Chuoi ket noi:

<connectionStrings>

<add name="EShopV20"

connectionString="Server=.\sqlexpress;Database=EShopV20;Integrated Security=True"

providerName="System.Data.SqlClient" />

</connectionStrings>

\_Layout

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Online Shopping Mall</title>

@Styles.Render("~/Content/css")

@Scripts.Render("~/bundles/modernizr")

@Scripts.Render("~/bundles/jquery")

@Scripts.Render("~/bundles/bootstrap")

<script src="~/Scripts/eshop.js"></script>

@RenderSection("scripts", required: false)

</head>

<body>

<div class="container">

<header class="nn-header row">

<div class="nn-company">

<h1>Online shopping mall</h1>

<h5>The center point of the professional programming</h5>

</div>

<img class="pull-right" src="/images/header-object.png" />

</header>

@Html.Partial("\_Menu")

<div class="nn-sheet row">

<article class="col-sm-9">

<div class="nn-body">

@RenderBody()

</div>

</article>

<aside class="col-sm-3">

<!--CartInfo-->

@Html.Partial("\_CartInfo")

<!--/CartInfo-->

<!--Search-->

@Html.Partial("\_Search")

<!--/Search-->

<!--Category-->

@Html.Action("\_Category", "Home")

<!--/Category-->

<!--Supplier-->

@Html.Action("\_Supplier", "Home")

<!--/Supplier-->

<!--Special-->

@Html.Partial("\_Special")

<!--/Special-->

<!--\_Favorite-->

@Html.Action("\_Favorite", "Home")

<!--/\_Favorite-->

</aside>

</div>

<footer class="panel panel-default row">

<div class="panel-heading">

<p class="text-center">Nhất Nghệ &copy; 2017. All rights reserved.</p>

</div>

</footer>

</div>

</body>

</html>

Category

public ActionResult \_Category()

{

var model = db.Categories.ToList();

return PartialView(model);

}

<div class="panel panel-default">

<div class="panel-heading">Category</div>

<div class="list-group">

@foreach(var item in Model)

{

<a href="/Product/ListByCategory/@item.Id" class="list-group-item">@item.NameVN</a>

}

</div>

</div>

Supplier

public ActionResult \_Supplier()

{

var model = db.Suppliers.ToList();

return PartialView(model);

}

<div class="panel panel-default">

<div class="panel-heading">Supplier</div>

<div class="list-group">

@foreach(var item in Model)

{

<a href="/Product/ListBySupplier/@item.Id" class="list-group-item">@item.Name</a>

}

</div>

</div>

List

public ActionResult List()

{

var model = db.Products.ToList();

return View(model);

}

@model List<EShop.Models.Product>

@{

ViewBag.Title = "List of Products";

}

<h2>List of Products</h2>

@foreach (var item in Model)

{

<div class="col-sm-6 nn-product">

<div class="panel panel-default">

<div class="panel-heading">@item.Name</div>

<div class="panel-body">

<a href="/Product/Detail/@item.Id">

<img src="~/images/products/@item.Image" />

</a>

</div>

<div class="panel-footer">

<div class="row">

<div class="pull-left">@Html.FormatValue(item.UnitPrice, "{0:c}")/@item.UnitBrief</div>

<div class="pull-right" data-id="@item.Id">

@Html.Partial("\_Buttons")

</div>

</div>

</div>

@if (item.Special)

{

<img src="~/images/special\_icon.gif" />

}

else if (item.Latest)

{

<img src="~/images/new\_icon.gif" />

}

else if (item.Discount > 0)

{

<img src="~/images/promo\_icon.gif" />

}

</div>

</div>

}

@Html.Partial("\_Dialog")

<!--Dialog.cshtml-->

<!-- Modal -->

<div id="send-dialog" class="modal fade" role="dialog">

<div class="modal-dialog">

<!-- Modal content-->

<div class="modal-content">

<div class="modal-header">

<button type="button" class="close" data-dismiss="modal">&times;</button>

<h4 class="modal-title">Gửi thông tin hàng hóa</h4>

</div>

<div class="modal-body">

<input id="item-id" type="hidden">

<div class="form-group">

<input id="from" placeholder="Người gửi" class="form-control" />

</div>

<div class="form-group">

<input id="to" placeholder="Người nhận" class="form-control" />

</div>

<div class="form-group">

<input id="subject" placeholder="Tiêu đề" class="form-control" />

</div>

<div class="form-group">

<textarea id="body" placeholder="Lời nhắn gửi" class="form-control" rows="3"></textarea>

</div>

</div>

<div class="modal-footer">

<button class="btn btn-default nn-btn-send">

<img src="~/images/loading.gif" />

Gửi đi

</button>

</div>

</div>

</div>

</div>

//Gui mail ProductController

public ActionResult Send(int Id, String From, String To, String Subject, String Body)

{

var url = Request.Url.AbsoluteUri.Replace("Send", "Detail/" + Id);

Body += "<hr><a href='" + url + "'>Chi tiết</a>";

XMailer.Send(To, Subject, Body, From);

return Content("Gửi mail thành công!");

}

$(".nn-send-mail").click(function () {

var id = $(this).parents("div").attr("data-id");

$("#item-id").val(id);

$(".nn-btn-send img").hide();

});

$(".nn-btn-send").click(function () {

var id = $("#item-id").val();

var from = $("#from").val();

var to = $("#to").val();

var subject = $("#subject").val();

var body = $("#body").val();

$(".nn-btn-send img").show();

$.ajax({

url: "/Product/Send",

data: { id: id, from: from, to: to, subject: subject, body: body },

type:"POST",

success: function (response) {

alert("Đã gửi mail thành công!");

$("button.close").click();

$(".nn-btn-send img").hide();

}

});

});

\_Buttons

<button class="btn btn-sm btn-primary nn-add-to-cart">

<span class="glyphicon glyphicon-shopping-cart"></span>

</button>

<button class="btn btn-sm btn-danger nn-add-to-favorite">

<span class="glyphicon glyphicon-heart"></span>

</button>

<button class="btn btn-sm btn-info nn-send-mail"

data-toggle="modal" data-target="#send-dialog">

<span class="glyphicon glyphicon-envelope"></span>

</button>

Detail

public ActionResult Detail(int Id)

{

var model = db.Products.Find(Id);

return View(model);

}

@{

ViewBag.Title = "Detail";

}

<script src="~/Scripts/eshop.js"></script>

<h2>Details of Product</h2>

<div class="panel panel-default nn-detail">

<div class="panel-heading">@Model.Name</div>

<div class="panel-body text-center">

<a href="/Product/Detail/@Model.Id">

<img src="~/images/products/@Model.Image" />

</a>

</div>

<div class="panel-body">

<ul>

<li>Discount: @Html.FormatValue(Model.Discount, "{0:p}")</li>

<li>Product Date: @Html.FormatValue(Model.ProductDate, "{0:dd-MM-yyyy}")</li>

<li>Special?: @(Model.Special ? "Yes" : "No")</li>

<li>Latest?: @(Model.Latest ? "Yes" : "No")</li>

<li>Available?: @(Model.Available ? "Yes" : "No")</li>

<li>Category: @Model.Category.NameVN</li>

<li>Supplier: @Model.Supplier.Name</li>

</ul>

<div>@Model.Description</div>

</div>

<div class="panel-footer">

<div class="row">

<div class="pull-left">@Html.FormatValue(Model.UnitPrice, "{0:c}")/@Model.UnitBrief</div>

<div class="pull-right" data-id="@Model.Id">

@Html.Partial("\_Buttons")

</div>

</div>

</div>

</div>

<div class="nn-thumbnail">

<ul class="nav nav-tabs">

<li class="active"><a data-toggle="tab" href="#home">Same Category</a></li>

<li><a data-toggle="tab" href="#menu1">Same Supplier</a></li>

<li><a data-toggle="tab" href="#menu2">Items Viewed</a></li>

</ul>

<div class="tab-content">

<div id="home" class="tab-pane fade in active">

<h3>HÀNG CÙNG LOẠI</h3>

@foreach (var item in Model.Category.Products)

{

<a href="/Product/Detail/@item.Id">

<img src="~/images/products/@item.Image" />

</a>

}

</div>

<div id="menu1" class="tab-pane fade">

<h3>HÀNG CÙNG CÙNG NHÀ CUNG CẤP</h3>

@foreach (var item in Model.Supplier.Products)

{

<a href="/Product/Detail/@item.Id">

<img src="~/images/products/@item.Image" />

</a>

}

</div>

<div id="menu2" class="tab-pane fade">

<h3>HÀNG ĐÃ XEM</h3>

@foreach (var item in ViewBag.Views)

{

<a href="/Product/Detail/@item.Id">

<img src="~/images/products/@item.Image" />

</a>

}

</div>

</div>

</div>

@Html.Partial("\_Dialog")

public ActionResult ListByCategory(int Id)

{

var list = db.Categories.Find(Id).Products.ToList();

return View("List", list);

}

public ActionResult ListBySupplier(String Id)

{

var list = db.Suppliers.Find(Id).Products.ToList();

return View("List", list);

}

Special

public ActionResult ListBySpecial(string Id)

{

List<Product> lst = null;

switch(Id)

{

case "Best":

lst = db.Products.OrderByDescending(p => p.OrderDetails.Sum(d => d.Quantity)).Take(10).ToList();

break;

case "Latest":

lst = db.Products.Where(p => p.Latest).ToList();

break;

case "Spec":

lst = db.Products.Where(p => p.Special).ToList();

break;

case "Favo":

lst = db.Products.OrderByDescending(p => p.Views).Take(10).ToList();

break;

case "Disc":

lst = db.Products.OrderByDescending(p => p.Discount > 0).Take(10).ToList();

break;

}

return View("List", lst);

}

<div class="panel panel-default">

<div class="panel-heading">Hang dac biet</div>

<div class="list-group">

<a href="/Product/ListBySpecial/Best" class="list-group-item">Hang ban chay</a>

<a href="/Product/ListBySpecial/Latest" class="list-group-item">Hang moi</a>

<a href="/Product/ListBySpecial/Spec" class="list-group-item">Hang dac biet</a>

<a href="/Product/ListBySpecial/Favo" class="list-group-item">Hang xem nhieu</a>

<a href="/Product/ListBySpecial/Disc" class="list-group-item">Hang giam gia</a>

</div>

</div>

Favorite

public ActionResult \_Favorite()

