Đại học Đà Nẵng

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT



BÁO CÁO CUỐI KỲ

LẬP TRÌNH C#

Tên đề tài:

TRANG WEBSITE BÁN ĐỒNG HỒ

Sinh viên thực hiện : NGUYỄN VĂN HẢI

Mã sinh viên : 1811505310213

Lớp : 18T2

Lớp HP : 220LTC03

GVHD : LÊ THỊ BÍCH TRA

Đà Nẵng, tháng 06/2021

MỤC LỤC

[Link Github: Source Code chương trình 5](#_Toc75162874)

[1. Thực hiện các yêu cầu cho trang Admin: 5](#_Toc75162875)

[a. Sử dụng template Admin mẫu, hiệu chỉnh lại cho phù hợp, và sử dụng các class css, bootstrap cho các chức năng bên dưới. (1đ) 5](#_Toc75162876)

[b. Thực hiện chức năng Kiểm tra đăng nhập (1đ) 5](#_Toc75162877)

[c. Hiển thị danh sách UserAccount 10](#_Toc75162878)

[+ Phân trang DS tài khoản người dùng (5 user/trang) và tìm kiếm (1đ) 10](#_Toc75162879)

[+ Xóa 1 người dùng có Status là Blocked (1đ) 12](#_Toc75162880)

[d. Hiển thị danh sách sản phẩm 16](#_Toc75162881)

[Yêu cầu: sắp xếp theo Số lượng tăng dần, Đơn giá giảm dần, dưới dạng bảng có các cột (1đ) 16](#_Toc75162882)

[ID || Tên Sản phẩm || Đơn giá || Số lượng || LoaiSP\_ID || Thao tác 16](#_Toc75162883)

[Click vào link Xem chi tiết (ở cột Thao tác) sẽ chuyển đến trang xem được chi tiết của sản phẩm (1đ) 18](#_Toc75162884)

[e. Thiết kế form, thực hiện chức năng thêm 1 sản phẩm: 20](#_Toc75162885)

[Form gồm có các Label kèm các Textbox: ID, Tên sản phẩm, đơn giá, số lượng và Loại Sản phẩm (dropdownlist: đổ dữ liệu của bảng Loại sản phẩm vào). (2đ) 20](#_Toc75162886)

[Yêu cầu thực hiện: 20](#_Toc75162887)

[+ Sử dụng Strongly typed Html Helper cho các thẻ trên form. 20](#_Toc75162888)

[+ Validation dữ liệu trên form (dùng DataAnnotation) 20](#_Toc75162889)

[2. Thực hiện các yêu cầu cho trang Người dùng (2 đ) 27](#_Toc75162890)

[a. Download 1 template bán hàng bất kỳ, hiệu chỉnh lại cho phù hợp với dữ liệu của bảng Sản phẩm. 27](#_Toc75162891)

[b. Đổ dữ liệu của bảng sản phẩm vào trang chủ. 27](#_Toc75162892)

[+ Hiển thị danh mục sản phẩm 27](#_Toc75162893)

[+ Hiển thị Top 8 sản phẩm giá rẽ 29](#_Toc75162894)

[+ Hiển thị danh sách sản phầm theo danh mục 31](#_Toc75162895)

[+ Hiển thị chi tiết sản phẩm 35](#_Toc75162896)

MỤC LỤC HÌNH ẢNH

[*Hình 1: Code giao diện Login* 5](#_Toc75141801)

[*Hình 2: Code LoginModdel xử lý chưa nhập dữ liệu* 5](#_Toc75141802)

[*Hình 3: UserDao Hàm xử lý Login* 6](#_Toc75141803)

[*Hình 4: Encripto Xử lý mã hóa MD5 của mật khẩu* 6](#_Toc75141804)

[*Hình 5: Login Model xử lý đăng nhập* 7](#_Toc75141805)

[*Hình 6: Login Model xử lý đăng xuất* 7](#_Toc75141806)

[*Hình 7: Kết quả chạy: Lỗi chưa nhập dữ liệu* 8](#_Toc75141807)

[*Hình 8: Kết quả chạy: Lỗi sai User name* 8](#_Toc75141808)

[*Hình 9: Kết quả chạy: Lỗi sai mật khẩu* 9](#_Toc75141809)

[*Hình 10: Đăng nhập thành công sẽ vào trang chủ* 9](#_Toc75141810)

[*Hình 11: UserDao: Xử lý Phân trang và tìm kiếm* 10](#_Toc75141811)

[*Hình 12: UserAccountController* 10](#_Toc75141812)

[*Hình 13: Code Giao diện quản lý User* 11](#_Toc75141813)

[*Hình 14: Kết quả* 12](#_Toc75141814)

[*Hình 15: Kết quả Tìm kiếm User Name “hai”* 12](#_Toc75141815)

[*Hình 16: Thẻ a gọi hàm* deleteOneRow() *của JS* 12](#_Toc75141816)

[*Hình 17: UserDao Xử lý xóa người dùng có Status là BLOCKED* 13](#_Toc75141817)

[*Hình 18: Dùng JS để xử lý xóa hàng, Gọi Ajax để thực hiện xóa dữ liệu* 13](#_Toc75141818)

[*Hình 19: Ajax gọi đến hàm* DeleteAJax2() *để thực hiện xóa dữ liệu* 14](#_Toc75141819)

[*Hình 20: Kết quả Xóa người dùng không BLOCKED* 14](#_Toc75141820)

[*Hình 21: Kết quả Xóa người dùng BLOCKED* 15](#_Toc75141821)

[*Hình 22: Truy vấn Danh sách sản phẩn, có tìm kiếm và phân trang* 16](#_Toc75141822)

[*Hình 23: Truy vấn Danh sách sản phẩn, có tìm kiếm và phân trang* 16](#_Toc75141823)

[*Hình 24: Code Giao diện Danh sách Product* 17](#_Toc75141824)

[*Hình 25: Kết quả danh sách sản phẩm* 18](#_Toc75141825)

[*Hình 26: ActionLink đến Action DetailProduct* 18](#_Toc75141826)

[*Hình 27: ProductDao truy vấn Detail Product bằng ID* 18](#_Toc75141827)

[*Hình 28: Product Controller Trả kết quả về View Deatil* 18](#_Toc75141828)

[*Hình 29: Code Giao diện của Chi tiết Sản phẩm* 19](#_Toc75141829)

[*Hình 30: Kết quả* 20](#_Toc75141830)

[*Hình 31: Danh Thể loại để sử dụng cho* dropdownlist 20](#_Toc75141831)

[*Hình 32: ProductController* 20](#_Toc75141832)

[*Hình 33: Tạo viewBag là DS Category* 21](#_Toc75141833)

[*Hình 34: ProductController: Xử lý Insert Sản phẩm* 21](#_Toc75141834)

[*Hình 35: Code Giao diện Thêm sản phẩm* 22](#_Toc75141835)

[*Hình 36: Danh sách ản phẩm ban đầu* 22](#_Toc75141836)

[*Hình 37: Form thêm sản phẩm* 23](#_Toc75141837)

[*Hình 38: Kết quả sau khi thêm* 23](#_Toc75141838)

[*Hình 39:* Validation *khi Chưa nhập tên (Bắt buộc)* 24](#_Toc75141839)

[*Hình 40:* Validation *khi nhập ký tự vào Price và Quantty* 24](#_Toc75141840)

[*Hình 41:* Update Sản phẩm 24](#_Toc75141841)

[*Hình 42:* Update Sản phẩm 25](#_Toc75141842)

[*Hình 43: Code Giao diện* Update Sản phẩm 26](#_Toc75141843)

[*Hình 43: Kết quả* 26](#_Toc75141844)

[*Hình 45: Category với các chức năng tương tự* 27](#_Toc75141845)

[*Hình 46: CategoryDao Truy vấn danh sách Danh mục* 27](#_Toc75141846)

[*Hình 47: Menu được tách ra nên RenderAction để gọi lại Menu* 27](#_Toc75141847)

[*Hình 48: Xử lý Mainmenu* 28](#_Toc75141848)

[*Hình 49: Code Giao diện MainMenu* 28](#_Toc75141849)

[*Hình 50: Kết quả* 29](#_Toc75141850)

[*Hình 51: Truy vấn Top Sản phẩm* 29](#_Toc75141851)

[*Hình 52: HomeController trả kết quả về View* 29](#_Toc75141852)

[*Hình 53: Code Giao diện* 30](#_Toc75141853)

[*Hình 54: Kết quả* 30](#_Toc75141854)

[*Hình 55: Truy vấn tất cả sản phẩm* 31](#_Toc75141855)

[*Hình 56: Truy vấn sản phẩm the loại sản phẩm* 31](#_Toc75141856)

[*Hình 57: Sử dụng ViewBag để Binding dữ liệu* 31](#_Toc75141857)

[*Hình 58: Click All Product hoặc các thể loại để đến Danh sách sản phẩm* 32](#_Toc75141858)

[*Hình 59: Kết quả All Product* 33](#_Toc75141859)

[*Hình 60: Kết quả thể loại ORIENT* 34](#_Toc75141860)

[*Hình 61: Kết quả thể loại SEIKO* 35](#_Toc75141861)

[*Hình 62: Truy vấn chi tiết sản phẩm theo ID sản phẩm* 35](#_Toc75141862)

[*Hình 63: ProductControler Action Detail Xử lý* 36](#_Toc75141863)

[*Hình 64: Code Giao diện* 36](#_Toc75141864)

[*Hình 65: Kết quả chi tiết sản phẩm có ID 15* 37](#_Toc75141865)

[*Hình 66: Kết quả chi tiết sản phẩm có ID 22* 37](#_Toc75141866)

## Link Github: [Source Code chương trình](https://github.com/vanhai260300/LTC-_NguyenVanHai)

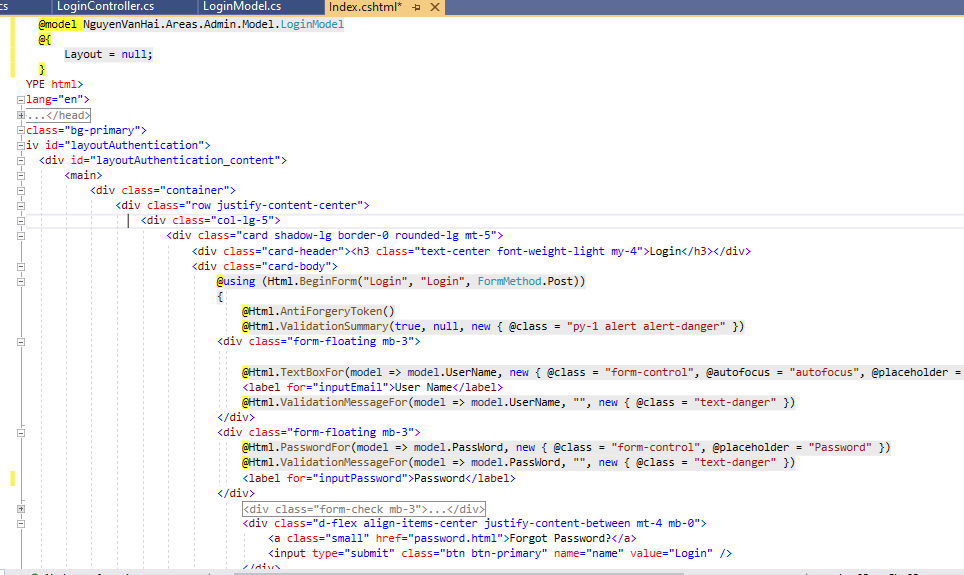
## Thực hiện các yêu cầu cho trang Admin:

## Sử dụng template Admin mẫu, hiệu chỉnh lại cho phù hợp, và sử dụng các class css, bootstrap cho các chức năng bên dưới. (1đ)

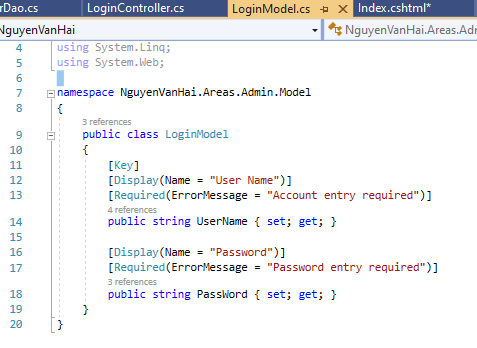
Link Tải Template Admin: <https://startbootstrap.com/template/sb-admin>

Code và hình ảnh được trình bày ở các yêu cầu tiếp theo.

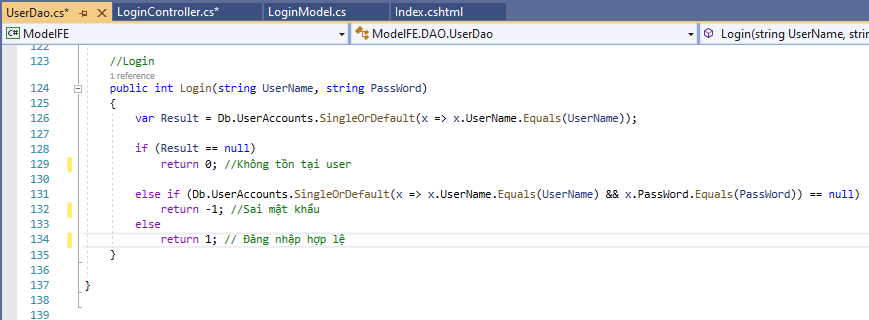
## Thực hiện chức năng Kiểm tra đăng nhập (1đ)



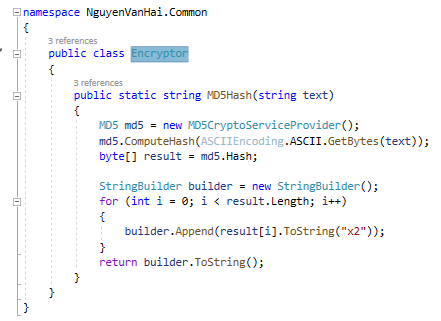
*Hình 1: Code giao diện Login*



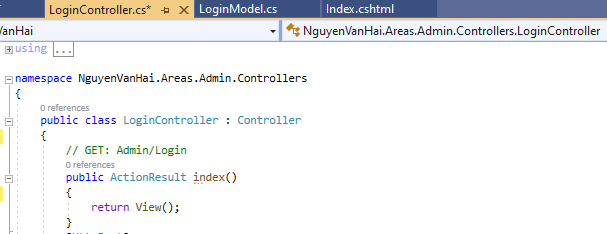
*Hình 2: Code LoginModdel xử lý chưa nhập dữ liệu*



*Hình 3: UserDao Hàm xử lý Login*

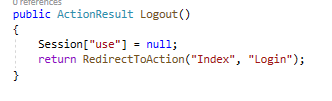


*Hình 4: Encripto Xử lý mã hóa MD5 của mật khẩu*

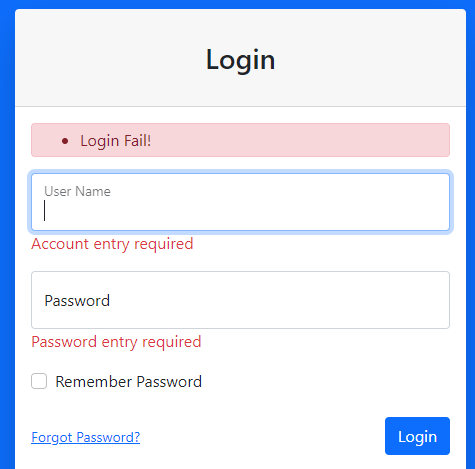




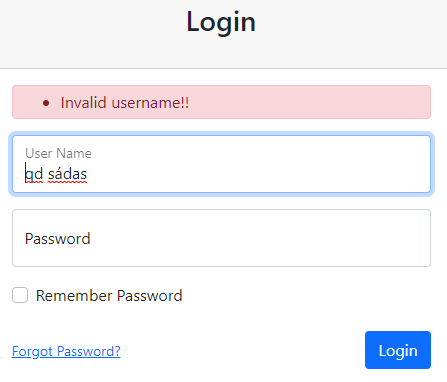
*Hình 5: Login Model xử lý đăng nhập*



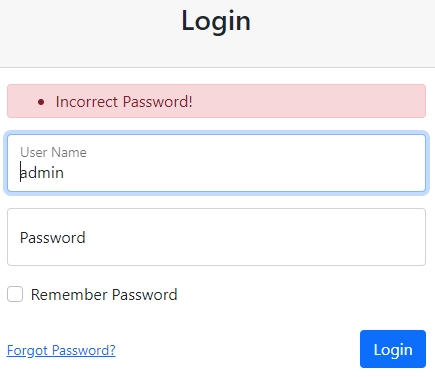
*Hình 6: Login Model xử lý đăng xuất*



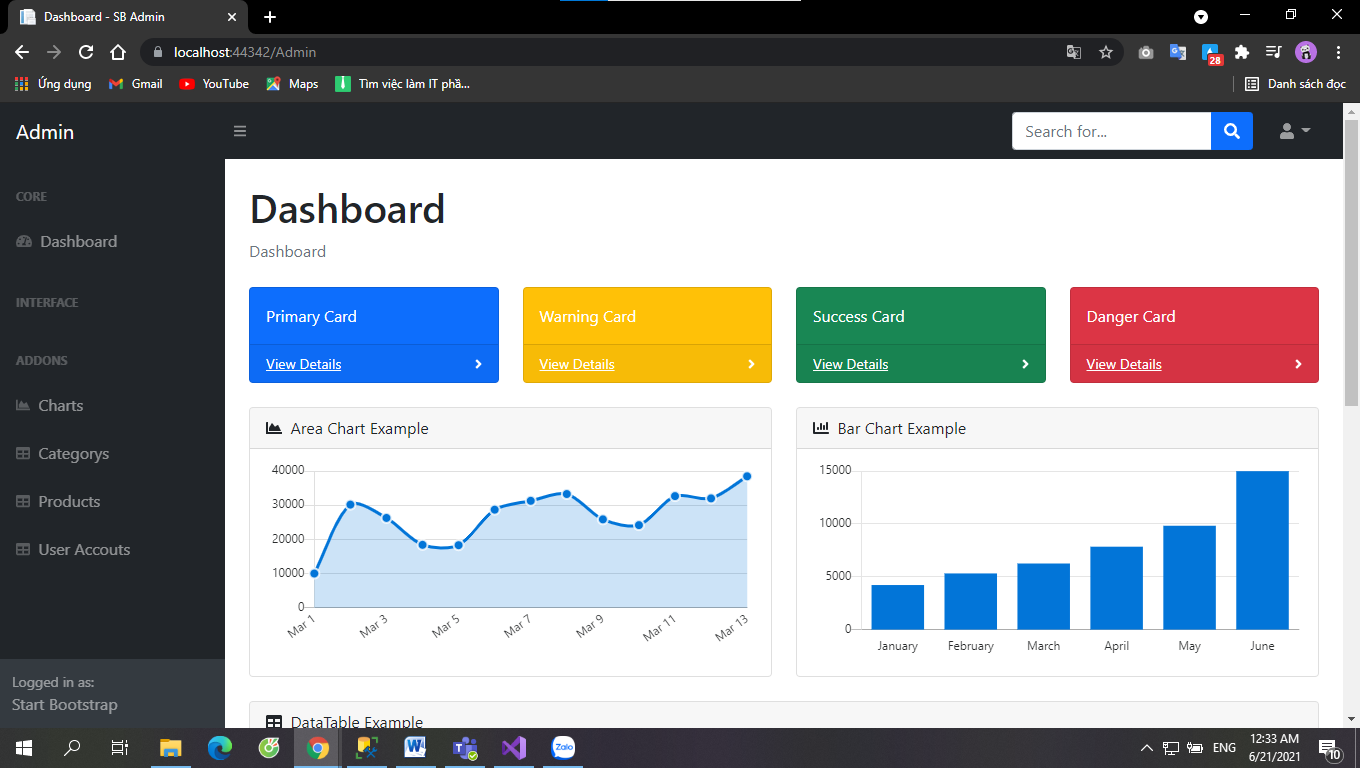
*Hình 7: Kết quả chạy: Lỗi chưa nhập dữ liệu*



*Hình 8: Kết quả chạy: Lỗi sai User name*



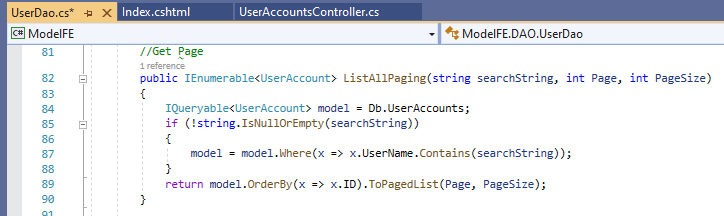
*Hình 9: Kết quả chạy: Lỗi sai mật khẩu*



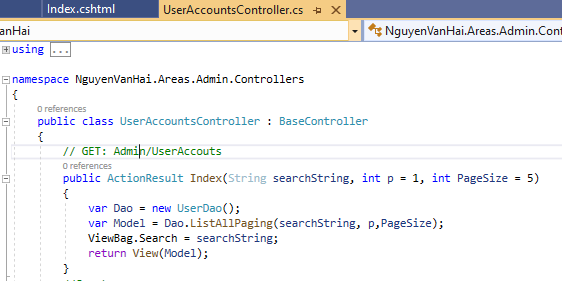
*Hình 10: Đăng nhập thành công sẽ vào trang chủ*

