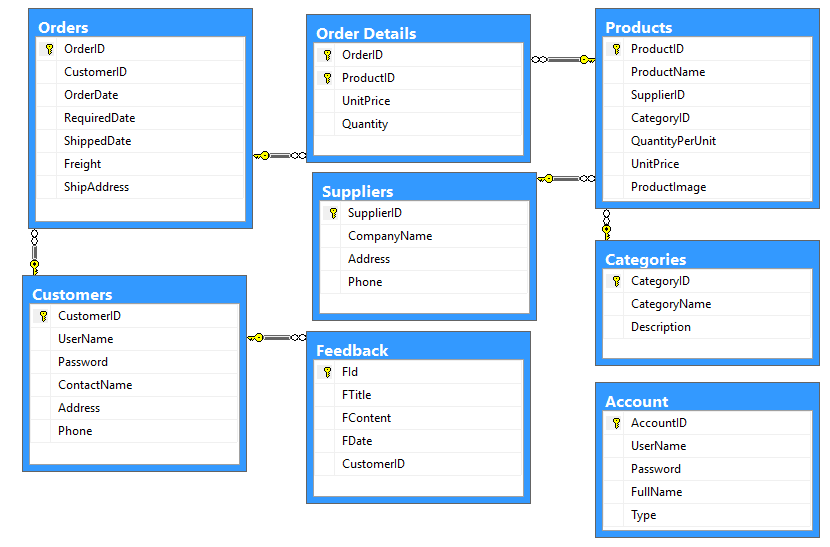
**HƯỚNG DẪN BÀI 3 (QUẢN LÝ BÁN HÀNG)**

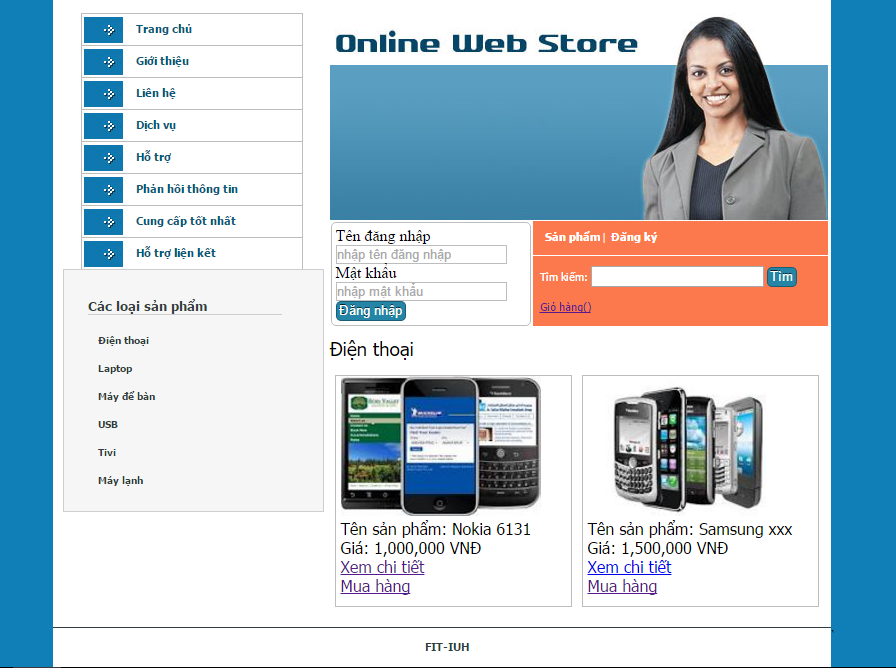
Dùng codefirst tạo CSDL quản lý bán hàng online đơn giản như sau:

Yêu cầu:

Bạn hãy thực hiện 1 website quản lý bán bàng với yêu cầu như sau:

* Phía người dùng:

1. Thực hiện Layout cho khách hàng như sau: (người dùng có thể xem danh sách các sản phẩm theo loại từ link của các loại sản phẩm)



@RenderBody

Load danh sách các loại sản phẩm từ bảng Category (**CategoriesPartial)**

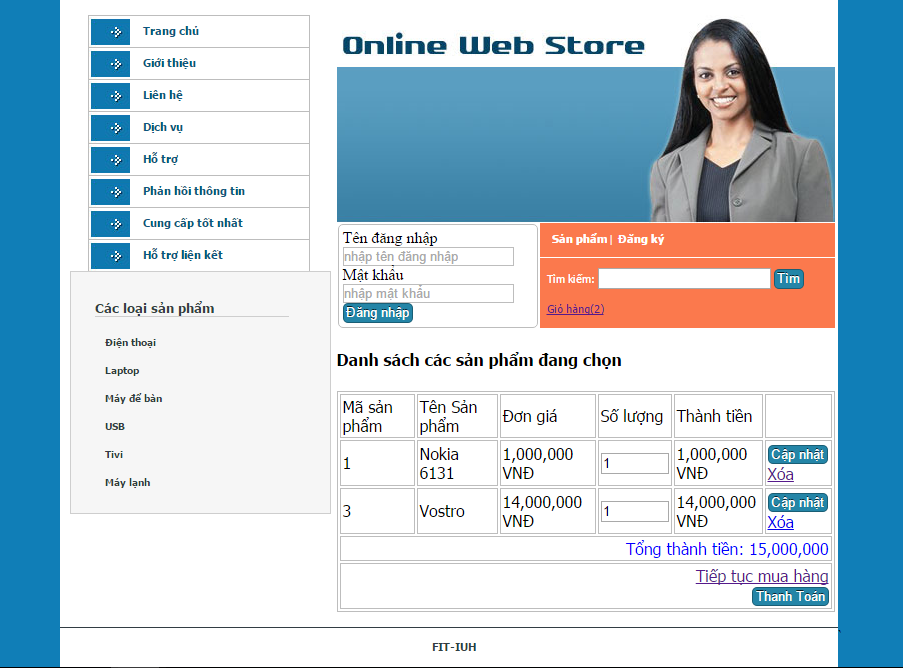
1. Tạo trang Index hiển thị thông tin các sản phẩm thừa kế từ Layout có giao diện như sau (dùng Ajax & jQuery để thực hiện load thêm sản phẩm lên trang khi sản phẩm quá nhiều):