{

var ids = XCookie.GetValue("Favorites");

var list = db.Products.ToList().Where(p => ids.Contains(p.Id.ToString()));

return PartialView(list);

}

public ActionResult AddToFavorite(int Id)

{

var ids = XCookie.GetValue("Favorites", Id.ToString());

if (!ids.Contains(Id.ToString()))

{

ids += "&" + Id.ToString();

}

XCookie.Add("Favorites", 30, ids);

return Content("");

}

<div class="panel panel-default">

<div class="panel-heading">

<span class="glyphicon glyphicon-heart"></span>

<strong>Hàng yêu thích</strong>

</div>

<div class="list-group">

@foreach (var item in Model)

{

<a href="/Product/Detail/@item.Id" class="list-group-item">@item.Name</a>

}

</div>

</div>

$(".nn-add-to-favorite").click(function () {

var id = $(this).parents("div").attr("data-id");

$.ajax({

url: "/Product/AddToFavorite/" + id,

success: function (response) {

location.reload();

}

});

});

//cartInfo

<div class="nn-cart">

<div class="panel panel-default">

<div class="panel-body">

<img class="col-sm-5 nn-cart-img" src="/images/shoppingcart.gif" />

<ul class="col-sm-7">

<li><span id="cart-cnt">@XSession.Cart.Count</span> items</li>

<li>$ <span id="cart-amt">@XSession.Cart.Amount</span></li>

<li><a href="/Cart/Index">View Cart</a></li>

</ul>

</div>

</div>

</div>

<style id="cart-effect"></style>

// hieu ung khi click vao trong the nn-add-to-cart se hien thi so luong va don gia

$(".nn-add-to-cart").click(function () {

var id = $(this).parents("div").attr("data-id");

$.ajax({

url: "/Cart/Add/" + id,

dataType: "json",

success: function (response) {

$("#cart-cnt").html(response.Count);

$("#cart-amt").html(response.Amount);

}

});

});

public ActionResult Add(int Id)

{

XSession.Cart.Add(Id);

var data = new

{

Count = XSession.Cart.Count,

Amount = String.Format("{0:#,###.#0}", XSession.Cart.Amount)

};

return Json(data, JsonRequestBehavior.AllowGet);

}

\_Buttons

<button class="btn btn-sm btn-primary nn-add-to-cart">

<span class="glyphicon glyphicon-shopping-cart"></span>

</button>

<button class="btn btn-sm btn-danger nn-add-to-favorite">

<span class="glyphicon glyphicon-heart"></span>

</button>

<button class="btn btn-sm btn-info nn-send-mail"

data-toggle="modal" data-target="#send-dialog">

<span class="glyphicon glyphicon-envelope"></span>

</button>

Hieu ung gio hang

$(function () {

$(".nn-add-to-cart").click(function () {

//lay anh o cartinfo

var img = $(this).parents(".panel").find(".panel-body img");

var src = img.attr("src");

var css = ".cart-effect{background-image:url('" + src + "');background-size:100% 100%;}";

$("#cart-effect").html(css);

var option = { to: ".nn-cart-img", className: "cart-effect" };

//tao hieu ung chuyen dong transfer

img.effect("transfer", option, 1500);

});

});

AccountController

public ActionResult Register()

{

return View();

}

[HttpPost]

public ActionResult Register(Customer user)

{

if (!ModelState.IsValid)

{

ModelState.AddModelError("", "Vui lòng sửa các lỗi sau đây");

return View();

}

// Upload

var file = Request.Files["UpPhoto"];

if (file.ContentLength > 0)

{

user.Photo = file.FileName;

file.SaveAs(Server.MapPath("~/Images/Customers/" + user.Photo));

}

else

{

user.Photo = "user.png";

}

// Insert

try

{

user.Activated = false;

dbc.Customers.Add(user);

dbc.SaveChanges();

// Gửi mail

var url = Request.Url.AbsoluteUri.Replace("Register", "Activate/" + user.Id.ToBase64());

var body = "Vui lòng click vào liên kết sau để <a href='"+url+"'>kích hoạt</a>";

XMailer.Send(user.Email, "Welcome to EShop", body);

ModelState.AddModelError("", "Đăng ký thành công!");

}

catch

{

ModelState.AddModelError("", "Đăng ký thất bại!");

}

return View();

}

//View cua AccountController

@model EShop.Models.Customer

@{

ViewBag.Title = "Đăng ký thành viên";

}

<div class="panel panel-default">

<div class="panel-heading">ĐĂNG KÝ THÀNH VIÊN</div>

<div class="panel-body">

@Html.ValidationSummary(true)

@using (Html.BeginForm("Register", "Account",

FormMethod.Post, new { enctype = "multipart/form-data" }))

{

<div class="form-group">

<label>Tên đăng nhập</label>

@Html.TextBoxFor(m => m.Id, new { @class = "form-control" })

@Html.ValidationMessageFor(m => m.Id)

</div>

<div class="form-group">

<label>Mật khẩu</label>

@Html.TextBoxFor(m => m.Password, new { @class = "form-control" })

@Html.ValidationMessageFor(m => m.Password)

</div>

<div class="form-group">

<label>Họ và tên</label>

@Html.TextBoxFor(m => m.Fullname, new { @class = "form-control" })

@Html.ValidationMessageFor(m => m.Fullname)

</div>

<div class="form-group">

<label>Email</label>

@Html.TextBoxFor(m => m.Email, new { @class = "form-control" })

@Html.ValidationMessageFor(m => m.Email)

</div>

<div class="form-group">

<label>Hình</label>

<input type="file" name="UpPhoto" class="form-control"/>

</div>

<div class="form-group">

<button class="btn btn-danger">Đăng ký</button>

</div>

}

</div>

</div>

public ActionResult Login()

{

return View();

}

[HttpPost]

public ActionResult Login(String id, String password, bool remember=false)

{

var user = dbc.Customers.Find(id);

if (user == null)

{

ModelState.AddModelError("", "Sai tên đăng nhâp!");

}

else if (user.Password != password)

{

ModelState.AddModelError("", "Sai mật khẩu!");

}

else if (!user.Activated)

{

ModelState.AddModelError("", "Tài khoản chưa kích hoạt!");

}

else

{

ModelState.AddModelError("", "Đăng nhâp thành công!");

XSession.User = user;

if (remember)

{

XCookie.Add("user", 30, id + '&' + password);

}

else

{

XCookie.Add("user", 0);

}

if (XSession.ReturnUrl != null)

{

return Redirect(XSession.ReturnUrl);

}

}

return View();

}

@using EShop.i18n.Account

@{

ViewBag.Title = "Đăng nhập";

}

<div class="panel panel-default">

<div class="panel-heading">ĐĂNG NHẬP</div>

<div class="panel-body">

@Html.ValidationSummary()

@using(Html.BeginForm()){

<div class="form-group">

<label>Tên đăng nhập</label>

@Html.TextBox("Id", null, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Mật khẩu</label>

@Html.TextBox("Password", null, new { @class = "form-control" })

</div>

<div class="form-group">

<label class="form-control">

@Html.CheckBox("Remember")

</label>

</div>

<div class="form-group">

<button class="btn btn-danger">@Login.Button</button>

</div>

}

</div>

</div>

public ActionResult Activate(String Id)

{

var user = dbc.Customers.Find(Id.FromBase64());

user.Activated = true;

dbc.SaveChanges();

return RedirectToAction("Login");

}

public ActionResult Forgot()

{

return View();

}

[HttpPost]

public ActionResult Forgot(String Id, String Email)

{

var user = dbc.Customers.Find(Id);

if (user == null)

{

ModelState.AddModelError("", "Sai tên đăng nhâp!");

}

else if (user.Email != Email)

{

ModelState.AddModelError("", "Sai địa chỉ email!");

}

else

{

XMailer.Send(Email, "Quên mật khẩu", user.Password);

ModelState.AddModelError("", "Mật khẩu đã được gửi qua email!");

}

return View();

}

<div class="panel panel-danger">

<div class="panel-heading">QUÊN MẬT KHẨU</div>

<div class="panel-body">

@Html.ValidationSummary()

@using(Html.BeginForm()){

<div class="form-group">

<label>Tên đăng nhập</label>

@Html.TextBox("Id", null, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Địa chỉ email</label>

@Html.TextBox("Email", null, new { @class = "form-control" })

</div>

<div class="form-group">

<button class="btn btn-danger">Lấy lại mật khẩu</button>

</div>

}

</div>

</div>

[Authenticate]

public ActionResult Change()

{

return View();

}

[HttpPost]

[Authenticate]

public ActionResult Change(String Id, String Password, String Password1, String Password2)

{

if (Password1 != Password2)

{

ModelState.AddModelError("", "Xác nhận mật khẩu mới không đúng!");

}

else

{

var user = dbc.Customers.Find(Id);

if (user == null)

{

ModelState.AddModelError("", "Sai tên đăng nhâp!");

}

else if (user.Password != Password)

{

ModelState.AddModelError("", "Sai mật khẩu!");

}

else

{

user.Password = Password1;

dbc.SaveChanges();

ModelState.AddModelError("", "Đổi mật khẩu thành công!");

}

}

return View();

}

@{

ViewBag.Title = "Đổi mật khẩu";

}

<div class="panel panel-danger">

<div class="panel-heading">ĐỔI MẬT KHẨU</div>

<div class="panel-body">

@Html.ValidationSummary()

@using(Html.BeginForm()){

<div class="form-group">

<label>Tên đăng nhập</label>

@Html.TextBox("Id", null, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Mật khẩu hiện tại</label>

@Html.TextBox("Password", null, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Mật khẩu mới</label>

@Html.TextBox("Password1", null, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Xác nhận mật khẩu mới</label>

@Html.TextBox("Password2", null, new { @class = "form-control" })

</div>

<div class="form-group">

<button class="btn btn-danger">Đổi mật khẩu</button>

</div>

}

</div>

</div>

[Authenticate]

public ActionResult Edit()

{

return View(XSession.User);

}

[HttpPost]

[Authenticate]

public ActionResult Edit(Customer user)

{

// Upload

var file = Request.Files["UpPhoto"];

if (file.ContentLength > 0)

