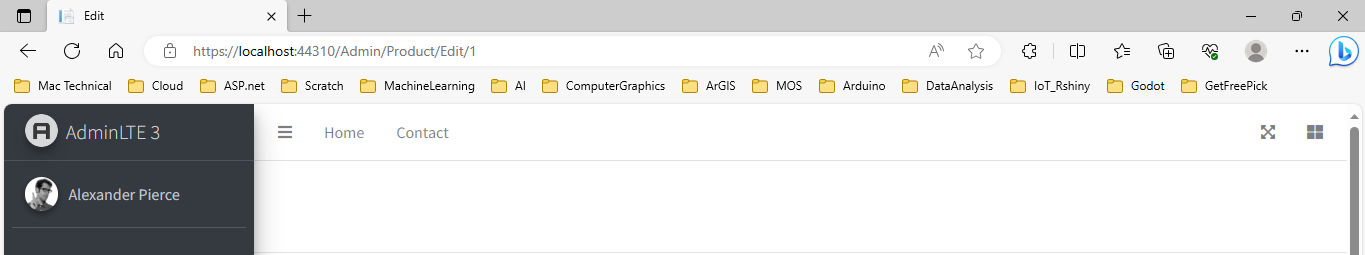
# Bài thực hành 6: xây dựng trang quản trị cho products

# Menu EDIT



Khi click chuyển đổi trạng thái trên nhóm menu chức năng,

Về giao diện trang Edit cũng giống như trang Create (chỉ Update một số trường thông tin).

## product CONTROLLER

Sao chép một số thành phần từ Categories, Suppliers

Trước hết là trả về danh sách Name (CatId, Supplier) dựa vào Id

/////////////////////////////////////////////////////////////////////////////////////

// GET: Admin/Product/Edit/5

public ActionResult Edit(int? id)

{

ViewBag.ListSupId = new SelectList(suppliersDAO.getList("Index"), "Id", "Name");

ViewBag.ListCatId = new SelectList(categoriesDAO.getList("Index"), "Id", "Name");

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

Products products = productDAO.getRow(id);

if (products == null)

{

return HttpNotFound();

}

return View(products);

}

### Đối với phương thức POST, GET

// POST: Admin/Product/Edit/5

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Edit(Products products)

{

if (ModelState.IsValid)

{

//Xu ly cho muc Slug

products.Slug = XString.Str\_Slug(products.Name);

//chuyen doi dua vao truong Name de loai bo dau, khoang cach = dau -

//xu ly cho phan upload hình ảnh

var img = Request.Files["img"];//lay thong tin file

if (img.ContentLength != 0)

{

string[] FileExtentions = new string[] { ".jpg", ".jpeg", ".png", ".gif" };

//kiem tra tap tin co hay khong

if (FileExtentions.Contains(img.FileName.Substring(img.FileName.LastIndexOf("."))))

//lay phan mo rong cua tap tin

{

string slug = products.Slug;

//ten file = Slug + phan mo rong cua tap tin

string imgName = slug + img.FileName.Substring(img.FileName.LastIndexOf("."));

products.Image = imgName;

string PathDir = "~/Public/img/product/";

string PathFile = Path.Combine(Server.MapPath(PathDir), imgName);

//cap nhat thi phai xoa file cu

//Xoa file

if (products.Image != null)

{

string DelPath = Path.Combine(Server.MapPath(PathDir), products.Image);

System.IO.File.Delete(DelPath);

}

//upload hinh

img.SaveAs(PathFile);

}

}//ket thuc phan upload hinh anh

//Xu ly cho muc UpdateAt

products.UpdateAt = DateTime.Now;

//Xu ly cho muc UpdateBy

products.UpdateBy = Convert.ToInt32(Session["UserId"]);

productDAO.Update(products);//chen them row vào Table Products

//Thong bao thanh cong

TempData["message"] = new XMessage("success", "Thêm sản phẩm thành công");

return RedirectToAction("Index");

}

ViewBag.ListSupId = new SelectList(suppliersDAO.getList("Index"), "Id", "Name");

ViewBag.ListCatId = new SelectList(categoriesDAO.getList("Index"), "Id", "Name");

return View(products);

}

## Edit.cshmtl

Lưu ý 3 vấn đề:

Bỏ qua trường CreateAt/By

@using (Html.BeginForm("Edit", "Supplier", FormMethod.Post, new { @enctype = "multipart/form-data" }))

{

@Html.AntiForgeryToken()

@Html.HiddenFor(model => model.Id);

@Html.HiddenFor(model => model.CreateAt);

@Html.HiddenFor(model => model.CreateBy);

Phần bắt buộc trong HTML form để hiển thị hình ảnh

@using (Html.BeginForm("Edit", "Product", FormMethod.Post, new { @enctype = "multipart/form-data" }))

