-----🙞🙜🕮🙞🙜-----

**PHẠM TIẾN MẠNH**

TÊN ĐỀ TÀI

**Xây dựng website bán quần áo sử dụng ASP.NET**

**ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**Hà Nội, Năm 2021**

**ĐỀ TÀI:**

**Xây dựng website bán quần áo sử dụng ASP.NET**

**Hà Nội, Năm 2021**

**CÁN BỘ HƯỚNG DẪN**

**GV. THS. Nguyễn Thị Thanh**

Bộ môn: Công nghệ Phần mềm

Số điện thoại: 0982148611

Email: thanhk50@gmail.com

**CÁN BỘ PHẢN BIỆN**

**NCS. ThS. Hoàng Anh Đức**

Bộ môn: Công nghệ Phần mềm

Email: hoanganhduc@humg.edu.vn

**SINH VIÊN THỰC HIỆN**

**Phạm Tiến Mạnh**

Mã sinh viên: 1521050044

Số điện thoại: 0368841616

Email:manhm111297@gmail.com

Lớp: Công nghệ Phần mềm K60

**ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC**

**Ngành Công nghệ thông tin**

**Chuyên ngành Công nghệ phần mềm**

**Mã số: 52480201**

PHIẾU GIAO NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

**1. Thông tin chung**

|  |  |
| --- | --- |
| Tên đề tài | Xây dựng website bán quần áo sử dụng ASP.Net |
| Họ và tên sinh viên: | Phạm Tiến Mạnh |
| Điện thoại liên lạc: | 0360041616 |
| Email: | Manhm111297@gmail.com |
| Lớp: | Công Nghệ Phần Mềm K60 |
| Hệ đào tạo: | Đại học chính quy |
| Đồ án tốt nghiệp được thực hiện tại: | Hà Nội |
| Thời gian làm ĐATN: | 2021 |

**2. Mục tiêu của ĐATN**

Nghiên cứu tổng quan về mô hình kỹ thuật công nghệ ASP.NET , ứng dụng để xây dựng Website bán quần áo.

**3. Các nhiệm vụ cụ thể của ĐATN**

- Nghiên cứu các kỹ thuật xây dựng ASP.NET

- Ứng dụng xây dựng Website bán quần áo online.

**4. Lời cam đoan của sinh viên:**

Tôi – Phạm Tiến Mạnh – cam kết ĐATN là công trình nghiên cứu của bản thân tôi dưới sự hướng dẫn của ThS. Nguyễn Thị Thanh

Các kết quả nêu trong ĐATN là trung thực, không phải là sao chép toàn văn của bất kỳ công trình nào khác.

|  |  |
| --- | --- |
|  | *Hà Nội, ngày 1 tháng 1 năm 2021*  Tác giả ĐATN  Phạm Tiến Mạnh |

**5. Xác nhận của giáo viên hướng dẫn về mức độ hoàn thành của ĐATN và cho phép bảo vệ:**

|  |  |
| --- | --- |
|  | *Hà Nội, ngày 4 tháng 3 năm2021*  Cán bộ hướng dẫn  ThS. Nguyễn Thị Thanh |

# TÓM TẮT NỘI DUNG ĐỒ ÁN TỐT NGHIỆP

Việc kinh doanh hầu như ai cũng muốn làm, nhưng không phải cũng đủ khả năng chi trả các chi phí cho việc mở một cửa hiệu để buôn bán, hơn nữa còn phải chi trả các chi phí cho việc quảng cáo, phát tờ rơi… . Hiện nay internet đã trở lên phổ biến mọi người đều có thể truy cập với các smartphone, vậy sao chúng ta không tận dụng điều đó như một hình thức kinh doanh? Vì vậy em chọn đề tài “xây dựng website bán quần áo” trên nền tảng ASP.Net cho đồ án tốt nghiệp của mình. Hi vọng từ những gì em đã được học tại trường Đại học Mỏ - Địa chất, để xây dựng một trang web giúp ích cho những bạn có nhu cầu buôn bán.

# LỜI CẢM ƠN

Trong thời gian làm đồ án tốt nghiệp, em đã nhận được nhiều sự giúp đỡ, đóng góp ý kiến và chỉ bảo nhiệt tình của thầy hướng dẫn và các thầy cô trong khoa - bộ môn Công nghệ phần mềm.

Em xin gửi lời cảm ơn chân thành đến ThS.Nguyễn Thị Thanh, giảng viên Bộ môn Công nghệ phần mềm - trường ĐH Mỏ - Địa Chất người đã tận tình hướng dẫn, chỉ bảo em trong suốt quá trình làm khoá luận.

Em cũng xin chân thành cảm ơn các thầy cô giáo trong Bộ môn Công nghệ phần mềm Trường Đại học Mỏ - Địa Chất đã dạy dỗ cho em kiến thức các môn chuyên ngành, giúp em có được cơ sở lý thuyết vững vàng và có nhiều góp ý qua những lần bảo vệ tiến độ để đồ án của em được tốt hơn.

Em xin chân thành cảm ơn !

# MỤC LỤC

[PHIẾU GIAO NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP I](#_Toc47820363)

[TÓM TẮT NỘI DUNG ĐỒ ÁN TỐT NGHIỆP III](#_Toc47820364)

[LỜI CẢM ƠN IV](#_Toc47820365)

[MỤC LỤC V](#_Toc47820366)

[DANH MỤC CÁC HÌNH VẼ VIII](#_Toc47820367)

[DANH MỤC CÁC BẢNG BIỂU IX](#_Toc47820368)

[DANH MỤC CÁC TỪ VIẾT TẮT X](#_Toc47820369)

[MỞ ĐẦU 1](#_Toc47820370)

[CHƯƠNG 1 : TỔNG QUAN VỀ ASP.NET 2](#_Toc47820371)

[1.1 Tổng quan về ASP.NET MVC 2](#_Toc47820372)

[1.2 Những ưu điểm của ASP NET 3](#_Toc47820373)

[1.3 Quá trình xử lý tệp tin ASPX 4](#_Toc47820374)

[CHƯƠNG 2 : PHÂN TÍCH THIẾT KẾ HỆ THỐNG VÀ XÂY DỰNG CƠ SỞ DỮ LIỆU 5](#_Toc47820375)

[2.1 Biểu đồ use case ( Tổng quát ) 5](#_Toc47820376)

[2.2 Use case quản lý đơn hàng 6](#_Toc47820377)

[2.3 Use case quản lý danh mục 7](#_Toc47820378)

[2.4 Use case quản lý sản phẩm 8](#_Toc47820379)

[2.5 Use case quản lý màu 9](#_Toc47820380)

[2.6 Use case quản lý size 10](#_Toc47820381)

[2.7 Mô tả chi tiết các use case 11](#_Toc47820382)

[2.7.1 Đăng nhập 11](#_Toc47820383)

[2.7.2 Quản lý sản phẩm 11](#_Toc47820384)

[2.7.3 Quản lý hóa đơn 12](#_Toc47820385)

[2.7.4 Quản lý thể loại 12](#_Toc47820386)

[2.7.5 Đặt hàng 13](#_Toc47820387)

[2.7.6 Quản lý giỏ hàng 13](#_Toc47820388)

[2.7.7 Tìm kiếm sản phẩm 14](#_Toc47820389)

[2.7.8 Comment 14](#_Toc47820390)

[2.8 Biểu đồ trình tự (đăng nhập) 16](#_Toc47820391)

[2.9 Biểu đồ trình tự (tìm kiếm) 17](#_Toc47820392)

[2.10 Biểu đồ trình tự (quản lý sản phẩm) 18](#_Toc47820393)

[2.11 Biểu đồ trình tự (đặt hàng) 22](#_Toc47820394)

[CHƯƠNG 3 : XÂY DỰNG CƠ SỞ DỮ LIỆU 25](#_Toc47820395)

[3.1 Xây dựng các Relationship 25](#_Toc47820396)

[3.1.1 Các Relationship 25](#_Toc47820397)

[3.1.2 Biểu đồ Diagram 26](#_Toc47820398)

[3.2 Các table 27](#_Toc47820399)

[3.2.1 dbo.Account 27](#_Toc47820400)

[3.2.2 dbo.Category 27](#_Toc47820401)

[3.2.3 dbo.Color 28](#_Toc47820402)

[3.2.4 dbo.DiscountCode 28](#_Toc47820403)

[3.2.5 dbo.Contact 29](#_Toc47820404)

[3.2.6 dbo.DiscountCode 30](#_Toc47820405)

[3.2.7 dbo.Employee 31](#_Toc47820406)

[3.2.8 dbo.Order 32](#_Toc47820407)

[3.2.9 dbo.OrderDetail 33](#_Toc47820408)

[3.2.10 dbo.OrderStatus 34](#_Toc47820409)

[3.2.11 dbo.PaymentID 34](#_Toc47820410)

[3.2.12 dbo.Product 35](#_Toc47820411)

[3.2.13 dbo.ProductSize 36](#_Toc47820412)

[CHƯƠNG 4 : XÂY DỰNG GIAO DIỆN VÀ CHẠY THỬ NGHIỆM 37](#_Toc47820413)

[4.1 Giới thiệu thư viện Bootstrap 37](#_Toc47820414)

[4.1.1 Định nghĩa 37](#_Toc47820415)

[4.1.2 Tại sao nên sử dụng Bootstrap 37](#_Toc47820416)

[4.2 Một số giao diện đã xây dựng 39](#_Toc47820417)

[4.3 Thiết kế code-behind 45](#_Toc47820418)

[4.3.1 Giới thiệu mô hình 3 lớp 45](#_Toc47820419)

[4.3.2 Định nghĩa 45](#_Toc47820420)

[4.3.3 Ưu điểm 45](#_Toc47820421)

[4.4 Xây dựng Code-behind 45](#_Toc47820422)

[HUMG-Clothes.DTO 45](#_Toc47820423)

[4.4.1 HUMG-Clothes.DAL 46](#_Toc47820424)

[4.4.2 HUMG-Clothes.BLL 46](#_Toc47820425)

[4.4.3 HUMG-Clothes.GUI 47](#_Toc47820426)

[TÀI LIỆU THAM KHẢO 48](#_Toc47820427)

# DANH MỤC CÁC hình vẽ

[Hình 1.1 Cơ chế ASPX 4](#_Toc47603552)