{

user.Photo = file.FileName;

file.SaveAs(Server.MapPath("~/Images/Customers/" + user.Photo));

}

// Update

try

{

dbc.Entry(user).State = System.Data.EntityState.Modified;

dbc.SaveChanges();

ModelState.AddModelError("", "Cập nhật thành công!");

}

catch

{

ModelState.AddModelError("", "Cập nhật thất bại!");

}

return View();

}

@model EShop.Models.Customer

@{

ViewBag.Title = "Cập nhật tài khoản";

}

<div class="panel panel-default">

<div class="panel-heading">CẬP NHẬT TÀI KHOẢN</div>

<div class="panel-body">

@Html.ValidationSummary()

@using (Html.BeginForm("Edit", "Account",

FormMethod.Post, new { enctype = "multipart/form-data" }))

{

<div class="form-group">

<label>Tên đăng nhập</label>

@Html.TextBoxFor(m => m.Id, new { @class = "form-control", @readonly="true" })

</div>

<div class="form-group">

<label>Mật khẩu</label>

@Html.TextBoxFor(m => m.Password, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Họ và tên</label>

@Html.TextBoxFor(m => m.Fullname, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Email</label>

@Html.TextBoxFor(m => m.Email, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Hình</label>

<input type="file" name="UpPhoto" class="form-control"/>

@Html.HiddenFor(m => m.Photo)

</div>

<div class="form-group">

@Html.HiddenFor(m => m.Activated)

<button class="btn btn-danger">Cập nhật</button>

</div>

}

</div>

</div>

using EShop.Models;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

namespace EShop.Controllers

{

public class CartController : Controller

{

public ActionResult Add(int Id)

{

XSession.Cart.Add(Id);

var data = new

{

Count = XSession.Cart.Count,

Amount = String.Format("{0:#,###.#0}", XSession.Cart.Amount)

};

return Json(data, JsonRequestBehavior.AllowGet);

}

public ActionResult Remove(int Id)

{

XSession.Cart.Remove(Id);

var data = new

{

Count = XSession.Cart.Count,

Amount = String.Format("{0:#,###.#0}", XSession.Cart.Amount)

};

return Json(data, JsonRequestBehavior.AllowGet);

}

$(".nn-remove-from-cart").click(function () {

var id = $(this).attr("data-id");

$.ajax({

url: "/Cart/Remove/" + id,

dataType: "json",

success: function (response) {

$("#cart-cnt").html(response.Count);

$("#cart-amt").html(response.Amount);

}

});

$(this).parents("tr").hide(500);

});

public ActionResult Update(int Id, int Qty)

{

XSession.Cart.Update(Id, Qty);

var data = new

{

Count = XSession.Cart.Count,

Amount = String.Format("{0:#,###.#0}", XSession.Cart.Amount)

};

return Json(data, JsonRequestBehavior.AllowGet);

}

$(".nn-update-cart").click(function () {

var id = $(this).attr("data-id");

var qty = $(this).val();

var price = $(this).attr("data-price");

var discount = $(this).attr("data-discount");

var amount = qty \* price \* (1 - discount);

amount = Math.round(amount \* 100) / 100;

var td = $(this).parents("tr").find(".nn-amount");

$.ajax({

url: "/Cart/Update",

data: { id: id, qty: qty },

dataType: "json",

success: function (response) {

$("#cart-cnt").html(response.Count);

$("#cart-amt").html(response.Amount);

td.html("$"+amount);

}

});

});

public ActionResult Clear()

{

XSession.Cart.Clear();

return View("Index");

}

public ActionResult Index()

{

return View();

}

}

}

View Cart

//Trang index cua cart

@{

ViewBag.Title = "Index";

}

<h2>Your Shopping Cart</h2>

@if (XSession.Cart.Count == 0)

{

<h3 class="text-center alert alert-danger">YOUR IS EMTPTY</h3>

}

else

{

<table class="table">

<thead>

<tr>

<th>Id</th>

<th>Name</th>

<th>Unit Price</th>

<th>Discount</th>

<th>Quantity</th>

<th>Amount</th>

<th></th>

</tr>

</thead>

<tbody>

@foreach (var p in XSession.Cart.Items)

{

<tr>

<td>@p.Id</td>

<td>@p.Name</td>

<td>@Html.FormatValue(p.UnitPrice, "{0:c}")</td>

<td>@Html.FormatValue(p.Discount, "{0:p}")</td>

<td>

<input value="@p.Quantity" type="number" min="1" style="width:50px"

class="nn-update-cart"

data-id="@p.Id"

data-price="@p.UnitPrice"

data-discount="@p.Discount" />

</td>

<td class="nn-amount">@Html.FormatValue((p.Quantity \* p.UnitPrice \* (1 - p.Discount)), "{0:c}")</td>

<td>

<button class="btn btn-sm btn-default nn-remove-from-cart" data-id="@p.Id">

<span class="glyphicon glyphicon-trash"></span>

</button>

</td>

</tr>

}

</tbody>

</table>

<div>

<a class="btn btn-default" href="/Cart/Clear">

<span class="glyphicon glyphicon-trash"></span> Clear

</a>

<a class="btn btn-default" href="/Order/Checkout">

Checkout <span class="glyphicon glyphicon-hand-right"></span>

</a>

</div>

}

using EShop.Filters;

using EShop.Models;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

namespace EShop.Controllers

{

[Authenticate]

public class OrderController : EShopController

{

// GET: Order

public ActionResult Checkout()

{

var model = new Order();

model.OrderDate = DateTime.Now;

model.CustomerId = XSession.User.Id;

model.Amount = XSession.Cart.Amount;

model.Receiver = XSession.User.Fullname;

model.RequireDate = DateTime.Now.AddDays(5);

return View(model);

}

[HttpPost]

public ActionResult Checkout(Order order)

{

dbc.Orders.Add(order);

foreach (var p in XSession.Cart.Items)

{

var detail = new OrderDetail

{

Discount = p.Discount,

Order = order,

ProductId = p.Id,

Quantity = p.Quantity,

UnitPrice = p.UnitPrice

};

dbc.OrderDetails.Add(detail);

}

dbc.SaveChanges();

ModelState.AddModelError("", "Đặt hàng thành công!");

XSession.Cart.Clear();

return View();

}

//view checkout

@model Doan\_13.Models.Order

@{

ViewBag.Title = "Đặt hàng";

}

<div class="panel panel-danger">

<div class="panel-heading">CÁC MẶT HÀNG ĐÃ CHỌN</div>

<div class="panel-body">

<table class="table">

<thead>

<tr>

<th>Id</th>

<th>Name</th>

<th>Unit Price</th>

<th>Discount</th>

<th>Quantity</th>

<th>Amount</th>

<th></th>

</tr>

</thead>

<tbody>

@foreach (var p in XSession.Cart.Items)

{

<tr>

<td>@p.Id</td>

<td>@p.Name</td>

<td>@Html.FormatValue(p.UnitPrice, "{0:c}")</td>

<td>@Html.FormatValue(p.Discount, "{0:p}")</td>

<td>@p.Quantity</td>

<td>@Html.FormatValue((p.Quantity \* p.UnitPrice \* (1 - p.Discount)), "{0:c}")</td>

</tr>

}

</tbody>

</table>

</div>

</div>

<div class="panel panel-default">

<div class="panel-heading">ĐẶT HÀNG</div>

<div class="panel-body">

@Html.ValidationSummary()

@using (Html.BeginForm())

{

<div class="row">

<div class="form-group col-sm-4">

<label>Mã đơn hàng</label>

@Html.TextBoxFor(m => m.Id, new { @class = "form-control", @readonly = "true" })

</div>

<div class="form-group col-sm-4">

<label>Khách hàng</label>

@Html.TextBoxFor(m => m.CustomerId, new { @class = "form-control", @readonly = "true" })

</div>

<div class="form-group col-sm-4">

<label>Ngày đặt</label>

@Html.TextBoxFor(m => m.OrderDate, "{0:MM/dd/yyyy}", new { @class = "form-control", @readonly = "true" })

</div>

</div>

<div class="row">

<div class="form-group col-sm-4">

<label>Ngày cần</label>

@Html.TextBoxFor(m => m.RequireDate, "{0:MM/dd/yyyy}", new { @class = "form-control datepicker" })

</div>

<div class="form-group col-sm-4">

<label>Người nhận</label>

@Html.TextBoxFor(m => m.Receiver, new { @class = "form-control" })

</div>

<div class="form-group col-sm-4">

<label>Địa chỉ nhận</label>

@Html.TextBoxFor(m => m.Address, new { @class = "form-control" })

</div>

</div>

<div class="row">

<div class="form-group col-sm-4">

<label>Tổng tiền</label>

@Html.TextBoxFor(m => m.Amount, new { @class = "form-control", @readonly = "true" })

</div>

</div>

<div class="row">

<div class="form-group col-sm-12">

<label>Ghi chú</label>

@Html.TextAreaFor(m => m.Description, new { @class = "form-control", rows = "4" })

</div>

<div class="form-group col-sm-12">

<button class="btn btn-danger">Đặt hàng</button>

</div>

</div>

}

</div>

</div>

public ActionResult List()

{

var model = dbc.Orders

.Where(o => o.CustomerId == XSession.User.Id)

.OrderByDescending(o => o.OrderDate)

.ToList();

return View(model);

}

//View List

@model List<EShop.Models.Order>

@{

ViewBag.Title = "List";

}

<h2>List</h2>

<table class="table">

<thead>

<tr>

<th>Id</th>

<th>Order Date</th>

<th>Receiver</th>

<th>Address</th>

<th>Amount</th>

<th></th>

</tr>

</thead>

<tbody>

@foreach (var o in Model)

{

<tr>

<td>@o.Id</td>

<td>@o.OrderDate</td>

<td>@o.Receiver</td>

<td>@o.Address</td>

<td>@o.Amount</td>

<td>

<a class="btn btn-sm btn-default" href="/Order/Detail/@o.Id">

<span class="glyphicon glyphicon-search"></span> Chi tiết

</a>

</td>

</tr>

}

</tbody>

</table>

public ActionResult Detail(int Id)

{

var model = dbc.Orders.Find(Id);

return View(model);

}

//view chi tiet don hang order

@model EShop.Models.Order

@{

ViewBag.Title = "Đặt hàng";