Phần hiển thị hình ảnh trong thẻ img

<div class="form-group">

<label>Ảnh đại diện <span class="text-danger"></span></label>

<div class="col-md-10">

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

</div>

</div>

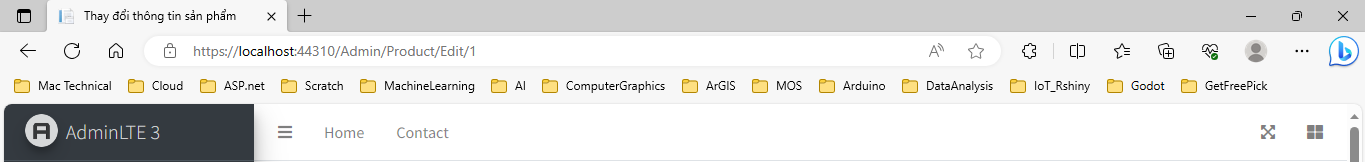
### Tiêu đề

@model MyClass.Model.Products

@{

ViewBag.Title = "Thay đổi thông tin sản phẩm";

Layout = "~/Areas/Admin/Views/Shared/\_LayoutAdmin.cshtml";



### Nội dung trang

Sao chép từ trang Create.cshtml

@model MyClass.Model.Products

@{

ViewBag.Title = "Thay đổi thông tin sản phẩm";

Layout = "~/Areas/Admin/Views/Shared/\_LayoutAdmin.cshtml";

}

@using (Html.BeginForm("Edit","Product",FormMethod.Post,new{@enctype="multipart/form-data"}))

{

@Html.AntiForgeryToken()

@Html.HiddenFor(model => model.Id)

@Html.HiddenFor(model => model.CreateAt);

@Html.HiddenFor(model => model.CreateBy);

@Html.ValidationSummary(true, "", new { @class = "text-danger" })

<!-- Content Wrapper. Contains page content -->

<div class="content-wrapper">

<!-- Main content -->

<section class="content my-2">

<!-- Default box -->

<div class="card">

<div class="card-header">

<div class="row">

<div class="col-md-6">

<strong class="text-danger">THAY ĐỔI THÔNG TIN SẢN PHẨM</strong>

</div>

<div class="col-md-6 text-right">

<a href="@Url.Action("Index", "Product")" class="btn-sm btn-info"> <i class="fa fa-long-arrow-alt-left"></i> Quay lại</a>

</div>

</div>

</div>

<div class="card-body">

@\*Phan NOI DUNG\*@

<div class="row">

@\* Phần nhập tay \*@

<div class="col-md-9">

<div class="form-group">

<label>Tên sản phẩm <span class="text-danger"> (\*)</span></label>

<div class="col-md-10">

@Html.EditorFor(model=>model.Name,new{htmlAttributes=new{@class="form-control"}})

@Html.ValidationMessageFor(model => model.Name, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

<label>Mô tả chi tiết <span class="text-danger"> (\*)</span></label>

<div class="col-md-10">

@Html.TextAreaFor(model => model.MetaDesc, new { @class = "form-control", @rows = "2" })

@Html.ValidationMessageFor(model => model.MetaDesc, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

<label>Từ khóa <span class="text-danger"> (\*)</span></label>

<div class="col-md-10">

@Html.TextAreaFor(model => model.MetaKey, new { @class = "form-control", @rows = "2" })

@Html.ValidationMessageFor(model => model.MetaKey, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

<label>Chi tiết sản phẩm <span class="text-danger"> (\*)</span></label>

<div class="col-md-10">

@Html.EditorFor(model => model.Detail, new { htmlAttributes = new { @class = "form-control" } })

@Html.ValidationMessageFor(model => model.Detail, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

<label>Ảnh đại diện <span class="text-danger"></span></label>

<div class="col-md-10">

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

</div>

</div>

</div>

@\* Phần lựa chọn \*@

<div class="col-md-3">

<div class="form-group">

<label>Số lượng <span class="text-danger">(\*)</span></label>

<div class="col-md-10">

@Html.EditorFor(model => model.Amount, new { htmlAttributes = new { @class = "form-control" } })

@Html.ValidationMessageFor(model => model.Amount, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

<label>Giá bán <span class="text-danger">(\*)</span></label>

<div class="col-md-10">

@Html.EditorFor(model => model.Price, new { htmlAttributes = new { @class = "form-control" } })

@Html.ValidationMessageFor(model => model.Price, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

<label>Giá giảm <span class="text-danger">(\*)</span></label>

<div class="col-md-10">

@Html.EditorFor(model => model.SalePrice, new { htmlAttributes = new { @class = "form-control" } })

@Html.ValidationMessageFor(model => model.SalePrice, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

