BỘ GIÁO DỤC ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC NHA TRANG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN PHÁT TRIỂN ỨNG DỤNG WEB**

**ĐỀ TÀI**

**XÂY DỰNG HỆ THỐNG QUẢN LÝ BÁN HÀNG HIỆU GIÀY PHƯỚC AN TRÊN NỀN TẢNG ASP.NET MVC 5**

**Giảng viên: Bùi Chí Thành**

**Nhóm 5**

**Nguyễn Lê Thành Tâm  
Nguyễn Khánh Duy**

**Nguyễn Minh Trí**

*Nha Trang, ngày 10 tháng 03 năm 2022*

Trang mục lục

Mục lục hình ảnh (nếu cần )

Mục lục bảng (nếu cần)

Danh sách chữ cái viết tắt (nếu cần)

**LỜI MỞ ĐẦU**

Bước sang cuộc cách mạng công nghệ 4.0, mọi lĩnh vực trong đời sống từ kinh tế, xã hội, quốc phòng, an ninh cho tới kinh doanh, giải trí gần như đề có sự góp mặt và hỗ trợ của công nghệ thông tin. Sự áp dụng công nghệ vào các lĩnh vực ấy giúp đơn giản hóa các khâu công việc phức tạp mà còn đem lại hiệu quả cao, nhất là trong lĩnh vực kinh doanh, mua bán. Quản lý các thông tin như: khách hàng, nhân viên, sản phẩm, hóa đơn, doanh thu, giao hàng, nhận hàng, thanh toán,...luôn là điều khiến các doanh nghiệp chú tâm. Công nghệ ra đời và giúp việc quản lý nó trở nên tối ưu hơn. Xuất phát từ ý tưởng cũng như nhu cầu thực tiễn đó, nhóm chúng em đã nghiên cứu và xây dựng nên một hệ thống quản lý bán hàng cho hiệu giày Phước An trên nền tảng ASP.NET với công nghệ MVC5.

**TÓM TẮT ĐỀ TÀI**

Công nghệ Web ứng dụng trên nền .NET Framework của Microsoft ngày càng được sử dụng rộng rãi với sự phát triển mạnh mẽ của các công nghệ như: PHP, JSP,... Tuy nhiên công nghệ ASP.NET vẫn khẳng định được tính ưu việt của mình. Tiếp đến là sự phát triển ồ ạt từ việc kinh doanh truyền thống sang kinh doanh trực tuyến, sự phát triển đó đòi hỏi về sự quản lý chặt chẽ cũng như khả năng sử lý một lượng lớn thông tin phức tạp để duy trì và phát triển doanh nghiệp.

Chính vì thế, để có thể tiết kiệm được thời gian, nhân công, và chi phí đầu tư cũng như tính tiện dụng, chúng em đã chọn đề tài xây dựng hệ thống hỗ trợ quản lý bán hàng mà đối tượng chính ở đây là hiệu giày Phước An, Nha Trang.

Trong suốt quá trình nghiên cứu và xây dựng dựa trên khảo sát, tham khảo, hỏi ý kiến và các vấn đề cần giải quyết từ cửa hàng, trao đổi giữa các thành viên,...Chúng em đã đút kết và xây dựng thành công hệ thống quản lý bán hàng đáp ứng đầy đủ các yêu cầu được đặt ra. Hệ thống có thể vận hành dễ dàng, cấu trúc đơn giản cũng như dễ dàng được học và sử dụng trong thực tế.

# CHƯƠNG I: TỔNG QUAN

## GIỚI THIỆU TỔNG QUAN

Với sự phát triển vượt bậc của công nghệ thông tin ngày nay, Microsoft là một trong nhưng đơn vị tiên phong cho ra đời một nền tảng ứng dụng mới tên là Web Application. Được phát triển trên nền tảng .NET Framework, vận hàng bằng ứng dụng truy cập Website thông thường trên đa nền tảng, đa thiết bị và quá trình vận hành được thực hiện mọi lúc, mọi nơi. Đây là một công nghệ tiên tiến có thể thay thế hoàn toàn Winform, công nghệ này có thể đáp ứng cho nhu cầu công việc ngày nay, đặc biệt là trong khâu quản lý.

## LÝ DO CHỌN ĐỀ TÀI

Với mục đích xây dựng được một hệ thống quản lý bán hàng giày dép cho cơ sở Phước An để giải quyết nhu cầu bán hàng. Hệ thống sẽ vận hành được trên mọi thiết bị có khả năng truy cập Website.

Hệ thống được xây dựng với mong muốn nâng cao năng xuất và hiệu quả công việc, tiết kiệm thời gian, nhân công và các chi phí để đầu tư các thiết bị máy móc nếu có. Tích lũy thêm kiến thức và kinh nghiệm, phát triển khả năng của các cá nhân.

## PHẠM QUY NGHIÊN CỨU

Quản lý bán hàng cho hiệu giày Phước An bao gồm các thông tin: khách hàng, nhân viên, nhà cung cấp, sản phẩm, loại sản phẩm, thanh toán,... của hiệu giày. Quy trình khách hàng đặt hàng, mua hàng, giao hàng và nhận hàng.

## PHƯƠNG PHÁP NGHIÊN CỨU

* + Khảo sát và quan sát thực tế quá trình buôn bán tại hiệu giày Phước An.
  + Trực tiếp đặt hàng tại cửa hàng để xem xét quy trình hoạt động cửa hàng.
  + Tìm và nghiên cứu các website mua bán giày khác để đưa ra được một quy trình thực hiện đặt hàng, mua bán sản phẩm hợp lý.
  + Tham khảo các mô hình, hệ thống sẵn có từ các trang bán hàng nổi tiếng.

## MỤC TIÊU CẦN ĐẠT ĐƯỢC

# CHƯƠNG II CƠ SỞ LÝ THUYẾT

## GIỚI THIỆU CÔNG NGHỆ ASP.NET MVC5

### LỊCH SỬ RA ĐỜI

MVC (viết tắt của Model – View – Controller) được phát minh tại Xerox Parc vào những năm 70 bởi TrygveReenskaug. MVC lần đầu tiên được công khai là trong Smalltalk-80 và được phát triển dần với các phiên bản hoàn thiện hơn.

Lần lược các MVC framework ra đời dựa trên mô hình MVC như CodeIgniter, Zend, ASP.NET MVC,...

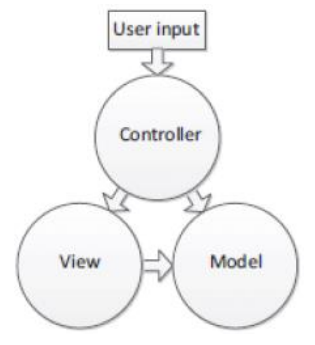
### 1.2 TỔNG QUAN VỀ MÔ HÌNH ASP.NET MVC

ASP.NET MVC được thiết kế gọn nhẹ, chi ứng dụng thành 3 tầng (layer): Model, View, và Controller. Việc chia như vậy là cho việc xây dựng ứng dụng trở nên ít phức tạp hơn, giúp lập trình viên dễ dàng kiểm soát các thành phần trong khi phát triển, lợi ích cho việc kiểm tra, bảo trì và nâng cấp sau này.

- Model: có chức năng chứa dữ liệu chia sẽ chung giữa Controller và View.

- View: là giao diện dành cho người sử dụng. View tiếp nhận Model từ Controller để sinh ra giao diện phù hợp.

- Controller: Xử lý các yêu cầu người dùng đưa đến thông qua view. Xử lý và xây dựng các Model phù hợp và chuyển Model sang cho View.



Hình 1 Mô hình MVC

## ĐẶC ĐIỂM MÔ HÌNH MVC5

Với MVC5 thì cải tiến hơn so với ASP.NET MVC4, Bootstrap được thay thế mẫu MVC mặc định. Chứng thực người dùng Authentication Filter được tuỳ chỉnh hoặc chứng thực từ hãng thứ 3 cung cấp. Với Filter overrides, chúng ta có thể Filter override trên Method hoặc Controller.

* Thuộc tính Routing được tích hợp vào MVC5.
* Tiếp tục hỗ trợ các tính năng trong ASP.NET : Hỗ trợ sử dụng các các tập tin:.ASPX, .ASCX, .Master như là thành phần View.
* Có khả năng mở rộng, mọi thứ trong MVC được thiết kế để dễ thay thế, dễ dàng

tùy biến. Ánh xạ URL mạnh mẽ, cho phép xây dựng ứng dụng với những URL sạch.

Không sử dụng mô hình post-back từ giao diện gửi đến server. Thay vào đó, chủ

động đưa những post-back từ View đến thẳng lớp Controller.

* Hỗ trợ nhiều công cụ tạo View (Support for Multiple View Engines)

- Cho phép chọn công cụ tạo view. Hộp thoại New Project cho phép xác định view engine mặc định cho một project.

- Các loại view engine

* Web Forms (ASPX)
* Razor
* Hay một view engine nguồn mở như Spark, NHaml, NDjango.
* Hỗ trợ định tuyến

- ASP.NET MVC Framework có một bộ máy ánh xạ URL thật sự mạnh mẽ.

