TRƯỜNG ĐẠI HỌC QUY NHƠN



ĐỀ TÀI

ỨNG DỤNG WEB

HỖ TRỢ TRƯNG BÀY NÔNG SẢN

Giảng viên : Nguyễn Thanh Bình

Nhóm : Nguyễn Công Thành

Nguyễn Thị Kiều Duyên

Lớp : CNTT – K40A

**MỤC LỤC**

**LỜI MỞ ĐẦU 1**

1. **GIỚI THIỆU 2**
2. **GIAO DIỆN VÀ XỬ LÝ 2**
3. **Giao diện trang chủ(chung) 2**
4. Giao diện trang chủ 3
5. Code xử lý load dữ liệu 3
6. Giao diện tìm kiếm
7. Xử lý tìm kiếm
8. **Giao diện đăng nhập** **8**
9. Giao diện đăng nhập admin 8
10. Code xử lý đăng nhập 8
11. **Giao diện admin 11**
    1. Quản lý nông sản 11
12. Giao diện quản lý nông sản 12
13. Xử lý load dữ liệu 12
14. Xử lý thêm mới nông sản 15
15. Xử lý cập nhập nông sản 19
16. Xử lý xóa nông sản 23
    1. Quản lý loại nông sản 24
17. Giao diện quản lý loại nông sản 24
18. Xử lý load dữ liệu 25
19. Xử lý thêm loại nông sản 27
20. Xử lý cập nhập loại nông sản 28
21. Xử lý xóa loại nông sản 29
22. **KẾT LUẬN 33**

**LỜI MỞ ĐẦU**

Trong bối cảnh hội nhập hiện nay, sự cạnh tranh không chỉ xuất hiện khi hàng hóa Việt Nam đi ra thế giới mà còn ở ngay thị trường trong nước, khi hàng hóa thế giới tiến vào Việt Nam. Việc đảm bảo hàng hóa có chất lượng tốt là điều rất quan trọng cho khả năng cạnh tranh. Tuy nhiên, doanh nghiệp cần đảm bảo cơ sở cho niềm tin của người tiêu dùng. Hơn nữa vấn nạn hàng giả, hàng nhái, hàng hóa không rõ nguồn gốc tràn lan trên thị trường gây hại trực tiếp đến sức khỏe của người tiêu dùng.

