ tự xuất hiện của danh mục, quyết định vị trí hiển thị trong giao diện người dùng. Metakey (nvarchar(150)): Là một chuỗi dùng để lưu trữ các từ khóa quan trọng liên quan đến danh mục, hỗ trợ tối ưu hóa SEO. Metadesc (nvarchar(150)): Là mô tả meta của danh mục, chứa thông tin ngắn về nội dung của danh mục để cải thiện SEO. Created\_at (smalldatetime): Thời điểm tạo danh mục, được ghi lại dưới định dạng smalldatetime. Created\_by (int): Chứa ID của người tạo danh mục. Updated\_at (smalldatetime): Thời điểm cập nhật danh mục cuối cùng, cũng sử dụng định dạng smalldatetime. Updated\_by (int): Là ID của người cập nhật danh mục. Status (int): Trạng thái của danh mục, được biểu diễn dưới dạng số nguyên. Bảng này đem lại cấu trúc tổ chức cho việc quản lý và hiển thị các danh mục trong hệ thống, giúp tối ưu hóa quá trình quản lý và tìm kiếm thông tin.

### 3.6.2. Bảng contact.

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Ý nghĩa |
| 1 | ID | int | khóa chính |
| 2 | fullname | nvarchar(100) | tên chi tiết |
| 3 | email | varchar(100) | email |
| 4 | phone | varchar(15) | số điện thoại |
| 5 | title | nvarchar(255) | tiêu đề |
| 6 | detail | ntext | chi tiết |
| 7 | created\_at | smalldatetime | ngày tạo |
| 8 | created\_by | int | người tạo |
| 9 | updated\_at | smalldatetime | ngày cập nhật |
| 10 | updated\_by | int | người cập nhật |
| 11 | status | int | trạng thái |

*Hình 3. 24. Bảng contact.*

**Mô tả:** Bảng trên được thiết kế để lưu trữ thông tin về các mục liên quan đến chi tiết người dùng hoặc các thông tin liên hệ trong hệ thống. Dưới đây là mô tả chi tiết các thuộc tính của bảng: ID (int): Đây là trường dữ liệu kiểu số nguyên, chứa giá trị định danh duy nhất cho mỗi mục trong bảng, và đồng thời đóng vai trò là khóa chính. Fullname (nvarchar(100)): Là trường dữ liệu kiểu chuỗi, được sử dụng để lưu trữ tên chi tiết của người liên quan, với độ dài tối đa là 100 ký tự. Email (varchar(100)): Trường này chứa địa chỉ email của người liên quan, được lưu trữ dưới dạng chuỗi với độ dài tối đa 100 ký tự. Phone (varchar(15)): Là trường kiểu chuỗi, dùng để lưu trữ số điện thoại liên quan, với độ dài tối đa là 15 ký tự. Title (nvarchar(255)): Trường này chứa tiêu đề liên quan đến mục dữ liệu, có kiểu dữ liệu là chuỗi và độ dài tối đa là 255 ký tự. Detail (ntext): Là trường dữ liệu kiểu văn bản lớn (ntext), được sử dụng để lưu trữ các thông tin chi tiết, mô tả, hoặc nội dung dài hơn. Created\_at (smalldatetime): Lưu trữ thời điểm mục dữ liệu được tạo, dưới định dạng smalldatetime. Created\_by (int): Chứa ID của người tạo mục dữ liệu, là một số nguyên. Updated\_at (smalldatetime): Ghi lại thời điểm cập nhật mục dữ liệu cuối cùng, cũng dưới định dạng smalldatetime. Updated\_by (int): Chứa ID của người cập nhật mục dữ liệu, là một số nguyên. Status (int): Là trường chứa thông tin về trạng thái của mục dữ liệu, có thể là các giá trị số nguyên thể hiện trạng thái khác nhau trong hệ thống. Bảng này được thiết kế để cung cấp một cơ sở dữ liệu đồng bộ và có tổ chức cho thông tin liên quan đến chi tiết người dùng hoặc các thông tin liên hệ khác.

### 3.6.3. Bảng link.

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Ý nghĩa |
| 1 | ID | int | khóa chính |
| 2 | slug | varchar(MAX) | slug |
| 3 | tableId | int | id của bảng |
| 4 | type | varchar(200) | loại bảng |
| 5 | parentId | int | id cấp cha |

*Hình 3. 25. Bảng link.*

**Mô tả:** Bảng trên có mục đích lưu trữ thông tin về các bảng trong hệ thống và quan hệ giữa chúng. Dưới đây là mô tả chi tiết các thuộc tính của bảng: ID (int): Đây là trường chứa giá trị số nguyên, đóng vai trò là khóa chính của bảng. Mỗi dòng trong bảng có một ID duy nhất để định danh. Slug (varchar(MAX)): Trường này là kiểu dữ liệu chuỗi (varchar) có độ dài tối đa là MAX, được sử dụng để lưu trữ thông tin "slug". Slug thường được sử dụng trong các đường dẫn thân thiện với SEO hoặc URL để đại diện cho bảng. TableId (int): Là trường chứa giá trị số nguyên, đại diện cho ID của bảng mà mục dữ liệu liên quan đến. Type (varchar(200)): Trường này là kiểu dữ liệu chuỗi với độ dài tối đa là 200 ký tự, chứa thông tin về loại bảng. Loại bảng có thể là một phân loại hoặc mô tả về mục đích sử dụng của bảng. ParentId (int): Là trường chứa giá trị số nguyên, đại diện cho ID của bảng cấp cha nếu bảng đang xét thuộc về một bảng lớn hơn. Bảng này giúp tổ chức và theo dõi các bảng trong hệ thống, đồng thời cung cấp thông tin về quan hệ giữa chúng. Trường Slug hỗ trợ việc tạo các đường dẫn thân thiện và dễ đọc trong các ứng dụng web, trong khi các trường khác như TableId, Type và ParentId giúp xác định và mô tả các quan hệ giữa các bảng.

### 3.6.4. Bảng menu.

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Ý nghĩa |
| 1 | ID | int | khóa chính |
| 2 | name | nvarchar(255) | tên |
| 3 | type | varchar(255) | loại |
| 4 | link | varchar(255) | link menu |
| 5 | tableid | int | id bảng |
| 6 | parentid | int | id cáp cha |
| 7 | orders | int | thứ tự sắp xếp |
| 8 | position | varchar(255) | vị trí |
| 9 | created\_at | smalldatetime | ngày tạo |
| 10 | created\_by | int | người tạo |
| 11 | updated\_at | smalldatetime | ngầy cập nhật |
| 12 | updated\_by | int | người cập nhật |
| 13 | status | int | trạng thái |

*Hình 3. 26. Bảng menu.*

**Mô tả:** Bảng này lưu trữ thông tin về các mục menu trong hệ thống, giúp tổ chức và hiển thị chúng một cách có tổ chức. Dưới đây là mô tả ngắn gọn các thuộc tính của bảng: ID (int): Khóa chính của bảng, chứa giá trị số nguyên để định danh duy nhất cho mỗi mục menu. Name (nvarchar(255)): Là tên của mục menu, được lưu trữ dưới dạng chuỗi với độ dài tối đa là 255 ký tự. Type (varchar(255)): Chứa thông tin về loại mục menu, có kiểu dữ liệu chuỗi với độ dài tối đa là 255 ký tự. Link (varchar(255)): Lưu trữ đường link tương ứng với mục menu, có kiểu dữ liệu chuỗi và độ dài tối đa là 255 ký tự. TableId (int): Chứa ID của bảng liên quan nếu mục menu này liên kết với một bảng trong hệ thống. ParentId (int): Là ID của mục menu cấp cha, nếu có. Giúp xác định mối quan hệ cấp độ giữa các mục menu. Orders (int): Thứ tự sắp xếp của mục menu, quyết định vị trí hiển thị trong giao diện người dùng. Position (varchar(255)): Mô tả vị trí của mục menu, có kiểu dữ liệu chuỗi với độ dài tối đa là 255 ký tự. Created\_at (smalldatetime): Ngày và thời gian mục menu được tạo. Created\_by (int): ID của người tạo mục menu. Updated\_at (smalldatetime): Ngày và thời gian mục menu được cập nhật lần cuối. Updated\_by (int): ID của người cập nhật mục menu. Status (int): Trạng thái của mục menu, có thể là các giá trị số nguyên biểu thị trạng thái khác nhau trong hệ thống.