## Hiển thị danh sách UserAccount

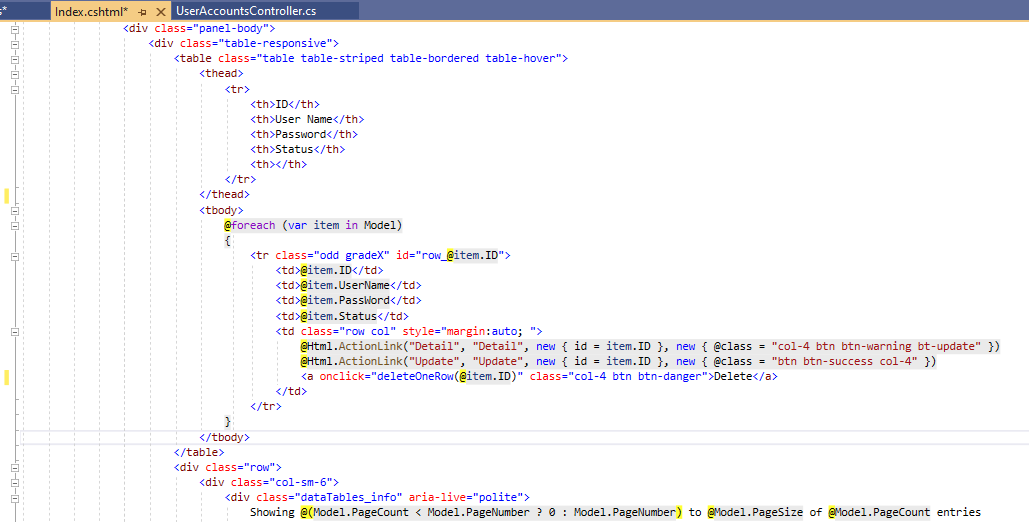
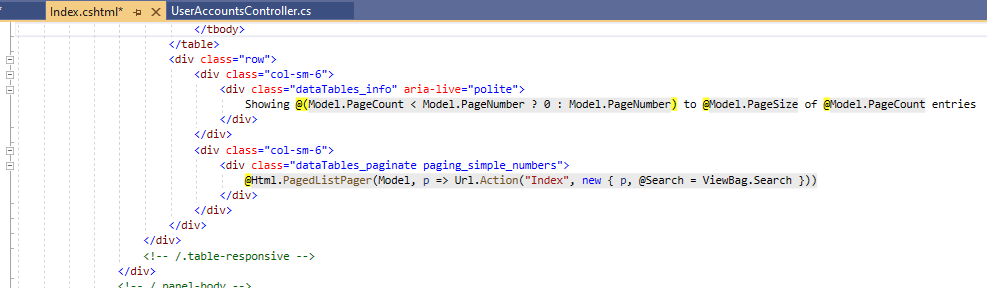
### Phân trang DS tài khoản người dùng (5 user/trang) và tìm kiếm (1đ)



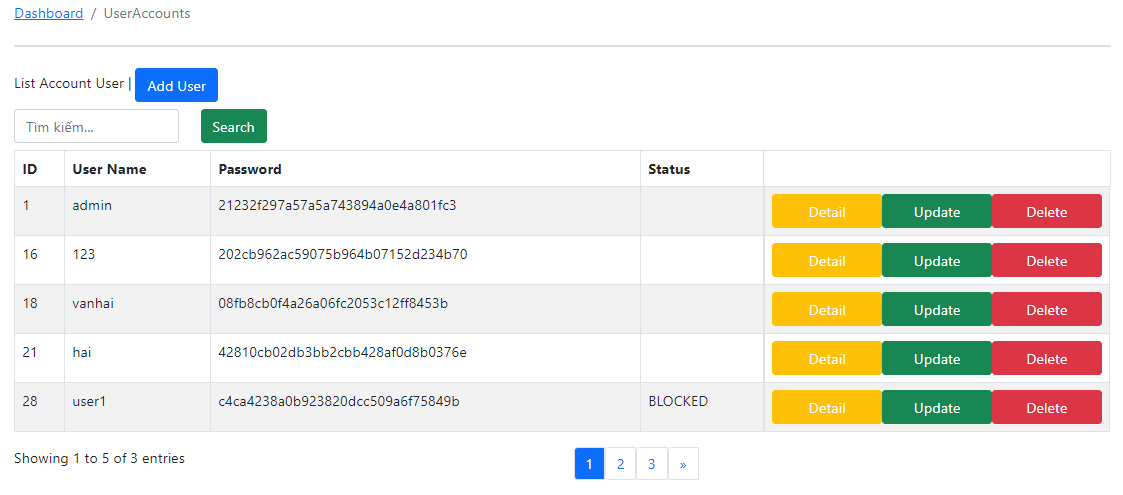
*Hình 11: UserDao: Xử lý Phân trang và tìm kiếm*



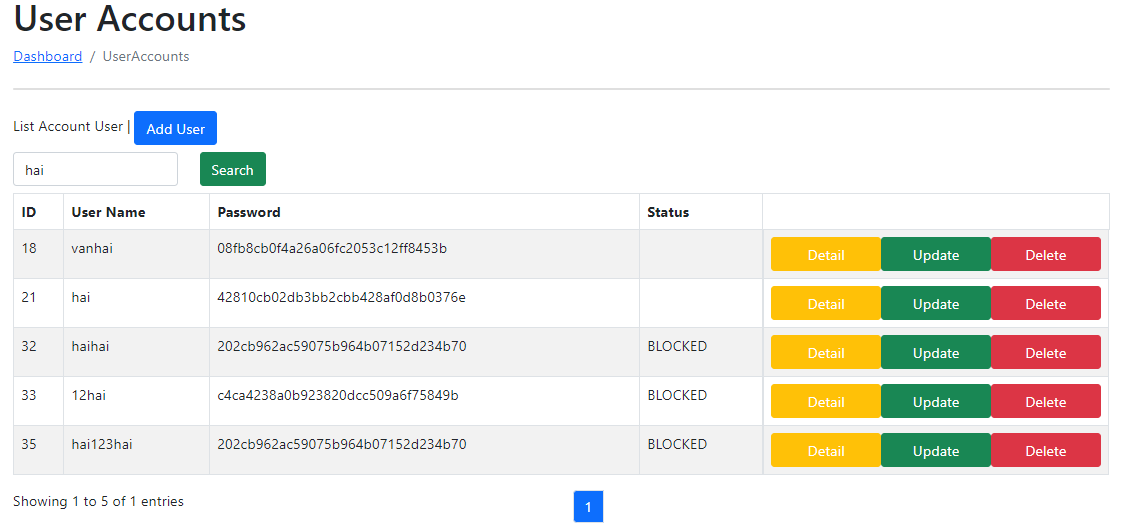
*Hình 12: UserAccountController*

*Hình 13: Code Giao diện quản lý User*

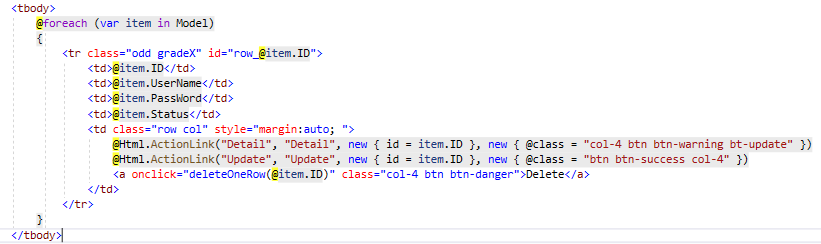


*Hình 14: Kết quả*

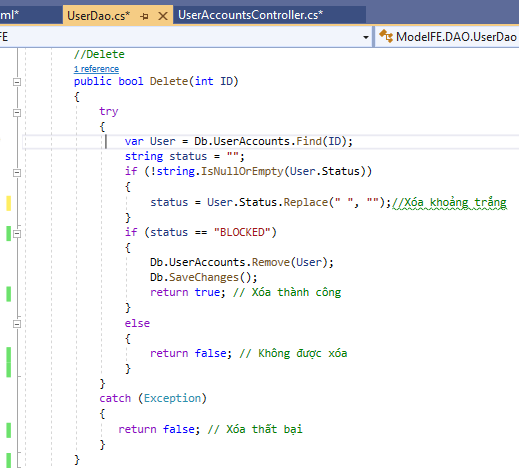


*Hình 15: Kết quả Tìm kiếm User Name “hai”*

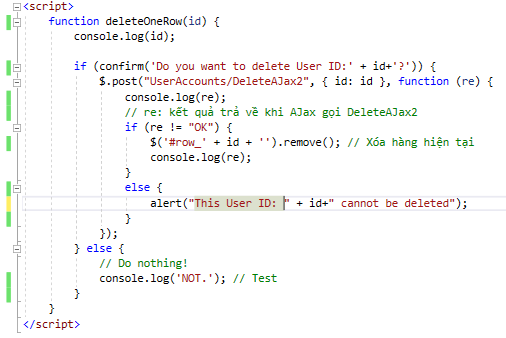
### Xóa 1 người dùng có Status là Blocked (1đ)



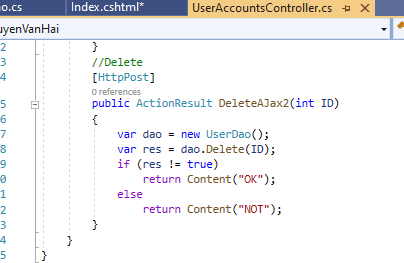
*Hình 16: Thẻ a gọi hàm* deleteOneRow() *của JS*



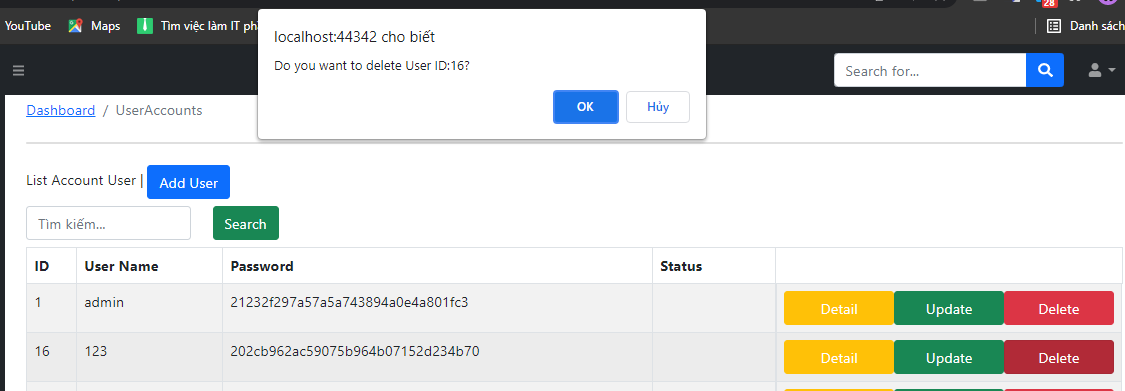
*Hình 17: UserDao Xử lý xóa người dùng có Status là BLOCKED*