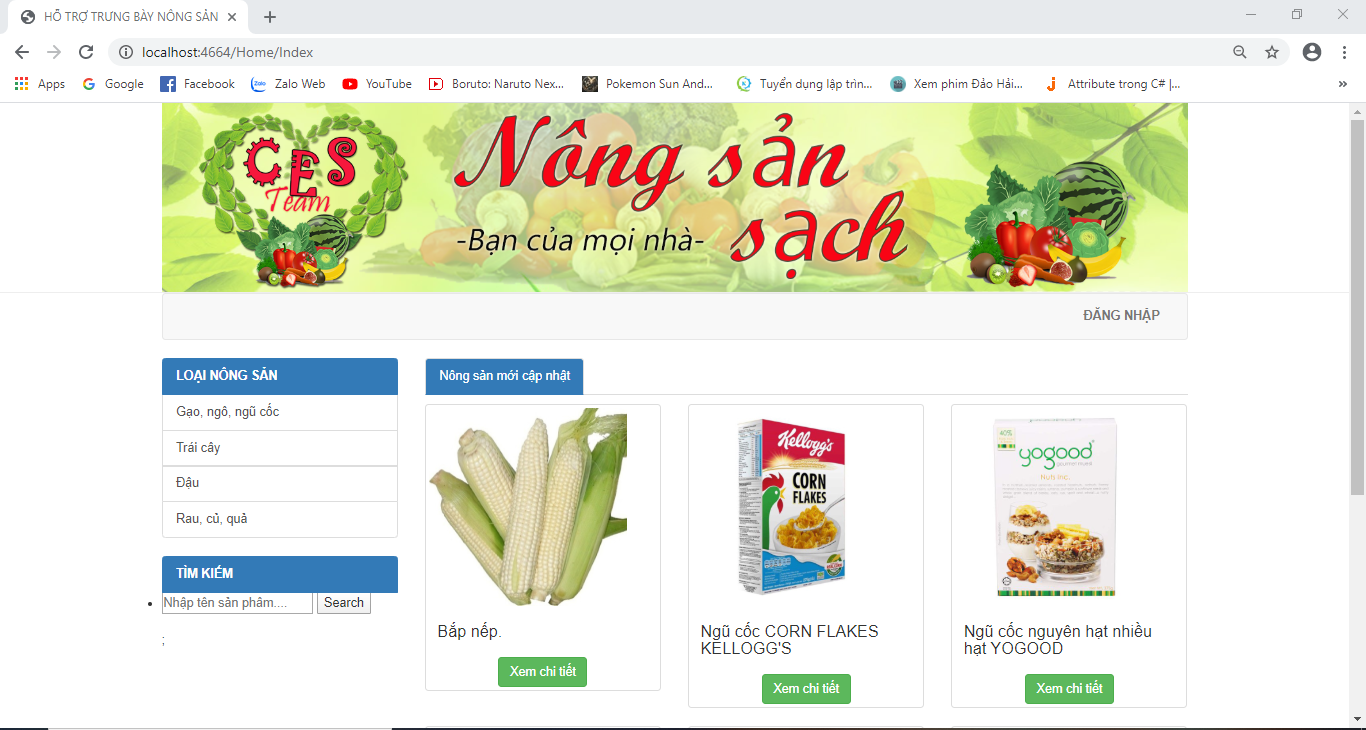
Hiện nay, vấn đề an toàn thực phẩm đang nhức nhối hơn bao giờ hết với nhiều vụ ngộ độc thực phẩm nghiêm trọng. Người tiêu dùng lo lắng và hoài nghi mọi thứ, doanh nghiệp bị mất uy tín.Web trưng bày nông sản cũng chính là một phần thông điệp về chất lượng nông sản mà nhà cung cấp muốn gửi tới khách hàng một cách cụ thể nhất.

Vì vậy, nhóm tụi em muốn làm 1 chút gì đó giúp mọi người yên tâm hơn trong việc lựa chọn nông sản sạch, giúp các nhà sản xuất hay cung cấp có thể quảng bá sản phẩm của mình một cách nhanh nhất. Từ đó, xua tan nỗi lo mua hàng kém chất lượng- mục đích mà bất cứ người tiêu dùng nào khi bỏ tiền ra mua đều mong muốn. Cho nên nhóm em đã quyết định sẽ chọn đề tài “Ứng dụng Web hỗ trợ trưng bày sản phẩm”