- Bộ máy này cung cấp phương pháp rất linh hoạt trong việc ánh xạ URLs

sang các Controller Classes.

- Bạn có thể dễ dàng định ra các quy luật, cài đặt đường đi, ASP.NET dựa

vào các quy luật đường đi đó để xác định Controller và action cần phải thực thi.

- ASP.NET còn có khả năng phân tích URL, chuyển các thông số trong

URL thành các tham số trong lời gọi hàm của Controller.

* Model Binding

- Model Binding là tính năng thế mạnh của ASP.NET MVC (và bây giờ nó

cũng được áp dụng cho cả Web Forms trong phiên bản ASP.NET 4.5).

- Hỗ trợ viết phương thức nhận một đối tượng tùy biến như một tham số

- Với sự hỗ trợ của Model Binding, bây giờ bạn chỉ cần tập trung vào việc

cài đặt các nghiệp vụ logic, không cần phải bận tâm về việc suy nghĩ làm

cách nào để ánh xạ dữ liệu từ người dùng sang các đối ượng .NET.

* Razor View.

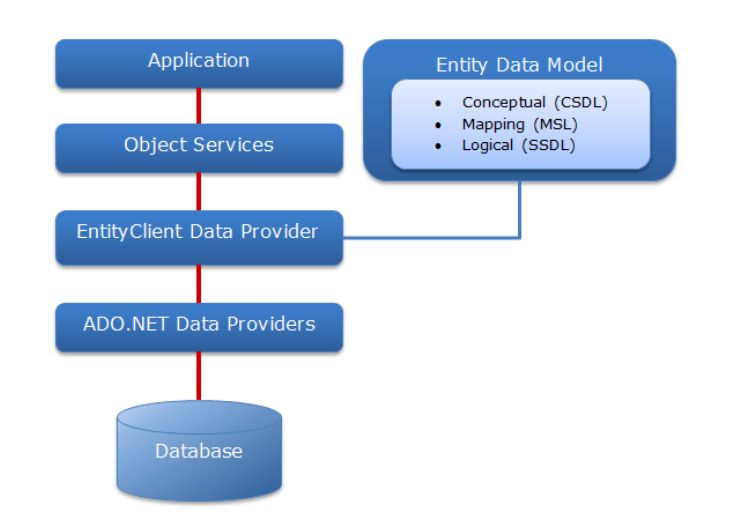
Từ ASP.net MVC 3 đi kèm với một công cụ View mới có tên là Razor với lợi ích:

* + Cú pháp Razor là sạch sẽ và xúc tích, đòi hỏi một số lượng tối thiểu các tổ hợp phím.
  + Việc tìm hiểu Razor tương đối dễ dàng vì nó dựa trên ngôn ngữ C# và Visual Basic.
  + Visual Studio bao gồm IntelliSense và mã cú pháp Razor được màu hóa.
  + Với Razor views có thể kiểm tra từng đơn vị mà không đòi hỏi bạn phải chạy các ứng dụng hoặc phải chạy website

## GIỚI THIỆU ENTITY FRAMEWORK

### 3.1 TỔNG QUAN

Entity Framework (EF) là một framework ánh xạ quan hệ đối tượng (ORM)  
dành cho ADO.NET, là 1 phần của .NET Framework. EF cho phép các nhà phát triển  
Web tương tác với dữ liệu quan hệ theo phương pháp hướng đối tượng đặc trưng. Lợi  
ích lớn nhất của EF là giúp lập trình viên giảm thiểu việc lập trình mã nguồn cần thiết  
để truy cập và tương tác với cơ sở dữ liệu. EF được Microsoft hỗ trợ phát triển lâu dài  
và bền vững, vì vậy EF là 1 framework mạnh nhất hiện nay để phát triển ứng dụng  
Web với sự hỗ trợ đông đảo của các nhà phát triển Web.



### 3.2 TẦNG APPLICATION

Application (ứng dụng) là tầng chứa giao diện trang Web (HTML, CSS,  
Javascript, hình ảnh, …) và các đoạn mã nguồn (C#, VB) để tương tác dữ liệu với các  
tầng khác trong mô hình thông qua Object Services

3.3 TẦNG OBJECT SERVICES

Object Services (tạm dịch là các dịch vụ đối tượng) là tầng chứa quá trình  
tương tác giữa ứng dụng và database, hay nói cách khác nó là nơi chủ yếu để truy cập  
dữ liệu từ database và trả ngược kết quả về giao diện. Object Services cung cấp các  
tiện ích để truy vết các thay đổi và quản lý nhận dạng, đồng thời là các quan hệ và thay  
đổi ở database.

### 3.4 TẦNG ENTITYCLIENT DATA PROVIDER

EntityClient là một data provider mới của ADO.NET dùng để truy xuất đến  
database. Được xây dựng bên trên các ADO.NET data provider cơ bản, EntityClient  
không truy xuất trực tiếp dữ liệu mà thông qua các data provider khác dựa vào các  
thông tin dữ liệu từ Entity Data Model.  
EntityClient cũng bao gồm các lớp giống như các ADO.NET data provider  
khác và tên lớp được đặt với tiền tố Entity. Ví dụ bạn có thể tạo kết nối bằng  
EntityConnection, tạo các câu truy vấn bằng EntityCommand và đọc kết quả bằng EntityDataReader.  
Một điểm khác biệt với các data provider khác là EntityClient sử dụng Entity  
SQL để truy vấn dữ liệu. Các lệnh Entity SQL sẽ được chuyển thành một cấu trúc lệnh  
dạng cây (command tree) và chuyển xuống cho các data provider khác

### 3.5 TẦNG ADO.NET DÂT PROVIDERS

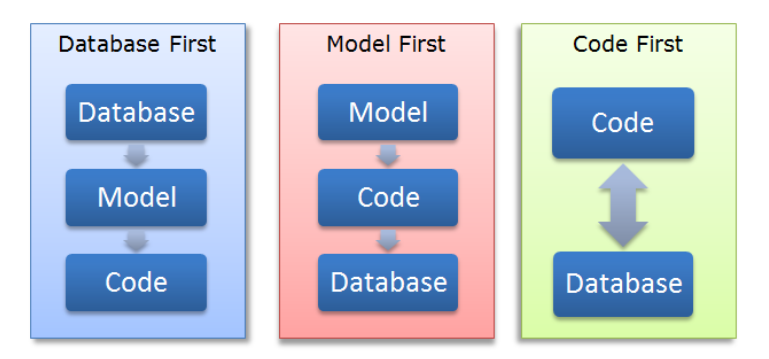
Đây là tầng thấp nhấp để dịch các truy vấn L2E (LINQ to Entity) thông qua cây  
lệnh thành các câu lệnh SQL và thực thi các câu lệnh trong hệ thống DBMS (database  
management system – hệ quản lý dữ liệu) nào đó. Tầng này kết với database sử dụng  
ADO.NET.  
EntityClient là một data provider mới của ADO.NET dùng để truy xuất đến  
database. Được xây dựng bên trên các ADO.NET data provider cơ bản, EntityClient  
không truy xuất trực tiếp dữ liệu mà thông qua các data provider khác dựa vào các  
thông tin dữ liệu từ Entity Data Model

### 3.6 TẦNG EDM (ENTITY DATA MODEL)

Entity Data Model (EDM) là mô hình dữ liệu được mô tả thông qua các ngôn  
ngữ theo chuẩn XML. EDM được chia làm 3 lớp là: Conceptual, Mapping và Logical.  
Mỗi lớp này được định nghĩa bởi ngôn ngữ riêng theo định dạng XML:  
- Conceptual – Conceptual Schema Definition Language (CSDL): là ngôn ngữ  
định nghĩa các entity, relationship, hàm trong tập tin với phần mở rộng .csdl. Có thể  
tạo được các entity class (object layer).  
- Mapping – Mapping specification language (MSL): định nghĩa các ánh xạ giữa  
lớp conceptual và logical, nội dung này được lưu trong tập tin .msl.  
- Logical – Store Schema Definition Language (SSDL): định nghĩa mô hình lưu  
trữ của dữ liệu, lưu trữ trong tập tin .ssdl

### 3.7 CÁC SỬ DỤNG TRONG ENTITY FRAMEWORK

Là thành phần trung gian giữa ứng dụng và database, Entity Data Model (EDM) giúp tạo liên kết và mô tả giữa nguồn dữ liệu vật lý và các đối tượng bussiness trong ứng dụng. Dựa vào sự lựa chọn thành phần nào sẽ được tạo ra trước, một ứng dụng Entity  
Framework (EF) có thể sử dụng 1 trong 3 cách tiếp cận: Database First, Model First và Code First.  
Các sơ đồ dưới đây minh hoạt mối quan hệ và thứ tự tạo ra giữa các thành phần dữ liệu  
trong ứng dụng của Entity Framework. Phần Model tương ứng với tập tin .edmx, Code tương ứng với tập tin .cs (hoặc .vb nếu bạn dùng VB.NET)