### 3.6.5. Bảng order.

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Ý nghĩa |
| 1 | ID | int | khóa chính |
| 2 | code | nvarchar(255) | code |
| 3 | userid | int | id người dùng |
| 4 | created\_ate | smalldatetime | ngày tạo |
| 5 | exportdate | smalldatetime | ngày xuất |
| 6 | deliveryaddress | nvarchar(255) | địa chỉ giao hàng |
| 7 | deliveryname | nvarchar(100) | tên người nhận |
| 8 | deliveryphone | varchar(255) | sdt người nhận |
| 9 | deliveryemail | varchar(255) | email người nhận |
| 10 | deliveryPaymentMethod | nvarchar(255) | loại thanh toán |
| 11 | StatusPayment | int | trạng thái thanh toán |
| 12 | updated\_at | smalldatetime | ngày cập nhật |
| 13 | updated\_by | int | người cập nhật |
| 14 | status | int | trạng thái |

*Hình 3. 27. Bảng order.*

**Mô tả:** Bảng này dường như liên quan đến quản lý đơn hàng hoặc giao hàng trong hệ thống. Dưới đây là mô tả ngắn gọn các thuộc tính của bảng: ID (int): Khóa chính của bảng, chứa giá trị số nguyên để định danh duy nhất cho mỗi đơn hàng. Code (nvarchar(255)): Mã đặt hàng hoặc mã giao dịch, được lưu trữ dưới dạng chuỗi với độ dài tối đa là 255 ký tự. UserID (int): ID của người dùng đặt hàng, liên kết đơn hàng với người dùng cụ thể trong hệ thống. Created\_at (smalldatetime): Ngày và thời gian mà đơn hàng được tạo. ExportDate (smalldatetime): Ngày xuất hàng, là thời điểm mà đơn hàng được xuất ra khỏi kho để giao cho người nhận. DeliveryAddress (nvarchar(255)): Địa chỉ giao hàng, chứa thông tin về nơi nhận hàng của người đặt hàng. DeliveryName (nvarchar(100)): Tên người nhận hàng, lưu trữ dưới dạng chuỗi với độ dài tối đa là 100 ký tự. DeliveryPhone (varchar(255)): Số điện thoại của người nhận hàng, có kiểu dữ liệu chuỗi với độ dài tối đa là 255 ký tự. DeliveryEmail (varchar(255)): Địa chỉ email của người nhận hàng, có kiểu dữ liệu chuỗi với độ dài tối đa là 255 ký tự. DeliveryPaymentMethod (nvarchar(255)): Loại phương thức thanh toán được sử dụng cho đơn hàng. StatusPayment (int): Trạng thái thanh toán của đơn hàng, có thể là các giá trị số nguyên biểu thị trạng thái thanh toán khác nhau. Updated\_at (smalldatetime): Ngày và thời gian mà thông tin đơn hàng được cập nhật lần cuối. Updated\_by (int): ID của người cập nhật thông tin đơn hàng. Status (int): Trạng thái chung của đơn hàng, có thể là các giá trị số nguyên biểu thị trạng thái tổng quát của đơn hàng trong hệ thống.

### 3.6.6. Bảng ordersdetail.

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Ý nghĩa |
| 1 | ID | int | khóa chính |
| 2 | orderid | int | id đơn đặt |
| 3 | productid | int | id sản phẩm |
| 4 | price | float | giá tiền |
| 5 | quantity | int | số lượng |
| 6 | priceSale | int | giá giảm |
| 7 | amount | float | tỉ lệ |

*Hình 3. 28. Bảng ordersdetail.*

**Mô tả:** Bảng này dường như liên quan đến chi tiết sản phẩm trong mỗi đơn đặt hàng. Dưới đây là mô tả ngắn gọn các thuộc tính của bảng: ID (int): Khóa chính của bảng, chứa giá trị số nguyên để định danh duy nhất cho mỗi chi tiết sản phẩm trong đơn đặt hàng. OrderID (int): ID của đơn đặt hàng, liên kết chi tiết sản phẩm với đơn đặt hàng cụ thể trong hệ thống. ProductID (int): ID của sản phẩm, liên kết chi tiết sản phẩm với sản phẩm cụ thể trong hệ thống. Price (float): Giá tiền của sản phẩm, có kiểu dữ liệu số thực (float). Quantity (int): Số lượng sản phẩm được đặt, có kiểu dữ liệu số nguyên. PriceSale (int): Giá giảm (nếu có) được áp dụng cho sản phẩm, có kiểu dữ liệu số nguyên. Amount (float): Tổng giá trị của mỗi chi tiết sản phẩm, được tính dựa trên giá tiền, số lượng và giá giảm (nếu có). Bảng này giúp lưu trữ thông tin chi tiết về các sản phẩm được đặt trong mỗi đơn đặt hàng, bao gồm giá tiền, số lượng, và các thông tin khác liên quan đến giá và thanh toán.

### 3.6.7. Bảng post.

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Ý nghĩa |
| 1 | ID | int | id |
| 2 | topid | int | chủ đề id |
| 3 | title | nvarchar(MAX) | tiêu đề bài viết |
| 4 | slug | varchar(255) | slug |
| 5 | detail | ntext | chi tiết bài viết |
| 6 | img | varchar(255) | ảnh |
| 7 | type | varchar(50) | loại |
| 8 | metakey | nvarchar(150) | từ khóa meta |
| 9 | metadesc | nvarchar(150) | thứ tự meta |
| 10 | created\_at | smalldatetime | ngày tạo |
| 11 | created\_by | int | người tạo |
| 12 | updated\_at | smalldatetime | ngày sửa |
| 13 | updated\_by | int | người sửa |
| 14 | status | int | trạng thái |

*Hình 3. 29. Bảng post.*

**Mô tả:** Bảng này có vẻ liên quan đến quản lý nội dung bài viết hoặc tin tức trong hệ thống. Dưới đây là mô tả ngắn gọn về các thuộc tính của bảng: ID (int): Khóa chính của bảng, chứa giá trị số nguyên để định danh duy nhất cho mỗi bài viết. TopID (int): ID của chủ đề hoặc danh mục mà bài viết thuộc về. Title (nvarchar(MAX)): Tiêu đề của bài viết, được lưu trữ dưới dạng chuỗi với độ dài tối đa. Slug (varchar(255)): Trường này chứa một chuỗi ngắn (slug) được sử dụng trong các URL hoặc đường dẫn thân thiện với SEO để đại diện cho bài viết. Detail (ntext): Trường này là kiểu dữ liệu văn bản lớn (ntext), được sử dụng để lưu trữ nội dung chi tiết của bài viết. Img (varchar(255)): Đường dẫn hoặc tên file ảnh đại diện cho bài viết. Type (varchar(50)): Chứa thông tin về loại bài viết, có thể là các giá trị như "Tin tức", "Bài viết"... Metakey (nvarchar(150)): Từ khóa meta của bài viết, được sử dụng để tối ưu hóa SEO. Metadesc (nvarchar(150)): Mô tả meta của bài viết, chứa thông tin ngắn về nội dung của bài viết để cải thiện SEO. Created\_at (smalldatetime): Ngày và thời gian bài viết được tạo. Created\_by (int): ID của người tạo bài viết. Updated\_at (smalldatetime): Ngày và thời gian bài viết được sửa đổi lần cuối. Updated\_by (int): ID của người sửa đổi bài viết. Status (int): Trạng thái của bài viết, có thể là các giá trị số nguyên biểu thị trạng thái khác nhau trong hệ thống.