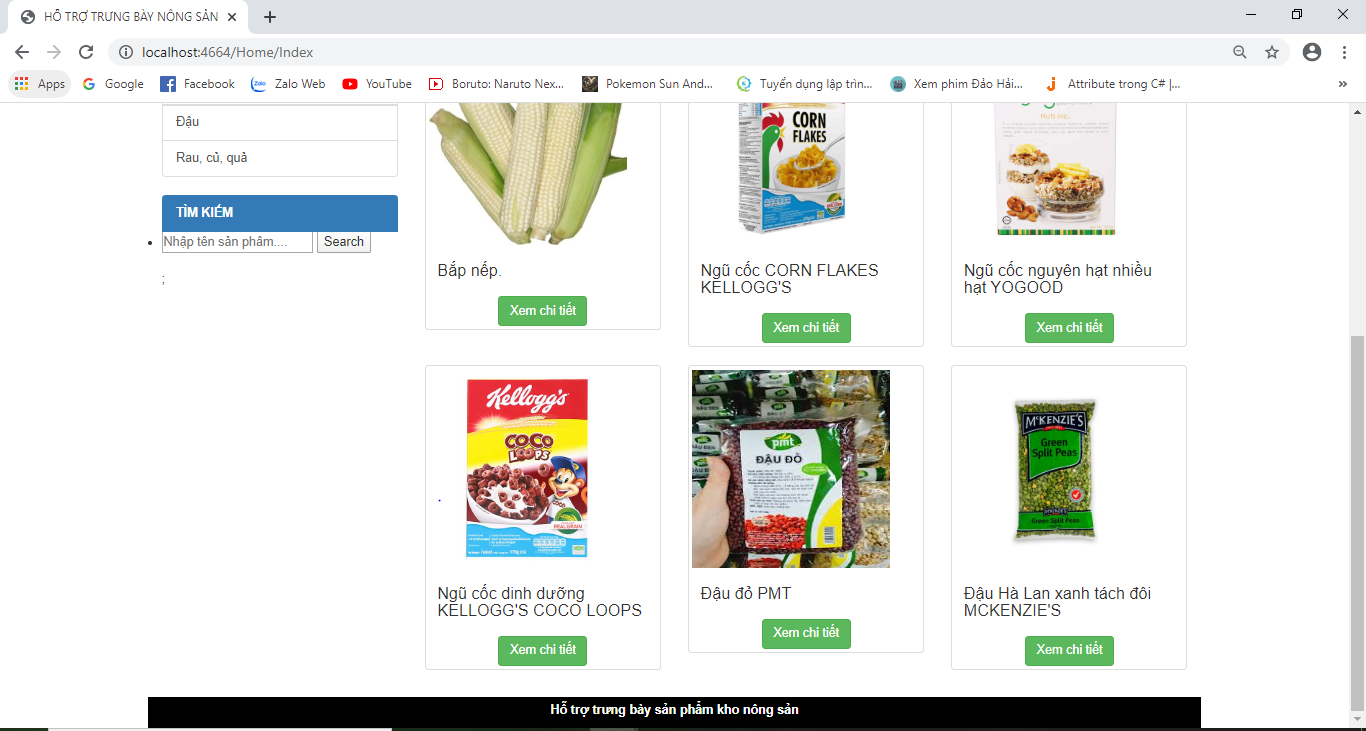
Nhóm em xin chân thành cảm ơn!

* + - 1. **GIỚI THIỆU:**
* Ứng dụng Web hỗ trợ trưng bày nông sản, giúp mọi người có thể trực tiếp tìm hiểu, thu thập thông tin về nông sản một cách đầy đủ và chi tiết nhất. Qua đó, hạn chế mua phải hàng kém chất lượng dẫn đến những ảnh hưởng trực tiếp đến sức khỏe của người dùng.
* Ứng dụng có thể:
* Phân loại được từng loại nông sản khác nhau, để tìm kiếm một cách nhanh chóng và thuận tiện hơn.
* Quản lý được tên nông sản, tên loại một cách chặt chẽ, dễ dàng, ít tốn thời gian, chi phí.
* Liệt kê được số nông sản ứng với từng loại
* Ưu điểm của ứng dụng:
* Giao diện đơn giản dễ sử dụng.
* Dễ dàng thêm, sửa, xóa các nông sản đối với admin.
* Ứng dụng còn cho biết rõ nhà cung cấp nông sản để người dùng an tâm hơn khi có nhu cầu mua đối với nông sản.
  + - 1. **THIẾT KẾ GIAO DIỆN:**
    1. **Giao diện trang chủ(chung):**