[Hình 1.2 Quá trình xử lý tập tin ASPX 5](#_Toc47603553)

[Hình 2.1 Biểu đồ use case tổng quát 6](#_Toc47603554)

[Hình 2.2 Biểu đồ use case quản lý đơn hàng 7](#_Toc47603555)

[Hình 2.3 Biểu đồ use case quản lý danh mục 8](#_Toc47603556)

[Hình 2.4 Biểu đồ use case quản lý sản phẩm 9](#_Toc47603557)

[Hình 2.5 Biểu đồ use case quản lý màu 10](#_Toc47603558)

[Hình 2.6 Biểu đồ use case quản lý size 11](#_Toc47603559)

[Hình 2.7 Biểu đồ trình tự (đăng nhập) 17](#_Toc47603560)

[Hình 2.8 Biểu đồ trình tự (tìm kiếm) 18](#_Toc47603561)

[Hình 2.9 Biểu đồ trình tự (quản lý sản phẩm) 22](#_Toc47603562)

[Hình 2.10 Biểu đồ trình tự (đặt hàng) 25](#_Toc47603563)

[Hình 2.11 Biểu đồ Diagram 27](#_Toc47603564)

[Hình 3.1 Thư viện Bootstrap 38](#_Toc47603565)

[Hình 3.2 Giao diện Login 39](#_Toc47603566)

[Hình 3.3 Giao diện quản lý đơn hang 40](#_Toc47603567)

[Hình 3.4 Giao diện quản lý sản phẩm 41](#_Toc47603568)

[Hình 3.5 Giao diện thêm mới sản phẩm 42](#_Toc47603569)

[Hình 3.6 Giao diện quản lý danh mục 43](#_Toc47603570)

[Hình 3.7 Giao diện chi tiết đơn hàng 44](#_Toc47603571)

# DANH MỤC CÁC BẢNG BIỂU

[Bảng 2.1 Bảng dữ liệu tài khoản 28](#_Toc47696997)

[Bảng 2.2 Bảng dữ liệu thể loại 28](#_Toc47696998)

[Bảng 2.3 Bảng dữ liệu màu sản phẩm 29](#_Toc47696999)

[Bảng 2.4 Bảng dữ liệu mật khẩu 29](#_Toc47697000)

[Bảng 2.5 Bảng dữ liệu thông tin liên lạc 30](#_Toc47697001)

[Bảng 2.6 Bảng dữ liệu khách hàng 31](#_Toc47697002)

[Bảng 2.7 Bảng dữ liệu mật khẩu 32](#_Toc47697003)

[Bảng 2.8 Bảng dữ liệu nhân viên 33](#_Toc47697004)

[Bảng 2.9 Bảng dữ liệu đặt hàng 34](#_Toc47697005)

[Bảng 2.10 Bảng dữ liệu chi tiết đơn hàng 35](#_Toc47697006)

[Bảng 2.11 Bảng dữ liệu trạng thái đơn hàng 36](#_Toc47697007)

[Bảng 2.12 Bảng dữ liệu thanh toán 36](#_Toc47697008)

[Bảng 2.13 Bảng dữ liệu sản phẩm 37](#_Toc47697009)

[Bảng 2.14 Bảng dữ liệu kích thước sản phẩm 38](#_Toc47697010)

# DANH MỤC CÁC TỪ VIẾT TẮT

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Từ viết tắt | Từ tiếng Anh | Từ và nghĩa tiếng Việt |
| 1 | CSDL | Database | Cơ sở dữ liệu |
| 2 | DB | Database | Cơ sở dữ liệu |
| 3 | DTO | Data Tranfer Object | Đối tượng truyền dữ liệu |
| 4 | BLL | Business Logic Layer | Lớp logic nghiệp vụ |
| 5 | DAL | Data Access Layer | Lớp truy cập dữ liệu |

# MỞ ĐẦU

1. Tổng quan tình hình nghiên cứu thuộc lĩnh vực của đề tài

Ngày nay, với sự phát triển của ngành thương mại điện tử. Các website thương mại điện tử, bán hàng trực tuyến ngày càng phổ biến. Với lợi thế tiện lợi và nhanh chóng, cùng với nhu cầu mua sắm của mọi người ngày càng nhiều cho thấy việc mua hàng online hiện nay rất phổ biến.

Cũng chính vì những yếu tố kể trên, hiện nay đã có khá nhiều website buôn bán quần áo thời trang lập lên và thực hiện buôn bán online rất tốt... Đồng thời các website này cũng có thể liên kết trực tiếp với các website thương mại điện tử để có thể tận dụng được tối đa sự hiệu quả của mình.

2. Mục tiêu đề tài

Xây dựng Website bán quần áo :

* Người dùng vào trang web của cửa hàng và chọn đồ
* Đồ được chọn sẽ trng giỏ hàng và khách hàng tiến hành thanh toán
* Quản lý các đơn hàng, loại hàng,…
* Báo cáo sự cố

3. Nội dung thực hiện

* Tìm hiểu về Asp.Net
* Tìm hiểu và nghiên cứu các website bán hàng quần áo hiện nay
* Phân tích và thiết kế hệ thống
* Cài đặt và thử nghiệm website

4. Bố cục đề tài

* Chương 1 : Tổng quan
* Chương 2 : Phân tích thiết kế hệ thống
* Chương 3 : Xây dựng cơ sở dữ liệu
* Chương 4 :Tổng kết

Kết luận và hướng phát triển

1. : TỔNG QUAN VỀ ASP.NET

## Tổng quan về ASP.NET MVC

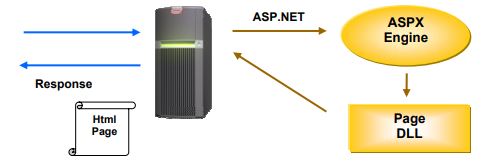
Từ khoảng cuối thập niên 90, ASP (Active Server Page) đã được nhiều lập trình viên lựa chọn để xây dựng và phát triển ứng dụng web động trên máy chủ sử dụng hệ điều hành Windows. ASP đã thể hiện được những ưu điểm của mình với mô hình lập trình thủ tục đơn giản, sử dụng hiệu quả các đối tượng COM: ADO (ActiveX Data Object) - xử lý dữ liệu, FSO (File System Object) - làm việc với hệ thống tập tin…, đồng thời, ASP cũng hỗ trợ nhiều ngôn ngữ: VBScript, JavaScript. Chính những ưu điểm đó, ASP đã được yêu thích trong một thời gian dài. Tuy nhiên, ASP vẫn còn tồn đọng một số khó khăn như Code ASP và HTML lẫn lộn, điều này làm cho quá trình viết code khó khăn, thể hiện và trình bày code không trong sáng, hạn chế khả năng sử dụng lại code. Bên cạnh đó, khi triển khai cài đặt, do không được biên dịch trước nên dễ bị mất source code.

Thêm vào đó, ASP không có hỗ trợ cache, không được biên dịch trước nên phần nào hạn chế về mặt tốc độ thực hiện. Quá trình xử lý Postback khó khăn, … Đầu năm 2002, Microsoft giới thiệu một kỹ thuật lập trình Web khá mới mẻ với tên gọi ban đầu là ASP+, tên chính thức sau này là ASP.Net. Với ASP.Net, không những không cần đòi hỏi bạn phải biết các tag HTML, thiết kế web, mà nó còn hỗ trợ mạnh lập trình hướng đối tượng trong quá trình xây dựng và phát triển ứng dụng Web. ASP.Net là kỹ thuật lập trình và phát triển ứng dụng web ở phía Server (Server-side) dựa trên nền tảng của Microsoft .Net Framework. Hầu hết, những người mới đến với lập trình web đều bắt đầu tìm hiểu những kỹ thuật ở phía Client (Client-side) như: HTML, Java Script, CSS (Cascading Style Sheets). Khi Web browser yêu cầu một trang web (trang web sử dụng kỹ thuật client-side), Web server tìm trang web mà Client yêu cầu, sau đó gởi về cho Client. Client nhận kết quả trả về từ Server và hiển thị lên màn hình. ASP.Net sử dụng kỹ thuật lập trình ở phía server thì hoàn toàn khác, mã lệnh ở phía server (ví dụ: mã lệnh trong trang ASP) sẽ được biên dịch và thi hành tại Web Server. Sau khi được Server đọc, biên dịch và thi hành, kết quả tự động được chuyển sang HTML/JavaScript/CSS và trả về cho Client. Tất cả các xử lý lệnh ASP.Net đều được thực hiện tại Server và do đó, gọi là kỹ thuật lập trình ở phía server.

## Những ưu điểm của ASP NET

ASP.Net cho phép bạn lựa chọn một trong các ngôn ngữ lập trình mà bạn yêu thích: Visual Basic.Net, J#, C#,…

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 là một bước nhảy vọt đáng kể so với kỹ thuật thông dịch của ASP.



Hình 1.1 Cơ chế ASPX

Response ASP.NET ASPX Engine Page DLL Html Page Tài liệu hướng dẫn giảng dạy Học phần 3 - Lập trình ứng dụng web với ASP.NET Trang 17/174 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.

ASP.Net sử dụng phong cách lập trình mới: Code behide. Tách code riêng, giao diện riêng Î Dễ đọc, dễ quản lý và bảo trì.

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

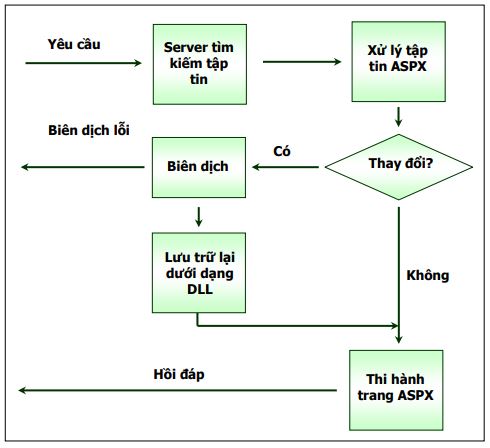
1. Không cần lock, không cần đăng ký DLL
2. Cho phép nhiều hình thức cấu hình ứng dụng

Hỗ trợ quản lý ứng dụng ở mức toàn cục

1. Global.aspx có nhiều sự kiện hơn
2. Quản lý session trên nhiều Server, không cần Cookies

## Quá trình xử lý tệp tin ASPX

Khi Web server nhận được yêu cầu từ phía client, nó sẽ tìm kiếm tập tin được yêu cầu thông qua chuỗi URL được gởi về, sau đó, tiến hành xử lý theo sơ đồ sau:



Hình 1.2 Quá trình xử lý tập tin ASPX

## Giới thiệu thư viện Bootstrap

### Định nghĩa

Bootstrap là framework HTML, CSS và JavaScript phổ biến nhất để phát triển các trang web có tính phản hồi, tích hợp di động. Nếu bạn mong muốn sở hữu một Website Responsive, tương thích với mọi trình duyệt và thiết bị di động, chắc chắn sẽ cần dùng đến công nghệ này.