### 3.6.8. Bảng product.

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Ý nghĩa |
| 1 | ID | int | id |
| 2 | catid | int | id danh mục |
| 3 | Submenu | int |  |
| 4 | name | nvarchar(MAX) | tên sản phẩm |
| 5 | slug | nvarchar(255) | slug |
| 6 | img | varchar(100) | ảnh sản phẩm |
| 7 | detail | ntext | chi tiết sản phẩm |
| 8 | number | int | số lượng |
| 9 | price | float | giá tiền |
| 10 | pricesale | float | giảm giá |
| 11 | metakey | nvarchar(150) | từ khóa meta |
| 12 | metadesc | nvarchar(MAX) | thời gian meta |
| 13 | created\_at | smalldatetime | ngày tạo |
| 14 | created\_by | int | người tạo |
| 15 | updated\_at | smalldatetime | ngày sửa |
| 16 | updated\_by | int | người sửa |
| 17 | status | int | trạng thái |
| 18 | sold | int |  |

*Hình 3. 30. Bảng product.*

**Mô tả:** Bảng này có vẻ liên quan đến quản lý thông tin về sản phẩm trong hệ thống. Dưới đây là mô tả ngắn gọn về các thuộc tính của bảng: ID (int): Khóa chính của bảng, chứa giá trị số nguyên để định danh duy nhất cho mỗi sản phẩm. CatID (int): ID của danh mục mà sản phẩm thuộc về. Submenu (int): ID của submenu nếu sản phẩm thuộc một submenu cụ thể trong danh mục. Name (nvarchar(MAX)): Tên của sản phẩm, được lưu trữ dưới dạng chuỗi với độ dài tối đa. Slug (nvarchar(255)): Trường này chứa một chuỗi ngắn (slug) được sử dụng trong các URL hoặc đường dẫn thân thiện với SEO để đại diện cho sản phẩm. Img (varchar(100)): Đường dẫn hoặc tên file ảnh đại diện cho sản phẩm. Detail (ntext): Trường này là kiểu dữ liệu văn bản lớn (ntext), được sử dụng để lưu trữ chi tiết về sản phẩm. Number (int): Số lượng sản phẩm hiện có. Price (float): Giá tiền của sản phẩm, có kiểu dữ liệu số thực (float). PriceSale (float): Giảm giá của sản phẩm, cũng có kiểu dữ liệu số thực (float). Metakey (nvarchar(150)): Từ khóa meta của sản phẩm, được sử dụng để tối ưu hóa SEO. Metadesc (nvarchar(MAX)): Thông tin mô tả meta của sản phẩm, chứa thông tin ngắn về nội dung để cải thiện SEO. Created\_at (smalldatetime): Ngày và thời gian sản phẩm được tạo. Created\_by (int): ID của người tạo sản phẩm. Updated\_at (smalldatetime): Ngày và thời gian sản phẩm được sửa đổi lần cuối. Updated\_by (int): ID của người sửa đổi sản phẩm. Status (int): Trạng thái của sản phẩm, có thể là các giá trị số nguyên biểu thị trạng thái khác nhau trong hệ thống. Sold (int): Số lượng sản phẩm đã bán ra.

### 3.6.9. Bảng role.

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Ý nghĩa |
| 1 | ID | int | khóa chính |
| 2 | parentId | int | id cấp cha |
| 3 | accessName | varchar(255) | Tên quyền |
| 4 | description | nvarchar(225) | mô tả |
| 5 | GropID | varchar(50) | nhóm quyền |

*Hình 3. 31. Bảng role.*

**Mô tả:** Bảng này có vẻ liên quan đến quản lý quyền truy cập trong hệ thống. Dưới đây là mô tả ngắn gọn về các thuộc tính của bảng: ID (int): Khóa chính của bảng, chứa giá trị số nguyên để định danh duy nhất cho mỗi quyền truy cập. ParentID (int): ID của quyền truy cập cấp cha, nếu có. Điều này có thể tạo ra mối quan hệ cấp độ giữa các quyền truy cập. AccessName (varchar(255)): Tên của quyền truy cập, được lưu trữ dưới dạng chuỗi với độ dài tối đa là 255 ký tự. Description (nvarchar(225)): Mô tả về quyền truy cập, được lưu trữ dưới dạng chuỗi với độ dài tối đa là 225 ký tự. GroupID (varchar(50)): Nhóm quyền, có thể là một chuỗi nhóm đặc trưng cho các quyền truy cập cùng loại hoặc chức năng. Bảng này có thể được sử dụng để xây dựng và quản lý cấu trúc quyền truy cập trong hệ thống, trong đó quyền truy cập có thể được tổ chức thành các nhóm và có mối quan hệ cấp độ. Các trường như Description giúp mô tả chi tiết về mục đích và nhiệm vụ của từng quyền truy cập.

### 3.6.10. Bảng slider.

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Ý nghĩa |
| 1 | ID | int | khóa chính |
| 2 | name | nvarchar(255) | tên slider |
| 3 | url | varchar(255) | link slider |
| 4 | position | varchar(100) | vị trí |
| 5 | img | varchar(100) | ảnh |
| 6 | orders | int | sắp xếp |
| 7 | created\_at | smalldatetime | ngầy tạo |
| 8 | created\_by | int | người tạo |
| 9 | updated\_at | smalldatetime | ngày cập nhật |
| 10 | updated\_by | int | người cập nhật |
| 11 | status | int | trạng thái |

*Hình 3. 32. Bảng slider.*

**Mô tả:** Bảng này có vẻ liên quan đến quản lý thông tin về các slider trên giao diện của hệ thống hoặc trang web. Dưới đây là mô tả ngắn gọn về các thuộc tính của bảng: ID (int): Khóa chính của bảng, chứa giá trị số nguyên để định danh duy nhất cho mỗi slider. Name (nvarchar(255)): Tên của slider, được lưu trữ dưới dạng chuỗi với độ dài tối đa là 255 ký tự. URL (varchar(255)): Đường link liên kết của slider, chứa thông tin về đích đến khi người dùng nhấp vào slider. Position (varchar(100)): Mô tả vị trí của slider trên giao diện, có thể là các giá trị như "Trang chủ", "Danh mục",... Img (varchar(100)): Đường dẫn hoặc tên file ảnh của slider. Orders (int): Thứ tự sắp xếp của slider, quyết định vị trí hiển thị trên giao diện. Created\_at (smalldatetime): Ngày và thời gian slider được tạo. Created\_by (int): ID của người tạo slider. Updated\_at (smalldatetime): Ngày và thời gian slider được cập nhật lần cuối. Updated\_by (int): ID của người cập nhật slider. Status (int): Trạng thái của slider, có thể là các giá trị số nguyên biểu thị trạng thái khác nhau trong hệ thống. Bảng này giúp quản lý và hiển thị thông tin về các slider trên giao diện người dùng, trong đó mỗi slider có thể được đặc trưng bởi tên, hình ảnh, vị trí, và đường link tương ứng.

### 3.6.11. Bảng topic.

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Ý nghĩa |
| 1 | ID | int | khóa chính |
| 2 | name | nvarchar(255) | tên topic |
| 3 | slug | varchar(255) | slug topipc |
| 4 | parentid | int | id cấp cha |
| 5 | orders | int | thứ tự |
| 6 | metakey | varchar(150) |  |
| 7 | metadesc | nvarchar(MAX) |  |
| 8 | created\_at | smalldatetime | ngày tạo |
| 9 | created\_by | int | người tạo |
| 10 | updated\_at | smalldatetime | ngày sửa |
| 11 | updated\_by | int | người sửa |
| 12 | status | int | trạng thái |

*Hình 3. 33. Bảng topic.*

Mô tả: Bảng này có vẻ liên quan đến quản lý thông tin về các chủ đề (topic) hoặc danh mục trong hệ thống. Dưới đây là mô tả ngắn gọn về các thuộc tính của bảng: ID (int): Khóa chính của bảng, chứa giá trị số nguyên để định danh duy nhất cho mỗi chủ đề. Name (nvarchar(255)): Tên của chủ đề, được lưu trữ dưới dạng chuỗi với độ dài tối đa là 255 ký tự. Slug (varchar(255)): Trường này chứa một chuỗi ngắn (slug) được sử dụng trong các URL hoặc đường dẫn thân thiện với SEO để đại diện cho chủ đề. ParentID (int): ID của chủ đề cấp cha, nếu có. Điều này có thể tạo ra mối quan hệ cấp độ giữa các chủ đề. Orders (int): Thứ tự sắp xếp của chủ đề, quyết định vị trí hiển thị trên giao diện. Metakey (varchar(150)): Từ khóa meta của chủ đề, được sử dụng để tối ưu hóa SEO. Metadesc (nvarchar(MAX)): Thông tin mô tả meta của chủ đề, chứa thông tin ngắn về nội dung để cải thiện SEO. Created\_at (smalldatetime): Ngày và thời gian chủ đề được tạo. Created\_by (int): ID của người tạo chủ đề. Updated\_at (smalldatetime): Ngày và thời gian chủ đề được sửa đổi lần cuối. Updated\_by (int): ID của người sửa đổi chủ đề. Status (int): Trạng thái của chủ đề, có thể là các giá trị số nguyên biểu thị trạng thái khác nhau trong hệ thống.