}

<div class="panel panel-default">

<div class="panel-heading">ĐẶT HÀNG</div>

<div class="panel-body">

@Html.ValidationSummary()

@using (Html.BeginForm("Cancel", "Order"))

{

<div class="row">

<div class="form-group col-sm-4">

<label>Mã đơn hàng</label>

@Html.TextBoxFor(m => m.Id, new { @class = "form-control", @readonly = "true" })

</div>

<div class="form-group col-sm-4">

<label>Khách hàng</label>

@Html.TextBoxFor(m => m.CustomerId, new { @class = "form-control", @readonly = "true" })

</div>

<div class="form-group col-sm-4">

<label>Ngày đặt</label>

@Html.TextBoxFor(m => m.OrderDate, "{0:MM/dd/yyyy}", new { @class = "form-control", @readonly = "true" })

</div>

</div>

<div class="row">

<div class="form-group col-sm-4">

<label>Ngày cần</label>

@Html.TextBoxFor(m => m.RequireDate, "{0:MM/dd/yyyy}", new { @class = "form-control datepicker", @readonly = "true" })

</div>

<div class="form-group col-sm-4">

<label>Người nhận</label>

@Html.TextBoxFor(m => m.Receiver, new { @class = "form-control", @readonly = "true" })

</div>

<div class="form-group col-sm-4">

<label>Địa chỉ nhận</label>

@Html.TextBoxFor(m => m.Address, new { @class = "form-control", @readonly = "true" })

</div>

</div>

<div class="row">

<div class="form-group col-sm-4">

<label>Tổng tiền</label>

@Html.TextBoxFor(m => m.Amount, new { @class = "form-control", @readonly = "true" })

</div>

</div>

<div class="row">

<div class="form-group col-sm-12">

<label>Ghi chú</label>

@Html.TextAreaFor(m => m.Description, new { @class = "form-control", rows = "4", @readonly = "true" })

</div>

<div class="form-group col-sm-12">

<button class="btn btn-danger">Hủy</button>

</div>

</div>

}

</div>

</div>

<div class="panel panel-danger">

<div class="panel-heading">CHI TIẾT ĐƠN HÀNG</div>

<div class="panel-body">

<table class="table">

<thead>

<tr>

<th>Id</th>

<th>Name</th>

<th>Unit Price</th>

<th>Discount</th>

<th>Quantity</th>

<th>Amount</th>

<th></th>

</tr>

</thead>

<tbody>

@foreach (var d in Model.OrderDetails)

{

<tr>

<td>@d.Product.Id</td>

<td>@d.Product.Name</td>

<td>@Html.FormatValue(d.UnitPrice, "{0:c}")</td>

<td>@Html.FormatValue(d.Discount, "{0:p}")</td>

<td>@d.Quantity</td>

<td>@Html.FormatValue((d.Quantity \* d.UnitPrice \* (1 - d.Discount)), "{0:c}")</td>

</tr>

}

</tbody>

</table>

</div>

</div>

public ActionResult Cancel(int Id)

{

var model = dbc.Orders.Find(Id);

dbc.Orders.Remove(model);

dbc.SaveChanges();

return RedirectToAction("List");

}

//}

public ActionResult Items()

{

var model = dbc.OrderDetails

.Where(d => d.Order.CustomerId == XSession.User.Id)

.Select(d => d.Product)

.Distinct()

.ToList();

return View("../Product/List", model);

}

}

}

//Phan quyen

\_Layout

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Website Administration</title>

@Styles.Render("~/Content/css")

@Scripts.Render("~/bundles/modernizr")

@Scripts.Render("~/bundles/jquery")

@Scripts.Render("~/bundles/bootstrap")

<script src="~/Scripts/eshop.js"></script>

@RenderSection("scripts", required: false)

</head>

<body>

<div class="container">

<div class="navbar navbar-inverse row">

<div class="navbar-header">

<button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<a href="/" class="navbar-brand">Trang chủ</a>

</div>

<div class="navbar-collapse collapse">

<ul class="nav navbar-nav">

<li class="dropdown">

<a href="#" class="dropdown-toggle" data-toggle="dropdown">Quản lý <span class="caret"></span></a>

<ul class="dropdown-menu">

<li><a href="/Admin/Product">Hàng hóa</a></li>

<li><a href="/Admin/Supplier">Nhà cung cấp</a></li>

<li><a href="/Admin/Category">Chủng loại</a></li>

<li><a href="/Admin/Customer">Khách hàng</a></li>

<li><a href="/Admin/Order">Hóa đơn</a></li>

</ul>

</li>

<li class="dropdown">

<a href="#" class="dropdown-toggle" data-toggle="dropdown">Thống kê <span class="caret"></span></a>

<ul class="dropdown-menu">

<li><a href="/Admin/Inventory/ByCategory">Theo loại</a></li>

<li><a href="/Admin/Inventory/BySupplier">Theo hãng</a></li>

<li class="divider"></li>

<li><a href="/Admin/Revenue/ByProduct">Theo từng mặt hàng</a></li>

<li><a href="/Admin/Revenue/ByCategory">Theo từng loại hàng</a></li>

<li><a href="/Admin/Revenue/BySupplier">Theo từng hãng</a></li>

<li><a href="/Admin/Revenue/ByCustomer">Theo từng khách hàng</a></li>

<li class="divider"></li>

<li><a href="/Admin/Revenue/ByYear">Theo từng năm</a></li>

<li><a href="/Admin/Revenue/ByQuarter">Theo từng quí</a></li>

<li><a href="/Admin/Revenue/ByMonth">Theo từng tháng</a></li>

</ul>

</li>

</ul>

<ul class="nav navbar-nav navbar-right">

<li><a href="#" class="dropdown-toggle" data-toggle="dropdown">Tài khoản <span class="caret"></span></a>

<ul class="dropdown-menu">

<li><a href="/Admin/Home/Logoff">Đăng xuất</a></li>

<li><a href="">Đổi mật khẩu</a></li>

<li class="divider"></li>

<li><a href="/Admin/Master">Quản lý tài khoản</a></li>

<li><a href="/Admin/Role">Quản lý vai trò</a></li>

<li><a href="/Admin/WebAction">Quản lý chức năng</a></li>

<li class="divider"></li>

<li><a href="/Admin/MasterRole">Phân vai trò</a></li>

<li><a href="/Admin/ActionRole">Phân quyền</a></li>

</ul>

</li>

</ul>

</div>

</div>

<div class="nn-sheet row">

<article class="nn-body">

@RenderBody()

</article>

</div>

<footer class="panel panel-default row">

<div class="panel-heading">

<p class="text-center">Nhất Nghệ &copy; 2017. All rights reserved.</p>

</div>

</footer>

</div>

</body>

</html>

HomeController

public ActionResult Login(String Reason = "")

{

if (Reason.Length > 0)

{

ModelState.AddModelError("", Reason);

}

return View();

}

[HttpPost]

public ActionResult Login(String Id, String Password)

{

var Master = dbc.Masters.Find(Id);

if (Master == null)

{

ModelState.AddModelError("", "Sai tên đăng nhập");

}

else if (Master.Password != Password)

{

ModelState.AddModelError("", "Sai mật khẩu");

}

else

{

Session["Master"] = Master;

ModelState.AddModelError("", "Đăng nhập thành công");

}

return View();

}

//view login

@{

ViewBag.Title = "Đăng nhập";

}

<div class="panel panel-default">

<div class="panel-heading">ĐĂNG NHẬP</div>

<div class="panel-body">

@Html.ValidationSummary()

@using(Html.BeginForm()){

<div class="form-group">

<label>Tên đăng nhập</label>

@Html.TextBox("Id", null, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Mật khẩu</label>

@Html.TextBox("Password", null, new { @class = "form-control" })

</div>

<div class="form-group">

<button class="btn btn-danger">Đăng nhập</button>

</div>

}

</div>

</div>

//MasterControoler

public ActionResult Index()

{

var model = new Master();

ViewBag.Items = dbc.Masters.ToList();

return View(model);

}

@model EShop.Models.Master

@{

ViewBag.Title = "Master Manager";

ViewBag.Controller = "/Admin/Master";

}

<h2>@ViewBag.Title</h2>

<ul class="nav nav-tabs">

<li class="active">

<a data-toggle="tab" href="#edit"><span class="glyphicon glyphicon-edit"></span> CẬP NHẬT</a>

</li>

<li>

<a data-toggle="tab" href="#list"><span class="glyphicon glyphicon-list-alt"></span> DANH SÁCH</a>

</li>

</ul>

<div class="tab-content">

<div id="edit" class="tab-pane fade in active">

@Html.Partial("\_Edit", Model)

</div>

<div id="list" class="tab-pane fade">

@Html.Partial("\_List")

</div>

</div>

\_Edit.cshtml

@model EShop.Models.Master

@using (Html.BeginForm())

{

<div class="panel panel-default">

<div class="panel-body">

<div class="text-right text-danger">@TempData["Message"]</div>

<div class="form-group">

<label>Id</label>

@Html.TextBoxFor(m => m.Id, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Password</label>

@Html.TextBoxFor(m => m.Password, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Full Name</label>

@Html.TextBoxFor(m => m.FullName, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Mobile</label>

@Html.TextBoxFor(m => m.Mobile, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Email</label>

@Html.TextBoxFor(m => m.Email, new { @class = "form-control" })

</div>

</div>

<div class="panel-footer">

<button class="btn btn-default" formaction="@ViewBag.Controller/Insert">

<span class="glyphicon glyphicon-save"></span> Create

</button>

<button class="btn btn-default" formaction="@ViewBag.Controller/Update">

<span class="glyphicon glyphicon-saved"></span> Update

</button>

<button class="btn btn-default" formaction="@ViewBag.Controller/Delete">

<span class="glyphicon glyphicon-trash"></span> Delete

</button>

<a class="btn btn-default" href="@ViewBag.Controller/Index">

<span class="glyphicon glyphicon-plus"></span> New

</a>

</div>

</div>

}

\_List.cshtml

<div class="panel panel-default">

<table class="table table-hover table-condensed">

<thead>

<tr>

<th>Id</th>

<th>Name</th>

<th>Password</th>

<th>Mobile</th>

<th>Email</th>

<th></th>

</tr>

</thead>

<tbody>

@foreach (var Item in ViewBag.Items as List<EShop.Models.Master>)