<label>Loại SP <span class="text-danger"></span></label>

<div class="col-md-10">

@Html.DropDownListFor(model => model.CatID, (SelectList)@ViewBag.ListCatId, "Loại SP", new { @class = "form-control" })

@Html.ValidationMessageFor(model => model.CatID, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

<label>Nhà CC <span class="text-danger"></span></label>

<div class="col-md-10">

@Html.DropDownListFor(model => model.SupplierId, (SelectList)@ViewBag.ListSupId, "Nhà CC", new { @class = "form-control" })

@Html.ValidationMessageFor(model => model.SupplierId, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

<label>Trạng thái <span class="text-danger"> (\*)</span></label>

<div class="col-md-10">

@Html.DropDownListFor(model => model.Status, new[] { new SelectListItem { Text = "Không xuất bản", Value = "2" }, new SelectListItem { Text = "Xuất bản", Value = "1" } }, new { @class = "form-control" })

@Html.ValidationMessageFor(model => model.Status, "", new { @class = "text-danger" })

</div>

</div>

</div>

</div>

<div class="form-group">

<div class="col-md-offset-2 col-md-10">

<button class="btn-sm btn-success" type="submit" name="create"> Lưu <i class="fa fa-save"></i> </button>

</div>

</div>

</div><!-- /.card-body -->

</div><!-- /.card -->

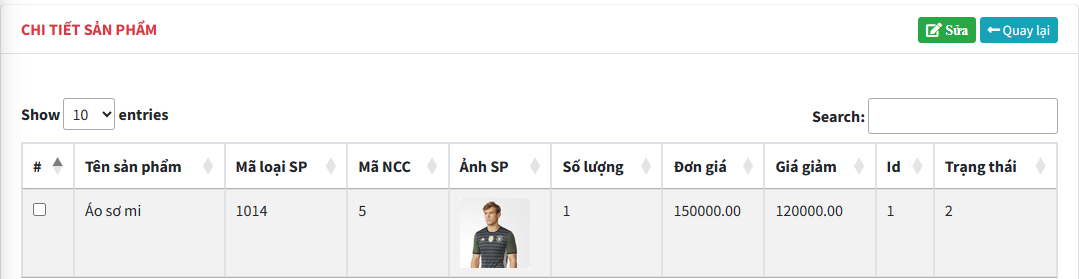
</section><!-- /.content -->

</div><!-- /.content-wrapper -->

}

# Thử nghiệm

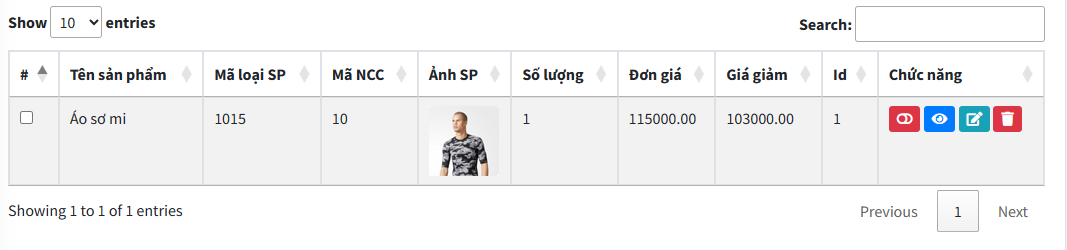
## Trước edit





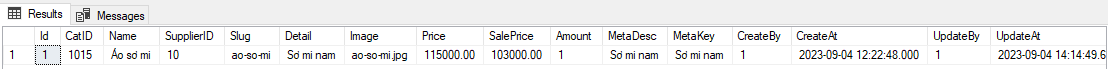


## Sau edit









# lưu ý

Đối với phần sản phẩm thì mỗi sản phẩm có thể có nhiều hình ảnh, và mỗi hình ảnh có thể thuộc về nhiều sản phẩm khác nhau nên phần hình ảnh sau khi cập nhật tuyệt đối không được xóa.

//cap nhat thi phai xoa file cu

////Xoa file

//if (products.Image != null)

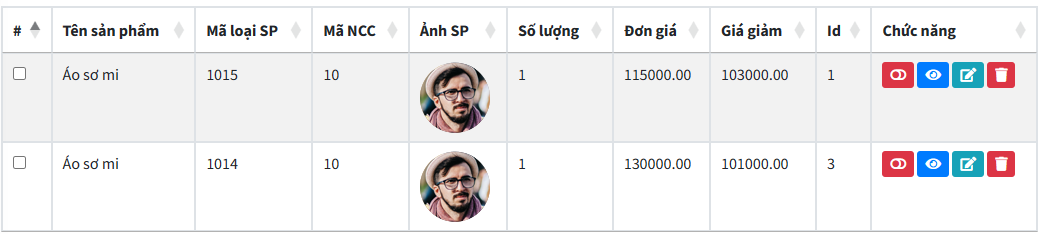
//{

// string DelPath = Path.Combine(Server.MapPath(PathDir), products.Image);