### 3.6.12. Bảng user.

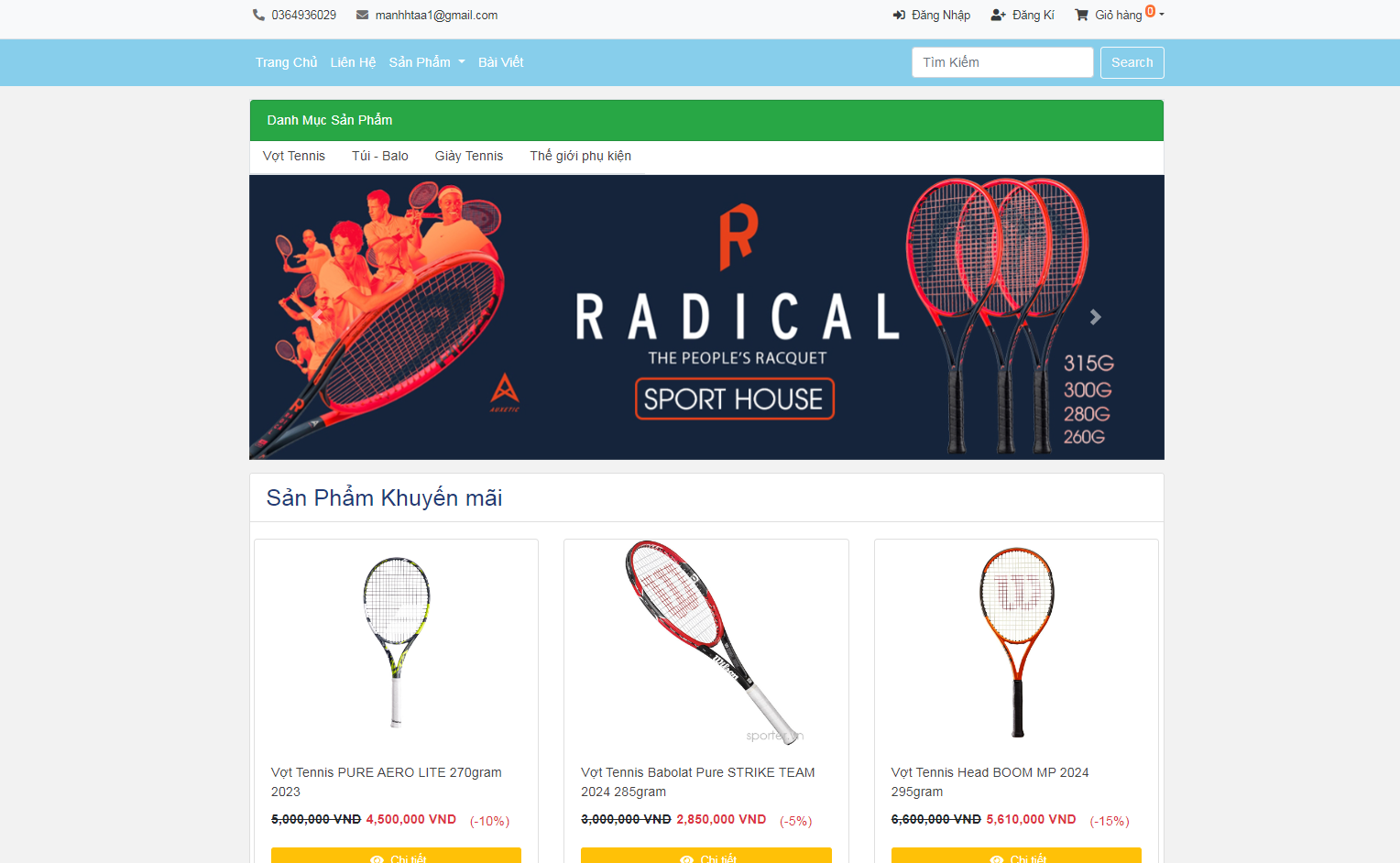
|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Ý nghĩa |
| 1 | ID | int | khóa chính |
| 2 | fullname | nvarchar(255) | tên người dùng |
| 3 | username | varchar(225) | tên tài khoản |
| 4 | password | varchar(64) | mật khẩu |
| 5 | email | varchar(255) | email |
| 6 | gender | nvarchar(5) | giới tính |
| 7 | phone | varchar(20) | số điện thoại |
| 8 | img | varchar(100) | ảnh |
| 9 | access | int |  |
| 10 | created\_at | smalldatetime | ngày tạo |
| 11 | created\_by | int | người tạo |
| 12 | updated\_at | smalldatetime | ngày sửa |
| 13 | updated\_by | int | người sửa |
| 14 | status | int | trạng thái |