{

<tr>

<td>@Item.Id</td>

<td>@Item.FullName</td>

<td>@Item.Password</td>

<td>@Item.Mobile</td>

<td>@Item.Email</td>

<td>

<a href="@ViewBag.Controller/Edit/@Item.Id" class="btn btn-sm btn-default">

<span class="glyphicon glyphicon-edit"></span> Edit

</a>

<a href="@ViewBag.Controller//Delete/@Item.Id" class="btn btn-sm btn-default">

<span class="glyphicon glyphicon-trash"></span> Delete

</a>

</td>

</tr>

}

</tbody>

</table>

</div>

public ActionResult Edit(String Id)

{

var model = dbc.Masters.Find(Id);

ViewBag.Items = dbc.Masters.ToList();

return View("Index", model);

}

public ActionResult Insert(Master entity)

{

try

{

dbc.Masters.Add(entity);

dbc.SaveChanges();

TempData["Message"] = "Thêm mới thành công!";

}

catch

{

TempData["Message"] = "Thêm mới thất bại!";

}

return RedirectToAction("Index");

}

public ActionResult Update(Master entity)

{

try

{

dbc.Entry(entity).State = System.Data.EntityState.Modified;

dbc.SaveChanges();

TempData["Message"] = "Cập nhật thành công!";

}

catch

{

TempData["Message"] = "Cập nhật thất bại!";

}

return RedirectToAction("Edit", new { entity.Id });

}

public ActionResult Delete(String Id)

{

try

{

var entity = dbc.Masters.Find(Id);

dbc.Masters.Remove(entity);

dbc.SaveChanges();

TempData["Message"] = "Xóa thành công!";

}

catch

{

TempData["Message"] = "Xóa thất bại!";

}

return RedirectToAction("Index");

}

//Role Controller

public ActionResult Index()

{

var model = new Role();

ViewBag.Items = dbc.Roles.ToList();

return View(model);

}

//view index

@model EShop.Models.Role

@{

ViewBag.Title = "Role Manager";

ViewBag.Controller = "/Admin/Role";

}

<h2>@ViewBag.Title</h2>

<ul class="nav nav-tabs">

<li class="active">

<a data-toggle="tab" href="#edit"><span class="glyphicon glyphicon-edit"></span> CẬP NHẬT</a>

</li>

<li>

<a data-toggle="tab" href="#list"><span class="glyphicon glyphicon-list-alt"></span> DANH SÁCH</a>

</li>

</ul>

<div class="tab-content">

<div id="edit" class="tab-pane fade in active">

@Html.Partial("\_Edit", Model)

</div>

<div id="list" class="tab-pane fade">

@Html.Partial("\_List")

</div>

</div>

\_Edit.cshtml

@model EShop.Models.Role

@using (Html.BeginForm())

{

<div class="panel panel-default">

<div class="panel-body">

<div class="text-right text-danger">@TempData["Message"]</div>

<div class="form-group">

<label>Id</label>

@Html.TextBoxFor(m => m.Id, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Password</label>

@Html.TextBoxFor(m => m.Name, new { @class = "form-control" })

</div>

</div>

<div class="panel-footer">

<button class="btn btn-default" formaction="@ViewBag.Controller/Insert">

<span class="glyphicon glyphicon-save"></span> Create

</button>

<button class="btn btn-default" formaction="@ViewBag.Controller/Update">

<span class="glyphicon glyphicon-saved"></span> Update

</button>

<button class="btn btn-default" formaction="@ViewBag.Controller/Delete">

<span class="glyphicon glyphicon-trash"></span> Delete

</button>

<a class="btn btn-default" href="@ViewBag.Controller/Index">

<span class="glyphicon glyphicon-plus"></span> New

</a>

</div>

</div>

}

\_List.cshtml

<div class="panel panel-default">

<table class="table table-hover table-condensed">

<thead>

<tr>

<th>Id</th>

<th>Name</th>

<th>Password</th>

<th>Mobile</th>

<th>Email</th>

<th></th>

</tr>

</thead>

<tbody>

@foreach (var Item in ViewBag.Items as List<EShop.Models.Role>)

{

<tr>

<td>@Item.Id</td>

<td>@Item.Name</td>

<td>

<a href="@ViewBag.Controller/Edit/@Item.Id" class="btn btn-sm btn-default">

<span class="glyphicon glyphicon-edit"></span> Edit

</a>

<a href="@ViewBag.Controller//Delete/@Item.Id" class="btn btn-sm btn-default">

<span class="glyphicon glyphicon-trash"></span> Delete

</a>

</td>

</tr>

}

</tbody>

</table>

</div>

public ActionResult Edit(String Id)

{

var model = dbc.Roles.Find(Id);

ViewBag.Items = dbc.Roles.ToList();

return View("Index", model);

}

public ActionResult Insert(Role entity)

{

try

{

dbc.Roles.Add(entity);

dbc.SaveChanges();

TempData["Message"] = "Thêm mới thành công!";

}

catch

{

TempData["Message"] = "Thêm mới thất bại!";

}

return RedirectToAction("Index");

}

public ActionResult Update(Role entity)

{

try

{

dbc.Entry(entity).State = System.Data.EntityState.Modified;

dbc.SaveChanges();

TempData["Message"] = "Cập nhật thành công!";

}

catch

{

TempData["Message"] = "Cập nhật thất bại!";

}

return RedirectToAction("Edit", new { entity.Id });

}

public ActionResult Delete(String Id)

{

try

{

var entity = dbc.Roles.Find(Id);

dbc.Roles.Remove(entity);

dbc.SaveChanges();

TempData["Message"] = "Xóa thành công!";

}

catch

{

TempData["Message"] = "Xóa thất bại!";

}

return RedirectToAction("Index");

}

//WebAction

public ActionResult Index()

{

var model = new WebAction();

ViewBag.Items = dbc.WebActions.ToList();

return View(model);

}

@model Doan\_13.Models.WebAction

@{

ViewBag.Title = "WebAction Manager";

ViewBag.Controller = "/Admin/WebAction";

}

<h2>@ViewBag.Title</h2>

<ul class="nav nav-tabs">

<li class="active">

<a data-toggle="tab" href="#edit"><span class="glyphicon glyphicon-edit"></span> CẬP NHẬT</a>

</li>

<li>

<a data-toggle="tab" href="#list"><span class="glyphicon glyphicon-list-alt"></span> DANH SÁCH</a>

</li>

</ul>

<div class="tab-content">

<div id="edit" class="tab-pane fade in active">

@Html.Partial("\_Edit", Model)

</div>

<div id="list" class="tab-pane fade">

@Html.Partial("\_List")

</div>

</div>

\_List

<div class="panel panel-default">

<table class="table table-hover table-condensed">

<thead>

<tr>

<th>Id</th>

<th>Name</th>

<th>Description</th>

<th></th>

</tr>

</thead>

<tbody>

@foreach (var Item in ViewBag.Items as List<EShop.Models.WebAction>)

{

<tr>

<td>@Item.Id</td>

<td>@Item.Name</td>

<td>@Item.Description</td>

<td>

<a href="@ViewBag.Controller/Edit/@Item.Id" class="btn btn-sm btn-default">

<span class="glyphicon glyphicon-edit"></span> Edit

</a>

<a href="@ViewBag.Controller//Delete/@Item.Id" class="btn btn-sm btn-default">

<span class="glyphicon glyphicon-trash"></span> Delete

</a>

</td>

</tr>

}

</tbody>

</table>

</div>

\_Edit

@model EShop.Models.WebAction

@using (Html.BeginForm())

{

<div class="panel panel-default">

<div class="panel-body">

<div class="text-right text-danger">@TempData["Message"]</div>

<div class="form-group">

<label>Id</label>

@Html.TextBoxFor(m => m.Id, new { @class = "form-control", @readonly = true })

</div>

<div class="form-group">

<label>Name</label>

@Html.TextBoxFor(m => m.Name, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Description</label>

@Html.TextBoxFor(m => m.Description, new { @class = "form-control" })

</div>

</div>

<div class="panel-footer">

<button class="btn btn-default" formaction="@ViewBag.Controller/Insert">

<span class="glyphicon glyphicon-save"></span> Create

</button>

<button class="btn btn-default" formaction="@ViewBag.Controller/Update">

<span class="glyphicon glyphicon-saved"></span> Update

</button>

<button class="btn btn-default" formaction="@ViewBag.Controller/Delete">

<span class="glyphicon glyphicon-trash"></span> Delete

</button>

<a class="btn btn-default" href="@ViewBag.Controller/Index">

<span class="glyphicon glyphicon-plus"></span> New

</a>

</div>

</div>

}

public ActionResult Edit(int Id)

{

var model = dbc.WebActions.Find(Id);

ViewBag.Items = dbc.WebActions.ToList();

return View("Index", model);

}

public ActionResult Insert(WebAction entity)

{

try

{

dbc.WebActions.Add(entity);

dbc.SaveChanges();

TempData["Message"] = "Thêm mới thành công!";

}

catch

{

TempData["Message"] = "Thêm mới thất bại!";

}

return RedirectToAction("Index");

}

public ActionResult Update(WebAction entity)

{

try

{

dbc.Entry(entity).State = System.Data.EntityState.Modified;

dbc.SaveChanges();

TempData["Message"] = "Cập nhật thành công!";

}

catch

{

TempData["Message"] = "Cập nhật thất bại!";

}

return RedirectToAction("Edit", new { entity.Id });

}

public ActionResult Delete(int Id)

{

try

{

var entity = dbc.WebActions.Find(Id);

dbc.WebActions.Remove(entity);

dbc.SaveChanges();

TempData["Message"] = "Xóa thành công!";

}

catch

{

TempData["Message"] = "Xóa thất bại!";

}

return RedirectToAction("Index");

}

//MasterRole

public ActionResult Index()

{

ViewBag.MasterId = new SelectList(dbc.Masters, "Id", "FullName");

ViewBag.Roles = dbc.Roles.ToList();

return View();

}

@{

ViewBag.Title = "Index";

}

<h2>Master Role Manager</h2>

@Html.DropDownList("MasterId", null, new { @class = "form-control" })

<ul>

@foreach (var role in ViewBag.Roles)