Hình 3.1 Thư viện Bootstrap

Bootstrap bao gồm các mã CSS + HTML cơ bản cho typography, forms, buttons, tables, grids, navigation, và nhiều thành phần khác trong một website. Điều này giúp cho các designer tránh việc phải lặp đi lặp lại trong quá trình tạo ra các class CSS và các đoạn mã HTML giống nhau trong khi thiết kế web. Style của các phần tử HTML trong Bootstrap khá đơn giản và thanh lịch. Ví dụ như phần đổ bóng trong input, highlight của bảng biểu, các mã CSS hiển thị cảnh báo, tabs, phân trang…

### Tại sao nên sử dụng Bootstrap

Lý do đầu tiên khiến Bootstrap dần trở nên phổ biến và xứng đáng được lựa chon đó là gần đây, các nhà lập trình đã bổ sung thêm tính năng Customize (Tùy chỉnh). Chức năng này giúp cho các nhà thiết kế linh hoạt hơn trong việc lựa chọn những thuộc tính, phần tử phù hợp với dự án họ đang theo đuổi. Chức năng này cũng cho phép bạn không cần phải tải toàn bộ mã nguồn về máy.

Bootstrap không đòi hỏi kiến thức quá phức tạp. Chỉ cần biết sơ qua HTML, CSS, Javascript, Jquery là bạn có thể sử dụng nó để tạo nên một website chuyên nghiệp, website theo yêu cầu đúng như mong muốn.

Boostrap được viết bởi những bộ não công nghệ tài năng trên khắp thế giới. Sự tương thích của trình duyệt với thiết bị đã được kiểm tra nhiều lần trước khi đưa vào sử dụng, nên khi lựa chọn Bootstrap, bạn có thể tin rằng mình sẽ tạo nên những sản phẩm hoàn hảo.

Do sử dụng Grid System nên Bootstrap mặc định hỗ trợ Responsive. Bootstrap được viết theo xu hướng Mobile First tức là ưu tiên cho việc tương thích với các giao diện trên thiết bị di động. Sử dụng công nghệ này cho website của bạn sẽ giúp website trở nên tương thích với tất cả kích thước màn hình – một điều cực kì quan trọng và cần thiết trong trải nghiệm người dùng.

Bootstrap hoạt động theo xu hướng mã nguồn mở nên bạn có thể vào mã nguồn của nó để thay đổi và chỉnh sửa tùy ý.

## Giới thiệu mô hình 3 lớp

### Định nghĩa

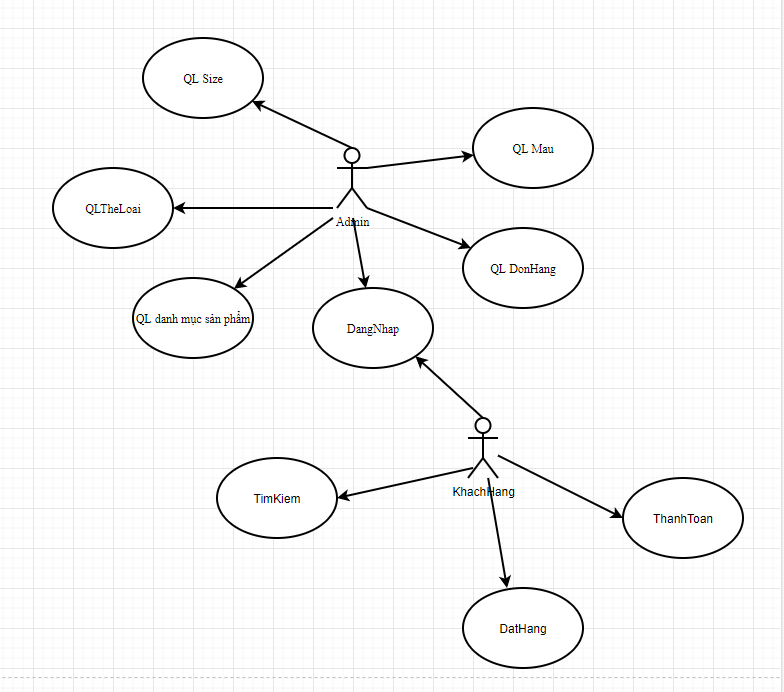
3-tiers là một kiến trúc kiểu client/server mà trong đó giao diện người dùng (UI-user interface), các quy tắc xử lý (BR-business rule hay BL-business logic), và việc lưu trữ dữ liệu được phát triển như những module độc lập, và hầu hết là được duy trì trên các nền tảng độc lập, và mô hình 3 tầng (3-tiers) được coi là một kiến trúc phần mềm và là một mẫu thiết kế.

### Ưu điểm

1. Việc phân chia thành từng lớp giúp cho code được tường minh hơn. Nhờ vào việc chia ra từng lớp đảm nhận các chức năng khác nhau và riêng biệt như giao diện, xử lý, truy vấn thay vì để tất cả lại một chỗ. Nhằm giảm sự kết dính.
2. Dễ bảo trì khi được phân chia, thì một thành phần của hệ thống sẽ dễ thay đổi. Việc thay đổi này có thể được cô lập trong 1 lớp, hoặc ảnh hưởng đến lớp gần nhất mà không ảnh hưởng đến cả chương trình.
3. Dễ phát triển, tái sử dụng: khi chúng ta muốn thêm một chức năng nào đó thì việc lập trình theo một mô hình sẽ dễ dàng hơn vì chúng ta đã có chuẩn để tuân theo. Và việc sử dụng lại  khi có sự thay đổi giữa hai môi trường ( Winform sang Webfrom ) thì chỉ việc thay đổi lại lớp GUI.
4. Dễ bàn giao. Nếu mọi người đều theo một quy chuẩn đã được định sẵn, thì công việc bàn giao, tương tác với nhau sẽ dễ dàng hơn và tiết kiệm được nhiều thời gian.
5. Dễ phân phối khối lượng công việc. Mỗi một nhóm, một bộ phận sẽ nhận một nhiệm vụ trong mô hình 3 lớp. Việc phân chia rõ ràng như thế sẽ giúp các lập trình viên kiểm soát được khối lượng công việc của mình.

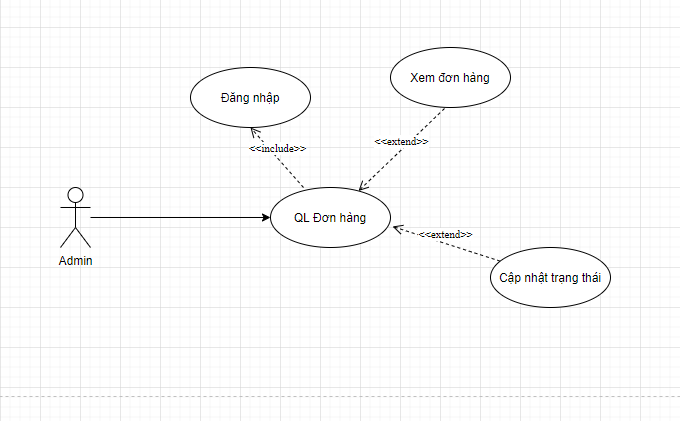
# : PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## Biểu đồ use case ( Tổng quát )



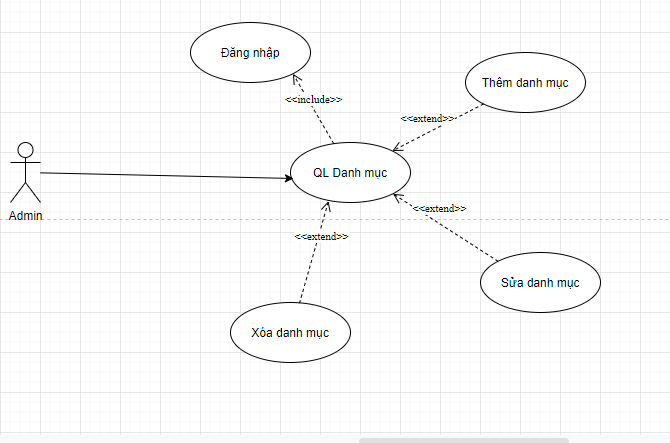
Hình 2.1 Biểu đồ use case tổng quát

## Use case quản lý đơn hàng



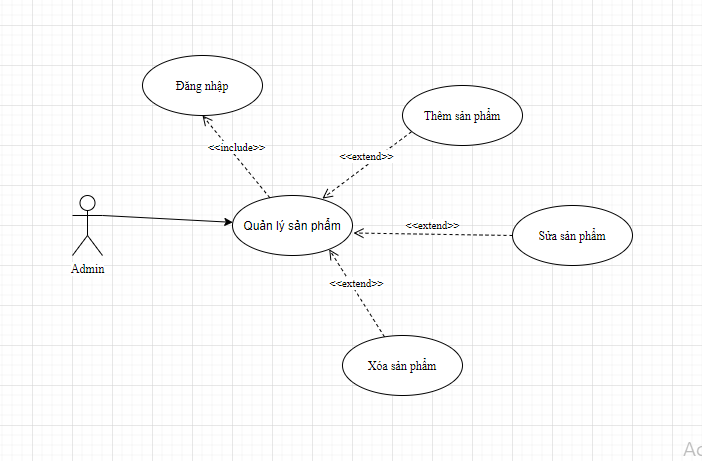
Hình 2.2 Biểu đồ use case quản lý đơn hàng