*Hình 3. 34. Bảng user.*

# CHƯƠNG 4: KẾT QUẢ THỰC HIỆN

## 4.1. Giao diện người dùng:

### 4.1.1. Giao diện trang chủ:

**

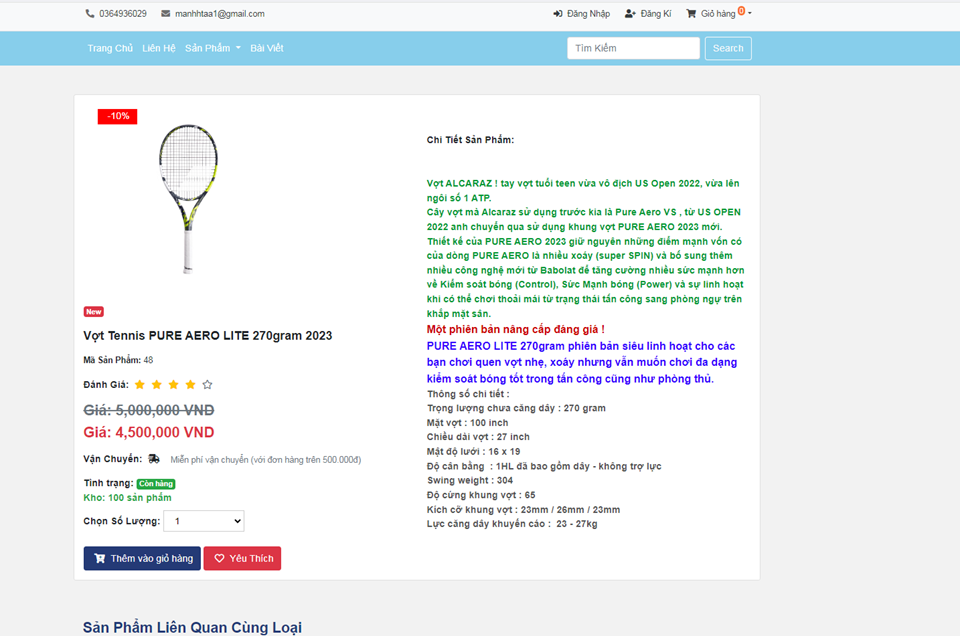
*Hình 4. 1. Giao diện trang chủ.*

### 4.1.2. Giao diện danh mục sản phẩm.



*Hình 4. 2. Giao diện danh mục sản phẩm.*

### 4.1.3. Giao diện chi tiết sản phẩm.

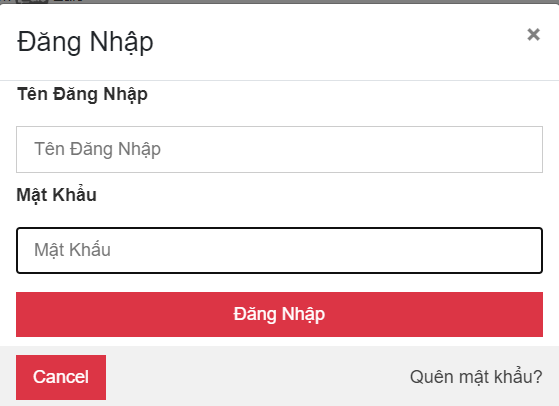
**

*Hình 4. 3. Giao diện chi tiết sản phẩm.*

### 4.1.4. Giao diện đăng kí.

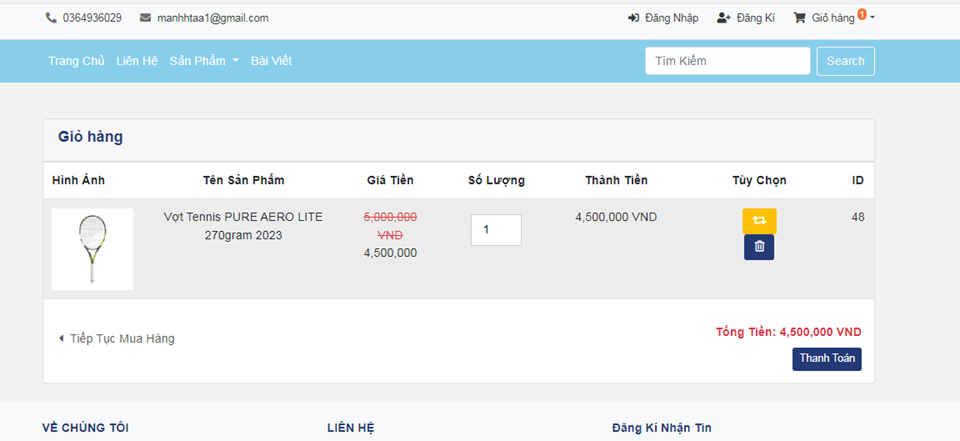
*Hình 4. 4. Giao diện đăng kí.*

### 4.1.5. Giao diện đăng nhập.



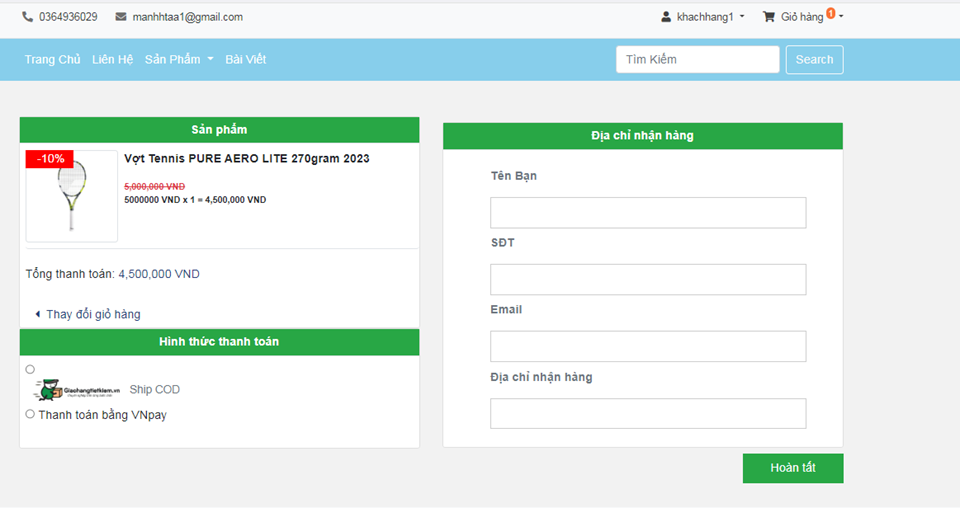
*Hình 4. 5. Giao diện đăng nhập.*

### 4.1.6. Giao diện giỏ hàng.



*Hình 4. 6. Giao diện giỏ hàng.*

### 4.1.7. Giao diện thanh toán.



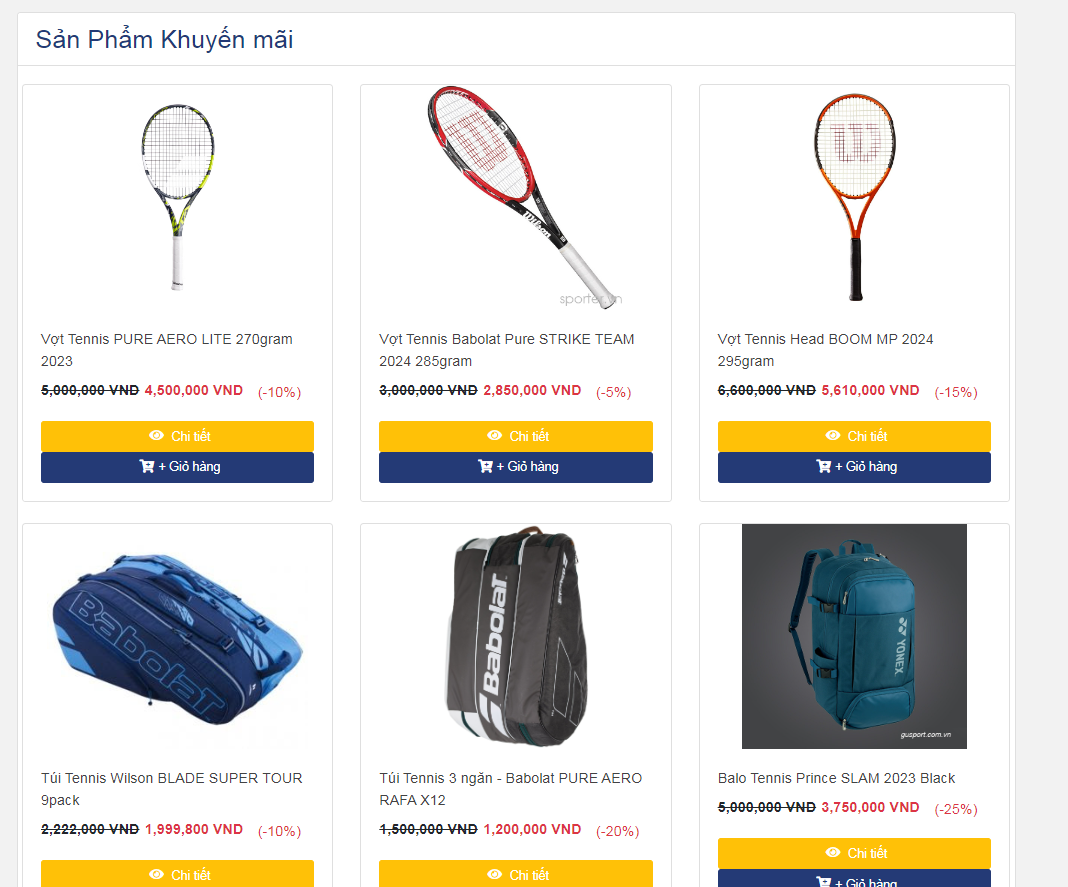
*Hình 4. 7. Giao diện thanh toán.*

### 4.1.8. Giao diện sau khi đặt hàng thành công.



*Hình 4. 8. Giao diện sau khi đặt hàng thành công.*

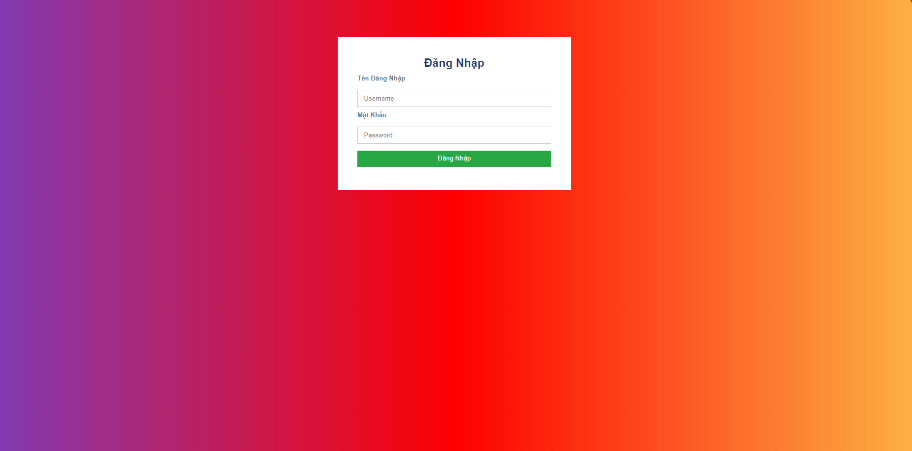
### 4.1.9. Giao diện sản phẩm khuyến mãi.