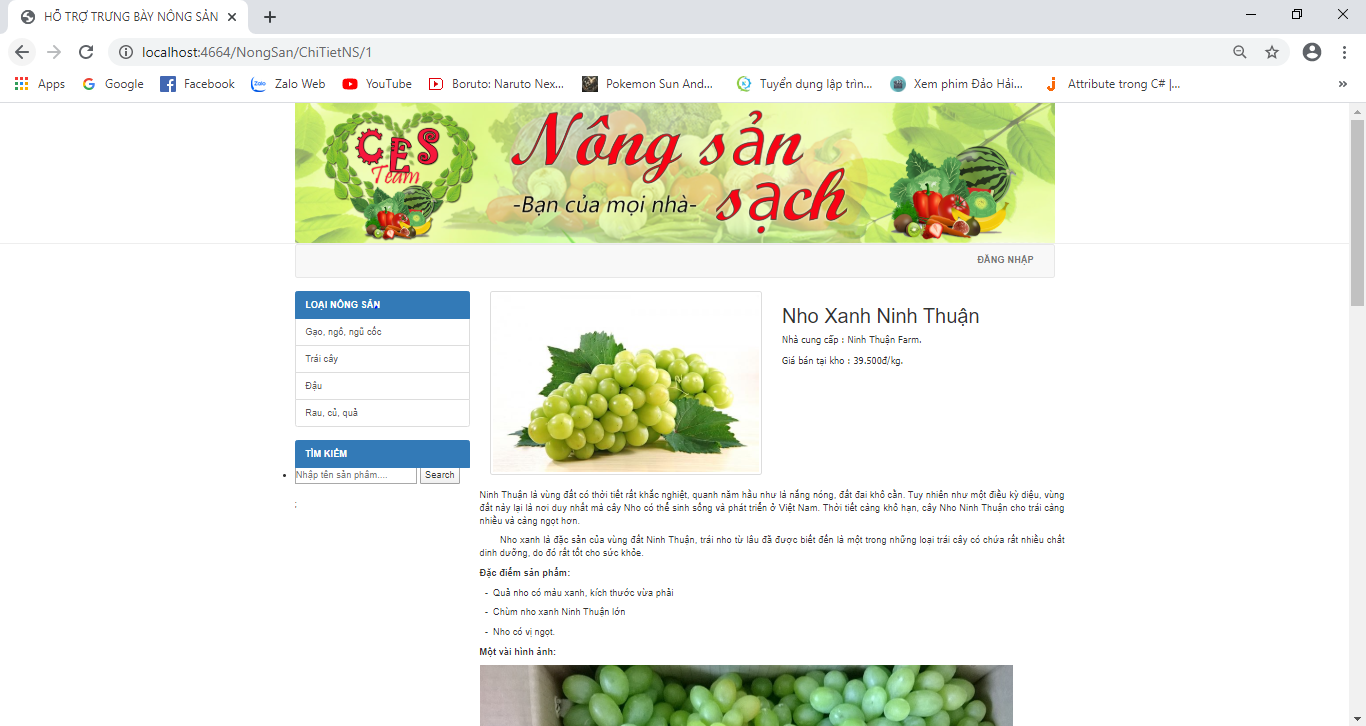
**a. Giao diện trang chủ:**



***Hình 1.1. Màn hình trang chủ***



***Hình 1.2. Màn hình trang chủ***



***Hình 1.3 Màn hình trang chủ***

**b. Code xử lý load dữ liệu:**

* ***Load dữ liệu :***

namespace NongSanVietNam.Models

{

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations;

using System.ComponentModel.DataAnnotations.Schema;

using System.Data.Entity.Spatial;

[Table("NongSan")]

public partial class NongSan

{

public int ID { get; set; }

[Required]

[Display(Name = "Tên nông sản")]

[StringLength(50)]

public string TenNS { get; set; }

[Display(Name = "Loại nông sản")]

public int IDLoaiNS { get; set; }

[StringLength(550)]

[Display(Name = "Mô tả ngắn")]

public string MoTaNgan { get; set; }

[Column(TypeName = "ntext")]

[Display(Name = "Bài viết chi tiết")]

public string MotaChiTiet { get; set; }

[Column(TypeName = "date")]

[Display(Name = "Ngày tạo")]

public DateTime NgayTao { get; set; }

[Required]

[Display(Name = "Người tạo")]

[StringLength(50)]

public string NguoiTao { get; set; }

[Column(TypeName = "date")]

[Display(Name = "Ngày cập nhật")]

public DateTime? NgayCapNhat { get; set; }

[StringLength(50)]

[Display(Name = "Người cập nhật")]

public string NguoiCapNhat { get; set; }

[StringLength(350)]

[Display(Name = "Ảnh đại diện")]

public string HinhAnhDaiDien { get; set; }

public virtual LoaiN LoaiN { get; set; }

}

}

***/////////////***

namespace NongSanVietNam.Models

{

using System;

using System.Data.Entity;

using System.ComponentModel.DataAnnotations.Schema;

using System.Linq;

public partial class NongSanVN : DbContext

{

public NongSanVN()

: base("name=NongSanVN")