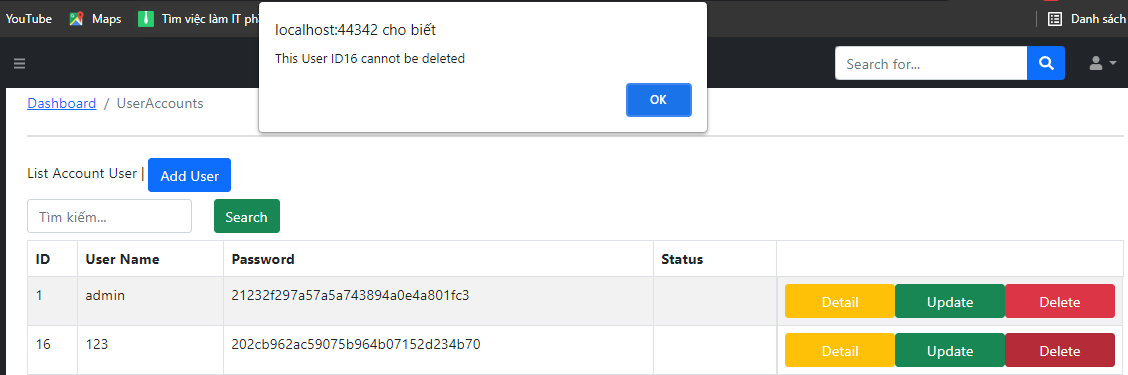


*Hình 18: Dùng JS để xử lý xóa hàng, Gọi Ajax để thực hiện xóa dữ liệu*

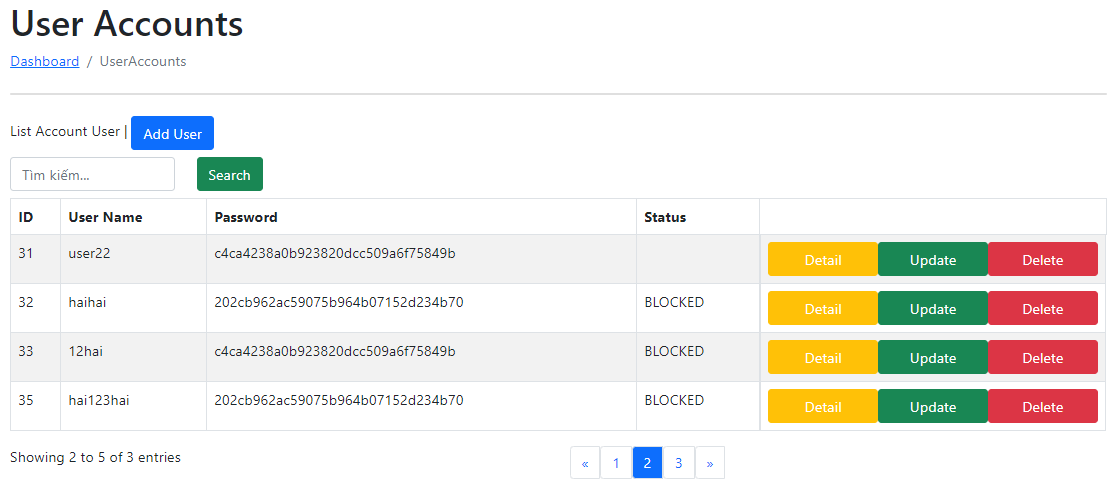
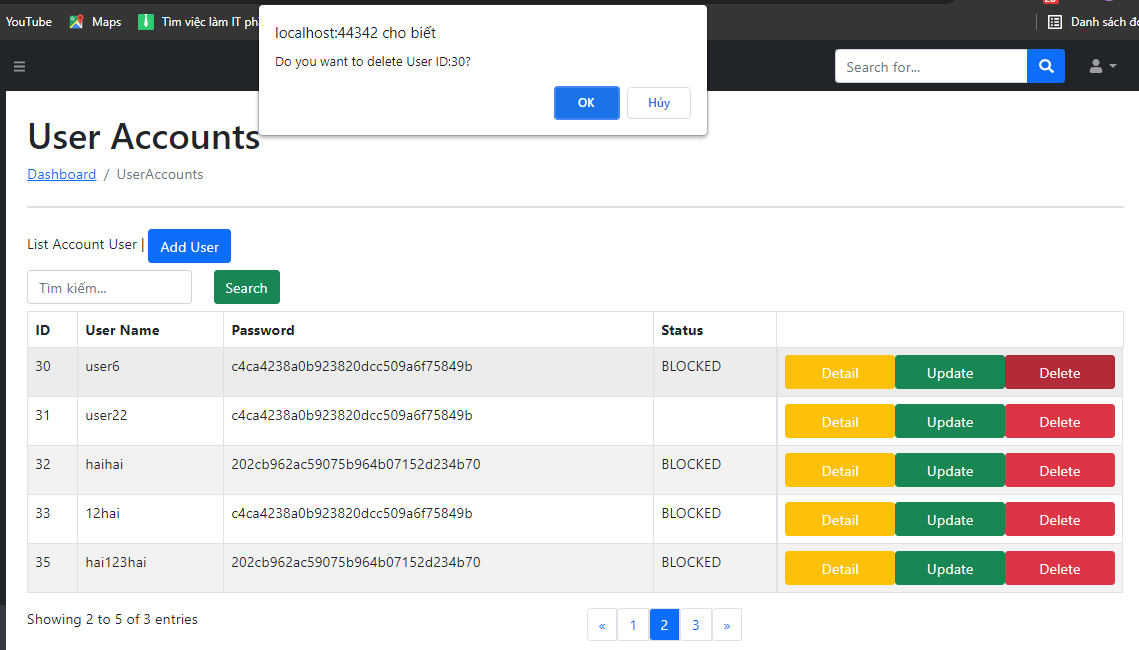