**

*Hình 4. 9. Giao diện sản phẩm khuyến mãi.*

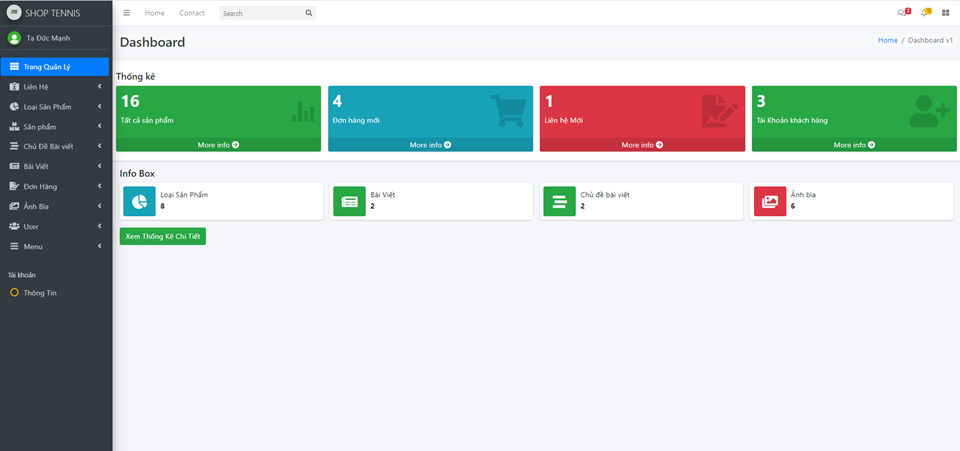
## 4.2. Giao diện trang quản lý.

### 4.2.1. Giao diện đăng nhập trang quản lý.



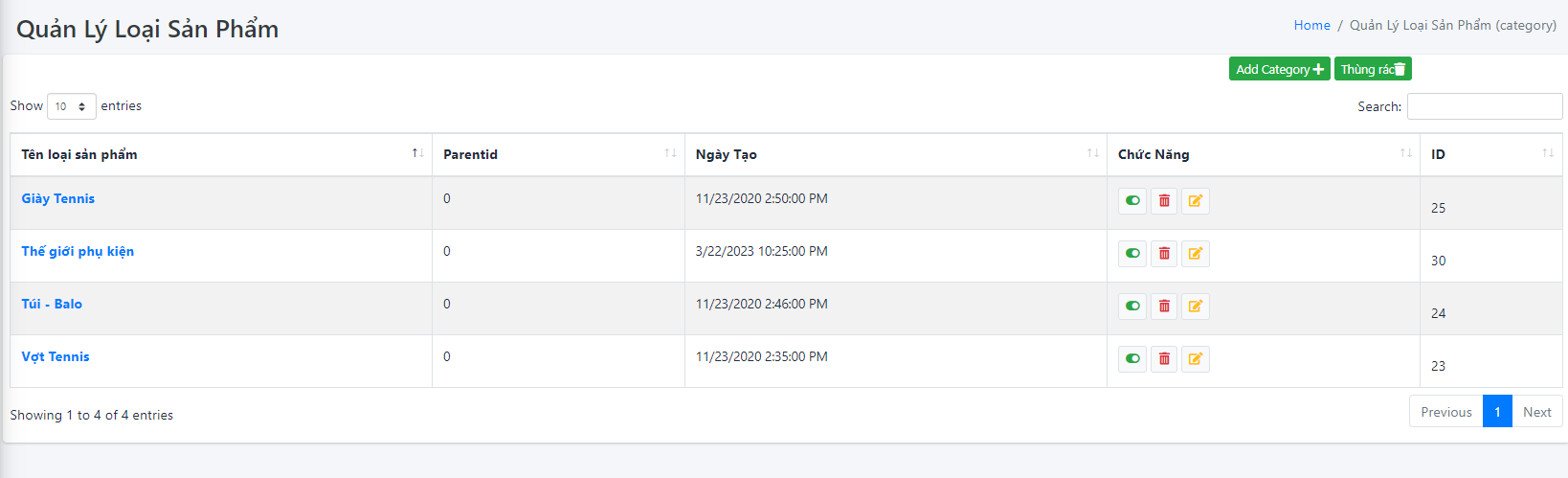
*Hình 4. 10. Giao diện đăng nhập trang quản lý.*

### 4.2.2. Giao diện trang chủ quản lý.



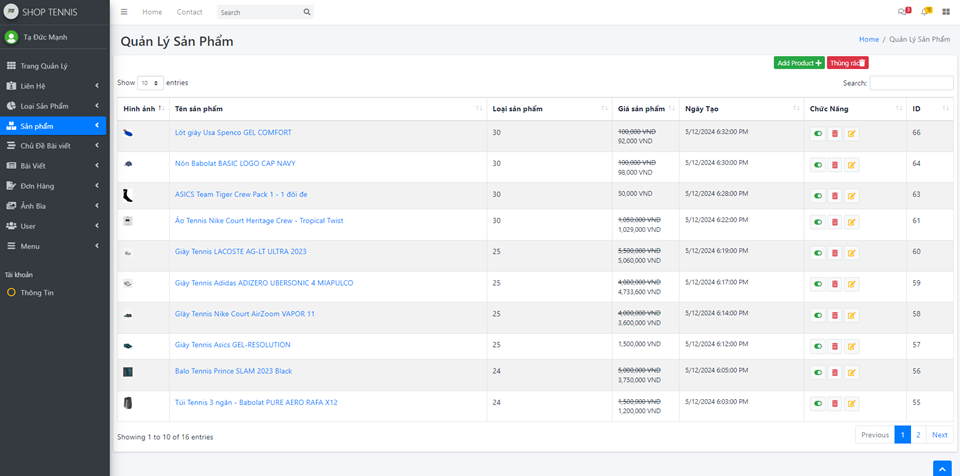
*Hình 4. 11. Giao diện trang chủ quản lý.*

### 4.2.3. Giao diện trang quản lý loại sản phẩm.



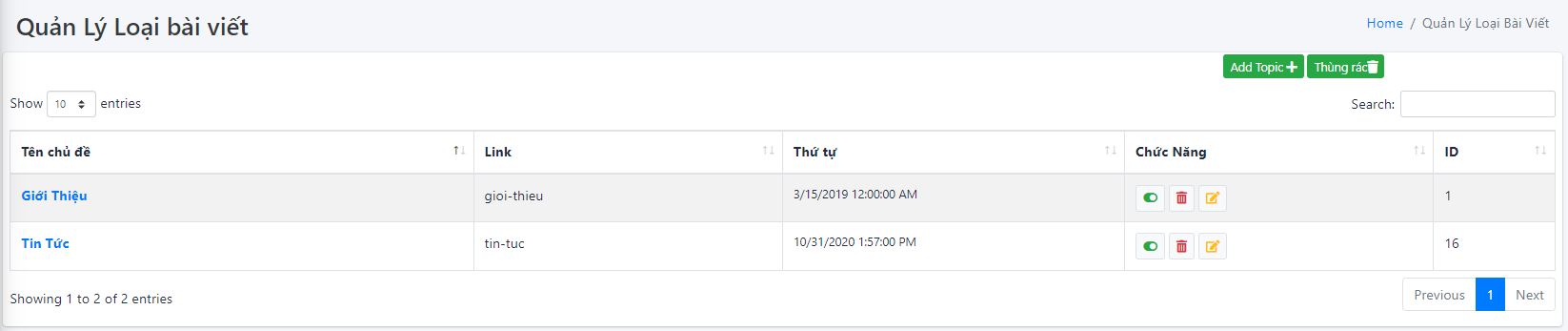
*Hình 4. 12. Giao diện trang quản lý loại sản phẩm.*