{

<li>

<label>

<input type="checkbox" value="@role.Id" /> @role.Name

</label>

</li>

}

</ul>

<script>

$(document).ready(function () {

$("#MasterId").change(function () {

var mid = $(this).val();

$.ajax({

url: "/Admin/MasterRole/GetRoleIds",

data: {MasterId: mid},

success: function (response) {

$(":checkbox").prop("checked", false);

for (var i = 0; i < response.length; i++) {

var roleId = response[i];

$(":checkbox[value=" + roleId + "]").prop("checked", true);

}

}

});

});

$("#MasterId").change();

$(":checkbox").click(function () {

var rid = $(this).val();

var mid = $("#MasterId").val();

$.ajax({

url: "/Admin/MasterRole/AddOrDelete",

data: { MasterId: mid, RoleId: rid },

success: function (response) {

alert("Cập nhật thành công");

}

});

});

});

</script>

//ActionRole

public ActionResult Index()

{

ViewBag.RoleId = new SelectList(dbc.Roles, "Id", "Name");

ViewBag.WebActions = dbc.WebActions.ToList();

return View();

}

@{

ViewBag.Title = "Index";

}

<h2>Master Role Manager</h2>

<div id="messsage"></div>

@Html.DropDownList("RoleId", null, new { @class = "form-control" })

<ul class="list-unstyled">

@foreach (var action in ViewBag.WebActions)

{

<li class="col-sm-3">

<label>

<input type="checkbox" value="@action.Id" /> @action.Name

</label>

</li>

}

</ul>

<script>

$(document).ready(function () {

$("#RoleId").change(function () {

var rid = $(this).val();

$.ajax({

url: "/Admin/ActionRole/GetActionIds",

data: {RoleId: rid},

success: function (response) {

$(":checkbox").prop("checked", false);

for (var i = 0; i < response.length; i++) {

var actionId = response[i];

$(":checkbox[value=" + actionId + "]").prop("checked", true);

}

}

});

});

$("#RoleId").change();

$(":checkbox").click(function () {

var aid = $(this).val();

var rid = $("#RoleId").val();

$.ajax({

url: "/Admin/ActionRole/AddOrDelete",

data: { WebActionId: aid, RoleId: rid },

success: function (response) {

$("#messsage").html(response);

}

});

});

});

</script>

public ActionResult GetActionIds(String RoleId)

{

var WebActionIds = dbc.ActionRoles

.Where(ar => ar.RoleId == RoleId)

.Select(ar => ar.WebActionId).ToList();

return Json(WebActionIds, JsonRequestBehavior.AllowGet);

}

public ActionResult AddOrDelete(ActionRole model)

{

var message = "";

try

{

var ActionRole = dbc.ActionRoles

.Single(mr => mr.RoleId == model.RoleId && mr.WebActionId == model.WebActionId);

dbc.ActionRoles.Remove(ActionRole);

message = "Xóa quyền";

}

catch

{

dbc.ActionRoles.Add(model);

message = "Thêm quyền";

}

dbc.SaveChanges();

return Content(message);

}

//Inventory Thong ke

public ActionResult ByCategory()

{

ViewBag.Group = "Category";

var data = dbc.Products

.GroupBy(p => p.Category)

.Select(g => new Report

{

Group = g.Key.NameVN,

Quantity = g.Sum(p => p.Quantity),

Value = g.Sum(p => p.UnitPrice \* p.Quantity),

MinPrice = g.Min(p => p.UnitPrice),

MaxPrice = g.Max(p => p.UnitPrice),

AvgPrice = g.Average(p => p.UnitPrice)

})

.ToList();

return View("Inventory", data);

}

public ActionResult BySupplier()

{

ViewBag.Group = "Supplier";

var data = dbc.Products

.GroupBy(p => p.Supplier)

.Select(g => new Report

{

Group = g.Key.Name,

Quantity = g.Sum(p => p.Quantity),

Value = g.Sum(p => p.UnitPrice \* p.Quantity),

MinPrice = g.Min(p => p.UnitPrice),

MaxPrice = g.Max(p => p.UnitPrice),

AvgPrice = g.Average(p => p.UnitPrice)

})

.ToList();

return View("Inventory", data);

}

//Revenue

public ActionResult ByCategory()

{

var data = dbc.OrderDetails

.GroupBy(d => d.Product.Category)

.Select(g => new Report

{

Group = g.Key.NameVN,

Quantity = g.Sum(p => p.Quantity),

Value = g.Sum(p => p.UnitPrice \* p.Quantity),

MinPrice = g.Min(p => p.UnitPrice),

MaxPrice = g.Max(p => p.UnitPrice),

AvgPrice = g.Average(p => p.UnitPrice)

})

.ToList();

return View("Revenue", data);

}

public ActionResult BySupplier()

{

var data = dbc.OrderDetails

.GroupBy(d => d.Product.Supplier)

.Select(g => new Report

{

Group = g.Key.Name,

Quantity = g.Sum(p => p.Quantity),

Value = g.Sum(p => p.UnitPrice \* p.Quantity),

MinPrice = g.Min(p => p.UnitPrice),

MaxPrice = g.Max(p => p.UnitPrice),

AvgPrice = g.Average(p => p.UnitPrice)

})

.ToList();

return View("Revenue", data);

}

public ActionResult ByYear()

{

var data = dbc.OrderDetails

.GroupBy(d => d.Order.OrderDate.Year)

.ToList()

.Select(g => new Report

{

Group = g.Key.ToString(),

Quantity = g.Sum(p => p.Quantity),

Value = g.Sum(p => p.UnitPrice \* p.Quantity),

MinPrice = g.Min(p => p.UnitPrice),

MaxPrice = g.Max(p => p.UnitPrice),

AvgPrice = g.Average(p => p.UnitPrice)

})

.ToList();

return View("Revenue", data);

}

public ActionResult ByQuarter()

{

var data = dbc.OrderDetails

.GroupBy(d => Math.Ceiling(d.Order.OrderDate.Month/3.0))

.ToList()

.Select(g => new Report

{

Group = g.Key.ToString(),

Quantity = g.Sum(p => p.Quantity),

Value = g.Sum(p => p.UnitPrice \* p.Quantity),

MinPrice = g.Min(p => p.UnitPrice),

MaxPrice = g.Max(p => p.UnitPrice),

AvgPrice = g.Average(p => p.UnitPrice)

})

.ToList();

return View("Revenue", data);

}

@{

ViewBag.Title = "Inventory";

}

<h2>Inventory by Category</h2>

<div class="panel panel-success">

<table class="table table-hover">

<thead>

<tr>

<th>Category</th>

<th>Quantity</th>

<th>Value</th>

<th>Min Price</th>

<th>Max Price</th>

<th>Avg Price</th>

</tr>

</thead>

<tbody>

@foreach (var Item in Model as List<EShop.Areas.Admin.Models.Report>)

{

<tr>

<td>@Item.Group</td>

<td>@Item.Quantity</td>

<td>@Item.Value</td>

<td>@Item.MinPrice</td>

<td>@Item.MaxPrice</td>

<td>@Item.AvgPrice</td>

</tr>

}

</tbody>

</table>

</div>

//Product

public ActionResult Index()

{

ViewBag.Items = dbc.Products.ToList();

var model = new Product();

ViewBag.CategoryId = new SelectList(dbc.Categories, "Id", "NameVN");

ViewBag.SupplierId = new SelectList(dbc.Suppliers, "Id", "Name");

return View(model);

}

@model EShop.Models.Product

@{

ViewBag.Title = "Product Manager";

ViewBag.Controller = "/Admin/Product";

}

<h2 class="alert alert-success">@ViewBag.Title</h2>

<ul class="nav nav-tabs">

<li class="active">

<a data-toggle="tab" href="#edit">

<span class="glyphicon glyphicon-edit"></span> Edit

</a>

</li>

<li>

<a data-toggle="tab" href="#list">

<span class="glyphicon glyphicon-edit"></span> List

</a>

</li>

</ul>

<div class="tab-content">

<div id="edit" class="tab-pane fade in active">

@Html.Partial("\_Edit", Model)

</div>

<div id="list" class="tab-pane fade">

@Html.Partial("\_Table")

</div>

</div>

\_Edit

@model EShop.Models.Product

@using (Html.BeginForm("Index", "Product",

FormMethod.Post, new { enctype = "multipart/form-data" }))

{

<div class="panel panel-danger">

<div class="panel-heading">@TempData["Message"]</div>

<div class="panel-body">

<div class="row">

<div class="form-group col-sm-4">

<label>Id</label>

@Html.TextBoxFor(m => m.Id, new { @class = "form-control" })

</div>

<div class="form-group col-sm-4">

<label>Name</label>

@Html.TextBoxFor(m => m.Name, new { @class = "form-control" })

</div>

<div class="form-group col-sm-4">

<label>Unit Price</label>

@Html.TextBoxFor(m => m.UnitPrice, new { @class = "form-control" })

</div>

</div>

<div class="row">

<div class="form-group col-sm-4">

<label>Unit Brief</label>

@Html.TextBoxFor(m => m.UnitBrief, new { @class = "form-control" })

</div>

<div class="form-group col-sm-4">

<label>Discount</label>

@Html.TextBoxFor(m => m.Discount, new { @class = "form-control" })

</div>

<div class="form-group col-sm-4">

<label>Quantity</label>

@Html.TextBoxFor(m => m.Quantity, new { @class = "form-control" })

</div>

</div>

<div class="row">

<div class="form-group col-sm-4">

<label>Product Date</label>

@Html.TextBoxFor(m => m.ProductDate, "{0:MM/dd/yyyy}", new { @class = "form-control datepicker" })

</div>

<div class="form-group col-sm-4">

<label>Features</label>

<div>

<label>@Html.CheckBoxFor(m => m.Available) Available</label>

<label>@Html.CheckBoxFor(m => m.Special) Special</label>

<label>@Html.CheckBoxFor(m => m.Latest) Latest</label>

</div>

</div>

<div class="form-group col-sm-4">

<label>Supplier</label>

@Html.DropDownList("SupplierId", null, new { @class = "form-control" })

</div>

</div>

<div class="row">

<div class="form-group col-sm-4">

<label>Category</label>

@Html.DropDownList("CategoryId", null, new { @class = "form-control" })

</div>

<div class="form-group col-sm-4">

<label>View Count</label>

@Html.TextBoxFor(m => m.Views, new { @class = "form-control" })