// System.IO.File.Delete(DelPath);

//}

Thứ 2, mỗi tên sản phẩm khi đổi tên hình mới sẽ thay đổi toàn bộ hình của các sản phẩm có cùng tên

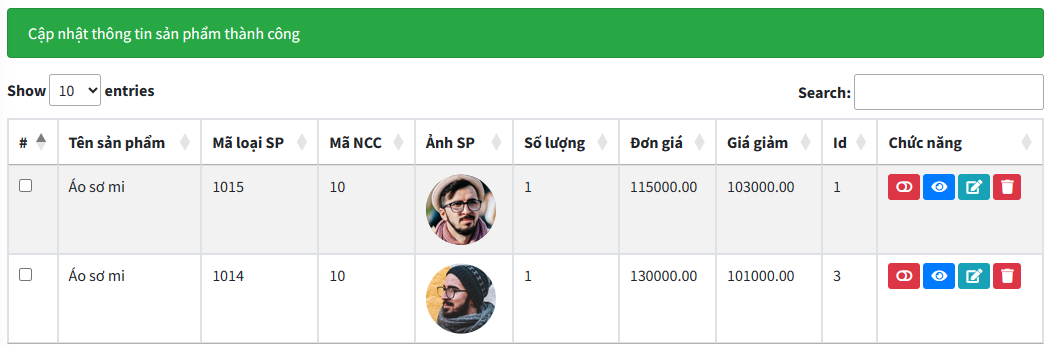


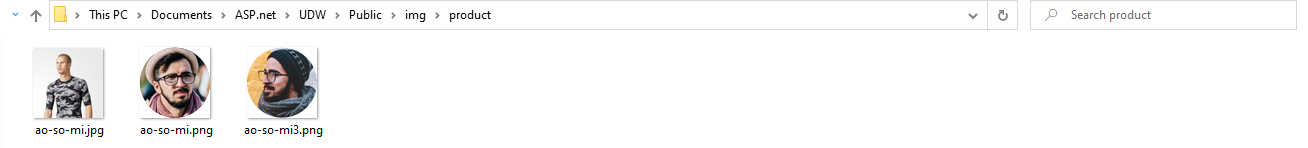
# Giải pháp:

Phần hình ảnh sẽ được gắn thêm phần Id cho mỗi Slug

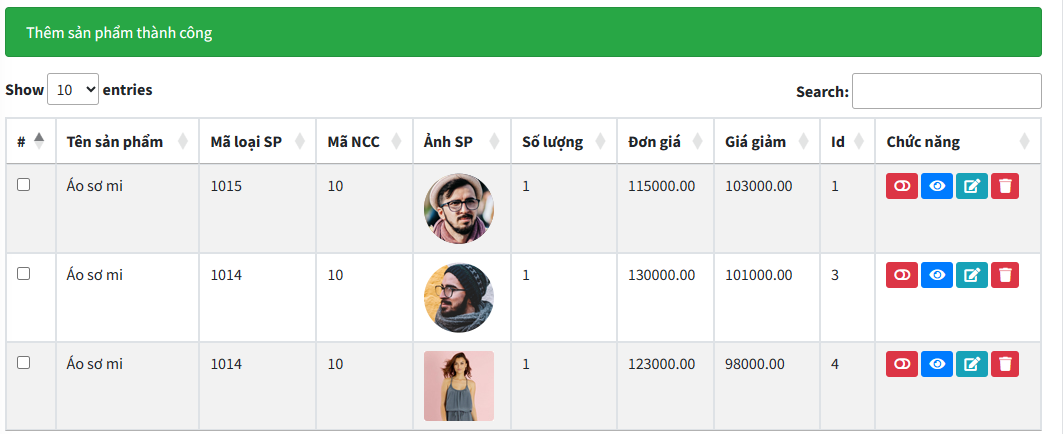
Thực hiện cho Index, Create, Edit

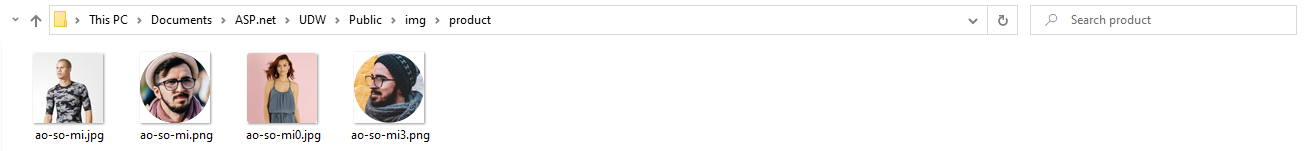
## Edit





## Create





## Edit