## Use case quản lý danh mục



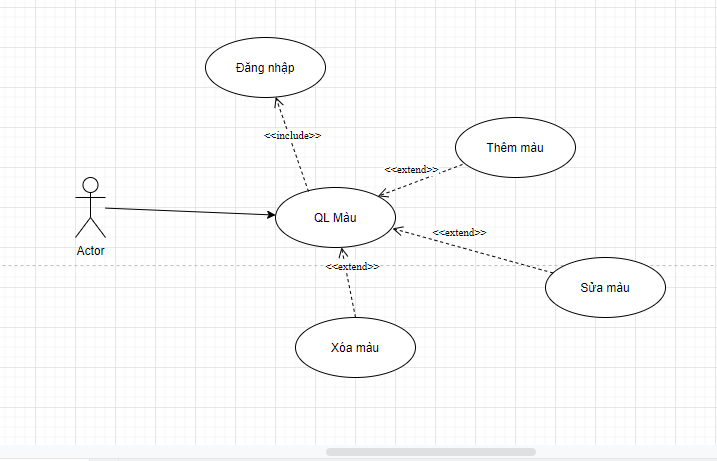
Hình 2.3 Biểu đồ use case quản lý danh mục

## Use case quản lý sản phẩm



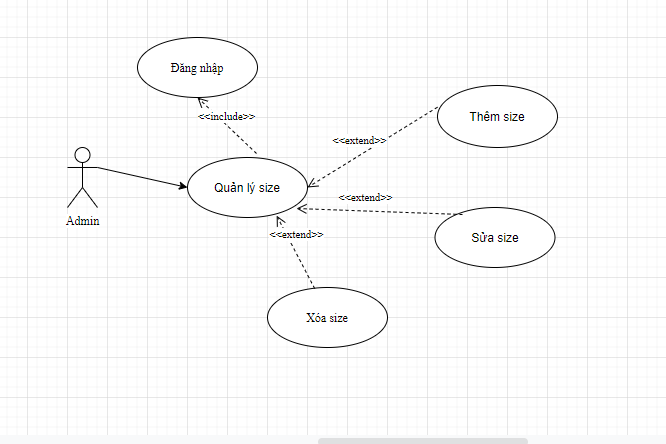
Hình 2.4 Biểu đồ use case quản lý sản phẩm

## Use case quản lý màu



Hình 2.5 Biểu đồ use case quản lý màu

## Use case quản lý size



Hình 2.6 Biểu đồ use case quản lý size

## Mô tả chi tiết các use case

### Đăng nhập

**Mô tả vắn tắt:**

Use case này cho phép thành viên (admin) đăng nhập vào hệ thống.

**Luồng sự kiện:**

* **Luồng cơ bản:**

1. Use case này bắt đầu thành viên chọn chức năng đăng nhập. Hệ thống sẽ hiển thị màn hình yêu cầu nhập email và mật khẩu.
2. Thành viên nhập email và mật khẩu, nhấn nút đăng nhập. Hệ thống kiểm tra email, mật khẩu, quyền trong bảng Users để hiển thị from tương ứng cho thành viên đăng nhập. Use case kết thúc.

* **Luồng rẽ nhánh:**

Tại luồng cơ bản 2 nếu người dùng nhập sai email và mật khẩu hệ thống hiển thị thông báo sai email hoặc mật khẩu và yêu cầu nhập lại. Thành viên tiếp tục nhập lại hoặc thoát. Use case kết thúc.

### Quản lý sản phẩm

**Mô tả vắn tắt:**

Use case này cho phép admin có thể thêm, sửa, xóa sản phẩm của hệ thống

**Luồng sự kiện:**

* **Luồng cơ bản:**

1. Use case này bắt đầu khi người dùng (admin) nhấn danh sách sản phẩm trong mục sản phẩm, hệ thống hiển thị danh sách các sản phẩm trong bảng San Pham lên màn hình.
2. **Thêm sản phẩm:** Khi người dùng nhấn nút thêm sản phẩm thì form thêm sản phẩm hiển thị, người dùng nhập thông tin sản phẩm (tên sản phẩm, hình ảnh, tên thể loại, loại, số lượng, giá, bảo hành, thông tin chung), các thông số sản phẩm và nhấn nút thêm sản phẩm, hệ thống kiểm tra thông tin và thêm vào bảng SanPham, cập nhật lại danh sách và hiển thị. Use case kết thúc.
3. **Sửa sản phẩm:** Khi người dùng nhấn nút sửa trên một dòng thông tin sản phẩm, form sửa sản phẩm hiển thị, người dùng sửa lại thông tin sản phẩm (số lượng, giá, bảo hành, thông tin chung,trạng thái), các thông số sản phẩm, hệ thống kiểm tra thông tin và cập nhật lại sản phẩm trong bảng SanPham, cập nhật lại danh sách hiển thị. Use case kết thúc.
4. **Xóa sản phẩm:** Khi người dùng nhấn nút xóa trên một dòng thông tin của sản phẩm, form xác nhận hiển thị, hệ thống cập nhật trạng thái sản phẩm trong bảng SanPham và hiển thị lại danh sách. Use case kết thúc.

* **Luồng rẽ nhánh:**

1. Tại luồng cơ bản 2 nếu thông tin người dùng nhập không hợp lệ hệ thống sẽ thông báo lỗi vài yêu cầu nhập lại, người dùng nhập lại tiếp túc hoặc nhấn thoát. Use case kết thúc.
2. Tại luồng cơ bản 3 nếu thông tin người dùng nhập không hợp lệ hệ thống sẽ thông báo lỗi vài yêu cầu nhập lại, người dùng nhập lại tiếp túc hoặc nhấn thoát. Use case kết thúc.
3. Tại luồng cơ bản 4 nếu trạng thái sản phẩm bằng 0 thì nút xóa sẽ không hiển thị.

### Quản lý hóa đơn

**Mô tả vắn tắt:**

Use case này cho phép người dùng (admin) sửa trạng thái hóa đơn.

**Luồng sự kiện:**

* **Luồng cơ bản:**

1. Use case này bắt đầu người dùng chọn mục danh sách hóa đơn. Hệ thống sẽ hiển thị danh sách đơn hàng trong bảng HoaDon.
2. **Thanh toán:** Người dùng nhấn nút thanh toán, hệ thống sẽ cập nhật trạng thái của hóa đơn đó thành 1 trong bảng HoaDon và hiện thị lại danh sách hóa đơn.

* **Luồng rẽ nhánh:**

Tại luồng cơ bản 2 nếu người dùng nhấn nút thoát Use case kết thúc.

### Quản lý thể loại

**Mô tả vắn tắt:**

Use case này cho phép người dùng (admin) có thể thêm, xóa thể loại.

**Luồng sự kiện:**

* **Luồng cơ bản:**

1. Use case này bắt đầu khi người dùng nhấn nút thể loại thì form thể loại sẽ hiển thị các thể loại trong bảng TheLoai.
2. **Thêm thể loại:** Người dùng nhấn nút thêm thể loại thì form thêm thêm thể loại hiển thị, người dùng nhập vào tên thể loại và nhấn nút thêm, hệ thống sẽ kiểm tra thông tin và thêm vào bảng TheLoai. Use case kết thúc.
3. **Xóa thể loại:** Người dùng nhấn nút xóa trên dòng thông tin thể loại. Trạng thái trong bảng TheLoai sẽ được cập nhật. Use case kết thúc.

* **Luồng rẽ nhánh:**

Tại luồng cơ bản 2 nếu thông tin người dùng nhập không hợp lệ thì hệ thống sẽ hiển thị thông báo lỗi và yêu cầu nhập lại, người dùng tiếp tục nhập lại hoặc nhấn thoát. Use case kết thúc.

### Đặt hàng

**Mô tả vắn tắt:**

Use case này cho phép khách hàng (người mua hàng) thanh toán sản phẩm trong giỏ hàng.

**Luồng sự kiện:**

* **Luồng cơ bản:**

1. Use case này bắt đầu khách hàng nhấn vào nút đặt hàng trong giỏ hàng. Hệ thống sẽ hiển thị form đặt hàng.
2. Khách hàng nhập thông tin cá nhân (tên, số điện thoại, email, địa chỉ, ghi chú), chọn hình thức thanh toán và nhấn vào nút xác nhận hệ thống kiểm tra thông tin và thêm vào bảng HoaDon và HoaDonChiTiet. Use case kết thúc.

* **Luồng rẽ nhánh:**

Tại luồng cơ bản 2 nếu thông tin khách hàng nhập vào không hợp lệ hệ thống sẽ thông báo lỗi và yêu cầu nhập lại, khách hàng tiếp tục nhập hoặc thoát. Use case kết thúc.

### Quản lý giỏ hàng

**Mô tả vắn tắt:**

Use case này cho phép khách hàng (người mua hàng) xem, sửa số lượng sản phẩm, xóa sản phẩm khỏi giỏ hàng.

**Luồng sự kiện:**

* **Luồng cơ bản:**

1. **Xem giỏ hàng**: Use case này bắt đầu khách hàng chọn giỏ hàng hoặc nhấn vào nút đặt hàng. Hệ thống sẽ truy xuất vào bảng “SanPham” và hiển thị lên giỏ hàng.
2. **Sửa số lượng sản phẩm trong giỏ hàng:** Khách hàng lựa chọn số lượng sản phẩm theo ý, hệ thống sẽ tự động cập nhật lại số lượng lấy từ bảng “SanPham” và giá để hiển thị. Use case kết thúc.
3. **Xóa sản phẩm trong giỏ hàng:**

Khách hàng click vào nút “ x”trong giỏ hàng, hệ thống sẽ xóa sản phẩm ra khỏi giỏ hàng đồng thời cập nhật lại bảng “SanPham” và hiển thị lại. Use case kết thúc.

* **Luồng rẽ nhánh:**

Tại luồng cơ bản 1 nếu giỏ hàng không có sản phẩm thì hệ thống sẽ hiển thì thông báo giỏ hàng trống.Nếu khách hàng nhấn nút thoát Use case kết thúc..

### Tìm kiếm sản phẩm

**Mô tả vắn tắt:**

Use case này cho khách hàng tìm kiếm nhanh chóng sản phẩm mình mong muốn.

**Luồng sự kiện:**

* **Luồng cơ bản:**

1. Use case này bắt đầu khi khách hàng kick con trỏ chuột vào ô nhập thông tin, khách hàng sẽ nhập tên sản phẩm. Khi khách hàng nhấn nút tìm kiếm thì hệ thống sẽ hiện thị danh sách sản phẩm từ bảng “SanPham” tương ứng với ô tìm kiếm.