</div>

<div class="form-group col-sm-4">

<label>Image</label>

<input type="file" name="UpImage" class="form-control" />

@Html.HiddenFor(m => m.Image)

</div>

</div>

<div class="row">

<div class="form-group col-sm-12">

<label>Description</label>

@Html.TextAreaFor(m => m.Description, new { @class = "form-control", rows = 4 })

</div>

</div>

</div>

<div class="panel-footer">

@Html.Partial("\_Buttons")

</div>

</div>

}

<button class="btn btn-default" formaction="@ViewBag.Controller/Insert">

<span class="glyphicon glyphicon-star"></span> Insert

</button>

<button class="btn btn-default" formaction="@ViewBag.Controller/Update">

<span class="glyphicon glyphicon-star"></span> Update

</button>

<button class="btn btn-default" formaction="@ViewBag.Controller/Delete">

<span class="glyphicon glyphicon-star"></span> Delete

</button>

<a class="btn btn-default" href="@ViewBag.Controller/Index">

<span class="glyphicon glyphicon-star"></span> New

</a>

\_Page

@foreach (var Item in ViewBag.Items as List<EShop.Models.Product>)

{

<tr>

<td>@Item.Id</td>

<td>@Item.Name</td>

<td>@Item.UnitPrice</td>

<td>@Item.UnitBrief</td>

<td>@Item.Discount</td>

<td>@Item.Special</td>

<td>@Item.Category.NameVN</td>

<td>

<a class="btn btn-sm btn-default" href="/Admin/Product/Edit/@Item.Id">

<span class="glyphicon glyphicon-star"></span> Edit

</a>

<a class="btn btn-sm btn-default" href="/Admin/Product/Delete/@Item.Id">

<span class="glyphicon glyphicon-star"></span> Delete

</a>

</td>

</tr>

}

\_Table

<ul class="pager">

<li><a href="#">First</a></li>

<li><a href="#">Prev</a></li>

<li><a></a></li>

<li><a href="#">Next</a></li>

<li><a href="#">Last</a></li>

</ul>

<div class="panel panel-success">

<table class="table table-hover">

<thead>

<tr>

<th>Id</th>

<th>Name</th>

<th>Unit Price</th>

<th>Unit Brief</th>

<th>Discount</th>

<th>Special?</th>

<th>Category</th>

<th></th>

</tr>

</thead>

<tbody id="page">

</tbody>

</table>

</div>

<script>

$(document).ready(function () {

var data = { PageSize: pageSize };

$.ajax({

url: "@ViewBag.Controller/GetPageCount",

data: data,

success: function (response) {

pageCount = response;

loadPage();

}

});

$(".pager li:eq(0)").click(function () {

pageNo = 0;

loadPage();

});

$(".pager li:eq(1)").click(function () {

pageNo--;

loadPage();

});

$(".pager li:eq(3)").click(function () {

pageNo++;

loadPage();

});

$(".pager li:eq(4)").click(function () {

pageNo = pageCount - 1;

loadPage();

});

});

var pageNo = 0;

var pageSize = 6;

var pageCount = 0;

function loadPage() {

$.ajax({

url: "@ViewBag.Controller/GetPage",

data: {PageNo: pageNo, PageSize: pageSize},

success: function (response) {

$("#page").html(response);

$(".pager a:eq(2)").html((pageNo + 1) + "/" + pageCount);

}

});

}

</script>

public ActionResult Edit(int Id)

{

ViewBag.Items = dbc.Products.ToList();

var model = dbc.Products.Find(Id);

ViewBag.CategoryId = new SelectList(dbc.Categories, "Id", "NameVN", model.CategoryId);

ViewBag.SupplierId = new SelectList(dbc.Suppliers, "Id", "Name", model.SupplierId);

return View("Index", model);

}

[ValidateInput(false)]

public ActionResult Insert(Product model)

{

var file = Request.Files["UpImage"];

if (file.ContentLength > 0)

{

model.Image = Guid.NewGuid() + file.FileName.Substring(file.FileName.LastIndexOf("."));

file.SaveAs(Server.MapPath("~/Images/Products/" + model.Image));

}

else

{

model.Image = "Product.png";

}

try

{

dbc.Products.Add(model);

dbc.SaveChanges();

TempData["Message"] = "Thêm mới thành công!";

return RedirectToAction("Index");

}

catch

{

TempData["Message"] = "Thêm mới thất bại!";

ViewBag.Items = dbc.Products.ToList();

return View("Index", model);

}

}

[ValidateInput(false)]

public ActionResult Update(Product model)

{

var file = Request.Files["UpImage"];

if (file.ContentLength > 0)

{

if (model.Image != "Product.png")

{

System.IO.File.Delete(Server.MapPath("~/Images/Products/" + model.Image));

}

model.Image = Guid.NewGuid() + file.FileName.Substring(file.FileName.LastIndexOf("."));

file.SaveAs(Server.MapPath("~/Images/Products/" + model.Image));

}

try

{

dbc.Entry(model).State = System.Data.EntityState.Modified;

dbc.SaveChanges();

TempData["Message"] = "Cập nhật thành công!";

return RedirectToAction("Edit", new { model.Id });

}

catch

{

TempData["Message"] = "Cập nhật thất bại!";

ViewBag.Items = dbc.Products.ToList();

return View("Index", model);

}

}

public ActionResult Delete(int Id)

{

var model = dbc.Products.Find(Id);

if (model.Image != "Product.png")

{

System.IO.File.Delete(Server.MapPath("~/Images/Products/" + model.Image));

}

try

{

dbc.Products.Remove(model);

dbc.SaveChanges();

TempData["Message"] = "Xóa thành công!";

return RedirectToAction("Index");

}

catch

{

TempData["Message"] = "Xóa thất bại!";

ViewBag.Items = dbc.Products.ToList();

return View("Index", model);

}

}

public ActionResult getPage(int PageNo = 0, int PageSize = 6)

{

ViewBag.Items = dbc.Products

.OrderBy(p=>p.Id)

.Skip(PageNo\*PageSize)

.Take(PageSize)

.ToList();

return PartialView("\_Page");

}

public ActionResult getPageCount(int PageSize = 6)

{

var RowCount = dbc.Products.Count();

var PageCount = Math.Ceiling(1.0 \* RowCount / PageSize);

return Content(PageCount.ToString());

}

//Category

public ActionResult Index()

{

ViewBag.Items = dbc.Categories.ToList();

var model = new Category();

return View(model);

}

//Index.cshtml

@model Doan\_13.Models.Category

@{

ViewBag.Title = "Category Manager";

ViewBag.Controller = "/Admin/Category";

}

<h2 class="alert alert-success">@ViewBag.Title</h2>

<ul class="nav nav-tabs">

<li class="active">

<a data-toggle="tab" href="#edit">

<span class="glyphicon glyphicon-edit"></span> Edit

</a>

</li>

<li>

<a data-toggle="tab" href="#list">

<span class="glyphicon glyphicon-edit"></span> List

</a>

</li>

</ul>

<div class="tab-content">

<div id="edit" class="tab-pane fade in active">

@Html.Partial("\_Edit", Model)

</div>

<div id="list" class="tab-pane fade">

@Html.Partial("\_Table")

</div>

</div>

\_Edit.cshtml

@model EShop.Models.Category

@using (Html.BeginForm())

{

<div class="panel panel-danger">

<div class="panel-heading">@TempData["Message"]</div>

<div class="panel-body">

<div class="form-group">

<label>Id</label>

@Html.TextBoxFor(m => m.Id, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Name</label>

@Html.TextBoxFor(m => m.Name, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Name VN</label>

@Html.TextBoxFor(m => m.NameVN, new { @class = "form-control" })

</div>

</div>

<div class="panel-footer">

@Html.Partial("\_Buttons")

</div>

</div>

}

\_Table.cshtml

<div class="panel panel-success">

<table class="table table-hover">

<thead>

<tr>

<th>Id</th>

<th>Name</th>

<th>Name VN</th>

<th></th>

</tr>

</thead>

<tbody>

@foreach (var Item in ViewBag.Items as List<EShop.Models.Category>)

{

<tr>

<td>@Item.Id</td>

<td>@Item.Name</td>

<td>@Item.NameVN</td>

<td>

<a class="btn btn-sm btn-default" href="@ViewBag.Controller/Edit/@Item.Id">

<span class="glyphicon glyphicon-star"></span> Edit

</a>

<a class="btn btn-sm btn-default" href="@ViewBag.Controller/Delete/@Item.Id">

<span class="glyphicon glyphicon-star"></span> Delete

</a>

</td>

</tr>

}

</tbody>

</table>

</div>

public ActionResult Edit(int Id)

{

ViewBag.Items = dbc.Categories.ToList();

var model = dbc.Categories.Find(Id);

return View("Index", model);

}

public ActionResult Insert(Category model)

{

try

{

dbc.Categories.Add(model);

dbc.SaveChanges();

TempData["Message"] = "Thêm mới thành công!";

return RedirectToAction("Index");

}

catch

{

TempData["Message"] = "Thêm mới thất bại!";

ViewBag.Items = dbc.Categories.ToList();

return View("Index", model);

}

}

public ActionResult Update(Category model)

{

try

{

dbc.Entry(model).State = System.Data.EntityState.Modified;

dbc.SaveChanges();

TempData["Message"] = "Cập nhật thành công!";

return RedirectToAction("Edit", new { model.Id });

}

catch

{

TempData["Message"] = "Cập nhật thất bại!";

ViewBag.Items = dbc.Categories.ToList();

return View("Index", model);

}

}

public ActionResult Delete(int Id)

{

var model = dbc.Categories.Find(Id);

try

{

dbc.Categories.Remove(model);

dbc.SaveChanges();

TempData["Message"] = "Xóa thành công!";

return RedirectToAction("Index");

}

catch

{

TempData["Message"] = "Xóa thất bại!";

ViewBag.Items = dbc.Categories.ToList();

return View("Index", model);

}

}

//Nha cung cap supplier

public ActionResult Index()

{

ViewBag.Items = dbc.Suppliers.ToList();

var model = new Supplier();

return View(model);

}

@model EShop.Models.Supplier

@{

ViewBag.Title = "Supplier Manager";