{

}

public virtual DbSet<LoaiN> LoaiNS { get; set; }

public virtual DbSet<NguoiDung> NguoiDungs { get; set; }

public virtual DbSet<NongSan> NongSans { get; set; }

public virtual DbSet<TinTuc> TinTucs { get; set; }

protected override void OnModelCreating(DbModelBuilder modelBuilder)

{

modelBuilder.Entity<LoaiN>()

.HasMany(e => e.NongSans)

.WithRequired(e => e.LoaiN)

.WillCascadeOnDelete(false);

modelBuilder.Entity<NguoiDung>()

.HasMany(e => e.TinTucs)

.WithRequired(e => e.NguoiDung)

.WillCascadeOnDelete(false);

}

}

}

* ***Load danh mục:***

@using NongSanVietNam.Models;

@model IEnumerable<NongSanVietNam.Models.NongSan>

@{

ViewBag.Title = "DanhMucNS";

Layout = "~/Views/Shared/Layout.cshtml";

}

<div class="row">

<div>

<ul class="nav nav-tabs">

<li role="presentation" class="active"><a href="#" style="background-color:#337ab7; color:white">@ViewBag.tenloai</a> </li>

</ul>

</div>

public ActionResult DanhMucNS(int id)

{

var ds = db.NongSans.Where(s => s.IDLoaiNS == id).ToList();

ViewBag.tenloai = db.LoaiNS.Where(s => s.IDLoaiNS == id).ToList().FirstOrDefault().TenLoai;

ViewBag.dssp = db.NongSans.Where(s => s.IDLoaiNS == id).ToList();

return View(ds);

}

* ***Load sản phẩm mới***

@model IEnumerable<NongSanVietNam.Models.NongSan>

<div class="row">

@{

foreach (var item in Model)

{

<div class="col-sm-6 col-md-4">

<div class="thumbnail">

<a class="fancybox" href='@item.HinhAnhDaiDien' data-fancybox-group="gallery" title='@item.TenNS'>

<div style="width:220px;height:220px">

<img src="@item.HinhAnhDaiDien" alt="..." width="220" height="220">

</div>

</a>

<div class="caption">

<h4>@item.TenNS</h4>

</div>

<div style="text-align:center">

@Html.ActionLink("Xem chi tiết", "ChiTietNS", "NongSan", new { id = item.ID }, new { @class = "btn btn-success", @role = "button" })

</div>

</div>

</div>

}

}

</div>

public PartialViewResult NongSanMoi()

{

var nsm = db.NongSans.OrderByDescending(s => s.ID).ToList().Take(6);

return PartialView(nsm);

}

* ***Load chi tiết thực phẩm***

@using NongSanVietNam.Models;

@model NongSanVietNam.Models.NongSan

@{

ViewBag.Title = "View";

Layout = "~/Views/Shared/Layout.cshtml";

}

<div class="row">

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

<div class="thumbnail">

<img src="@Model.HinhAnhDaiDien" alt="..." width="500" height="300">

</div>

</div>

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

<p>

<h2>@Model.TenNS</h2>

</p>

<p>

@Html.Raw(Model.MoTaNgan)

</p>

</div>

</div>

<div class="row">

<div>

<div>@Html.Raw(Model.MotaChiTiet)</div>

</div>

</div>

<div class="row">

<div>

<ul class="nav nav-tabs">

<li role="presentation" class="active"><a href="#" style="background-color:#337ab7; color:white">Nông sản cùng loại</a> </li>

</ul>

</div>

@{

foreach (var item in @ViewBag.nslq)

{

<div class="row">

<div class="col-sm-6 col-md-4">

<div class="thumbnail">

<a class="fancybox" href='@item.HinhAnhDaiDien' data-fancybox-group="gallery" title='@item.TenNS'>

<div style="width:220px;height:220px">

<img src="@item.HinhAnhDaiDien" alt="..." width="262" height="220">

</div>

</a>

<div class="caption">

<h4 style="text-align:center">@item.TenNS</h4>

</div>

<div style="text-align:center">

@Html.ActionLink("Xem chi tiết", "ChiTietNS", "NongSan", new { id = item.ID }, new { @class = "btn btn-success", @role = "button" })

</div>

</div>

</div>

</div>

}

}

</div>

public ActionResult ChiTietNS(int id)