## GIỚI THIỆU SQL SERVER

Là một hệ quản trị cơ sở dữ liệu do Microsoft phát triển. SQL Server là một hệ  
quản trị cơ sở dữ liệu quan hệ mạng máy tính hoạt động theo mô hình khách chủ cho  
phép đồng thời cùng lúc có nhiều người dùng truy xuất đến dữ liệu, quản lý việc truy  
nhập hợp lệ và các quyền hạn của từng người dùng trên mạng

## NGÔN NGỮ GIAO DIỆN

### HTML VA CSHTML

HTML viết tắt của HyperText Markup Language (tạm dịch Ngôn ngữ Đánh dấu  
Siêu văn bản) là ngôn ngữ đánh dấu được thiết kế để tạo nên các trang web với các  
mẫu thông tin trình bày trên World Wide Web. Cùng với CSS và JavaScript, HTML tạo thành bộ ba nền tảng kỹ thuật cho World Wide Web. Hiện nay, HTML đang được  
phát triển tiếp với phiên bản HTML5 hứa hẹn mang lại diện mạo mới cho trang web.  
Trong view của ASP.NET, sẽ sử dụng các thẻ HTML đồng thời có thể kết hợp  
với các khối lệnh ngôn ngữ C# bằng cách khai báo @{…} hoặc @<tên biến> để tạo ra  
một view template (file .cshtml)

### 5.2 RAZOR VIEW ENGINE

Như chúng ta đã biết, từ phiên bản Asp.net MVC 3, Microsoft lựa chọn View  
Engine mặc định là Razor View Engine, khi View được yêu cầu bởi Http Request thì  
Razor View Engine thực hiện biên dịch trang View này.  
Khi sử dụng Razor View Engine, lập trình viên không cần phải học một ngôn  
ngữ lập trình mới, mà thay vào đó Razor cung cấp các mẫu nhằm kết hợp giữa cú pháp  
của ngôn ngữ đánh dấu siêu văn bản (HTML) và ngôn ngữ lập trình như C#, VB ở nội  
dung của các trang View. Với các lập trình viên đã quen với các ngôn ngữ C# hoặc  
VB thì học và làm quen với Razor View Engine là rất đơn giản.

Để thông dịch các đoạn mã được viết bởi các ngôn ngữ như C#, VB trong các  
trang View, Razor đưa ra những quy tắc nhằm để xác định các đoạn mã đó được biên  
dịch ở phía Server (server-side code) và các đoạn mã đánh dấu (markup code) được  
thực hiện biên dịch ở phía client (như javascript, css, html, …), Razor sử dụng ký tự @  
nhằm phân biệt giữa server-side code và markup code.  
Khi tạo ra các Razor view chúng ta cần chú ý như sau :  
Sử dụng @{ code } để viết một khối mã lệnh  
Để viết một biểu thức inline (Viết biểu thức Razor trực tiếp trong đoạn mã của  
markup code, lập trình viên chỉ cần sử dụng @.  
Sử dụng từ khóa var để khai báo biến  
Sử dụng ; để kết thúc một câu lệnh  
Với các Razor view sử dụng ngôn ngữ C#, các file có đuôi mở rộng là .cshtml  
Với các Razor view sử dụng ngôn ngữ VB, các file có đuôi mở rộng là .vbhtml

### 5.3 CSS (Cascading Style Sheets)

CSS được viết tắc của cụm từ Cascading Style Sheets, hiểu đơn giản nó là tổng hợp  
các chuỗi dạng text, mỗi câu lệnh của CSS dùng để định dạng một phần tử của HTML về  
màu sắc, kích thước, font chữ...Nhờ có CSS mà giao diện của web, nội dung trên web trở  
nên rõ ràng hơn, hấp dẫn hơn với người đọc.

### 5.4 BOOTSTRAP

Bootstrap hiện đang là Framwork hỗ trợ tốt nhất và phổ biến nhất trong việc  
phát triễn các dự án về giao diện website cho cả thiết bị di động và máy tính bao gồm  
HTML, CSS và Java script. Bootstrap phát triễn qua nhiều phiên bản, phiên bản mới  
nhất lúc mình thực hiện tutorials là bản 3.3.2.  
Bootstrap được sử dụng rộng rải bởi lợi ích to lớn mà nó mang lại như giao diện  
website tương thích với các thiết bị smartphone, tablet và máy tính thông thường vì có  
thể tự động co giản kích thước. Bootstrap được xây dựng với các module hỗ trợ sẵn  
nên rất dễ dàng tìm hiểu, xây dựng giao diện một cách nhanh chóng

# CHƯƠNG III: PHÂN TÍCH HỆ THỐNG

## 1. XÁC ĐỊNH ACTOR

### 1.1 QUẢN TRỊ VIÊN HỆ THỐNG

Là thành viên có cấp độ cao nhất khi giao tiếp với hệ thống với toàn quyền thao  
tác với hệ thống.  
Thêm thành viên người dùng thao tác với hệ thống, cấp quyền cho người dùng thao tác với hệ thống, thêm xoá sửa thông tin sản phẩm, quản lí hoá đơn, thêm xoá sửa đơn hàng

### 1.2 NHÂN VIÊN

Nhân viên người dùng thao tác với hệ thống, cấp quyền cho người dùng  
thao tác với hệ thống, xem thông tin sản phẩm, quản lí hoá đơn, duyệt hoá đơn

## 2. XÁC ĐỊNH USECASE

### 2.1 QUẢN TRỊ VIÊN HỆ THỐNG

- Thay đổi cấu trúc giao diện hệ thống

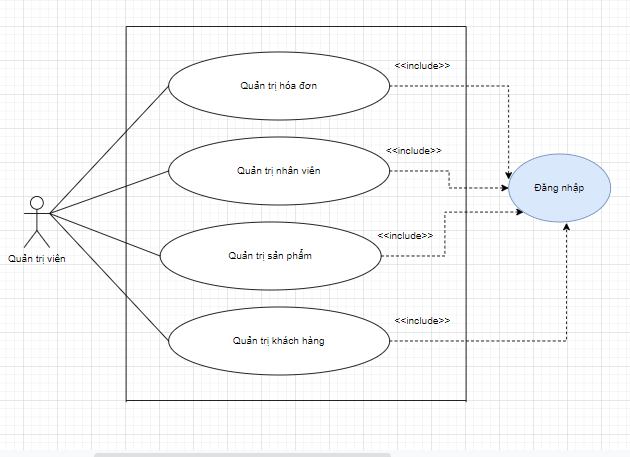
- Quản trị tài khoản quản trị viên hệ thống

- Quản trị danh sách sản phẩm giày, danh sách hoá đơn đặt của hệ thống

- Thực hiện tìm kiếm, thêm xoá sửa thông tin giày

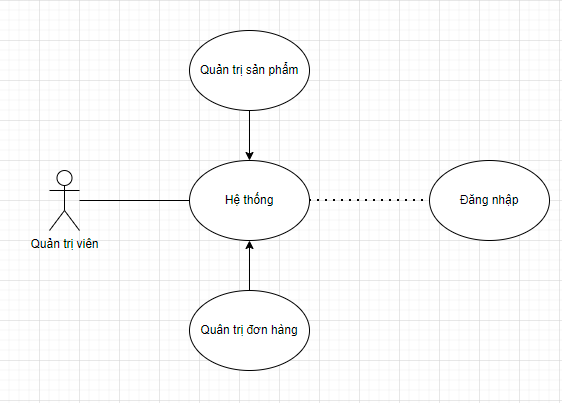
## 3. MÔ HÌNH TƯƠNG TÁC GIỮA NGƯỜI DÙNG VÀ HỆ THỐNG

### 3.1 SƠ ĐỒ TỔNG THỂ

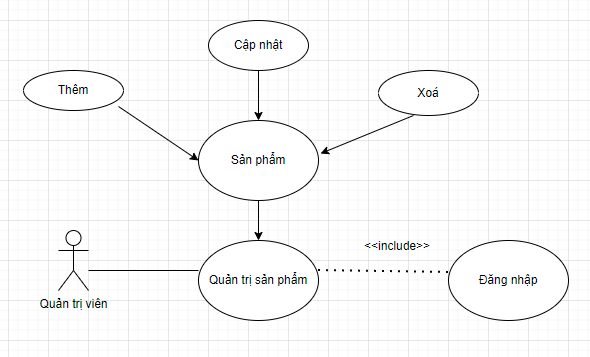


### 3.2 SƠ ĐỒ PHÂN RÃ CHỨC NĂNG CÁC ACTOR

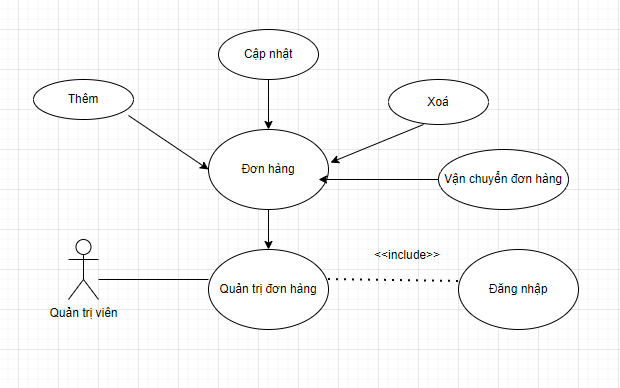
#### 3.2.1 Quản trị viên hệ thống



#### 3.2.2 Quản trị sản phẩm



#### 3.2.3 Quản trị đơn hàng



## 4. ĐẶC TẢ CÁC USE CASE TỔNG QUÁT

### 4.1 Usecase đăng nhập

|  |  |
| --- | --- |
| **Tên Use case** | Đăng nhập |
| **Mô tả use case** | Kiểm tra tên đăng nhập và mật khẩu đăng nhập của người dùng trong hệ thống |
| **Tác nhân chính** | Quản trị viên |
| **Mức** | 1 |
| **Người chịu trách nhiệm** | Quản trị viên |
| **Tiền điều kiện** | Không có |
| **Đảm bảo tối thiếu** | Tự động thoát khỏi hệ thống khi không sử dụng và đăng nhập lại |
| **Kích hoạt** | Truy cập vào trang giao diện chính của hệ thống |
| **Mở rộng** | Không có |