ViewBag.Controller = "/Admin/Supplier";

}

<h2 class="alert alert-success">@ViewBag.Title</h2>

<ul class="nav nav-tabs">

<li class="active">

<a data-toggle="tab" href="#edit">

<span class="glyphicon glyphicon-edit"></span> Edit

</a>

</li>

<li>

<a data-toggle="tab" href="#list">

<span class="glyphicon glyphicon-edit"></span> List

</a>

</li>

</ul>

<div class="tab-content">

<div id="edit" class="tab-pane fade in active">

@Html.Partial("\_Edit", Model)

</div>

<div id="list" class="tab-pane fade">

@Html.Partial("\_Table")

</div>

</div>

\_Edit.cshtml

@model EShop.Models.Supplier

@using (Html.BeginForm("Index", "Supplier",

FormMethod.Post, new { enctype = "multipart/form-data" }))

{

<div class="panel panel-danger">

<div class="panel-heading">@TempData["Message"]</div>

<div class="panel-body">

<div class="form-group">

<label>Id</label>

@Html.TextBoxFor(m => m.Id, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Name</label>

@Html.TextBoxFor(m => m.Name, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Email</label>

@Html.TextBoxFor(m => m.Email, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Phone</label>

@Html.TextBoxFor(m => m.Phone, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Logo</label>

<input type="file" name="UpLogo" class="form-control" />

@Html.HiddenFor(m => m.Logo)

</div>

</div>

<div class="panel-footer">

@Html.Partial("\_Buttons")

</div>

</div>

}

\_Table.cshtml

<div class="panel panel-success">

<table class="table table-hover">

<thead>

<tr>

<th>Id</th>

<th>Name</th>

<th>Name VN</th>

<th></th>

</tr>

</thead>

<tbody>

@foreach (var Item in ViewBag.Items as List<EShop.Models.Supplier>)

{

<tr>

<td>@Item.Id</td>

<td>@Item.Name</td>

<td>@Item.Email</td>

<td>@Item.Logo</td>

<td>@Item.Phone</td>

<td>

<a class="btn btn-sm btn-default" href="@ViewBag.Controller/Edit/@Item.Id">

<span class="glyphicon glyphicon-star"></span> Edit

</a>

<a class="btn btn-sm btn-default" href="@ViewBag.Controller/Delete/@Item.Id">

<span class="glyphicon glyphicon-star"></span> Delete

</a>

</td>

</tr>

}

</tbody>

</table>

</div>

public ActionResult Edit(String Id)

{

ViewBag.Items = dbc.Suppliers.ToList();

var model = dbc.Suppliers.Find(Id);

return View("Index", model);

}

public ActionResult Insert(Supplier model)

{

var file = Request.Files["UpLogo"];

if (file.ContentLength > 0)

{

model.Logo = Guid.NewGuid() + file.FileName.Substring(file.FileName.LastIndexOf("."));

file.SaveAs(Server.MapPath("~/Images/Suppliers/" + model.Logo));

}

else

{

model.Logo = "Supplier.png";

}

try

{

dbc.Suppliers.Add(model);

dbc.SaveChanges();

TempData["Message"] = "Thêm mới thành công!";

return RedirectToAction("Index");

}

catch

{

TempData["Message"] = "Thêm mới thất bại!";

ViewBag.Items = dbc.Suppliers.ToList();

return View("Index", model);

}

}

public ActionResult Update(Supplier model)

{

var file = Request.Files["UpLogo"];

if (file.ContentLength > 0)

{

if (model.Logo != "Supplier.png")

{

System.IO.File.Delete(Server.MapPath("~/Images/Suppliers/" + model.Logo));

}

model.Logo = Guid.NewGuid() + file.FileName.Substring(file.FileName.LastIndexOf("."));

file.SaveAs(Server.MapPath("~/Images/Suppliers/" + model.Logo));

}

try

{

dbc.Entry(model).State = System.Data.EntityState.Modified;

dbc.SaveChanges();

TempData["Message"] = "Cập nhật thành công!";

return RedirectToAction("Edit", new { model.Id });

}

catch

{

TempData["Message"] = "Cập nhật thất bại!";

ViewBag.Items = dbc.Suppliers.ToList();

return View("Index", model);

}

}

public ActionResult Delete(String Id)

{

var model = dbc.Suppliers.Find(Id);

if (model.Logo != "Supplier.png")

{

System.IO.File.Delete(Server.MapPath("~/Images/Suppliers/" + model.Logo));

}

try

{

dbc.Suppliers.Remove(model);

dbc.SaveChanges();

TempData["Message"] = "Xóa thành công!";

return RedirectToAction("Index");

}

catch

{

TempData["Message"] = "Xóa thất bại!";

ViewBag.Items = dbc.Suppliers.ToList();

return View("Index", model);

}

}

//Ordercontrooler

public ActionResult Index()

{

ViewBag.Items = dbc.Orders.ToList();

var model = new Order();

return View(model);

}

//Index.cshtml

@model EShop.Models.Order

@{

ViewBag.Title = "Order Manager";

ViewBag.Controller = "/Admin/Order";

}

<h2 class="alert alert-success">@ViewBag.Title</h2>

<ul class="nav nav-tabs">

<li class="active">

<a data-toggle="tab" href="#edit">

<span class="glyphicon glyphicon-edit"></span> Edit

</a>

</li>

<li>

<a data-toggle="tab" href="#list">

<span class="glyphicon glyphicon-edit"></span> List

</a>

</li>

</ul>

<div class="tab-content">

<div id="edit" class="tab-pane fade in active">

@Html.Partial("\_Edit", Model)

</div>

<div id="list" class="tab-pane fade">

@Html.Partial("\_Table")

</div>

</div>

\_Edit.cshtml

@model EShop.Models.Order

@using (Html.BeginForm())

{

<div class="panel panel-danger">

<div class="panel-heading">@TempData["Message"]</div>

<div class="panel-body">

<div class="form-group">

<label>Id</label>

@Html.TextBoxFor(m => m.Id, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Customer</label>

@Html.TextBoxFor(m => m.CustomerId, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Receiver</label>

@Html.TextBoxFor(m => m.Receiver, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Address</label>

@Html.TextBoxFor(m => m.Address, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Order Date</label>

@Html.TextBoxFor(m => m.OrderDate, new { @class = "form-control" })

</div>

<div class="form-group">

<label>RequireDate</label>

@Html.TextBoxFor(m => m.RequireDate, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Amount</label>

@Html.TextBoxFor(m => m.Amount, new { @class = "form-control" })

</div>

<div class="form-group">

<label>Notes</label>

@Html.TextAreaFor(m => m.Description, new { @class = "form-control", rows=4 })

</div>

</div>

@if (Model.OrderDetails.Count > 0)

{

<div class="panel-body">

@Html.Partial("\_OrderDetail", Model.OrderDetails.ToList())

</div>

}

<div class="panel-footer">

@Html.Partial("\_Buttons")

</div>

</div>

}

\_OrderDetail.cshtml

<div class="panel panel-success">

<div class="panel-heading">ORDER DETAILS</div>

<table class="table table-hover">

<thead>

<tr>

<th>Id</th>

<th>Name</th>

<th>Unit Price</th>

<th>Discount</th>

<th>Quantity</th>

<th>Amount</th>

</tr>

</thead>

<tbody>

@foreach (var Item in Model as List<EShop.Models.OrderDetail>)

{

<tr>

<td>@Item.Product.Id</td>

<td>@Item.Product.Name</td>

<td>@Item.UnitPrice</td>

<td>@Item.Discount</td>

<td>@Item.Quantity</td>

<td>@(Item.UnitPrice \* Item.Quantity \* (1 - Item.Discount))</td>

</tr>

}

</tbody>

</table>

</div>

\_Table.cshtml

<div class="panel panel-success">

<table class="table table-hover">

<thead>

<tr>

<th>Id</th>

<th>CustomerId</th>

<th>OrderDate</th>

<th>Receiver</th>

<th>Address</th>

<th>Amount</th>

<th></th>

</tr>

</thead>

<tbody id="order">

@foreach (var Item in ViewBag.Items as List<EShop.Models.Order>)

{

<tr data-id="@Item.Id">

<td>@Item.Id</td>

<td>@Item.CustomerId</td>

<td>@Item.OrderDate</td>

<td>@Item.Receiver</td>

<td>@Item.Address</td>

<td>@Item.Amount</td>

<td>

<a class="btn btn-sm btn-default" href="@ViewBag.Controller/Edit/@Item.Id">

<span class="glyphicon glyphicon-star"></span> Edit

</a>

<a class="btn btn-sm btn-default" href="@ViewBag.Controller/Delete/@Item.Id">

<span class="glyphicon glyphicon-star"></span> Delete

</a>

</td>

</tr>

}

</tbody>

</table>

</div>

<script>

$(document).ready(function () {

$("#order tr").dblclick(function () {

var id = $(this).attr("data-id");

location.href = "@ViewBag.Controller/Edit/"+id;

});

});

</script>

public ActionResult Edit(int Id)

{

ViewBag.Items = dbc.Orders.ToList();

var model = dbc.Orders.Find(Id);

return View("Index", model);

}

public ActionResult Insert(Order model)

{

try

{

dbc.Orders.Add(model);

dbc.SaveChanges();

TempData["Message"] = "Thêm mới thành công!";

return RedirectToAction("Index");

}

catch

{

TempData["Message"] = "Thêm mới thất bại!";

ViewBag.Items = dbc.Orders.ToList();

return View("Index", model);

}

}

public ActionResult Update(Order model)

{

try

{

dbc.Entry(model).State = System.Data.EntityState.Modified;

dbc.SaveChanges();

TempData["Message"] = "Cập nhật thành công!";

return RedirectToAction("Edit", new { model.Id });

}

catch

{

TempData["Message"] = "Cập nhật thất bại!";

ViewBag.Items = dbc.Orders.ToList();

return View("Index", model);

}

}

public ActionResult Delete(int Id)

{

var model = dbc.Orders.Find(Id);

try

{

dbc.Orders.Remove(model);

dbc.SaveChanges();

TempData["Message"] = "Xóa thành công!";

return RedirectToAction("Index");

}

catch

{

TempData["Message"] = "Xóa thất bại!";

ViewBag.Items = dbc.Orders.ToList();

return View("Index", model);

}

}