* **Luồng rẽ nhánh:** Khi khách hàng nhấn nút thoát hoặc trang chủ thì Use case kết thúc.

### Comment

**Mô tả vắn tắt:**

Use case cho phép khách hàng bình luân về sản phẩm.

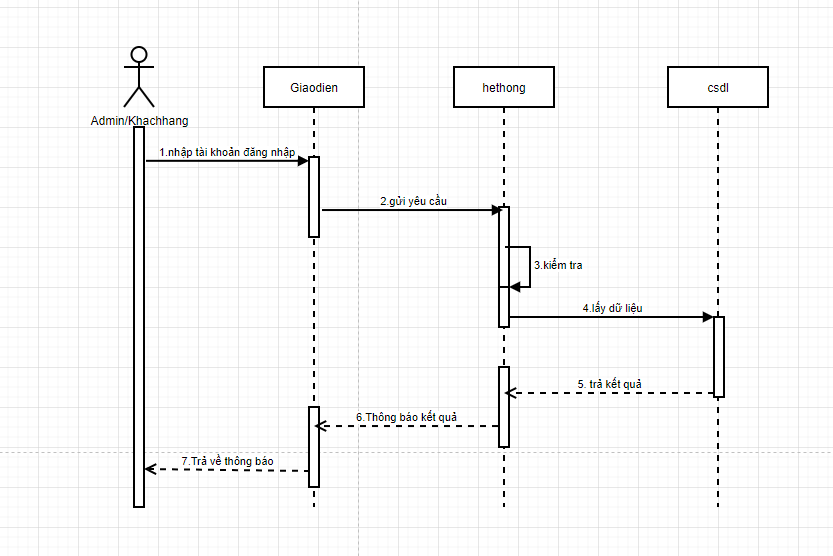
**Luồng sự kiện:**

* **Luồng cơ bản:**

Use case này bắt đầu khách hàng thành viên nhập thông tin ô bình luận và nhấn vào nút Comment, hệ thống sẽ hiện thị Comment của khách hàng lên màn hình.

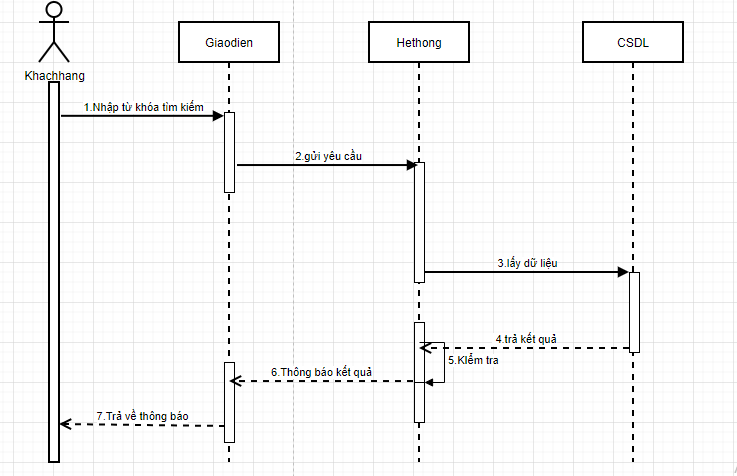
* **Luồng rẽ nhánh:** Nếu khách hàng nhấn nút trang chủ hoặc nút thoát thì Use case kết thúc.

## Biểu đồ trình tự (đăng nhập)



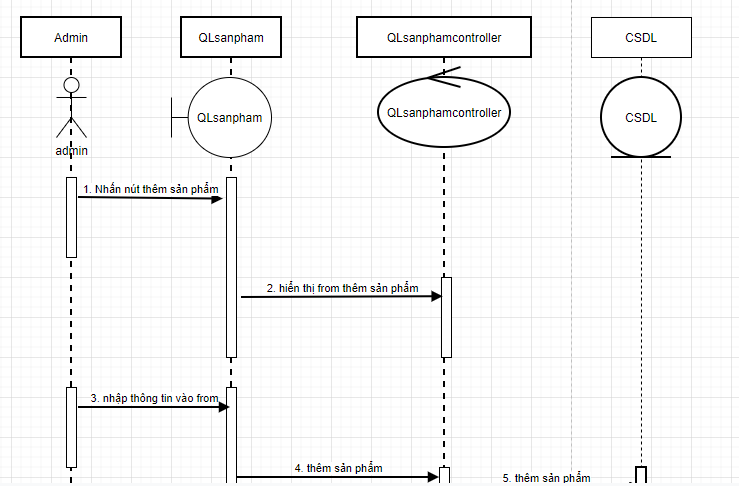
Hình 2.7 Biểu đồ trình tự (đăng nhập)

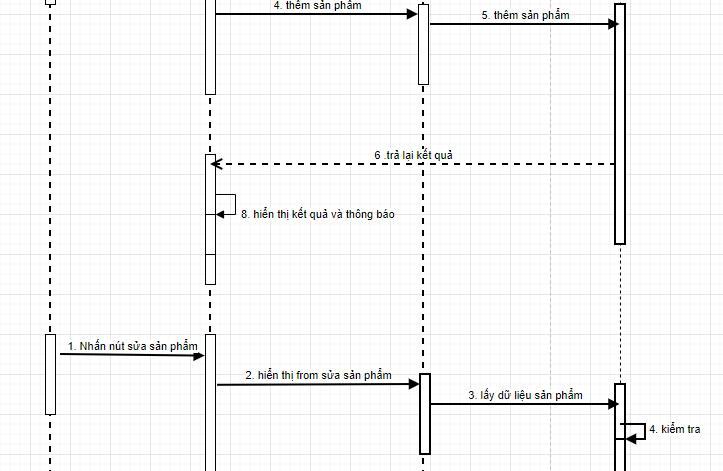
## Biểu đồ trình tự (tìm kiếm)

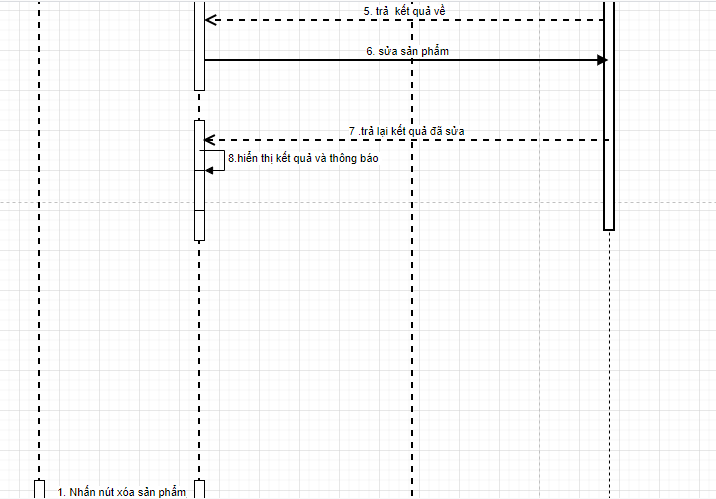


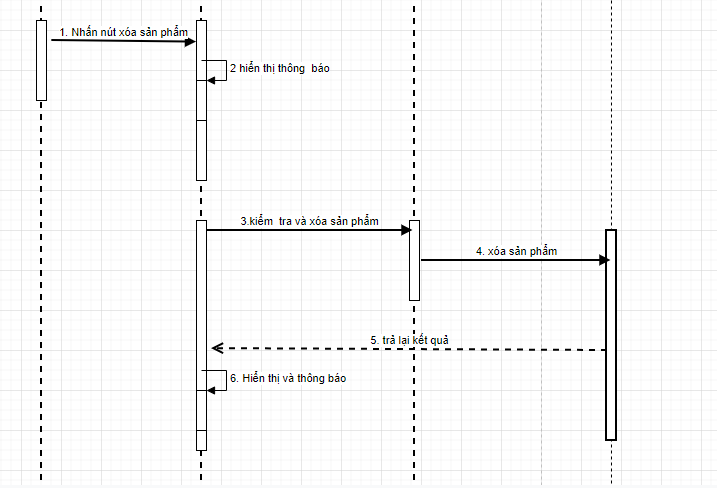
Hình 2.8 Biểu đồ trình tự (tìm kiếm)

## Biểu đồ trình tự (quản lý sản phẩm)



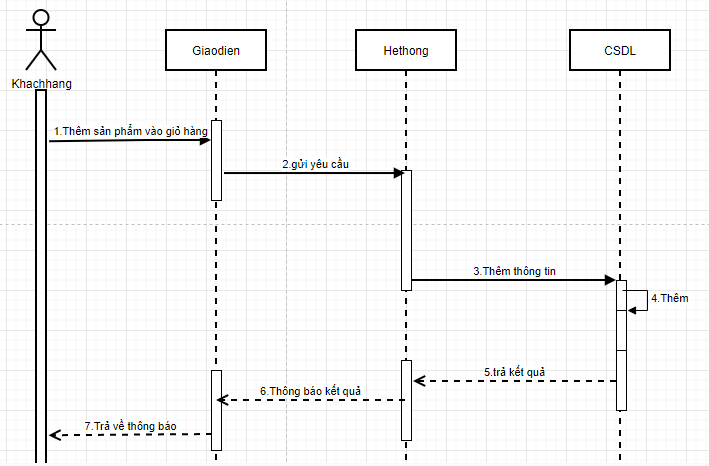


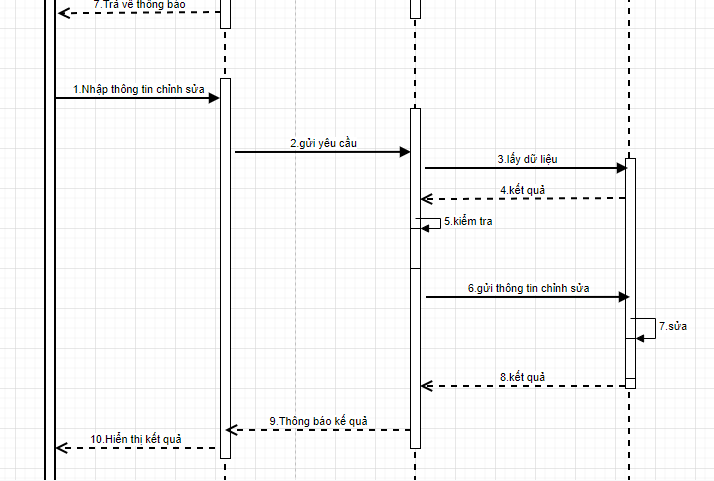


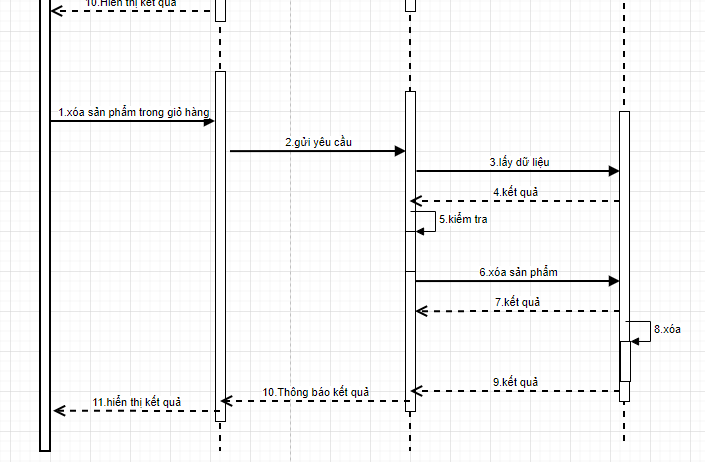


Hình 2.9 Biểu đồ trình tự (quản lý sản phẩm)