### 4.2 Usecase quản trị sản phẩm

|  |  |
| --- | --- |
| **Tên Use case** | Quản trị sản phẩm |
| **Mô tả use case** | Chức năng cho phép thực hiện quản trị sản phẩm trong hệ thống, phục vụ cho việc bán hàng hiệu quả, với các thao tác thêm xoá sửa sản phẩm giày |
| **Tác nhân chính** | Quản trị viên |
| **Mức** | 1 |
| **Người chịu trách nhiệm** | Quản trị viên |
| **Tiền điều kiện** | Đăng nhập vào hệ thống dưới vai trò quản trị |
| **Đảm bảo tối thiếu** | Thêm mới sản phẩm, cập nhật thông tin sản phẩm khi cần thay đổi, quản lí các thuộc tính khác của giày như kích thước, số lượng tồn kho, loại sản phẩm |
| **Đảm bảo thành công** | Lưu trữ thông tin vào hệ thống thành công |
| **Kích hoạt** | Từ trang quản trị chọn mục Sản phẩm |
| **Chuỗi sự kiện chính** | * + Thêm mã loại sản phẩm   + Thêm tên sản phẩm   + Nhập kích thước   + Đơn vị tính, số lượng kho   + Đơn giá   + Nhà cung cấp   Mô tả chi tiết sản phẩm |
| **Mở rộng** | Không có |

### 4.3 Usecase quản trị sản phẩm

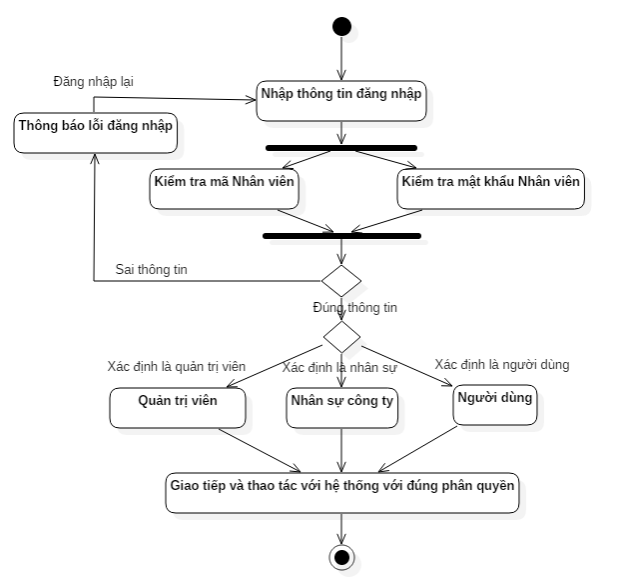
|  |  |
| --- | --- |
| **Tên Use case** | Quản trị đơn hàng |
| **Mô tả use case** | Chức năng cho phép thực hiện quản trị đơn hàng đã được đặt trong hệ thống, phục vụ cho việc bán hàng hiệu quả, với các thao tác cập nhật xoá duyệt đơn hàng |
| **Tác nhân chính** | Quản trị viên |
| **Mức** | 1 |
| **Người chịu trách nhiệm** | Quản trị viên |
| **Tiền điều kiện** | Đăng nhập vào hệ thống dưới vai trò quản trị |
| **Đảm bảo tối thiếu** | Cập nhật thông tin đơn hàng khi cần thay đổi, quản lí các thuộc tính khác của đơn hàng như hình thức thanh toán, trạng thái đơn hàng, tổng tiền, thông tin giao hàng, ngày đặt, ngày giao đơn hàng |
| **Đảm bảo thành công** | Lưu trữ thông tin vào hệ thống thành công |
| **Kích hoạt** | Từ trang quản trị chọn mục Sản phẩm |
| **Chuỗi sự kiện chính** | * + Sửa hình thức thanh toán   + Cập nhật ngày giao   + Cập nhật trạng thái đơn hàng, xem chi tiết đơn hàng |
| **Mở rộng** | In hóa đơn hoặc lưu sang định dạng file PDF |

### 4.4 Usecase tìm kiếm sản phẩm

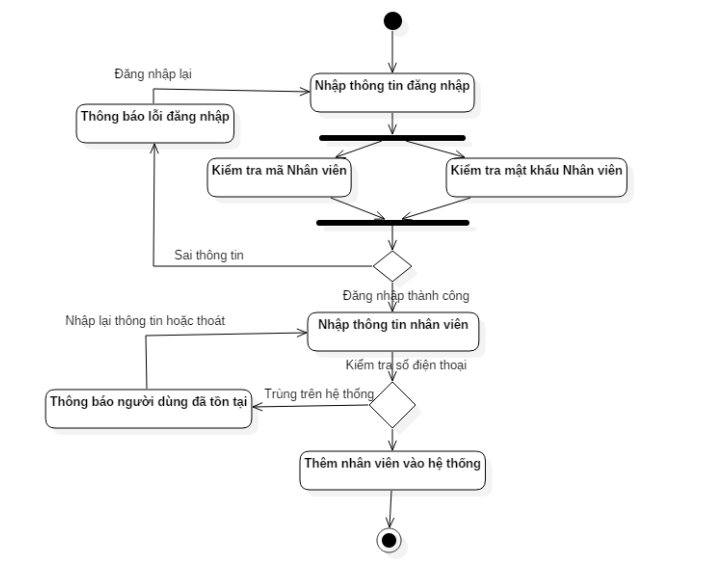
|  |  |
| --- | --- |
| **Tên Use case** | Tìm kiếm sản phẩm theo nhiều tiêu chí |
| **Mô tả use case** | Chức năng cho phép thực hiện tìm kiếm sản phẩm theo tên, theo mã sản phẩm, theo đơn giá và theo nhà cung cấp |
| **Tác nhân chính** | Quản trị viên |
| **Mức** | 1 |
| **Người chịu trách nhiệm** | Quản trị viên |
| **Tiền điều kiện** | Đăng nhập vào hệ thống dưới vai trò quản trị |
| **Đảm bảo tối thiếu** | Không có |
| **Đảm bảo thành công** | Thực hiện tìm kiếm thành công với các tiêu chí tìm kiếm |
| **Kích hoạt** | Từ trang quản trị chọn mục Sản phẩm |
| **Chuỗi sự kiện chính** | * + Chọn mục tìm kiếm   + Nhập các tiêu chí cần thực hiện tìm kiếm   + Nhấn Tìm kiếm hoặc Enter |
| **Mở rộng** | Thông báo khi không tìm thấy sản phẩm nào tương thích các tiêu chí tìm kiếm |