*Hình 19: Ajax gọi đến hàm* DeleteAJax2() *để thực hiện xóa dữ liệu*





*Hình 20: Kết quả Xóa người dùng không BLOCKED*

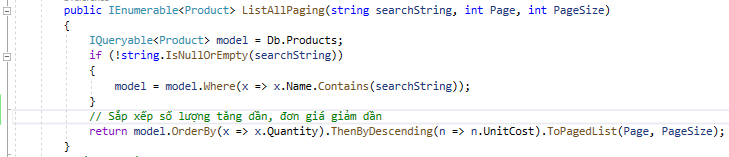


*Hình 21: Kết quả Xóa người dùng BLOCKED*

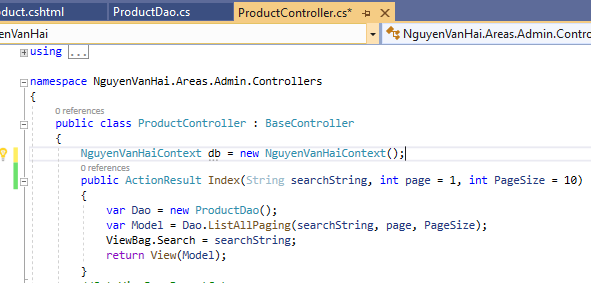
## Hiển thị danh sách sản phẩm

## Yêu cầu: sắp xếp theo Số lượng tăng dần, Đơn giá giảm dần, dưới dạng bảng có các cột (1đ)

## ID || Tên Sản phẩm || Đơn giá || Số lượng || LoaiSP\_ID || Thao tác

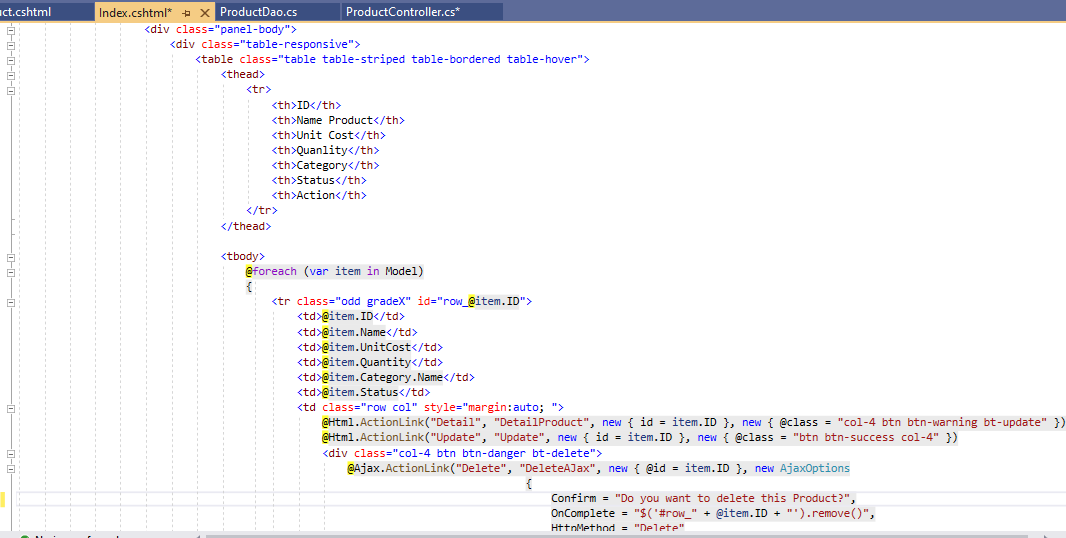


*Hình 22: Truy vấn Danh sách sản phẩn, có tìm kiếm và phân trang*

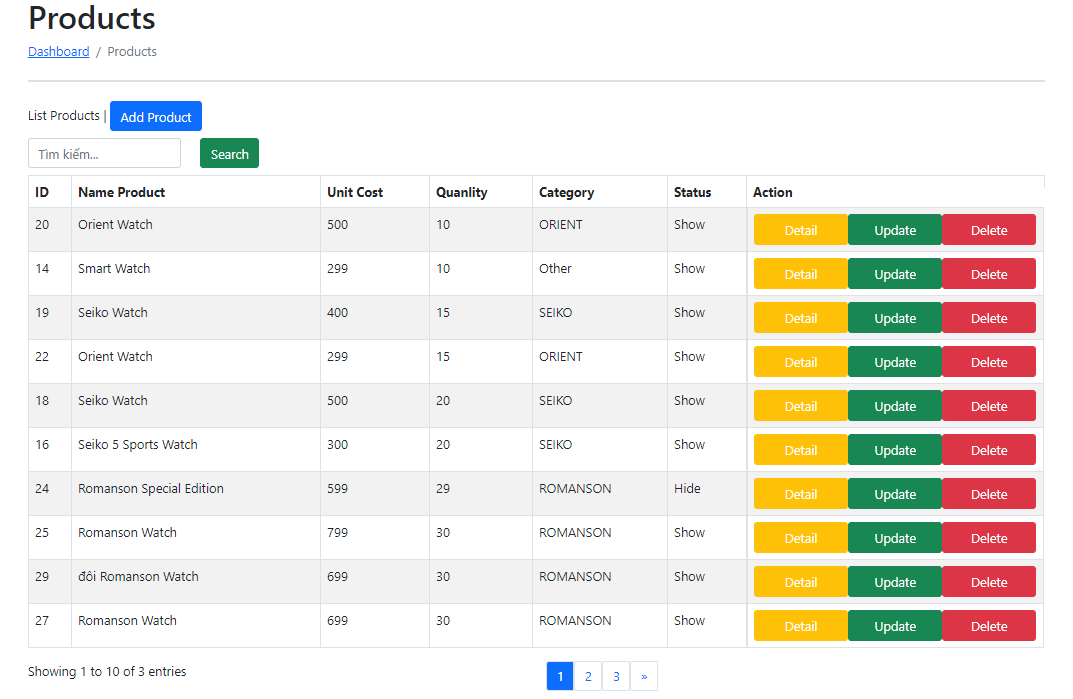


*Hình 23: Truy vấn Danh sách sản phẩn, có tìm kiếm và phân trang*



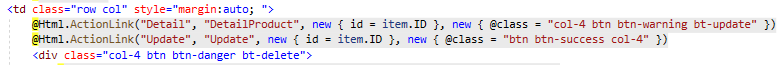


*Hình 24: Code Giao diện Danh sách Product*

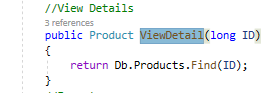


*Hình 25: Kết quả danh sách sản phẩm*

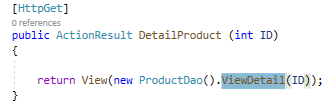
## Click vào link Xem chi tiết (ở cột Thao tác) sẽ chuyển đến trang xem được chi tiết của sản phẩm (1đ)



*Hình 26: ActionLink đến Action DetailProduct*



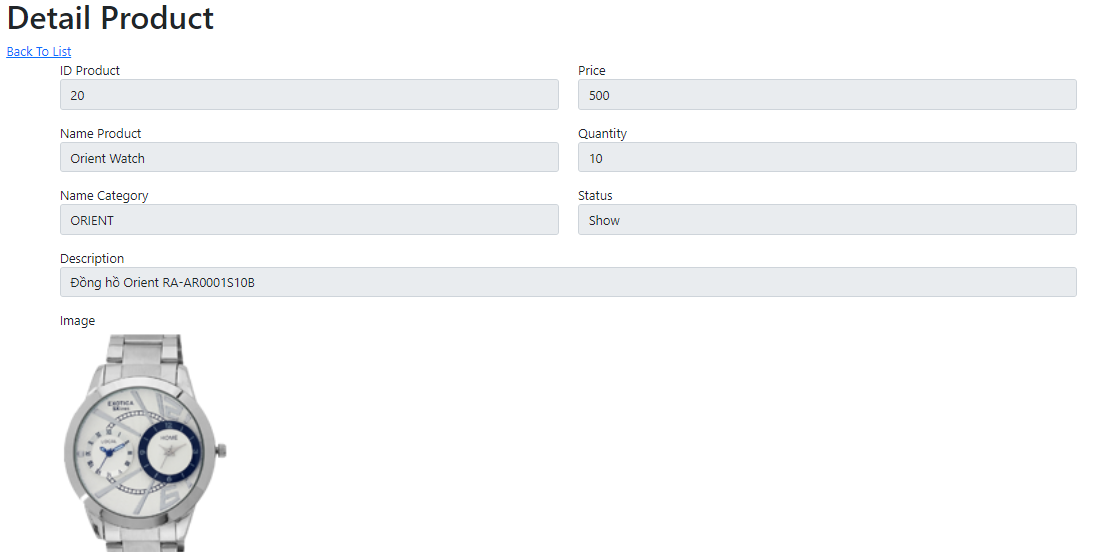
*Hình 27: ProductDao truy vấn Detail Product bằng ID*



*Hình 28: Product Controller Trả kết quả về View Deatil*

*Hình 29: Code Giao diện của Chi tiết Sản phẩm*



*Hình 30: Kết quả*

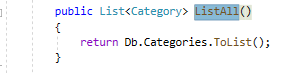
## Thiết kế form, thực hiện chức năng thêm 1 sản phẩm:

## Form gồm có các Label kèm các Textbox: ID, Tên sản phẩm, đơn giá, số lượng và Loại Sản phẩm (dropdownlist: đổ dữ liệu của bảng Loại sản phẩm vào). (2đ)

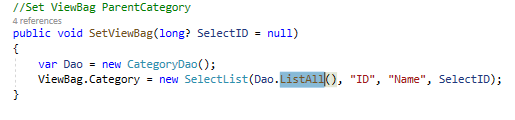
## Yêu cầu thực hiện:

### Sử dụng Strongly typed Html Helper cho các thẻ trên form.

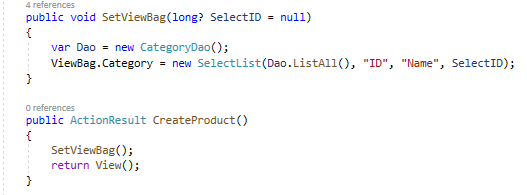
### Validation dữ liệu trên form (dùng DataAnnotation)



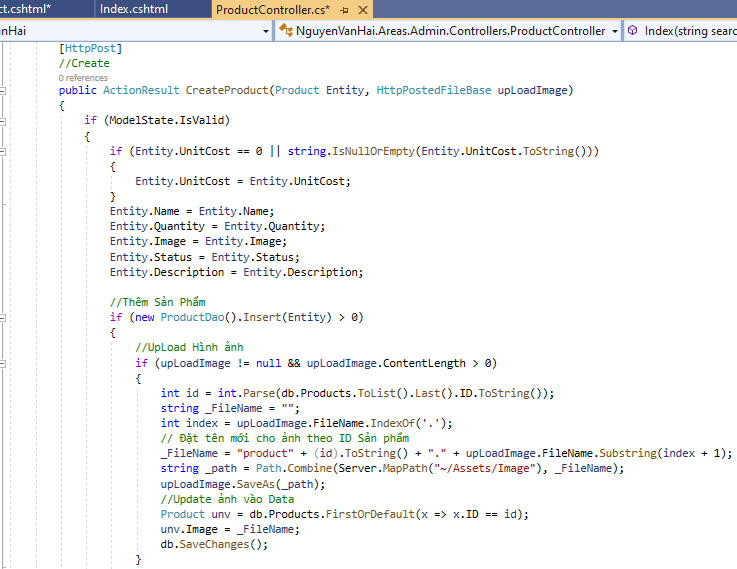
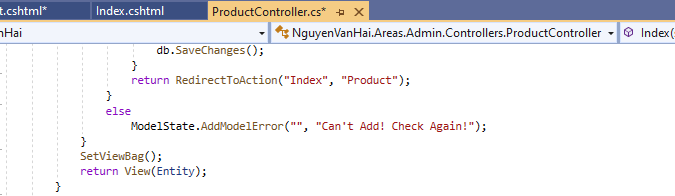
*Hình 31: Danh Thể loại để sử dụng cho* dropdownlist



*Hình 32: ProductController*



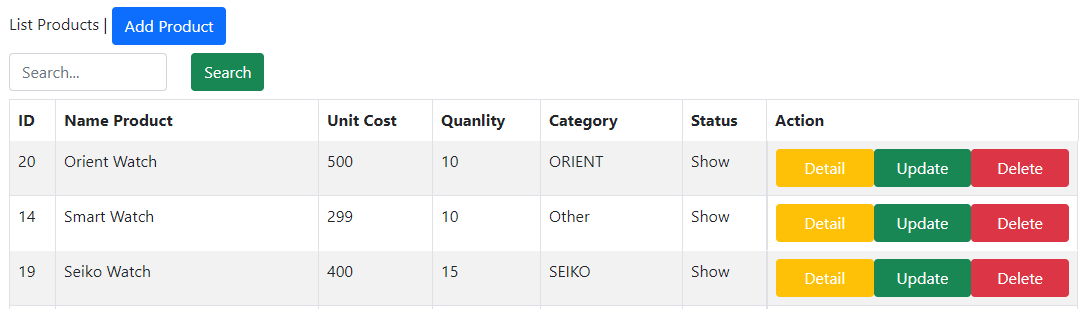
*Hình 33: Tạo viewBag là DS Category*

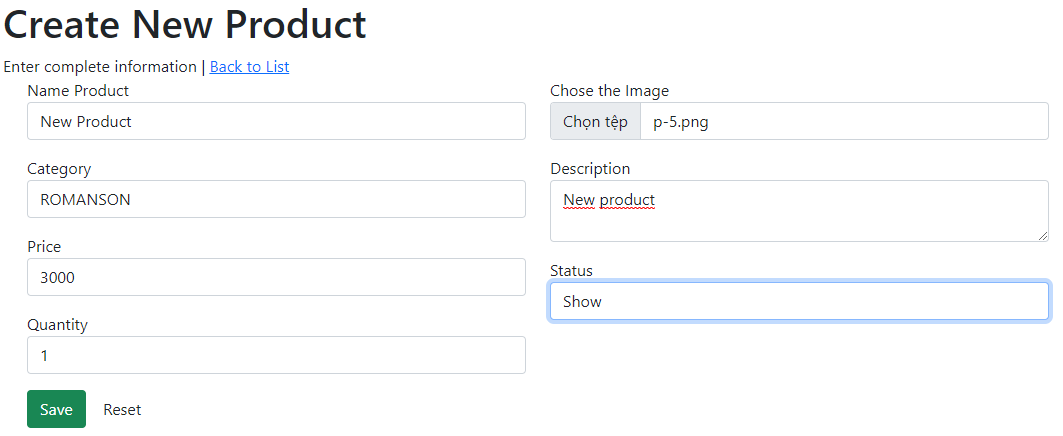
*Hình 34: ProductController: Xử lý Insert Sản phẩm*