{

var ns = db.NongSans.Where(s => s.ID == id).ToList().FirstOrDefault();

ViewBag.nslq = db.NongSans.Where(s => s.IDLoaiNS == ns.IDLoaiNS && s.ID != ns.ID).ToList().Take(3);

return View(ns);

}

[HttpGet]

public ActionResult TimKiemNS(string tenNS)

{

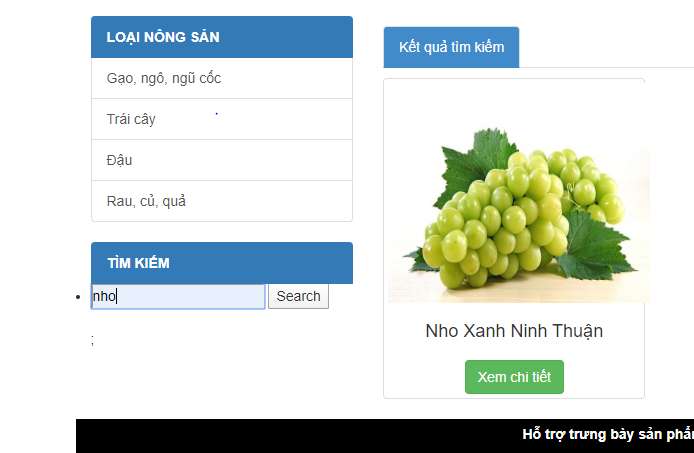
var ns = db.NongSans.Where(s => s.TenNS.Contains(tenNS)).ToList();

ViewBag.dsns = db.NongSans.Where(s => s.TenNS.Contains(tenNS)).ToList();

return View(ns);

}

**c.Giao diện tìm kiếm:**



***Hình 1.3 Màn hình tìm kiếm***

**d.Xử lý tìm kiếm :**

@model IEnumerable<NongSanVietNam.Models.NongSan>

@{

ViewBag.Title = "TimKiemNS";

Layout = "~/Views/Shared/Layout.cshtml";

}

<ul class="nav nav-tabs" style="padding-top:10px">

<li role="presentation" class="active" style="border-color:#428bca"><a href="#" style="background-color:#428bca;color:#ffffff">Kết quả tìm kiếm</a></li>

</ul><p></p>

@{

if (ViewBag.thongbao != null)

{

<div class="alert alert-warning" role="alert">

<span class="glyphicon glyphicon-exclamation-sign">

@ViewBag.thongbao

</span>

</div>

}

else

{

<div class="row">

@{

foreach (var item in Model)

{

<div class="col-sm-6 col-md-4">

<div class="thumbnail">

<a class="fancybox" href='@item.HinhAnhDaiDien' data-fancybox-group="gallery" title='@item.TenNS'>

<div style="width:220px;height:220px">

<img src="@item.HinhAnhDaiDien" alt="..." width="262" height="220">

</div>

</a>

<div class="caption">

<h4 style="text-align:center">@item.TenNS</h4>

</div>

<div style="text-align:center">

@Html.ActionLink("Xem chi tiết", "ChiTietNS", "NongSan", new { id = item.ID }, new { @class = "btn btn-success", @role = "button" })

</div>

</div>

</div>

}

}

</div>

}

}

public ActionResult TimkiemNS(string txtTimKiem)

{

List<NongSan> kqtk = db.NongSans.Where(s => s.TenNS.Contains(txtTimKiem)).ToList();

if (kqtk.Count != 0)

{

return PartialView(kqtk);

}

else

{

ViewBag.thongbao = "Không tìm thấy sản phẩm cần tìm";

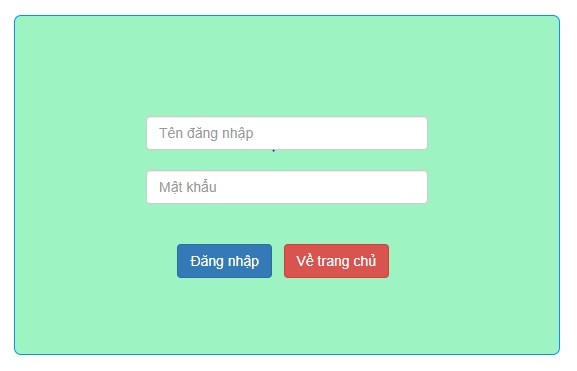
return View();