## 5. SƠ ĐỒ HOẠT ĐỘNG CÁC CHỨC NĂNG

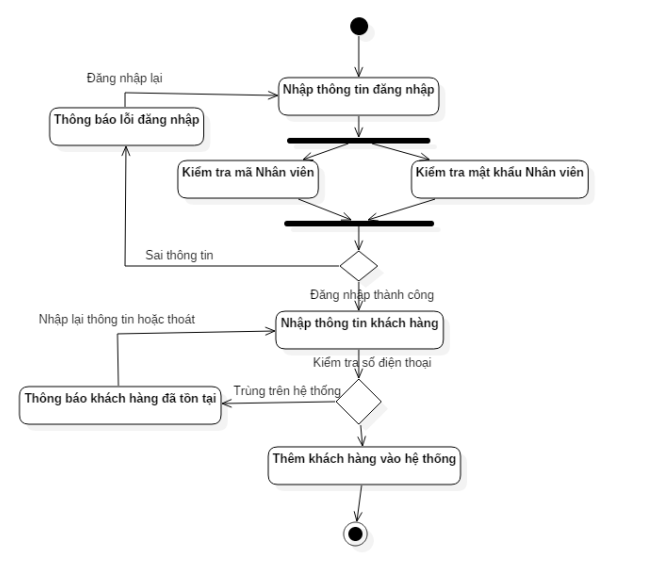
### 5.1 Chức năng đăng nhập



### 5.2 Chức năng thêm tài khoản

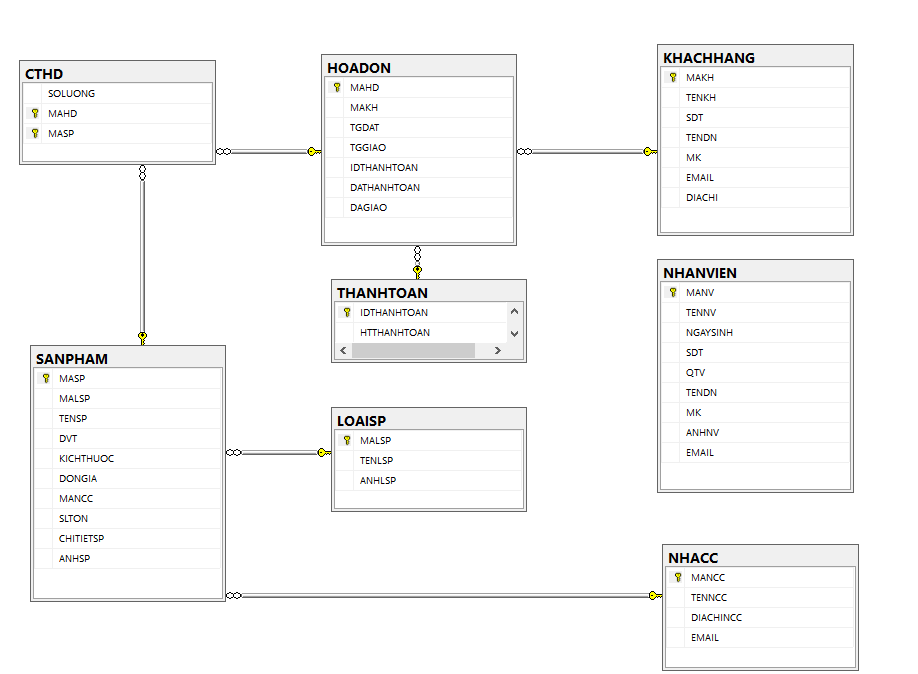


### 5.3 Chức năng thêm tài khoản người dùng



### 5.4 Chức năng thêm sản phẩm mới

## MÔ HÌNH CƠ SỞ DỮ LIỆU



## 7. NỘI DUNG CÁC BẢNG DỮ LIỆU

Bảng khách hàng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên cột | Kiểu dữ liệu | Not Null | Diễn giải |
| 1 | **MALSP** | VARCHAR(10) | NOT NULL | Mã loại sản phẩm |
| 2 | TENLSP | NVARCHAR(200) | NOT NULL | Tên loại sản phẩm |
| 3 | ANHLSP | VARCHAR(1024) | NOT NULL | Ảnh loại sản phẩm |

Bảng nhân viên

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên cột | Kiểu dữ liệu | Not Null | Diễn giải |
| 1 | **MANV** | VARCHAR(10) | NOT NULL | Mã nhân viên |
| 2 | TENNV | NVARCHAR(200) | NOT NULL | Tên nhân viên |
| 3 | NGAYSINH | DATE | NOT NULL | Ngày sinh |
| 4 | SDT | NVARCHAR(15) | NOT NULL | Số điện thoại |
| 5 | CHUCVU | NVARCHAR(200) | NOT NULL | Chức vụ |
| 6 | TENDN | VARCHAR(15) | NOT NULL | Tên doanh nghiệp |
| 7 | MK | VARCHAR(15) | NOT NULL | Mật khẩu |
| 8 | ANHNV | VARCHAR(1024) | NOT NULL | Ảnh nhân viên |
| 9 | EMAIL | VARCHAR(15) | NOT NULL | Địa chỉ Email |

Bảng nhà cung cấp

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên cột | Kiểu dữ liệu | Not Null | Diễn giải |
| 1 | **MANCC** | VARCHAR(10) | NOT NULL | Mã nhà cung cấp |
| 2 | TENNCC | NVARCHAR(200) | NOT NULL | Tên nhà cung cấp |
| 3 | DIACHINCC | NVARCHAR(200) | NOT NULL | Địa chỉ nhà cung cấp |
| 4 | EMAIL | VARCHAR(15) | NOT NULL | Địa chỉ Email |

Bảng loại sản phẩm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên cột | Kiểu dữ liệu | Not Null | Diễn giải |
| 1 | **MALSP** | VARCHAR(10) | NOT NULL | Mã loại sản phẩm |
| 2 | TENLSP | NVARCHAR(200) | NOT NULL | Tên loại sản phẩm |
| 3 | ANHLSP | VARCHAR(1024) | NOT NULL | Ảnh loại sản phẩm |

Bảng sản phẩm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên cột | Kiểu dữ liệu | Not Null | Diễn giải |
| 1 | **MASP** | VARCHAR(10) | NOT NULL | Mã sản phẩm |
| 2 | MALSP | VARCHAR(10) | NOT NULL | Mã loại sản phẩm |
| 3 | TENSP | NVARCHAR(200) | NOT NULL | Tên sản phẩm |
| 4 | DVT | NVARCHAR(20) | NOT NULL | Đơn vị tính |
| 5 | KICHTHUOC | SMALLINT | NOT NULL | Kích thước |
| 6 | DONGIA | DECIMAL(10,2) | NOT NULL | Đơn giá |
| 7 | MANCC | VARCHAR(10) | NOT NULL | Mã nhà cung cấp |
| 8 | SLTON | INT | NOT NULL | Số lượng tồn kho |
| 9 | CHITIETSP | NTEXT | NOT NULL | Chi tiết sản phẩm |
| 10 | ANHSP | VARCHAR(1024) | NOT NULL | Ảnh sản phẩm |

Bảng hóa đơn

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên cột | Kiểu dữ liệu | Not Null | Diễn giải |
| 1 | **MAHD** | VARCHAR(10) | NOT NULL | Mã hóa đơn |
| 2 | MANV | VARCHAR(10) | NOT NULL | Mã nhân viên |
| 3 | MAKH | VARCHAR(10) | NOT NULL | Mã khách hàng |
| 4 | TGDAT | DATETIME | NOT NULL | Thời gian đặt hàng |
| 5 | TGGIAO | DATETIME | NOT NULL | Thời gian giao hàng |
| 6 | IDTHANHTOAN | INT | NOT NULL | Thanh toán |
| 7 | DATHANHTOAN | BIT | NOT NULL | Đã thanh toán |
| 8 | DAGIAO | BIT | NOT NULL | Đã giao |

Bảng chi tiết hóa đơn

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên cột | Kiểu dữ liệu | Not Null | Diễn giải |
| 1 | SOLUONG | SMALLINT | NOT NULL | Số lượng |
| 2 | **MHD** | VARCHAR(10) | NOT NULL | Mã hóa đơn |
| 3 | **MASP** | VARCHAR(10) | NOT NULL | Mã sản phẩm |