* *

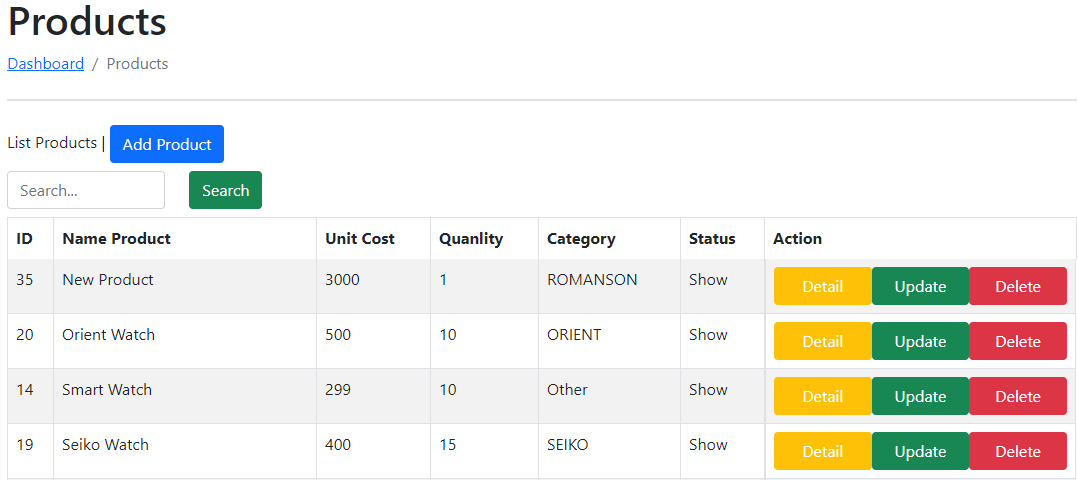
*Hình 35: Code Giao diện Thêm sản phẩm*



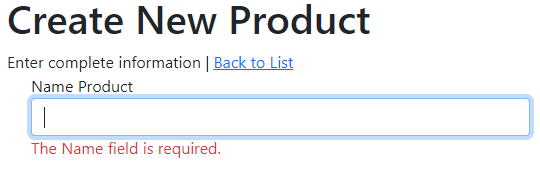
*Hình 36: Danh sách ản phẩm ban đầu*

** **

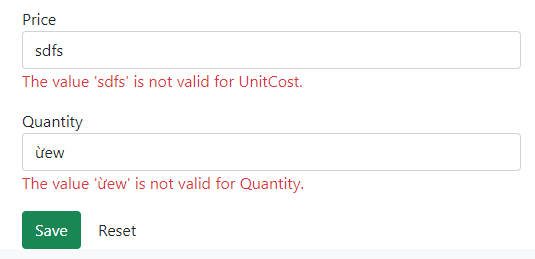
*Hình 37: Form thêm sản phẩm*



*Hình 38: Kết quả sau khi thêm*

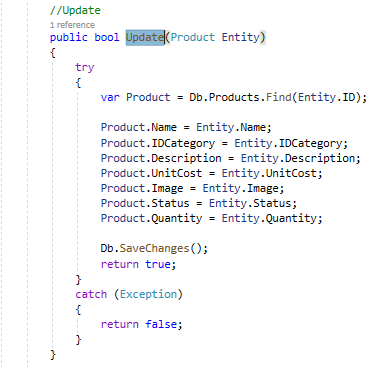


*Hình 39:* Validation *khi Chưa nhập tên (Bắt buộc)*

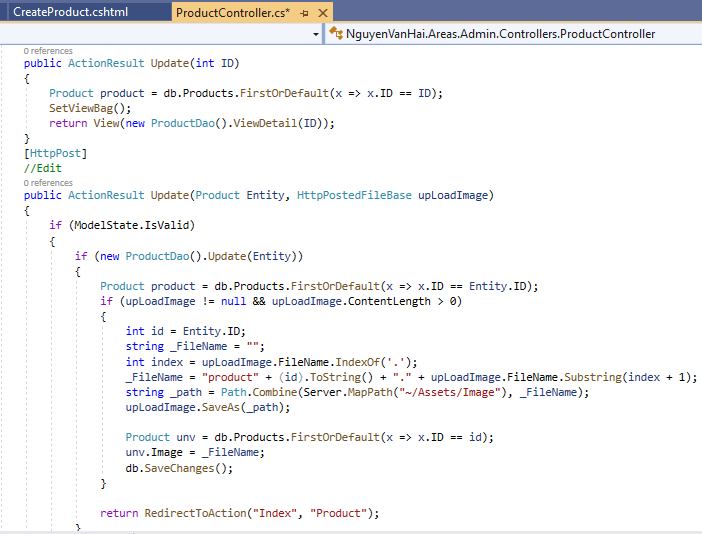


*Hình 40:* Validation *khi nhập ký tự vào Price và Quantty*

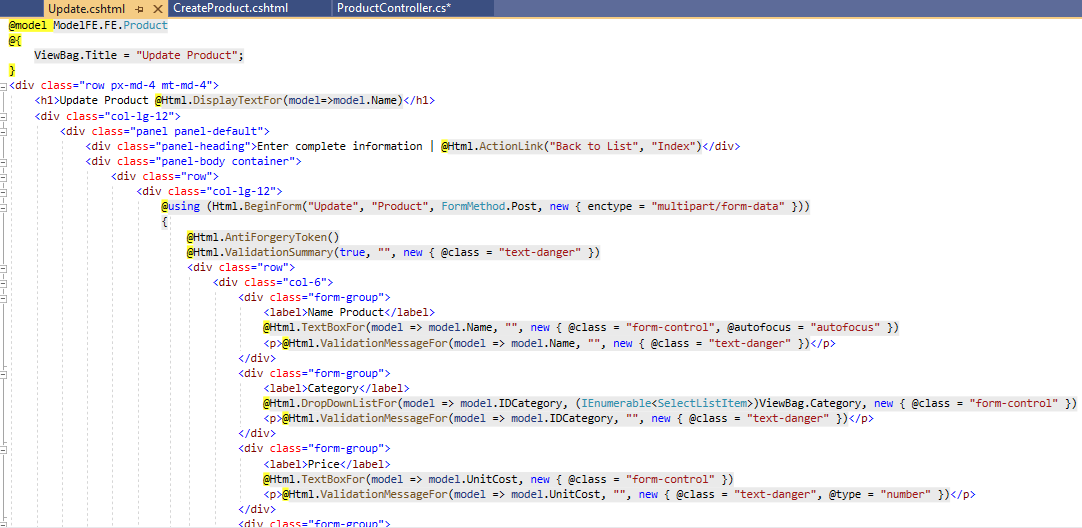
**Ngoài ra còn thực hiện chức năng Cập nhật sản phẩm và trang Quản lý Category với các chức năng tương tự của các bảng đã yêu cầu.**

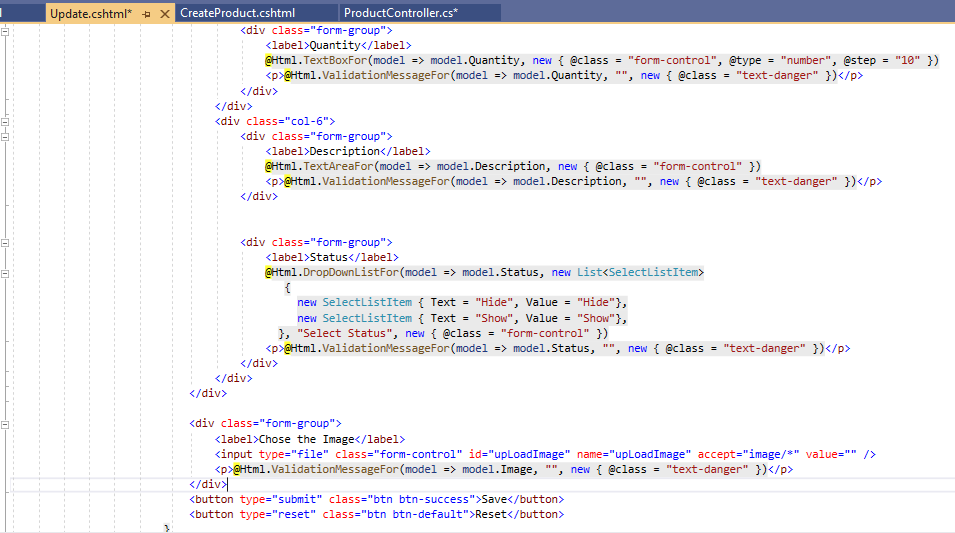


*Hình 41:* Update Sản phẩm

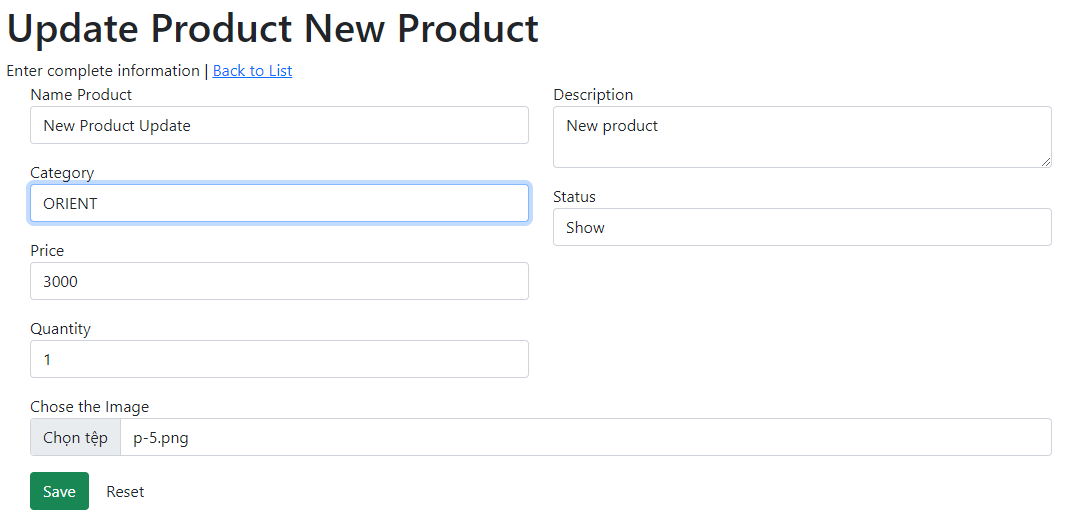
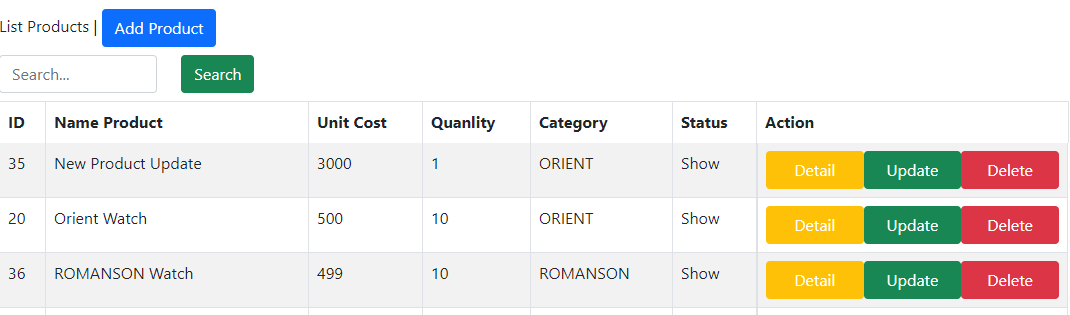
****