## Biểu đồ trình tự (đặt hàng)

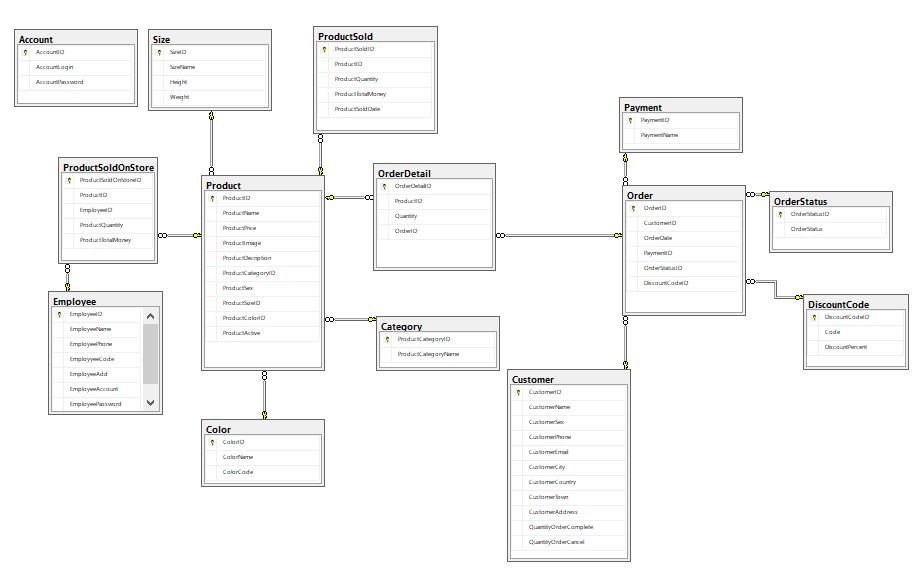






Hình 2.10 Biểu đồ trình tự (đặt hàng)

## Biểu đồ Diagram



Hình 2.11 Biểu đồ Diagram

# : XÂY DỰNG CƠ SỞ DỮ LIỆU

## Xây dựng các Relationship

### Các Relationship

FK\_Product\_Size

FK\_ProductSoldOnStore\_Product

FK\_ProductSoldOnStore\_Employee

FK\_ProductSold\_Product

FK\_OrderDetail\_Product

FK\_Product\_Category

FK\_Product\_Color

FK\_OrderDetail\_Order

FK\_Order\_Payment

FK\_Order\_Customer

FK\_Order\_OrderStatus

FK\_Order\_DiscountCode

## Các table

### dbo.Account

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên cột | Kiểu dữ liệu | Ràng buộc |
| 1 | AccountID | Int | Primary Key |
| 2 | AccountLogin | Nchar(20) | Not Null |
| 3 | AccountPassword | Nchar(200) | Not Null |

Bảng . Bảng dữ liệu tài khoản

### dbo.Category

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên cột | Kiểu dữ liệu | Ràng buộc |
| 1 | ProductCategoryID | Int | Primary Key |
| 2 | ProductCategoryName | Nchar(20) | Not Null |

Bảng . Bảng dữ liệu thể loại

### dbo.Color

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên cột | Kiểu dữ liệu | Ràng buộc |
| 1 | ColorID | Int | Primary Key |
| 2 | ColorName | Nchar(200) | Not Null |
| 3 | ColorCode | Nchar(10) | Not Null |

Bảng . Bảng dữ liệu màu sản phẩm

### dbo.DiscountCode

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên cột | Kiểu dữ liệu | Ràng buộc |
| 1 | DiscountCodeID | Int | Primary Key |
| 2 | Code | Nchar(10) | Not Null |
| 3 | DiscountPercent | int | Not Null |

Bảng . Bảng dữ liệu mật khẩu

### dbo.Contact

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên cột | Kiểu dữ liệu | Ràng buộc |
| 1 | ContactID | Int | Primary Key |
| 2 | ShopName | Nchar(400) | Not Null |
| 3 | Phone | Nchar(10) | Not Null |
| 4 | Phone1 | Nchar(10) |  |
| 5 | Address | Text |  |
| 6 | Address | Text |  |

Bảng . Bảng dữ liệu thông tin liên lạc

### dbo.DiscountCode

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên cột | Kiểu dữ liệu | Ràng buộc |
| 1 | DiscountCodeID | Int | Primary Key |
| 2 | Code | Nchar(10) | Not Null |
| 3 | DiscountPercent | int | Not Null |

Bảng . Bảng dữ liệu mật khẩu

### dbo.Employee

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên cột | Kiểu dữ liệu | Ràng buộc |
| 1 | EmployeeID | Int | Primary Key |
| 2 | EmployeeName | Nchar(200) | Not Null |
| 3 | EmployeePhone | Nchar(10) | Not Null |
| 4 | EmployeeCode | Nchar(20) |  |
| 5 | EmployeeAdd | Nchar(300) |  |
| 6 | EmployeeAccount | Nchar(15) | Not Null |
| 7 | EmployeePassword | Nchar(200) | Not Null |

Bảng . Bảng dữ liệu nhân viên

### dbo.Order

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên cột | Kiểu dữ liệu | Ràng buộc |
| 1 | OrderID | Int | Primary Key |
| 2 | CustomerID | Int | Foreign Key |
| 3 | OrderDate | DateTime | Not Null |
| 4 | PaymentID | Int | Foreign Key |
| 5 | OrderStatusID | Int | Foreign Key |
| 6 | DiscountCodeID | Int | Foreign Key |

Bảng . Bảng dữ liệu đặt hàng

### dbo.OrderDetail

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên cột | Kiểu dữ liệu | Ràng buộc |
| 1 | OrderDetailID | Int | Primary Key |
| 2 | ProductID | Int | Foreign Key |
| 3 | Quantity | Int |  |
| 4 | OrderID | Int | Foreign Key |

Bảng . Bảng dữ liệu chi tiết đơn hàng

### dbo.OrderStatus

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên cột | Kiểu dữ liệu | Ràng buộc |
| 1 | OrderStatusID | Int | Primary Key |
| 2 | OrderStatus | Nchar(200) |  |

Bảng . Bảng dữ liệu trạng thái đơn hàng

### dbo.PaymentID

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên cột | Kiểu dữ liệu | Ràng buộc |
| 1 | PaymentID | Int | Primary Key |
| 2 | PaymentName | Nchar(200) |  |

Bảng . Bảng dữ liệu thanh toán

### dbo.Product

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên cột | Kiểu dữ liệu | Ràng buộc |
| 1 | ProductID | Int | Primary Key |
| 2 | ProductName | Nchar(1000) | Not Null |
| 3 | ProductPrice | Float | Not Null |
| 4 | ProductImage | Nchar(1000) |  |
| 5 | ProductDescription | Text |  |
| 6 | ProductCategoryID | Int | Foreign Key |
| 7 | ProductGender | Bit |  |
| 8 | ProductSizeID | Int | Foreign Key |
| 9 | ProductColorID | Int | Foreign Key |
| 10 | ProductActive | bit |  |