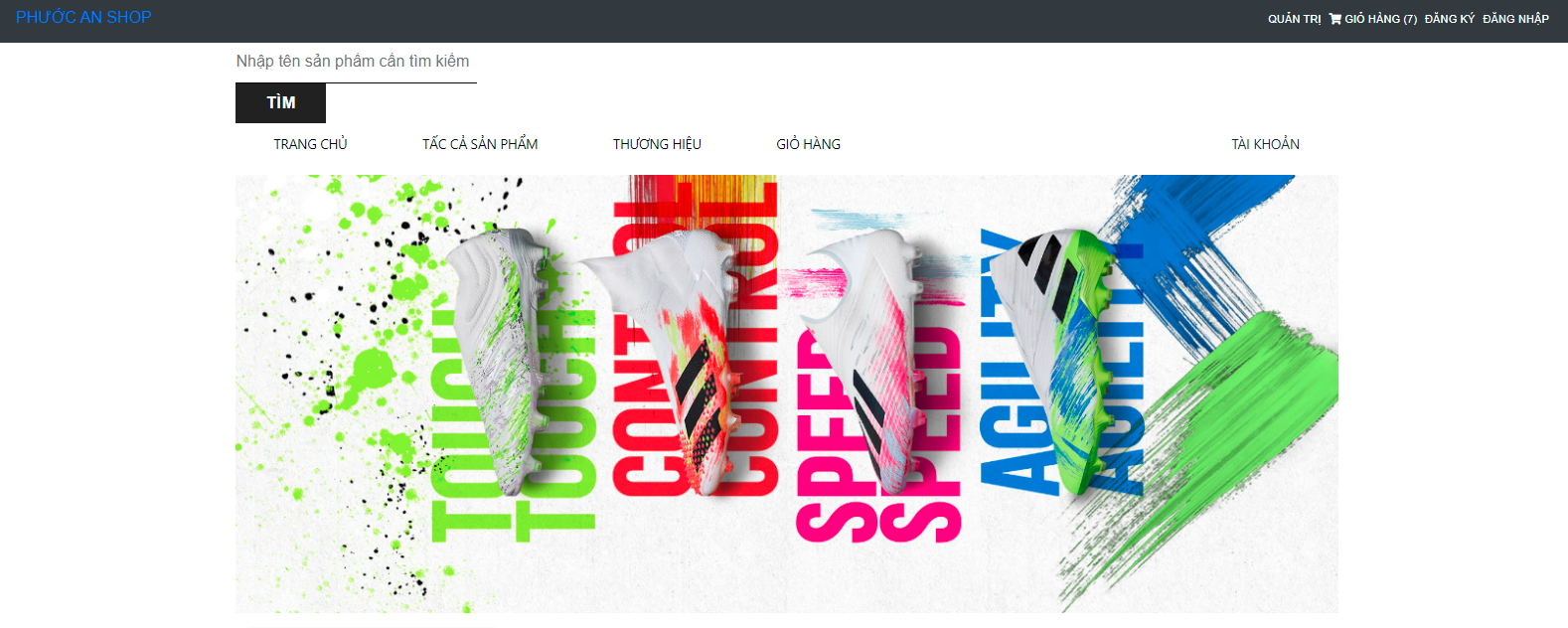
Bảng thanh toán

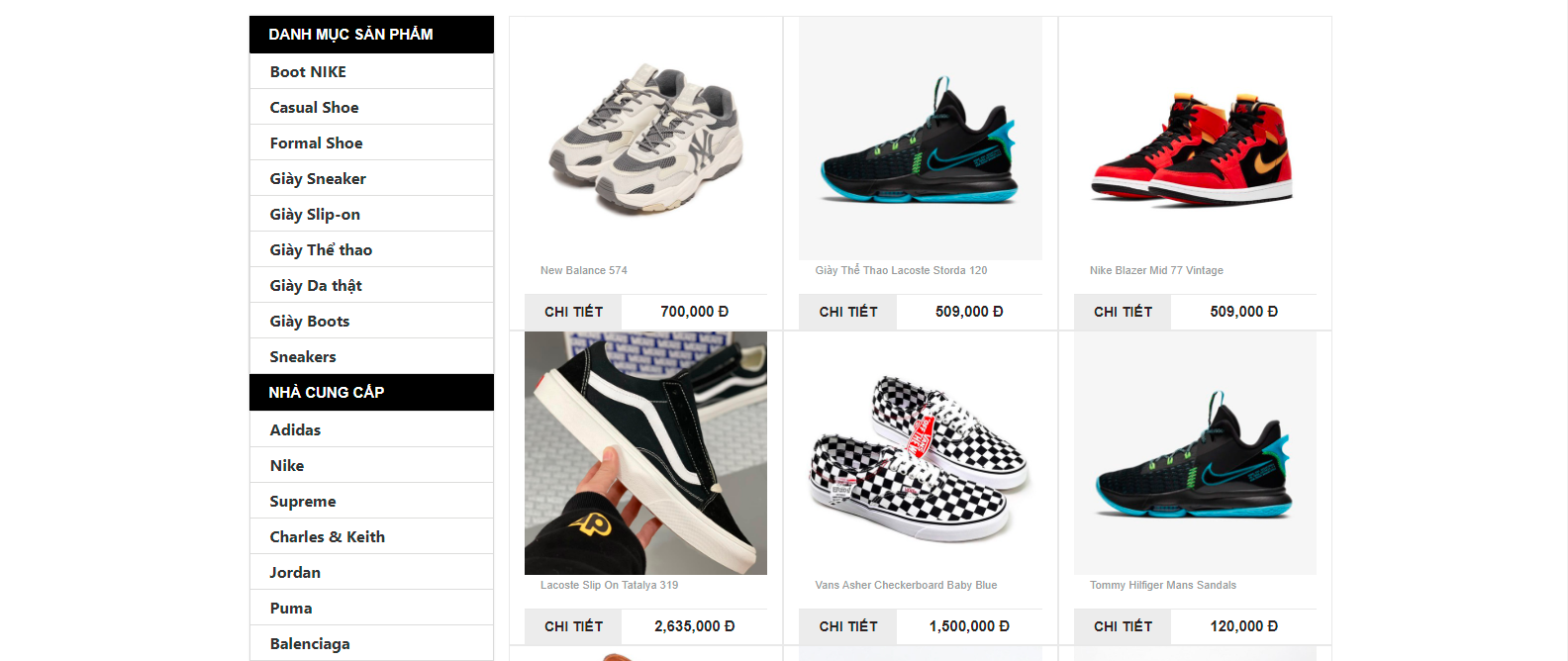
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên cột | Kiểu dữ liệu | Not Null | Diễn giải |
| 1 | **IDTHANHTOAN** | INT | NOT NULL | Thanh toán |
| 2 | HTTHANHTOAN | NAVARCHAR(20) | NOT NULL | Hình thức thanh toán |