*Hình 42:* Update Sản phẩm

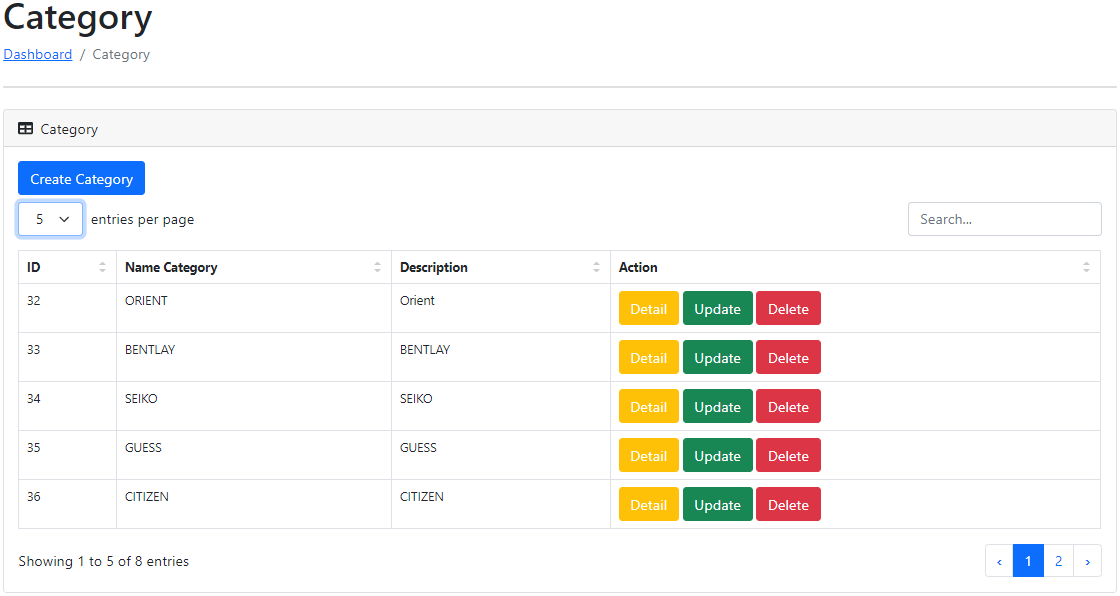




*Hình 43: Code Giao diện* Update Sản phẩm

*Hình 43: Kết quả*



*Hình 45: Category với các chức năng tương tự*

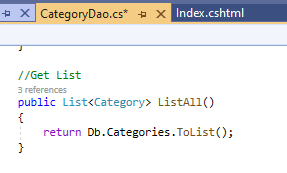
## Thực hiện các yêu cầu cho trang Người dùng (2 đ)

## Download 1 template bán hàng bất kỳ, hiệu chỉnh lại cho phù hợp với dữ liệu của bảng Sản phẩm.

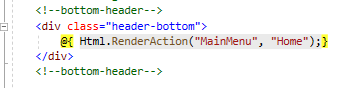
Link Download Template: [Luxury Watch](https://w3layouts.com/template/luxury-watches-a-flat-ecommerce-bootstrap-responsive-web-template/)

## Đổ dữ liệu của bảng sản phẩm vào trang chủ.

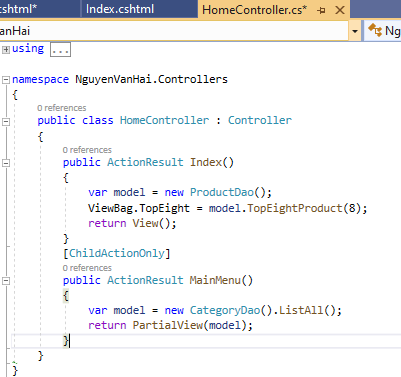
### Hiển thị danh mục sản phẩm



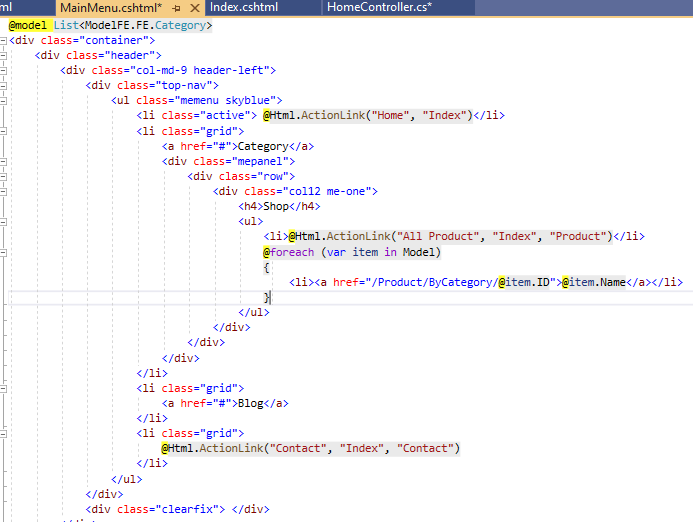
*Hình 46: CategoryDao Truy vấn danh sách Danh mục*



*Hình 47: Menu được tách ra nên RenderAction để gọi lại Menu*



*Hình 48: Xử lý Mainmenu*

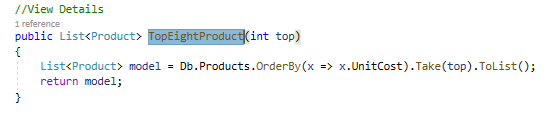


*Hình 49: Code Giao diện MainMenu*

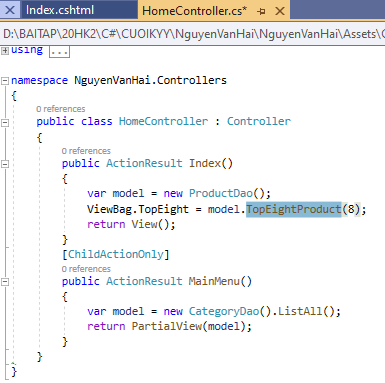
**

*Hình 50: Kết quả*

### Hiển thị Top 8 sản phẩm giá rẽ



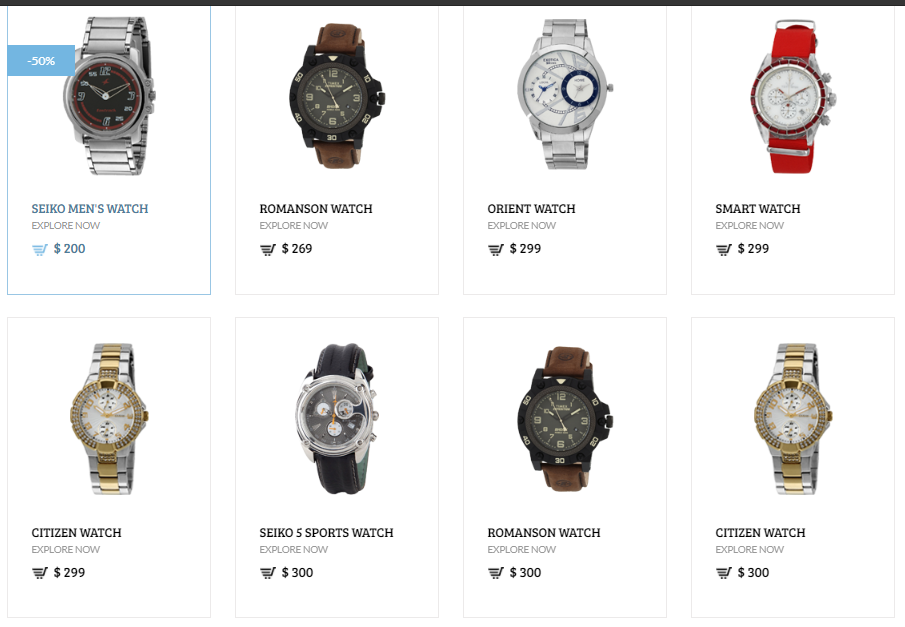
*Hình 51: Truy vấn Top Sản phẩm*



*Hình 52: HomeController trả kết quả về View*

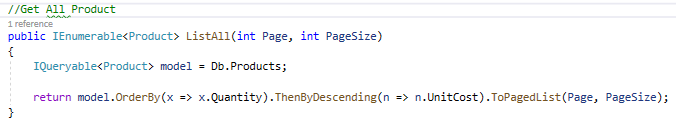


*Hình 53: Code Giao diện*

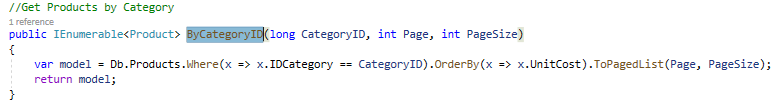


*Hình 54: Kết quả*

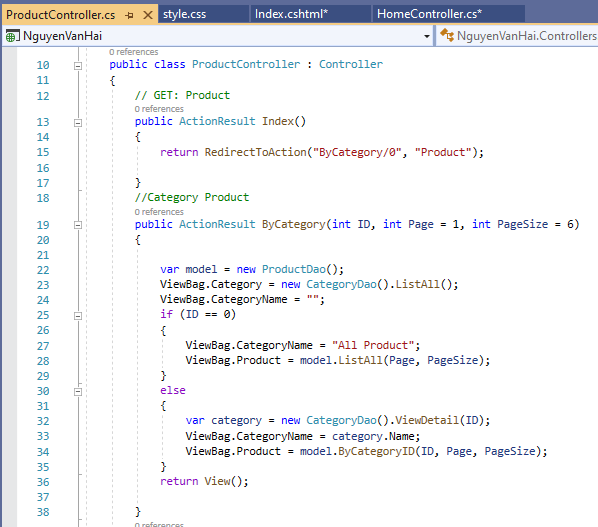
### Hiển thị danh sách sản phầm theo danh mục