1. Nếu người dùng muốn xem cho tiết sản phẩm thì click vào link xem chi tiết sản phẩm sẽ được hiển thị ở trang ProductDetail có giao diện như sau:



1. Nếu người dùng muốn mua hàng thì click vào link Mua hàng, thông tin giỏ hàng sẽ được cập nhật phía trên web như hình trên. Mỗi lần click vào link mua hàng, nếu mặt hàng mới thì giỏ hàng thêm sản phẩm vào với số lượng là 1, nếu mặt hàng đã có trong giỏ hàng thì số lượng sẽ được cộng thêm 1.
2. Nếu người dùng muốn xem giỏ hàng thì click vào link giỏ hàng, website sẽ chuyển hướng sang trang giỏ hàng cho người dùng xem thông tin các mặt hàng mình đang chọn, số lượng là bao nhiêu với giao diện như sau (Action ShopCart):



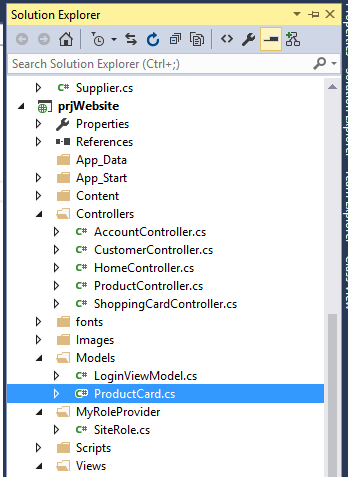
1. Khi người dùng muốn thay đổi số lượng sản phẩm cần mua, có thể nhập thay đổi trong textbox số lượng và click nút cập nhật. Khi người dùng cần xóa sản phẩm trong giỏ hàng, click link xóa. Nếu khách hàng muốn tiếp tục mua thì click link tiếp tục mua hàng. Người dùng muốn thanh toán hóa đơn để kết thúc việc mua hàng thì click nút Thanh toán, website sẽ chuyển hướng sang trang thanh toán (nếu người dùng chưa đăng nhập thì phải bắt buộc người dùng đăng nhập) với giao diện như sau:



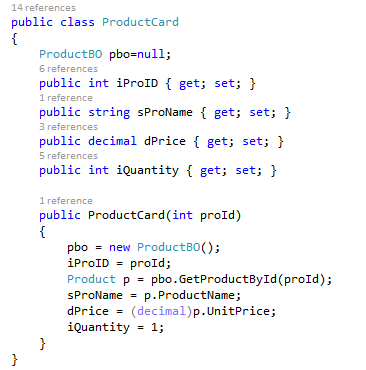
1. Trường hợp người dùng chưa có tài khoản thì click vào link “Đăng ký” để mở trang DangKy khách hàng mới (Sinh viên tự thiết kế giao diện và kiểm tra dữ liệu hợp lý)
2. Nếu người dùng đã đăng nhập, nhập địa chỉ giao hàng, nhấn nút Thanh toán, website sẽ thực hiện lưu vào cơ sở dữ liệu, đồng thời xóa đi thông tin các mặt hàng trong giỏ hàng.

Hướng dẫn thực hiện chức năng giỏ hàng:

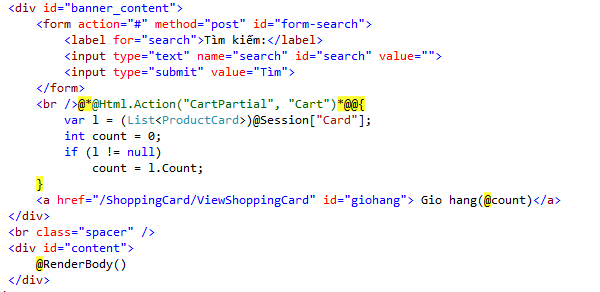
Trong thư mục Models tạo class ProductCard như sau (dùng để khai báo thông tin 1 mặt hàng trong giỏ hàng):



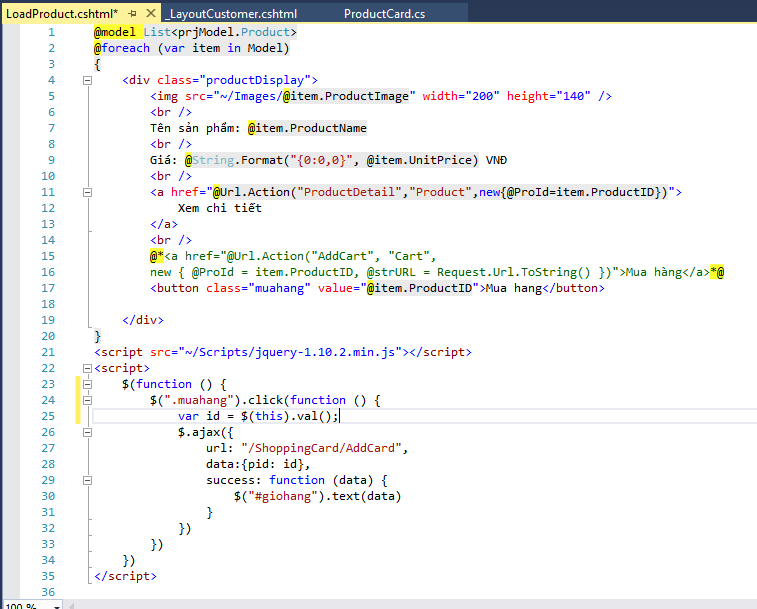
Class ProductCard:



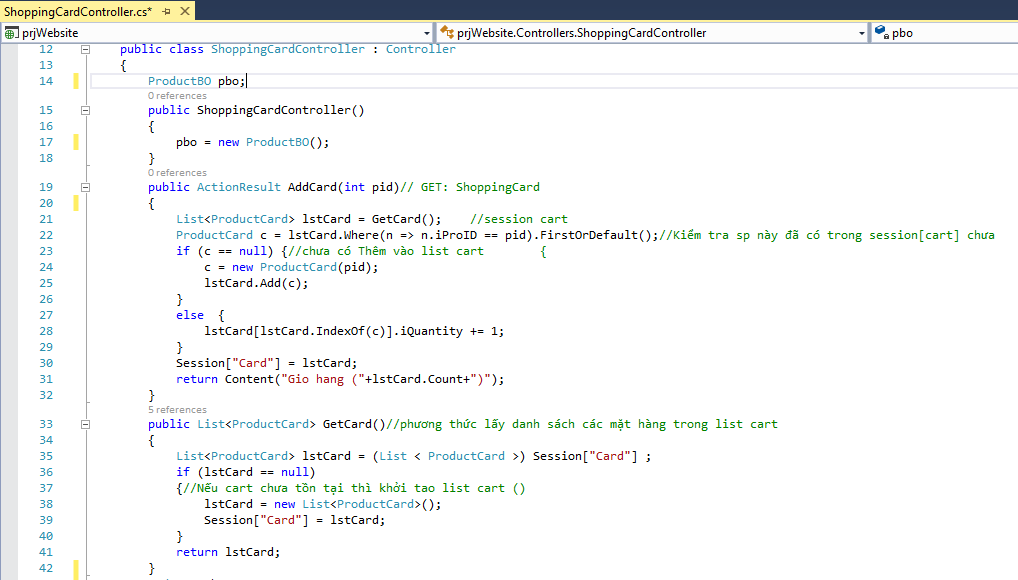
Mở Layout\_Customer thêm phần thông tin giỏ hàng như sau:



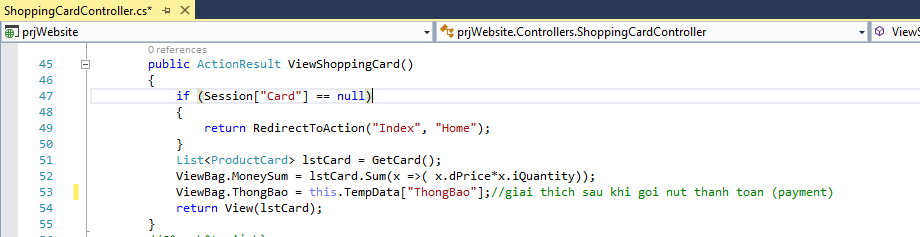
Mở lại View LoadProduct.cshtml thêm phần jQuery gọi hàm ajax để cập nhật giỏ hàng như sau:



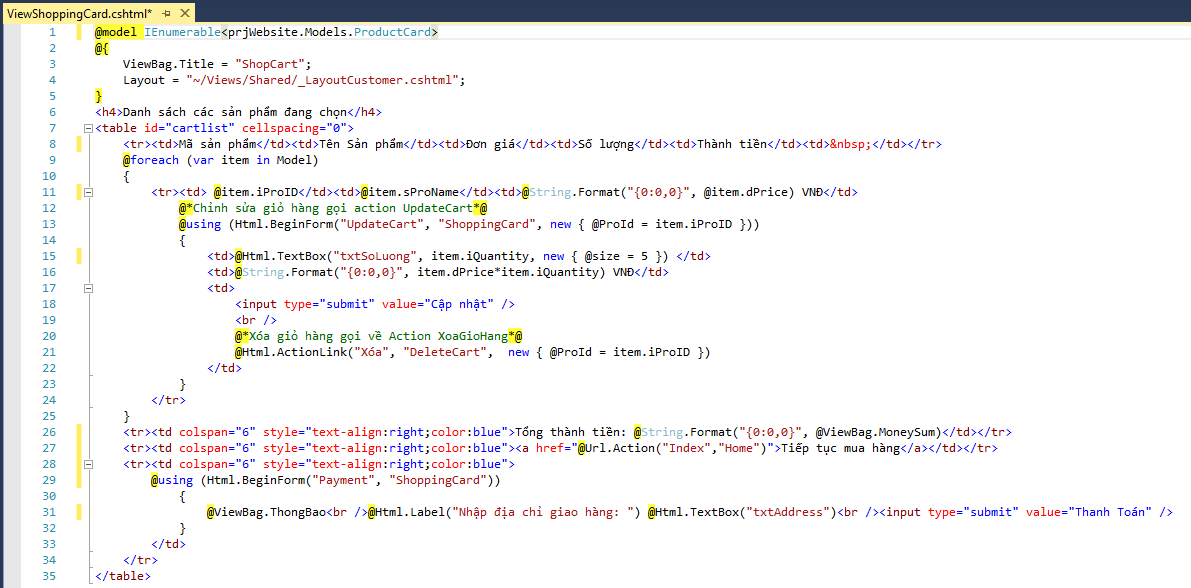
Tạo ShoppingCardController.cs có phương thức AddCard như sau:



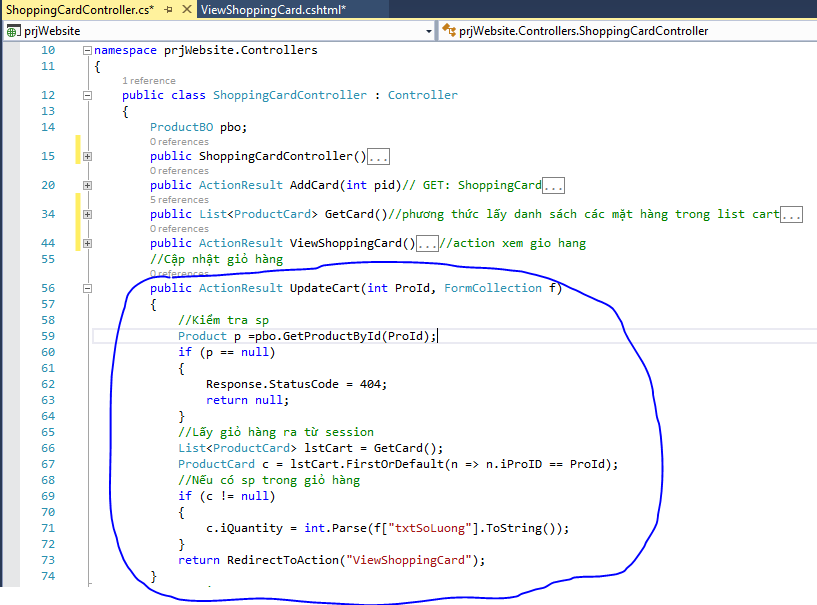
Link Giỏ hàng trong Layout gọi ViewShoppingCard -> mở ShoppingCardController thêm Action: ViewShoppingCard như sau:



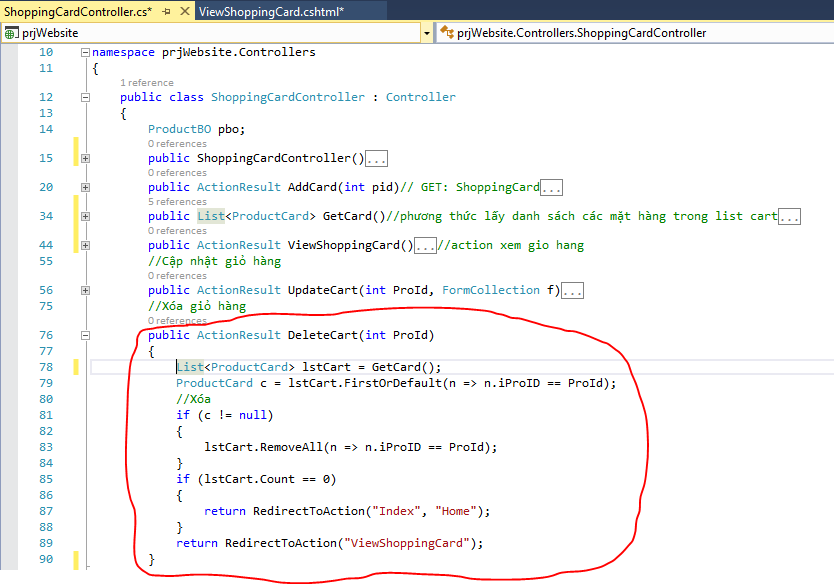
Tạo ViewShoppingCard.cshtml tương ứng:



Thêm Action: UpdateCard trong ShoppingCardController như sau:

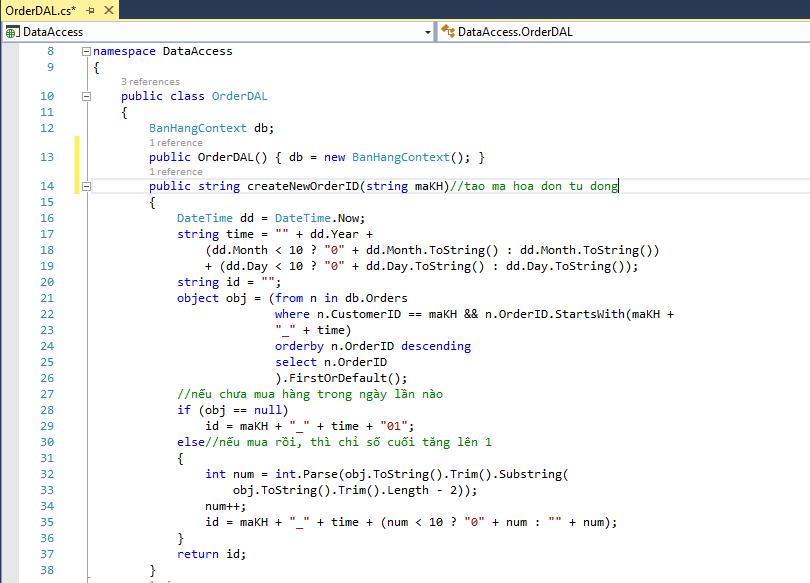


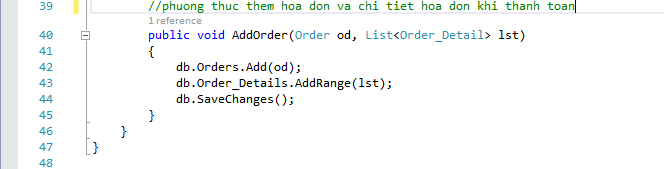
Thêm Action: DeleteCard trong ShoppingCardController như sau:



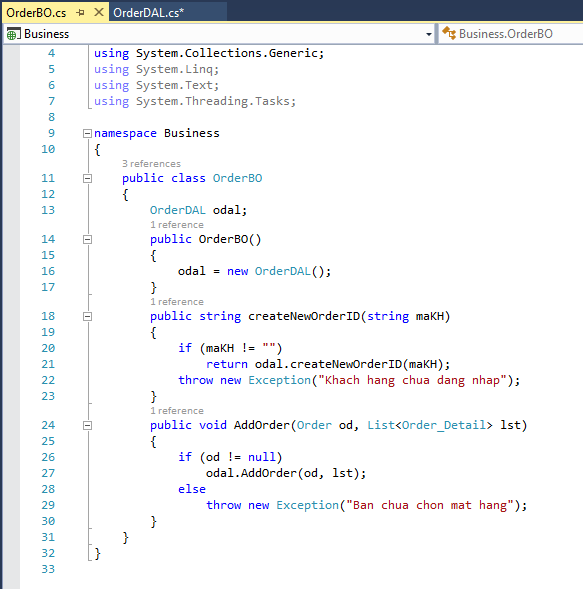
Nút thanh toán:

Trước tiên xây dựng phương thức Thanh toán, tạo mã hóa đơn tự động trong DataAccess:





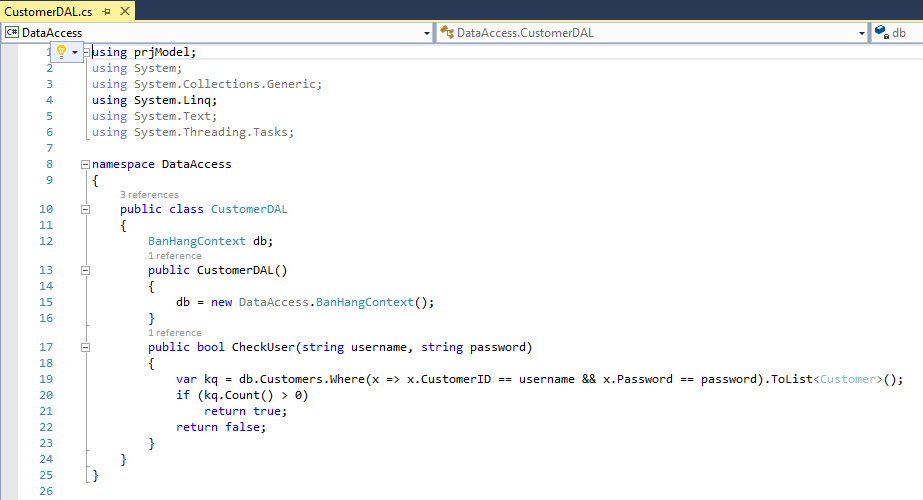
Tạo Class OrderBO: để cung cấp cho website:



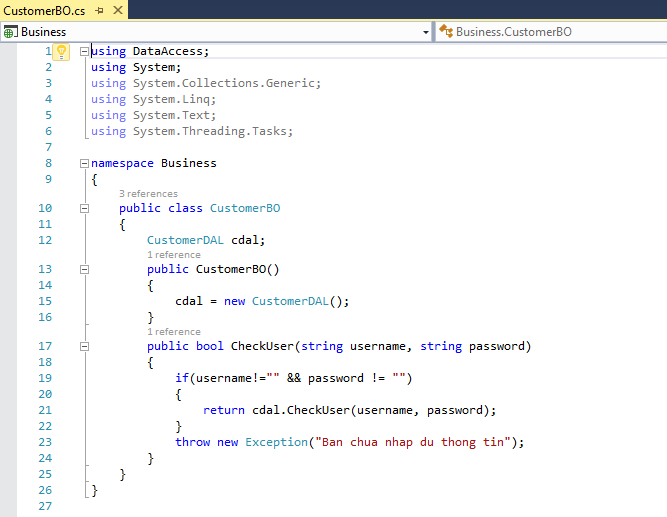
Người dùng khi mua hàng phải đăng nhập (Form đăng nhập đã để trong Layout):



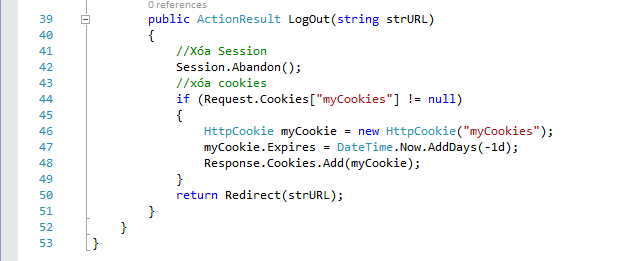
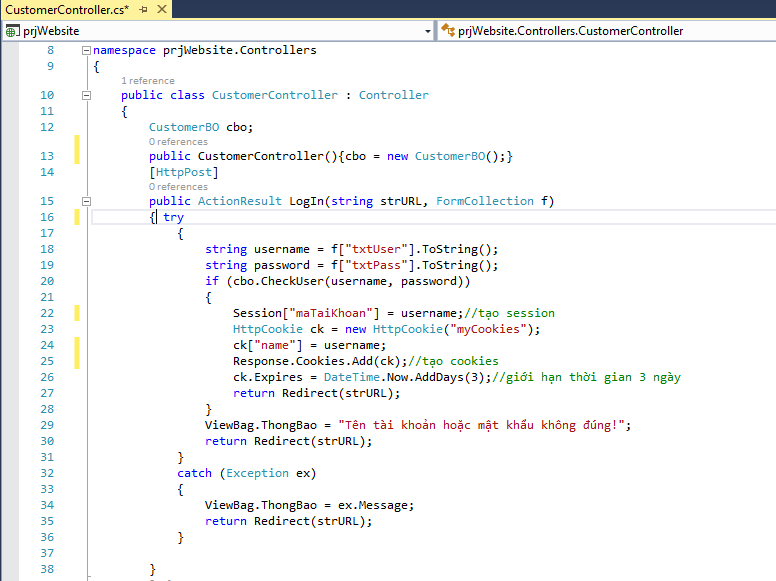
Tạo phương thức kiểm tra người dùng class CustomerDAL trong tầng DataAccess:



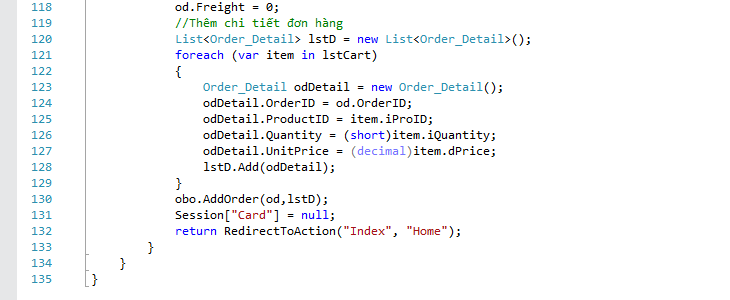
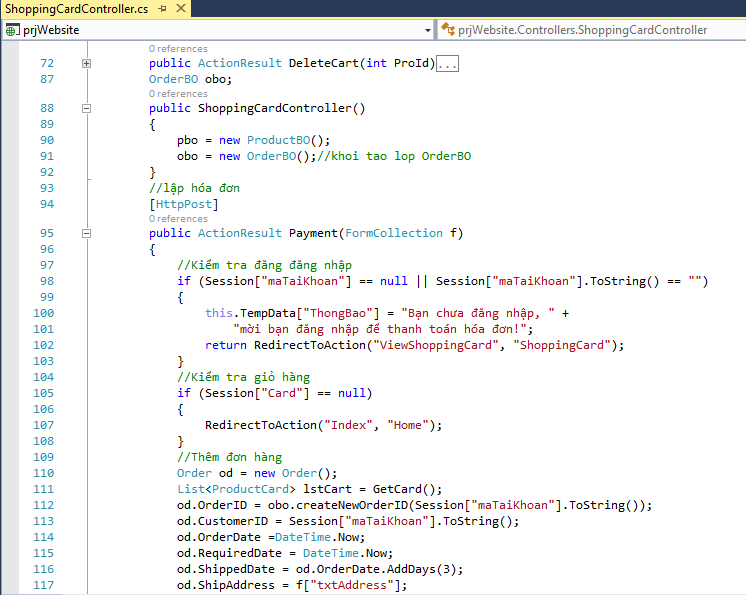
Tầng Business tạo class CustomerBO:



Tạo CustomerController:

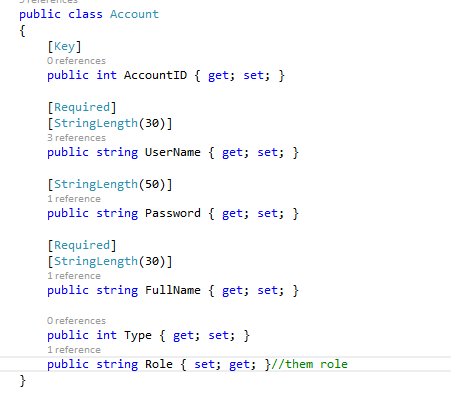


Tạo thêm phương thức thanh toán trong ShoppingCardController:

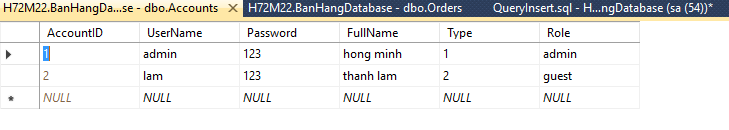


Hướng dẫn phần phân quyền:

Thêm thuộc tính Role trong bảng Account như sau:



Mở Sql Server nhập them Role cho bảng Account:



Viết class AccountDAL:



Class AccountBO:

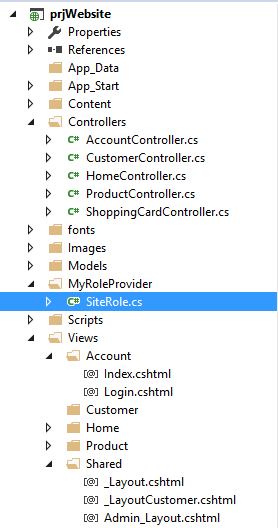


Mở tập Web.config trong prjWebsite nhập thêm trong thẻ <system.web> như sau:

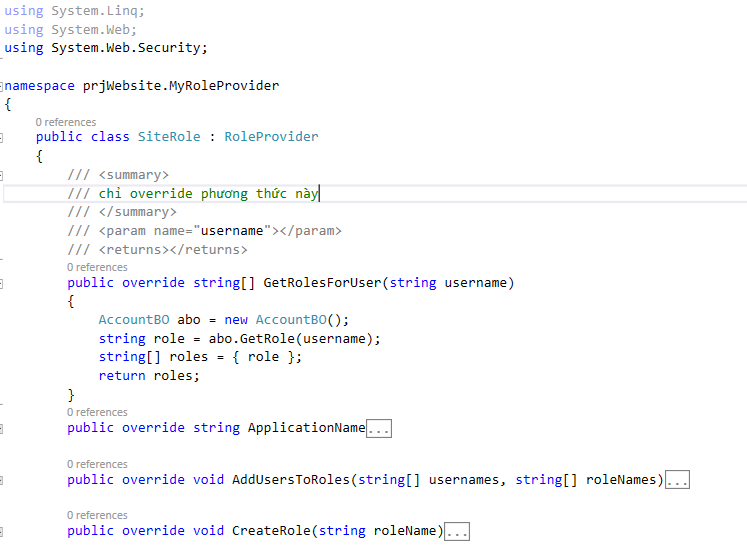


Trong đó:

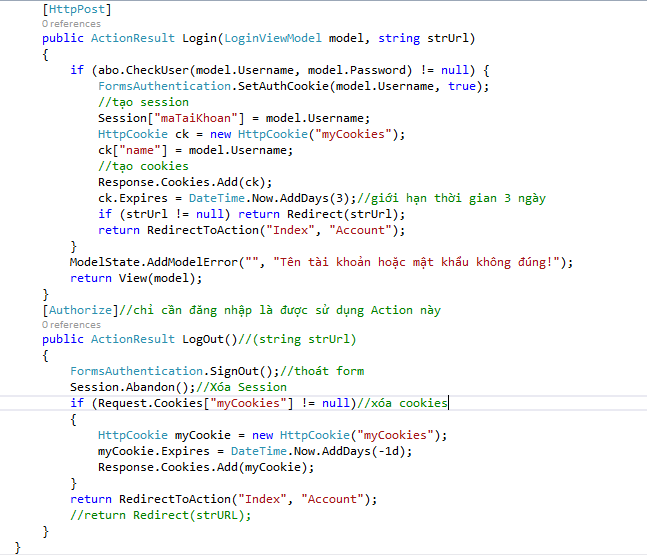
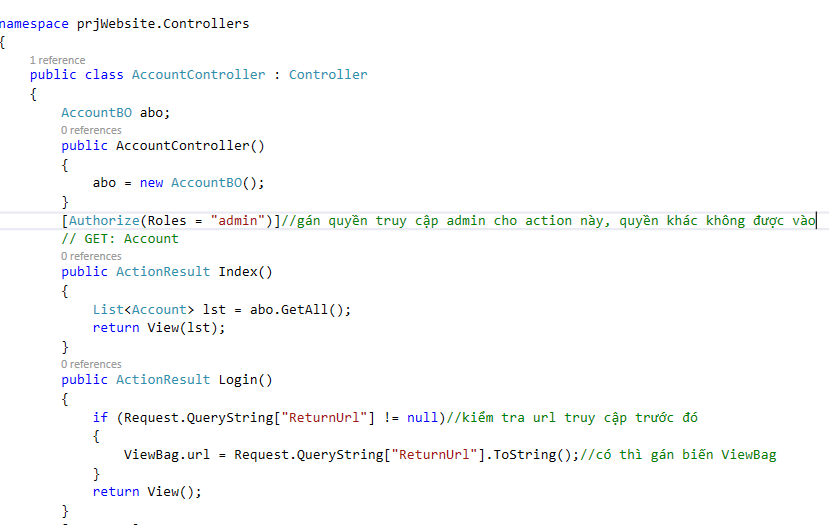
* Authentication : giới hạn người dung truy cập những Action đòi hỏi có quyền đăng nhập bằng Form và nếu truy cập vào các action này người dung phải chuyển qua trang Login của AccountController.
* RoleManager: mở quyền truy cập bằng role thông qua Class **SiteRole** trong thư mục **MyRoleProvider** 🡪 tạo thư mục MyRoleProvider trong prjWebsite, trong thư mục này tạo class **SiteRole** thừa kế từ class **RoleProvider như sau:**



Class SiteRole:



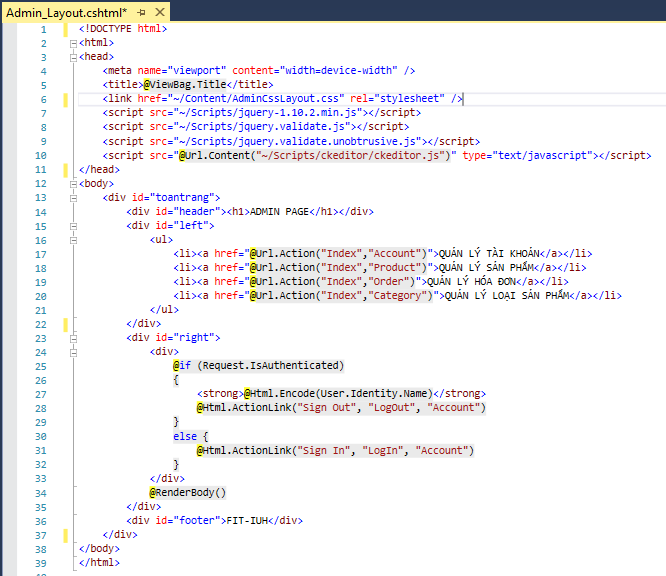
Tạo AccountController:



Tạo View Login từ Action Login:



Kết hợp với AdminLayout (tạo trong thư mục Shared):



File css (bạn có thể dung bootstrap):

body {

margin: 0;

padding: 0;

background: #FFFFFF;

font-size: 16px;

color:#11295b;

}

div#toantrang{

overflow:auto;

font-family:Tahoma;

background-color:#FFFFFF;

width: 1000px;

height: auto;

margin-bottom: 10px; margin-top: 10px;

margin-left: auto; margin-right: auto;

border:5px solid black;

}

#header {

width: 1000px;

height: 100px;

margin: 0 auto;

float:left;

text-align: center;

background-color:#e1f6f3;

border-bottom:1px solid black;

}

div#left{

width:200px;

height:auto;

float:left;

margin:0 auto;