### 4.2.4. Giao diện trang quản lý sản phẩm.



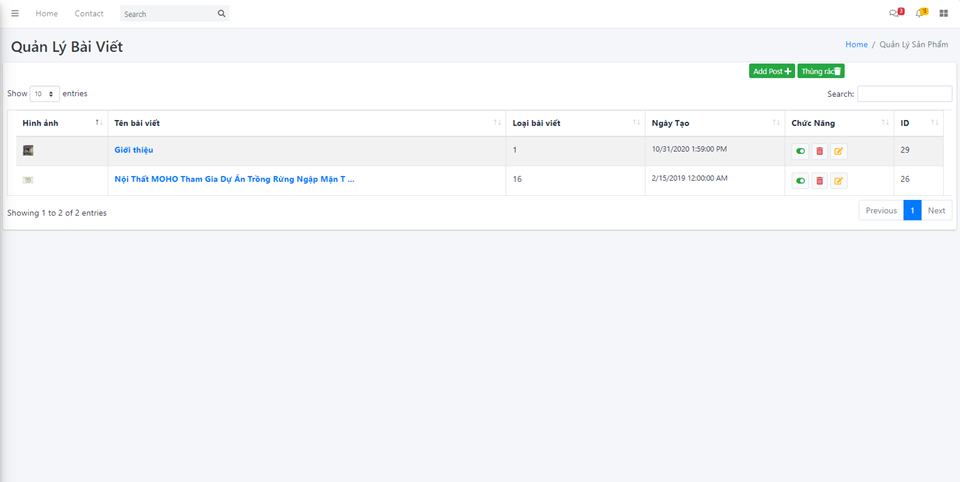
*Hình 4. 13. Giao diên trang quản lý sản phẩm.*

### 4.2.5. Giao diện trang quản lý chủ đề bài viết.



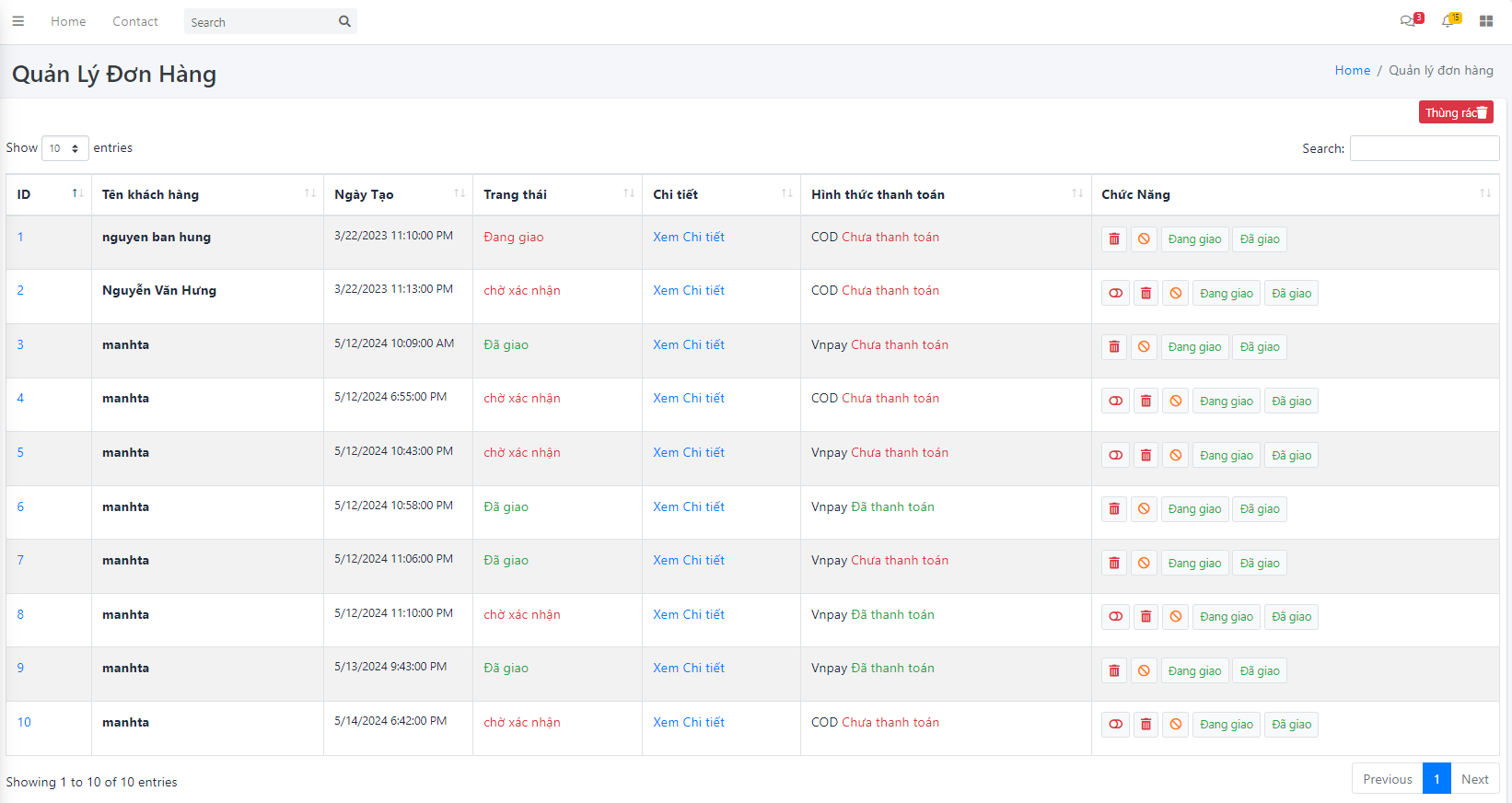
*Hình 4. 15. Giao diện trang quản lý chủ đề bài viết.*

### 4.2.6. Giao diện trang quản lý bài viết.



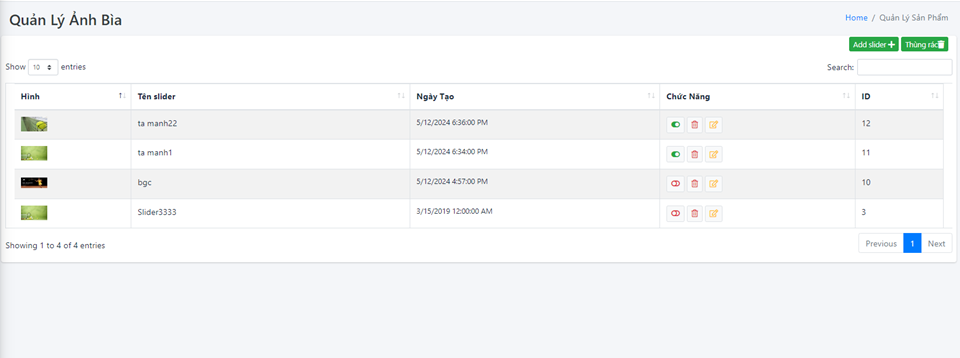
*Hình 4. 16. Giao diện trang quản lý bài viết.*