Bảng . Bảng dữ liệu sản phẩm

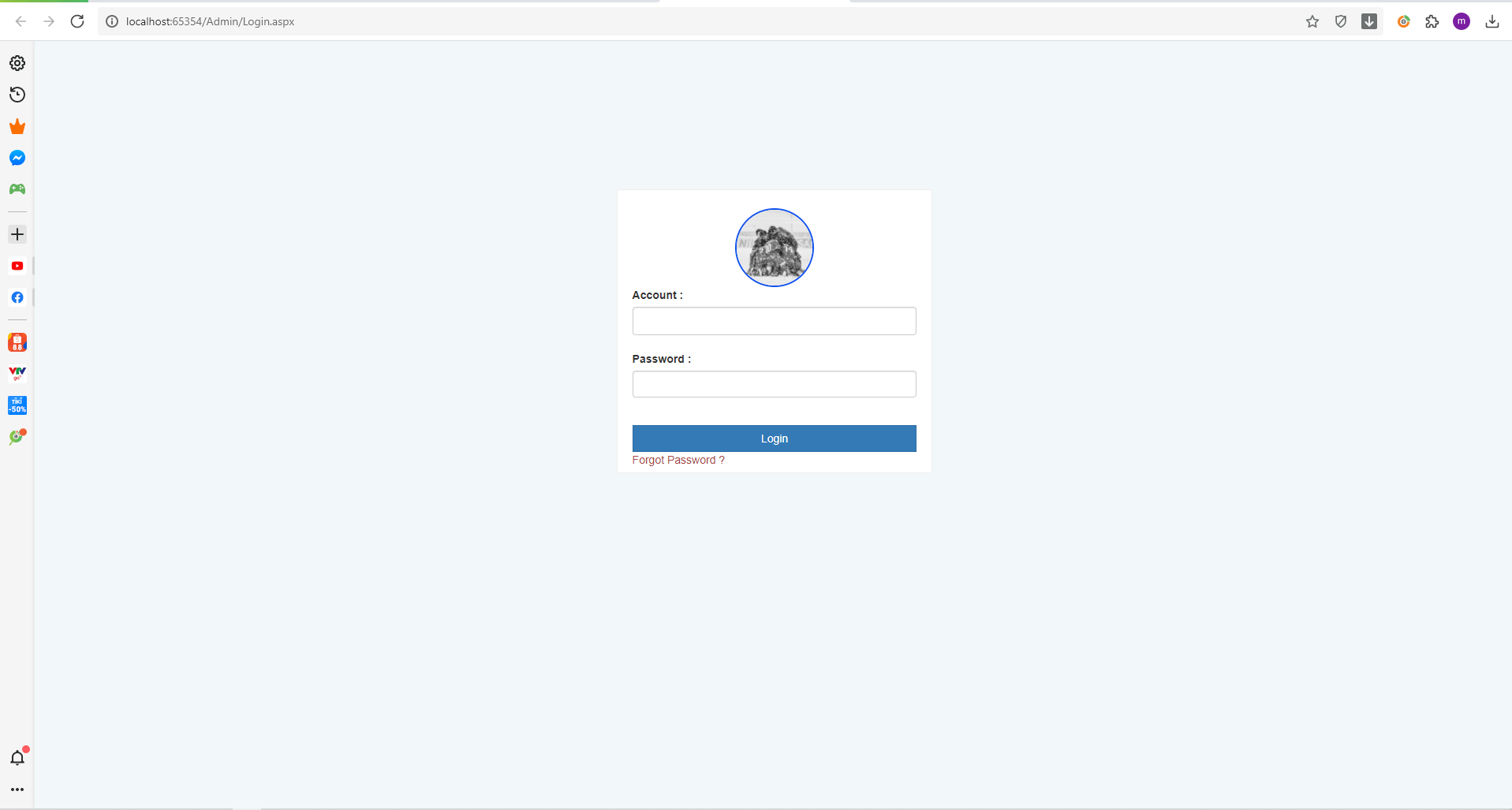
### dbo.ProductSize

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên cột | Kiểu dữ liệu | Ràng buộc |
| 1 | SizeID | Int | Primary Key |
| 2 | SizeName | Nchar(10) |  |
| 3 | Height | Nchar(7) |  |
| 4 | Weight | Nchar(7) |  |