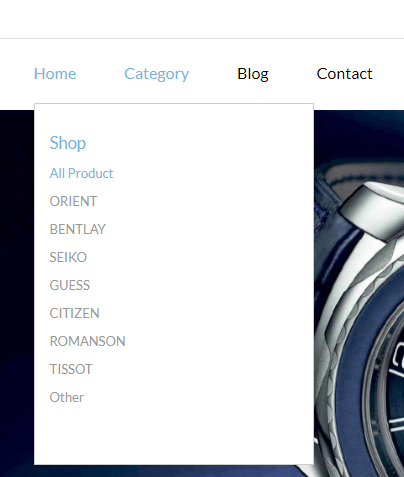
*Hình 55: Truy vấn tất cả sản phẩm*



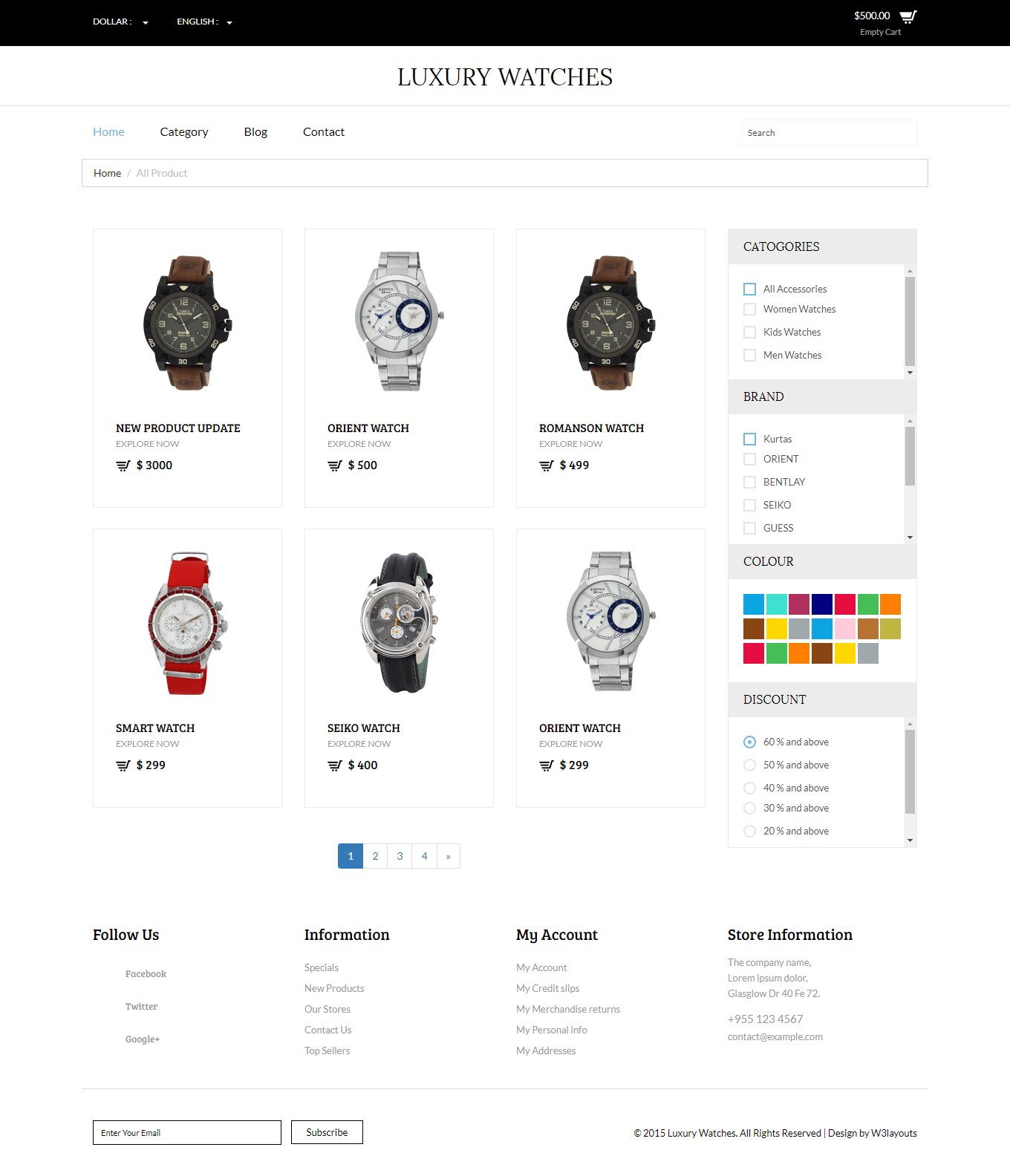
*Hình 56: Truy vấn sản phẩm the loại sản phẩm*



*Hình 57: Sử dụng ViewBag để Binding dữ liệu*



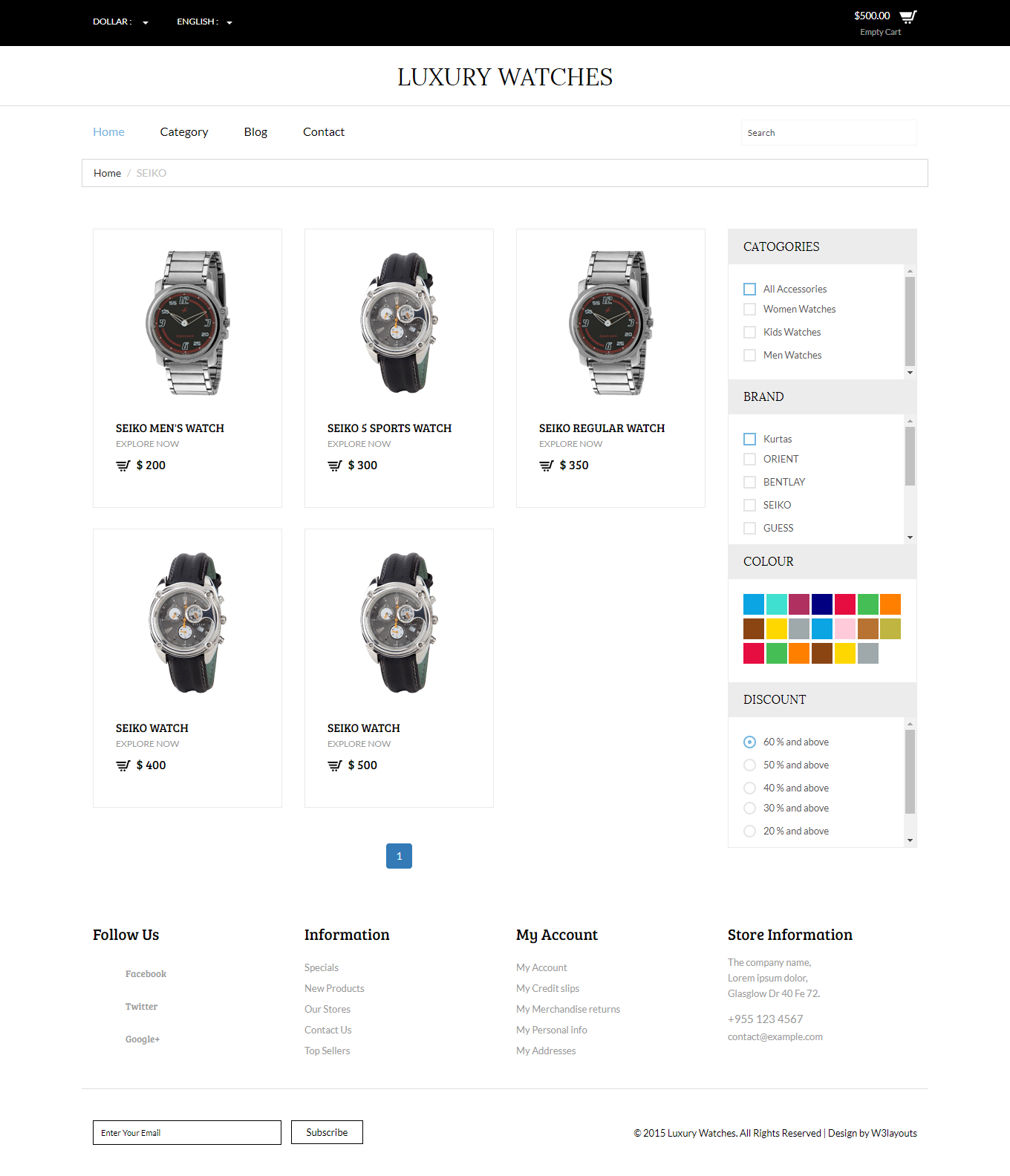
*Hình 58: Click All Product hoặc các thể loại để đến Danh sách sản phẩm*



*Hình 59: Kết quả All Product*

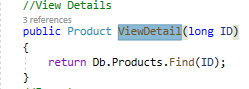


*Hình 60: Kết quả thể loại ORIENT*

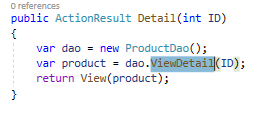


*Hình 61: Kết quả thể loại SEIKO*

### Hiển thị chi tiết sản phẩm

****

*Hình 62: Truy vấn chi tiết sản phẩm theo ID sản phẩm*

**

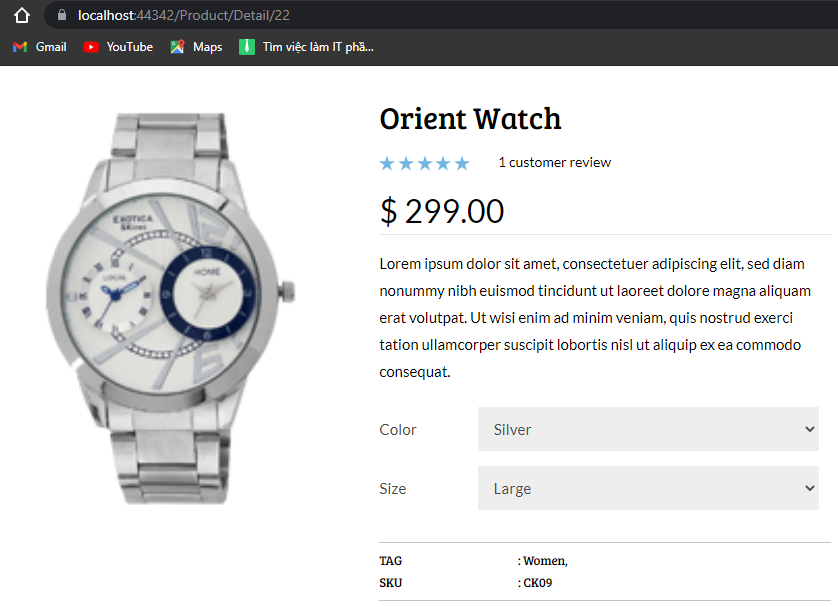
*Hình 63: ProductControler Action Detail Xử lý*

** **

*Hình 64: Code Giao diện*

**

*Hình 65: Kết quả chi tiết sản phẩm có ID 15*



*Hình 66: Kết quả chi tiết sản phẩm có ID 22*

# 