# CHƯƠNG IV: TRIỂN KHAI VÀ XÂY DỰNG HỆ THỐNG

## 1. TỔNG QUAN VỀ HỆ THỐNG

## 2. GIAO DIỆN CHÍNH

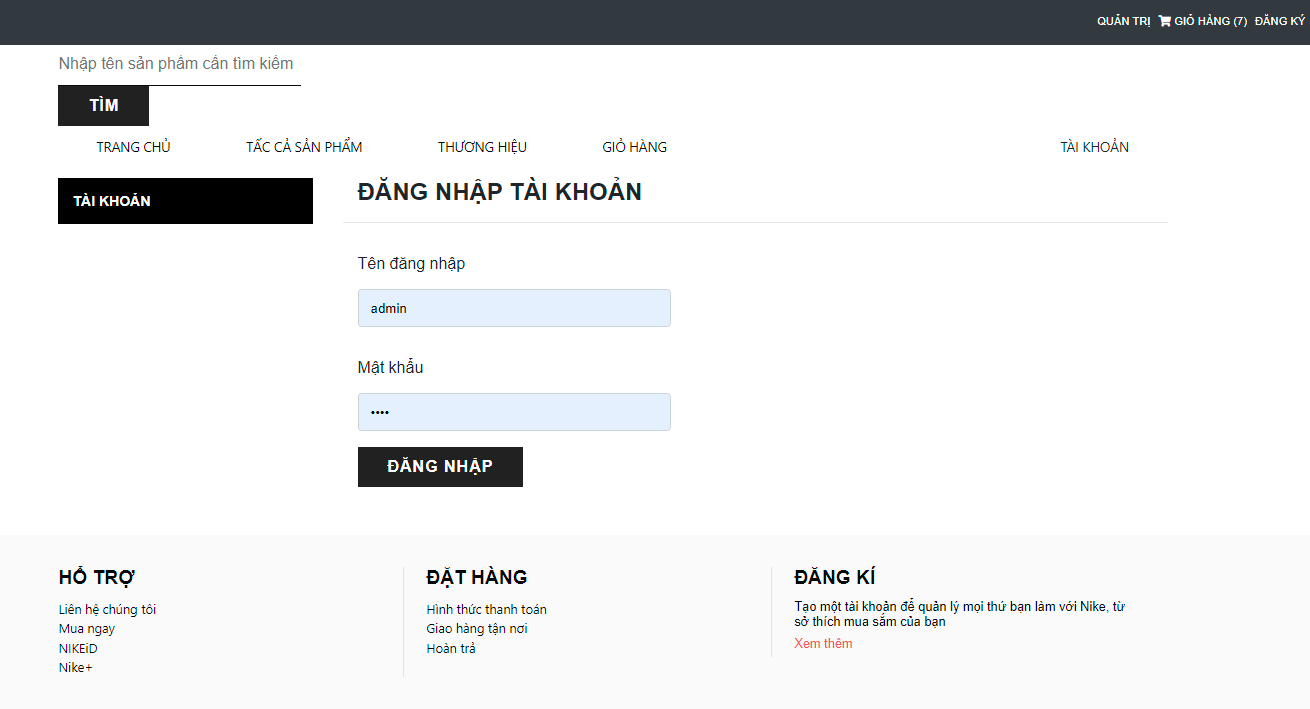




Ảnh Giao diện chính của hệ thống

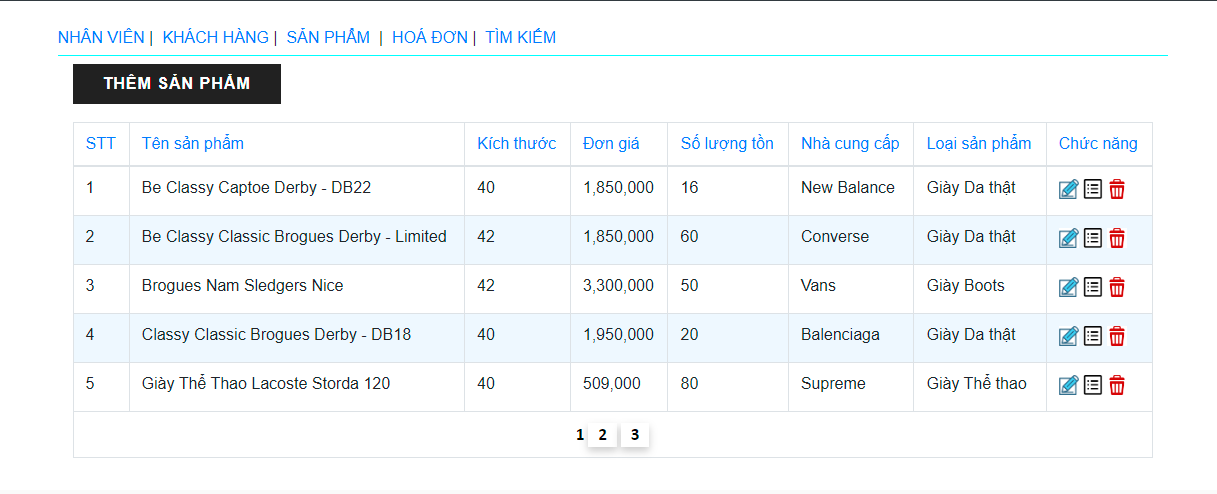
## 3. CHỨC NĂNG CỦA CÁC GIAO DIỆN

## 3.1 Giao diện đăng nhập của quản trị viên



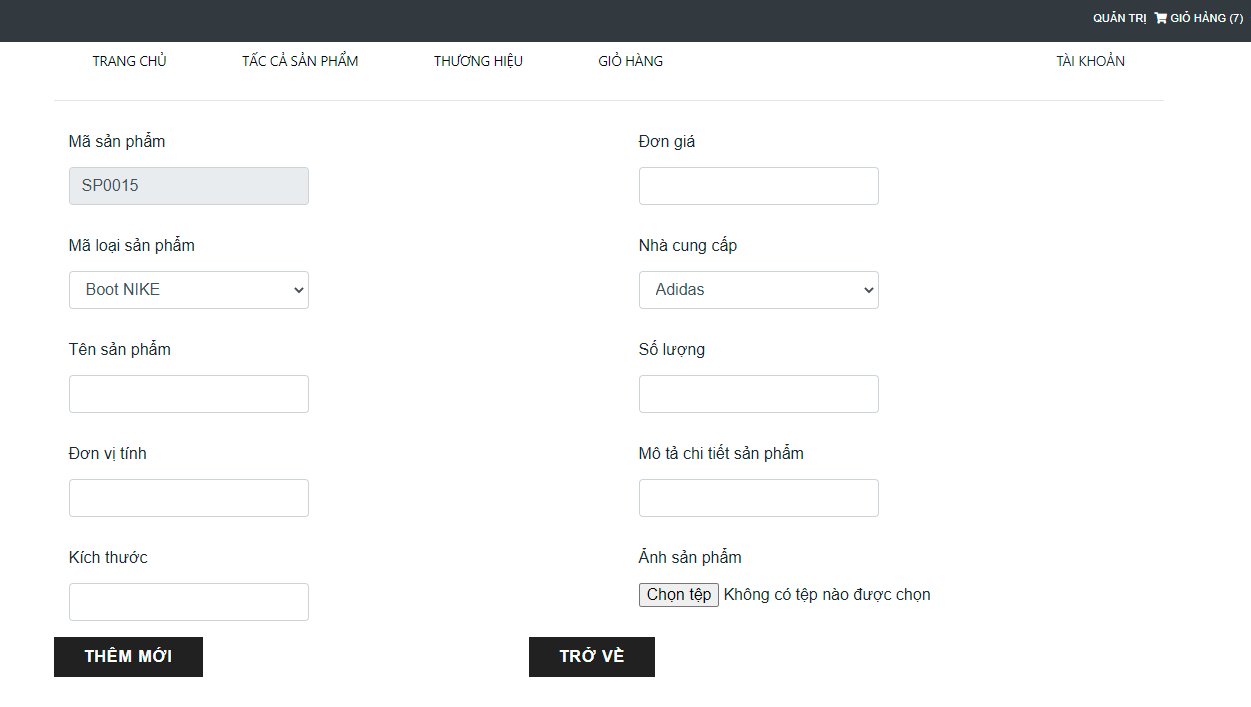
Ảnh Giao diện đăng nhập quản trị viên

## 3.2 Giao diện quản trị sản phẩm của quản trị viên



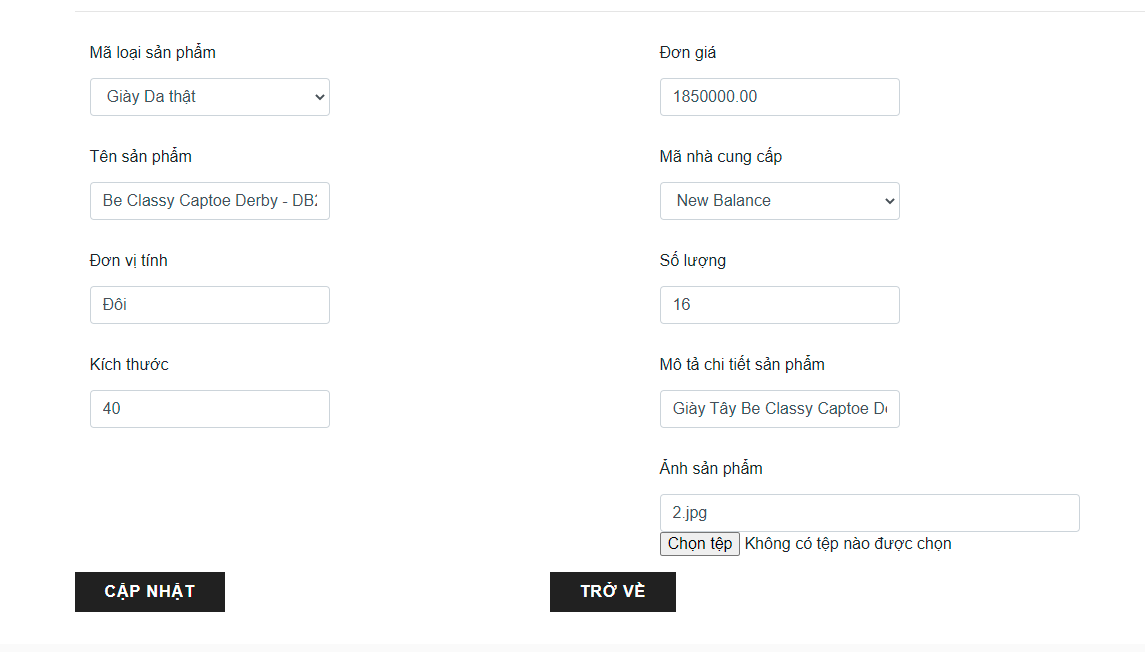
Ảnh Giao diện quản trị sản phẩm

## 3.3 Giao diện thêm sản phẩm mới



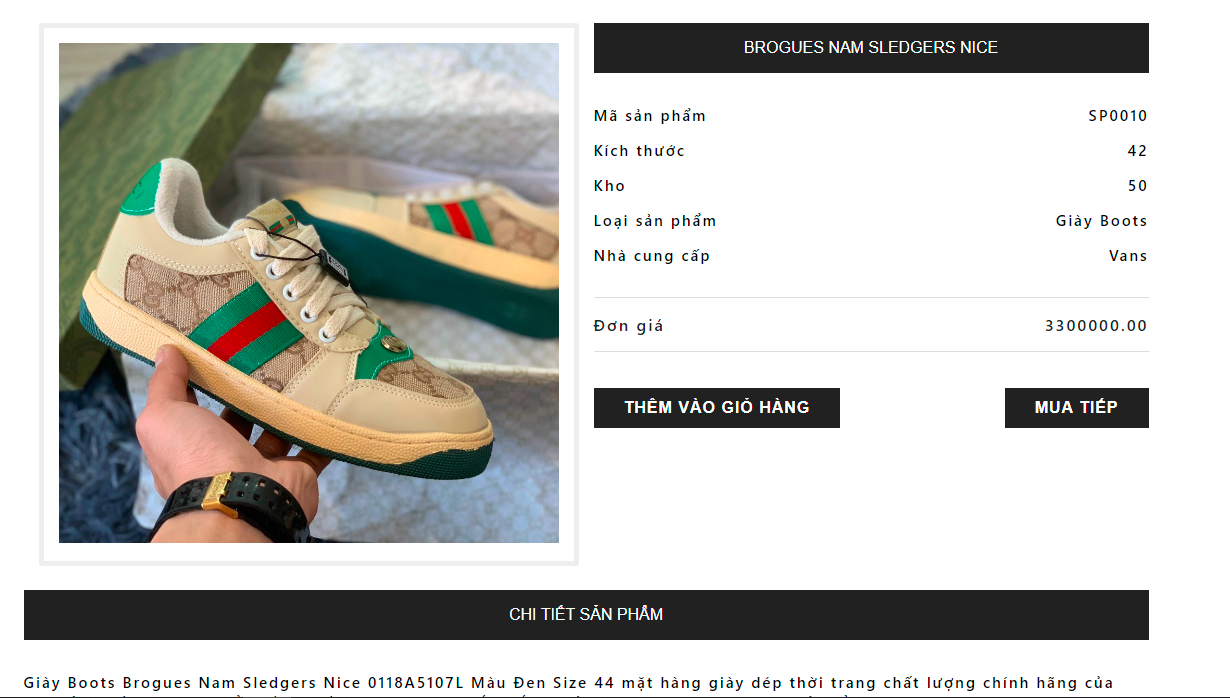