}

div#right{

float:left;

width:757px;

height:auto;

margin:0 auto;

padding:20px;

position:inherit;

border-left:1px solid black;

}

#footer {

clear:both;

width: 999px;

height: 30px;

margin: 0 auto;

float:left;

text-align: center;

background-color:#e1f6f3;

border-top:1px solid black;

}

a{

color:#11295b;

text-decoration:none;

}

a:hover{

color:#00ff21;

}

table{

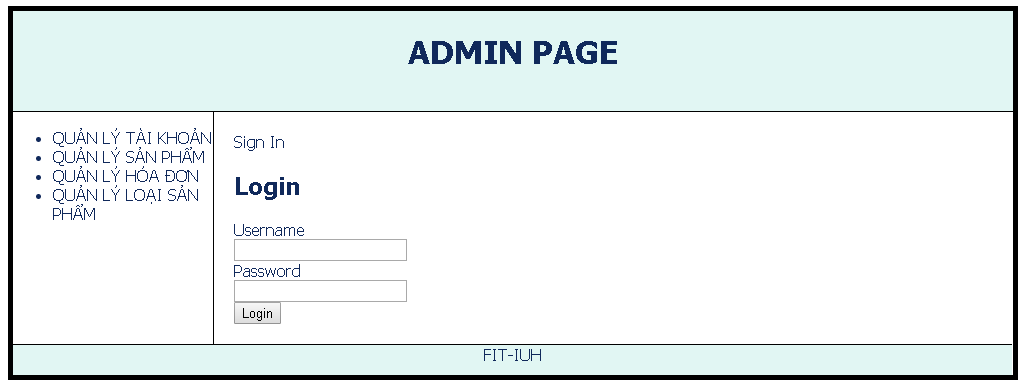
border:1px solid black;

width:100%;

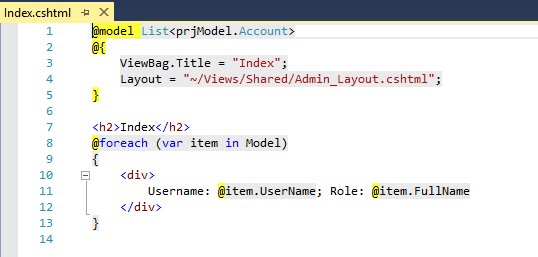
border-collapse:collapse;

}

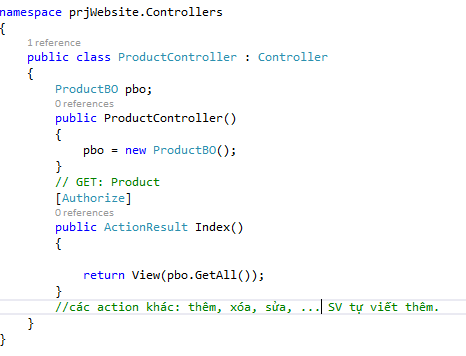
td,th{ border:1px solid black;}



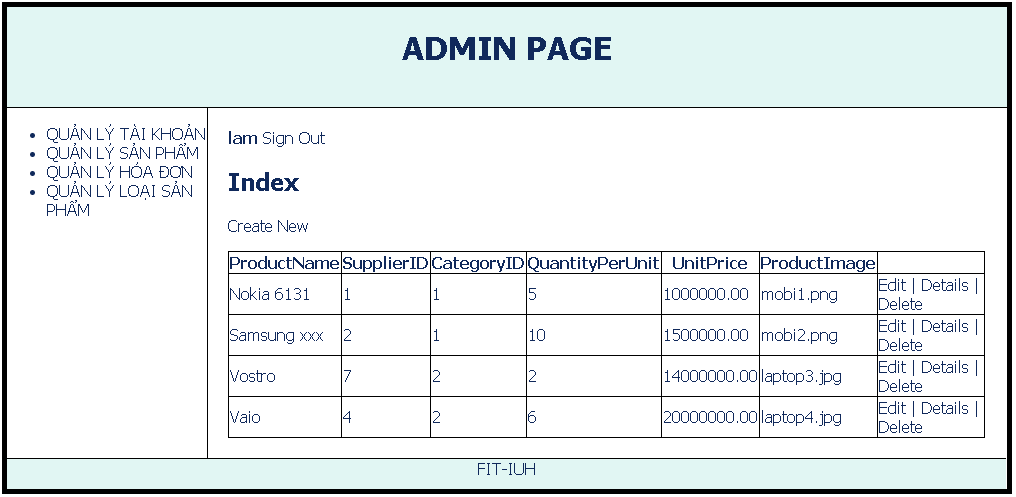
Nếu đăng nhập thành công (chú ý phải role: admin) chuyển qua trang Index của Account:



Tạo ProductController để quản lý thông tin các sản phẩm (nhớ gán quyền đăng nhập cho các Action):



Tạo các View tương ứng để quản lý sản phẩm:



public ActionResult Login(LoginViewModels model)

{

var userbll = new UserBLL();

if (ModelState.IsValid)

{

var result = userbll.Login(model.UserName, EncryptorMD5.MD5Hash(model.Passwd));

if (result == 1)

{

var user = userbll.GetByUserName(model.UserName);

var userSession = new UserLogin();

userSession.UserName = user.UserName;

userSession.UserID = user.ID;

Session.Add(ComomConstant.USER\_SESSION, userSession);

return RedirectToAction("Index", "HomeAdmin");

}

else if (result == 0)

{

ModelState.AddModelError(string.Empty, "TK Không tồn tại");

}

else if (result == -1)

{

ModelState.AddModelError("", "TK đang bị khóa");

}

else if (result == -2)

{

ModelState.AddModelError("", "sai pass");

}

}

return View("Login");

}