### 4.2.7. Giao diện trang quản lý đơn hàng.



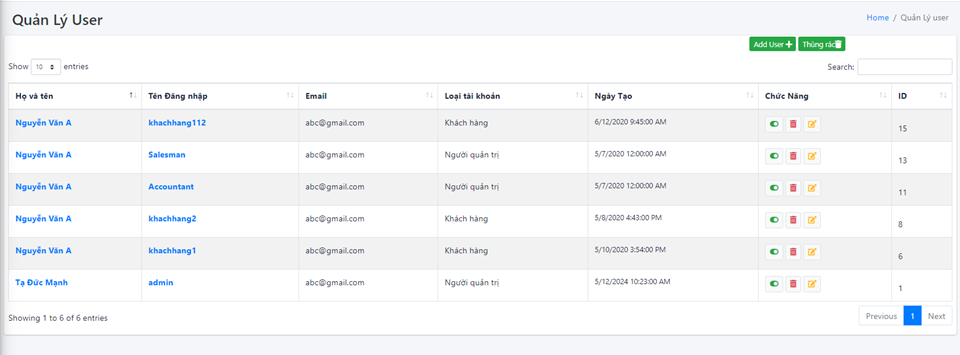
*Hình 4. 17. Giao diện trang quản lý đơn hàng.*

### 4.2.8. Giao diện trang quản lý ảnh bìa.



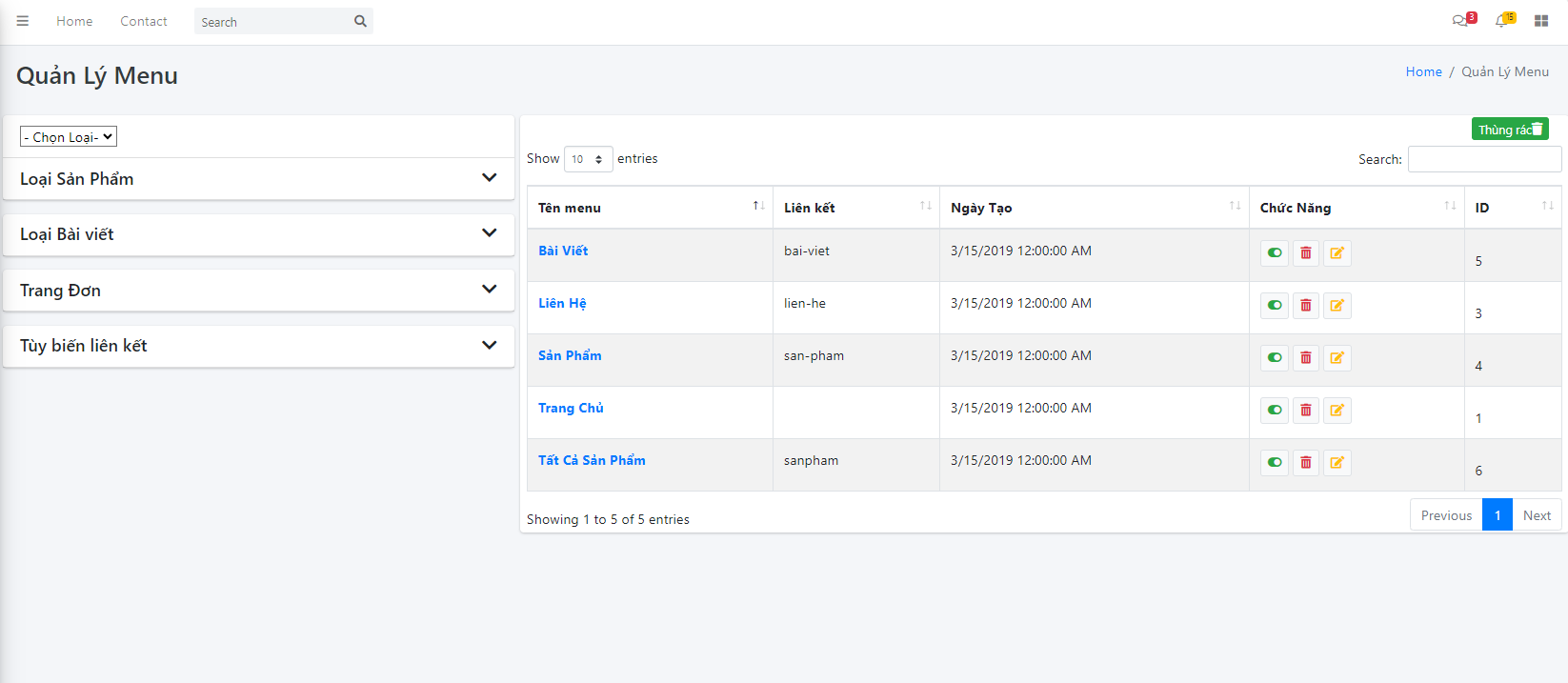
*Hình 4. 18. Giao diện trang quản lý ảnh bìa.*

### 4.2.9. Giao diện trang quản lý tài khoản người dùng.



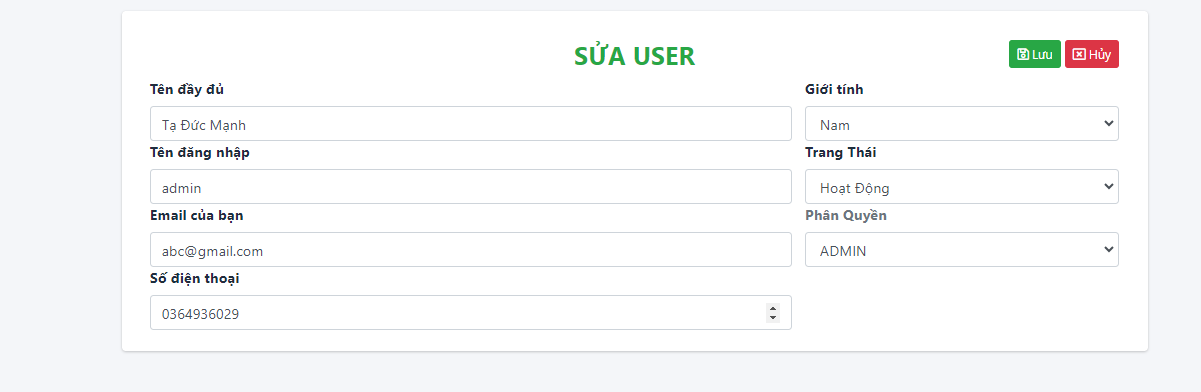
*Hình 4. 19. Giao diện trang quản lý tài khoản người dùng.*

### 4.2.10. Giao diện trang quản lý menu.



*Hình 4. 20. Giao diện trang quản lý menu.*

### 4.2.11. Giao diện trang thông tin tài khoản.



*Hình 4. 21. Giao diện trang thông tin tài khoản.*

# KẾT LUẬN

Trong thời gian thực hiện đề tài, với sự cố gắng nỗ lực hết mình em đã hoàn thành đề tài đúng thời gian quy định.

***Kết quả đạt được:***

* Xây dựng thành công Website bán đồng hồ, đáp ứng được yêu cầu của người sử dụng.
* Tìm hiểu tương đối kỹ về Website
* Tìm hiểu tương đối căn bản và đầy đủ về ASP.Net, CSS, JavaScript, HTML....
* Phân tích thiết kế hệ thống tương đối đầy đủ.
* Giao diện Website được thiết kế đơn giản, thân thiện và dễ sử dụng.

***Hạn chế:***

* Việc biểu diễn các thông tin trên Website chưa được linh hoạt.
* Các thao tác quản lý còn chậm, chưa nhanh so với winform.

# TÀI LIỆU THAM KHẢO

[1]. Dương Anh Đức, Giáo trình phân tích và thiết kế hướng đối tượng bằng UML.

[2]. Huỳnh Anh Đức, Kỹ thuật và ứng dụng UML, NXB lao động.

[3].Nguyễn Tiến – Ngô Quốc Việt, Kỹ thuật và ứng dụng bằng UML, NXB thống kê.

[4]. Phạm Hữu Khang ,C# 2005: Lập trình Windowns Form ( Tập 2).

[5]. Phạm Hữu Khang, C# 2005: Lập trình hướng đối tượng ( Tập 3).

[6]. Phạm Hữu Khang, C# 2005: Lập trình cơ sở dữ liệu( Tập 4).

[7]. <http://hiepsiit.com/>