Ảnh Giao diện thêm sản phẩm mới

## 3.4 Giao diện cập nhật sản phẩm



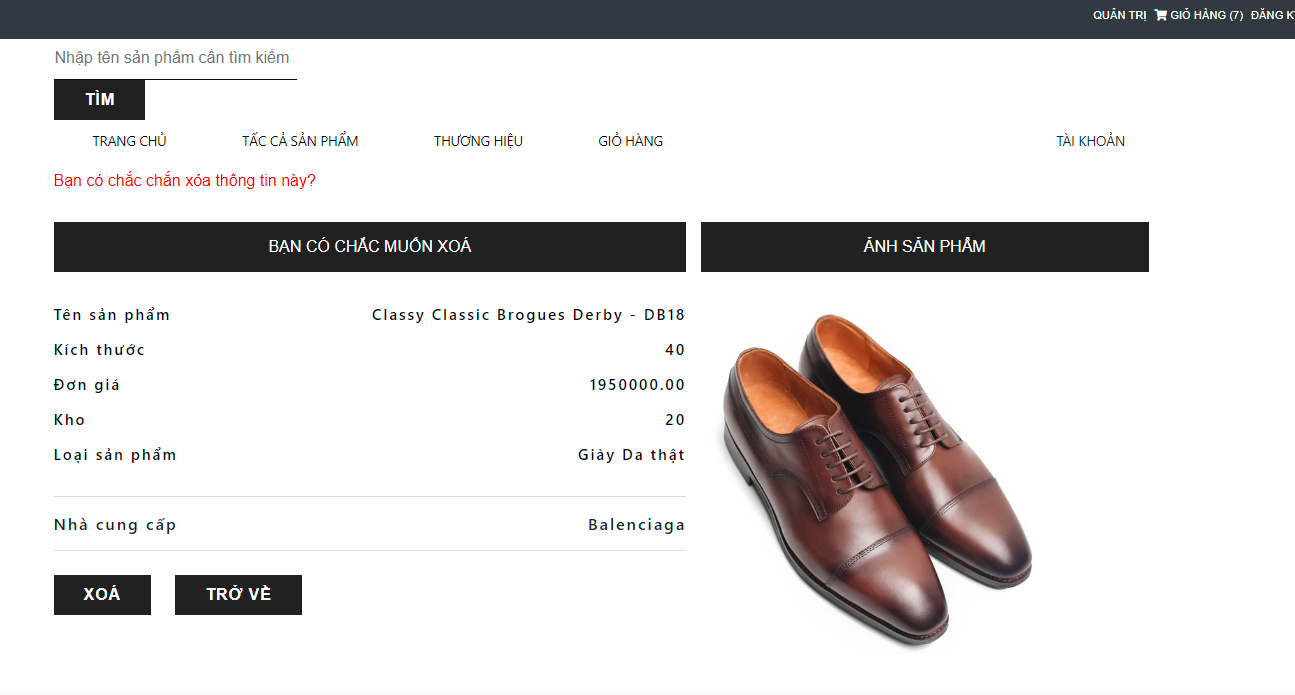
Ảnh Giao diện cập nhật sản phẩm

## 3.5 Giao diện chi tiết sản phẩm



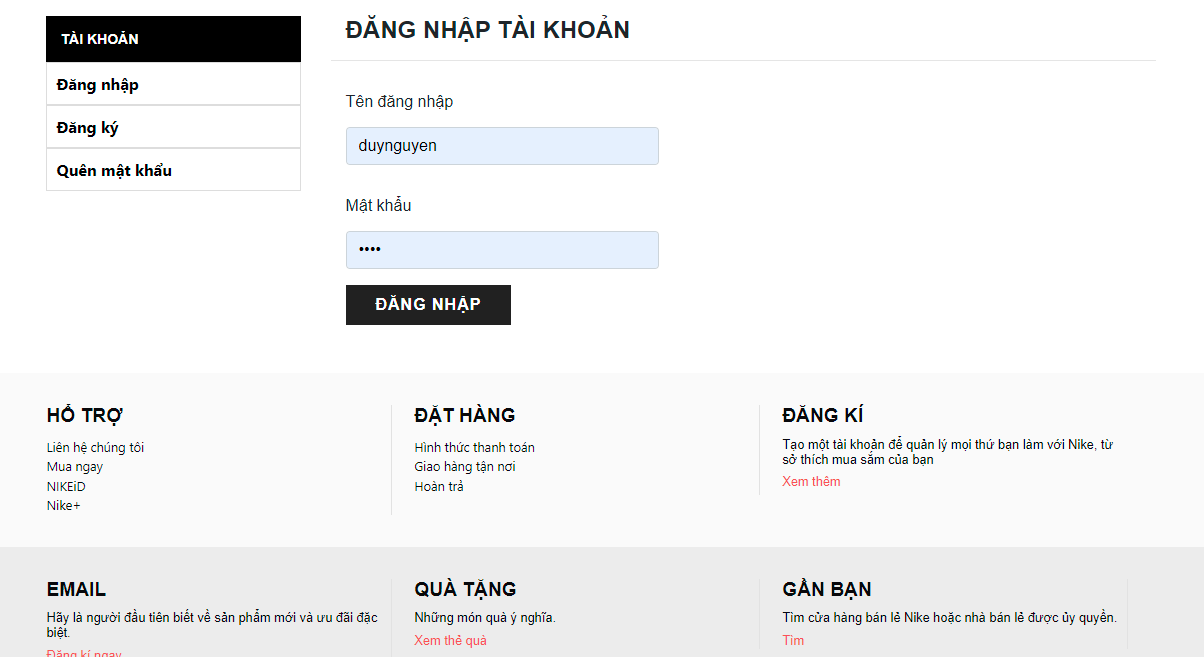
Ảnh Giao diện chi tiết sản phẩm

## 3.6 Giao diện xoá sản phẩm



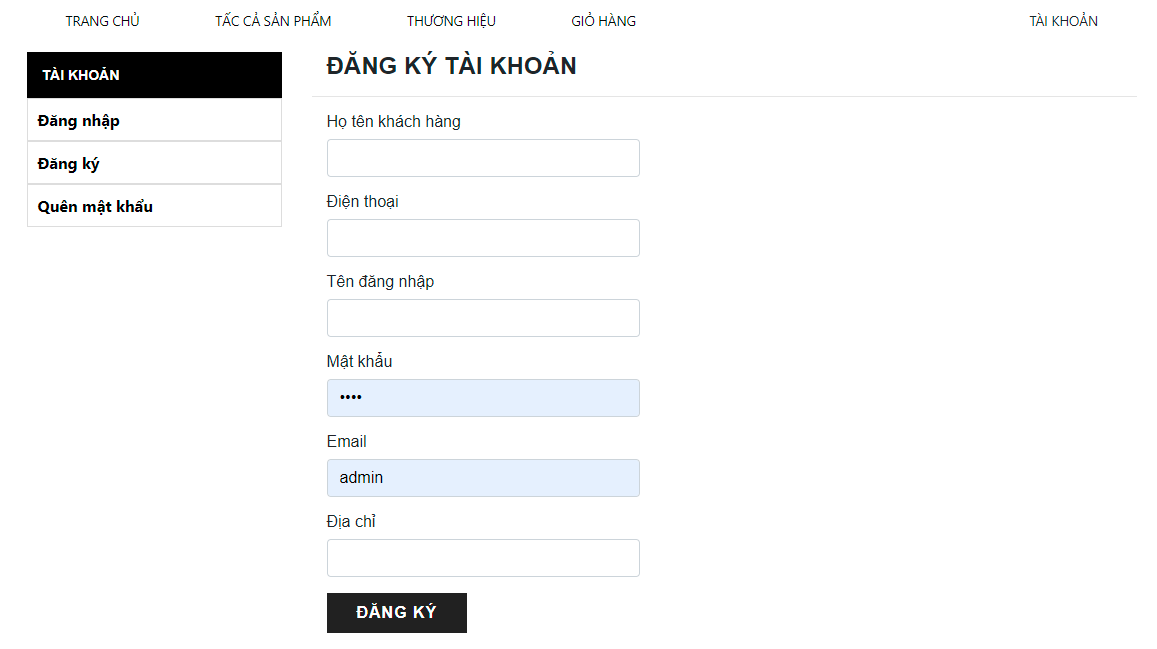
Ảnh Giao diện xoá sản phẩm

## 3.7 Giao diện đăng nhập của người dùng



Ảnh Giao diện đăng nhập người dùng

## 3.8 Giao diện đăng ký tài khoản mới của người dùng



Ảnh Giao diện đăng ký tài khoản của người dùng

# CHƯƠNG V: KẾT LUẬN

## KẾT QUẢ ĐẠT ĐƯỢC

## HẠN CHẾ

# TÀI LIỆU THAM KHẢO