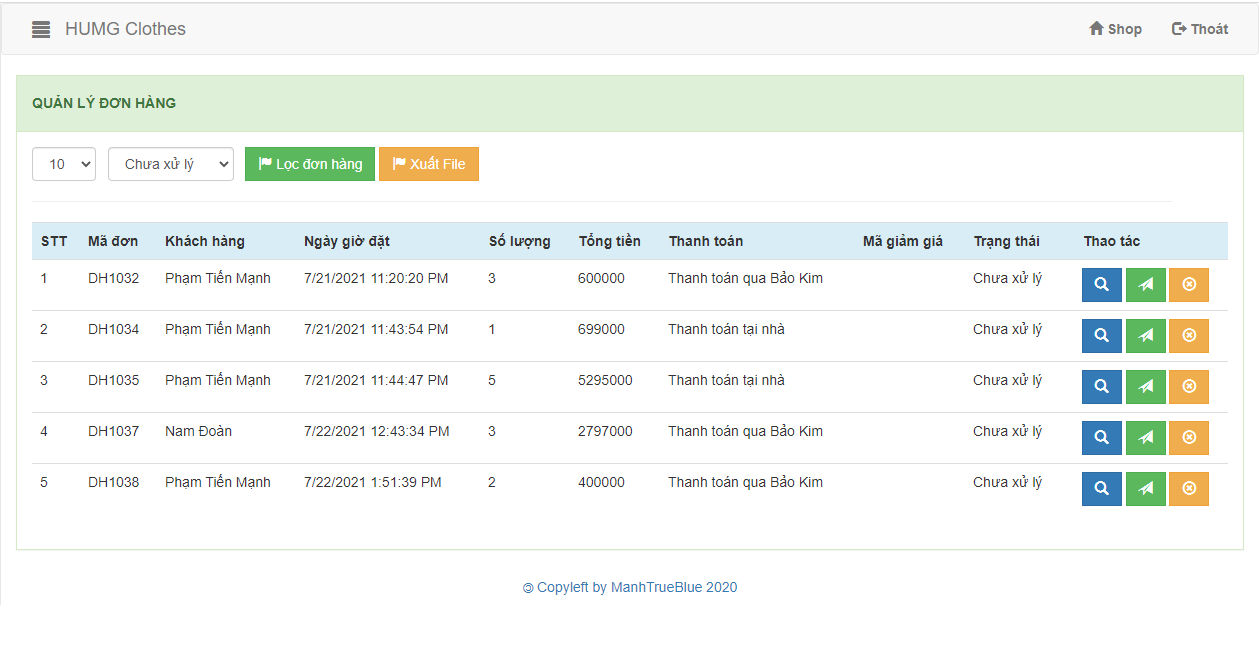
Bảng . Bảng dữ liệu kích thước sản phẩm

# : XÂY DỰNG GIAO DIỆN VÀ CHẠY THỬ NGHIỆM

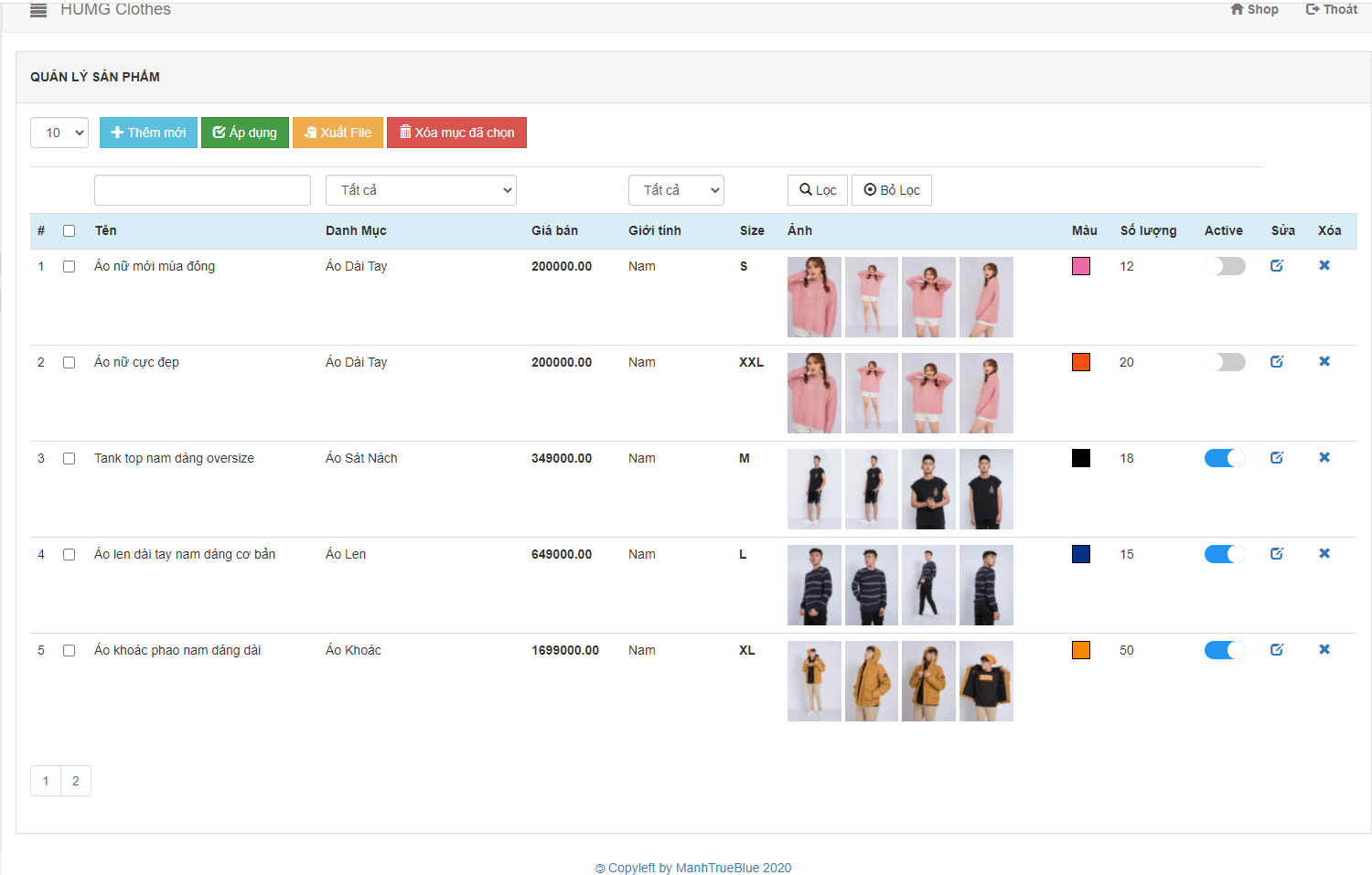
## Một số giao diện đã xây dựng



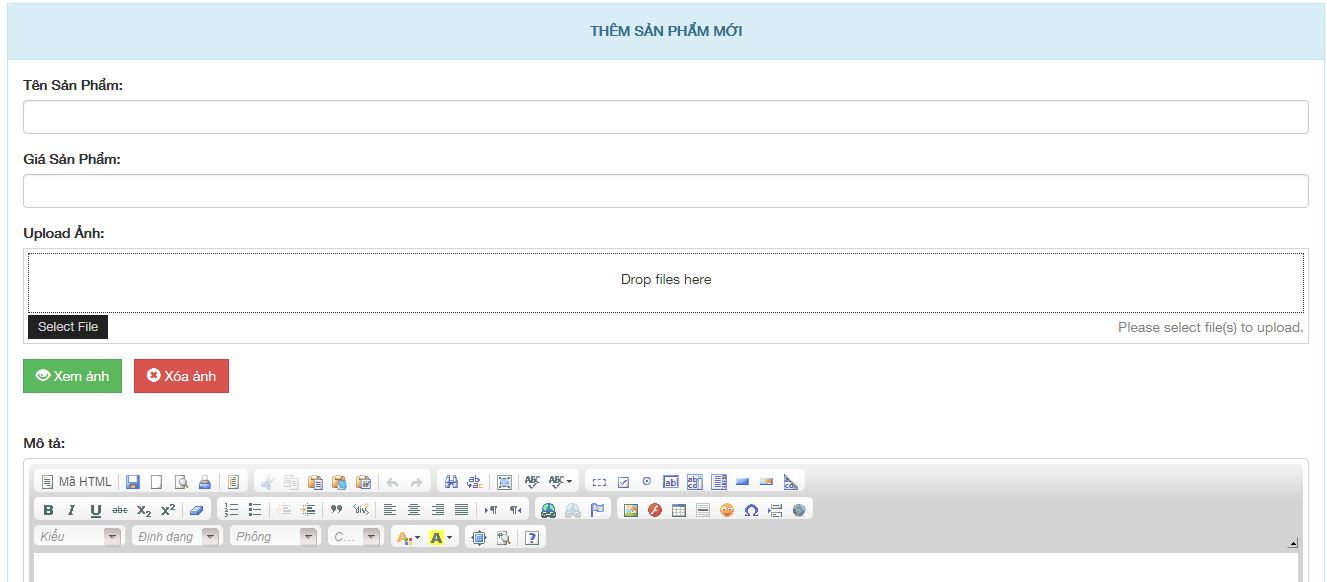
Hình 3.2 Giao diện Login



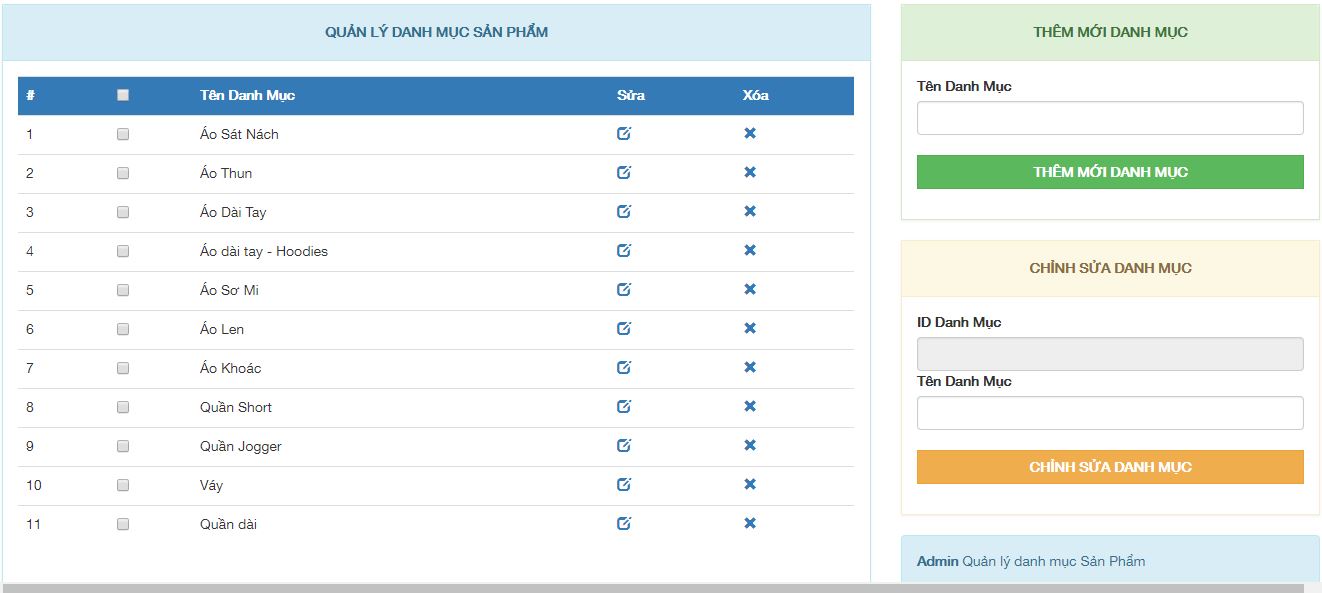
Hình 3.3 Giao diện quản lý đơn hàng



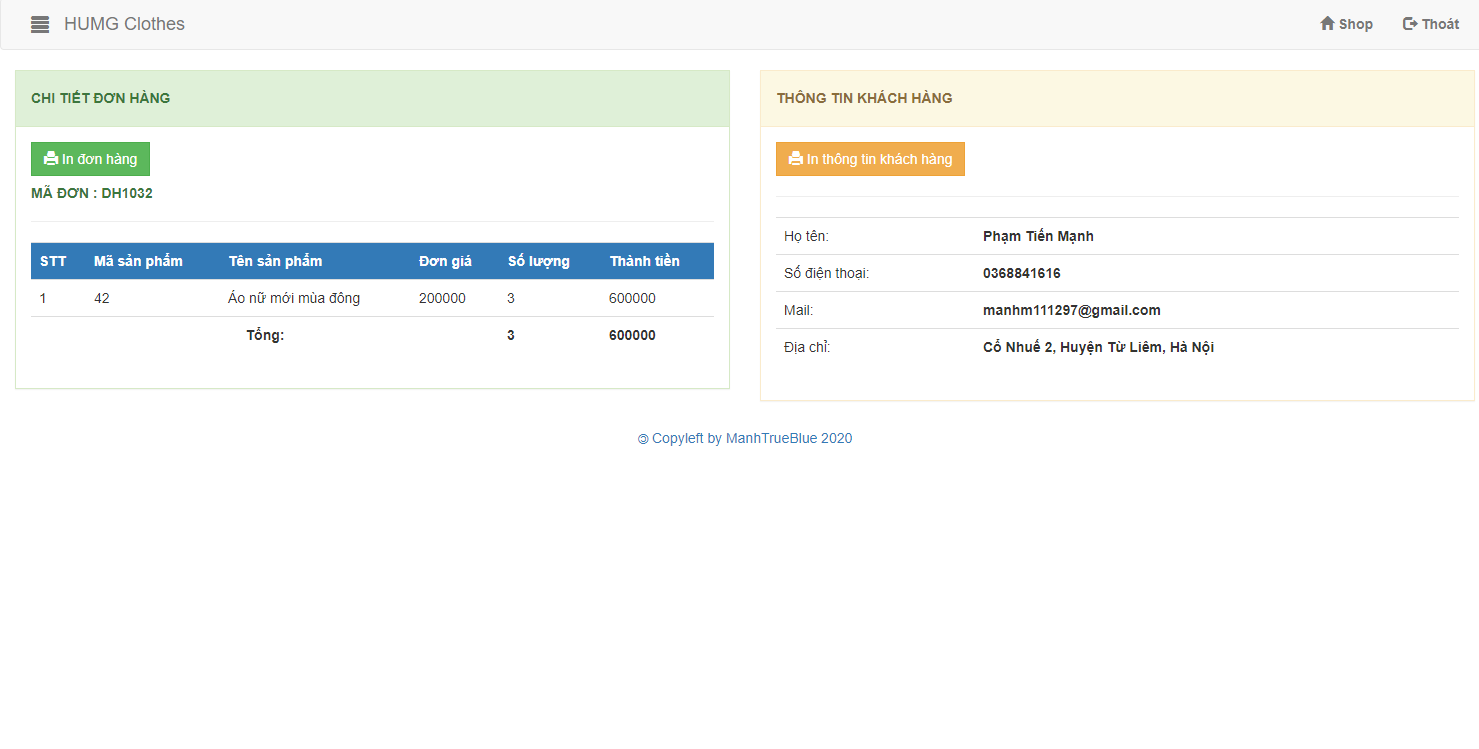
Hình 3.4 Giao diện quản lý sản phẩm



Hình 3.5 Giao diện thêm mới sản phẩm



Hình 3.6 Giao diện quản lý danh mục



Hình 3.7 Giao diện chi tiết đơn hàng

## Tổng kết

Xây dựng Website bán hàng quần áo trên nền tảng ASP.Net. Chương trình xây dựng với mục đích là vừa mang tính chất quảng bá sản phẩm, vừa phải là phần mềm quản lý kinh doanh trong đó xử lý chủ yếu các nghiệp vụ quản lý bán hàng và xử lý các đơn hàng của khách hàng thông qua mạng internet. Chương trình đã góp một phần không nhỏ cho các doanh nghiệp có thể tự quảng bá sản phẩm của mình đến người tiêu dùng, đồng thời giúp người tiêu dùng dễ dàng tìm mua sản phẩm theo sở thích mà không mất nhiều thời gian cho việc tìm kiếm.

Chương trình cho phép thực hiện giao dịch mua, bán sản phẩm trên website một cách dễ dàng, cho phép người quản trị hệ thống có thể thường xuyên thêm mới, cập nhật thông tin sản phẩm cũng như gỡ bỏ các sản phẩm cũ, không có trong danh mục sản phẩm đang bán, cho phép quản trị có thể quản lý được chính xác các đơn đặt hàng của khách hàng để có thể giao dịch kịp thời tới khách hàng của mình. Ngoài ra còn cho phép lưu trữ thông tin khách hàng một cách ngắn gọn nhưng vừa đủ để có thể thực hiện các giao dịch với khách hàng khi thực hiện các thủ tục mua bán hàng hóa thông qua website.

Tuy nhiên chương trình vẫn còn nhiều hạn chế, em kính mong thầy cô đóng góp ý kiến để chương trình được hoàn thiện hơn.

# TÀI LIỆU THAM KHẢO

x

|  |  |
| --- | --- |
| [1] | Andrew Tuner, *Introduction to Neogeography*. Sebastopol, CA USA: O'Reilly Media, 2006. |
| [2] | kdtqt.duytan.edu.vn. Tổng quan về mô hình MVC. [Online]. <http://kdtqt.duytan.edu.vn/Home/ArticleDetail/vn/33/1790/tong-quan-ve-mo-hinh-mvc> |
| [3] | microsoft. Tổng quan về ASP.NET MVC. [Online]. <https://support.microsoft.com/vi-vn/help/2645095> |
| [4] | tutorialspoint.com. ASP.NET MVC tổng quan. [Online]. <https://www.tutorialspoint.com/asp.net_mvc/asp.net_mvc_overview.htm> |
| [5] | Lê Văn Nam, *Lập trình.NET 2*. HaNoi: HUMG, 2018. |

x