}

}

**2. Giao diện đăng nhập :**

**a. Giao diện:**



***Hình 2.1 Màn hình trang đăng nhập***

1. **Code xử lý đăng nhập:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

using NongSanVietNam.Models;

namespace NongSanVietNam.Areas.Admin.Controllers

{

public class DefaultController : Controller

{

// GET: Admin/Default

NongSanVN db = new NongSanVN();

public ActionResult Index()

{

if (string.IsNullOrEmpty(Session["ten"].ToString()))

{

return RedirectToAction("Login");

}

else

{

return View();

}

}

public PartialViewResult Menu()

{

return PartialView();

}

public ActionResult Login()

{

return View();

}

[HttpPost]

public ActionResult Login(string txtTen, string txtMK)

{

var user = db.NguoiDungs.Where(s => s.TaiKhoan == txtTen && s.MatKhau == txtMK).ToList();

if(user.Count!=0)

{

Session["ten"] = txtTen;

Session["pass"] = txtMK;

return RedirectToAction("Index");

}

else

{

return View();//Response.Write("<script>alert('Đăng nhập không thành công')</script>");

}

}

public ActionResult Thoat()

{

Session["ten"] = "";

Session["pass"] = "";

return RedirectToAction("Login");

}

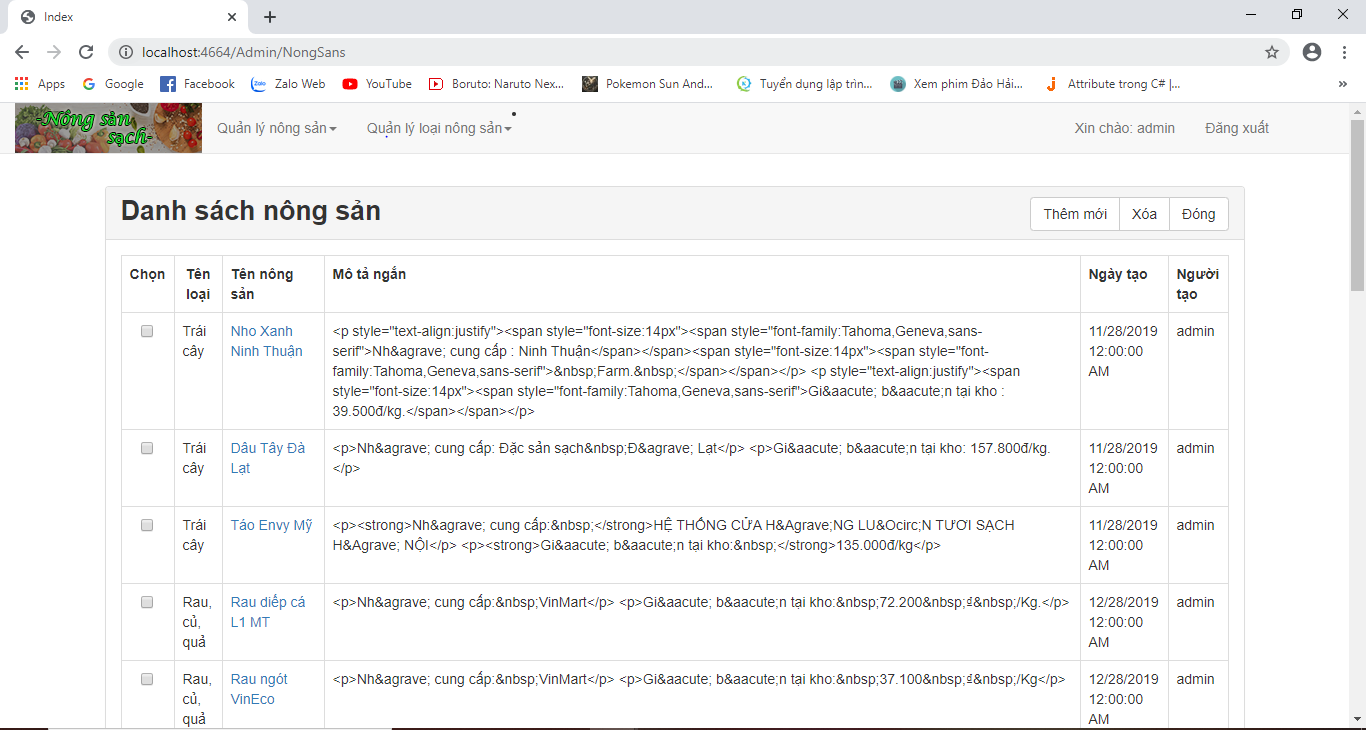
}

}

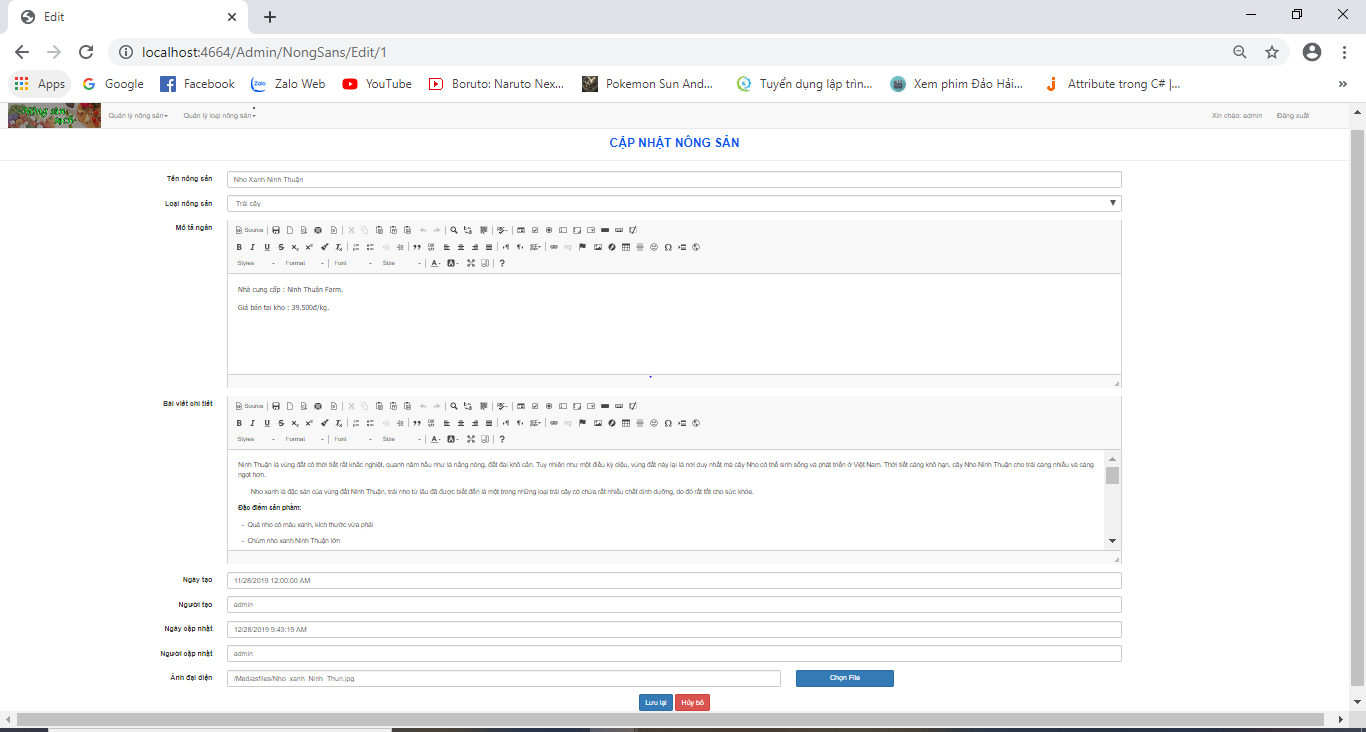
**3r. Giao diện admin :**

**4.1. Quản lý nông sản :**

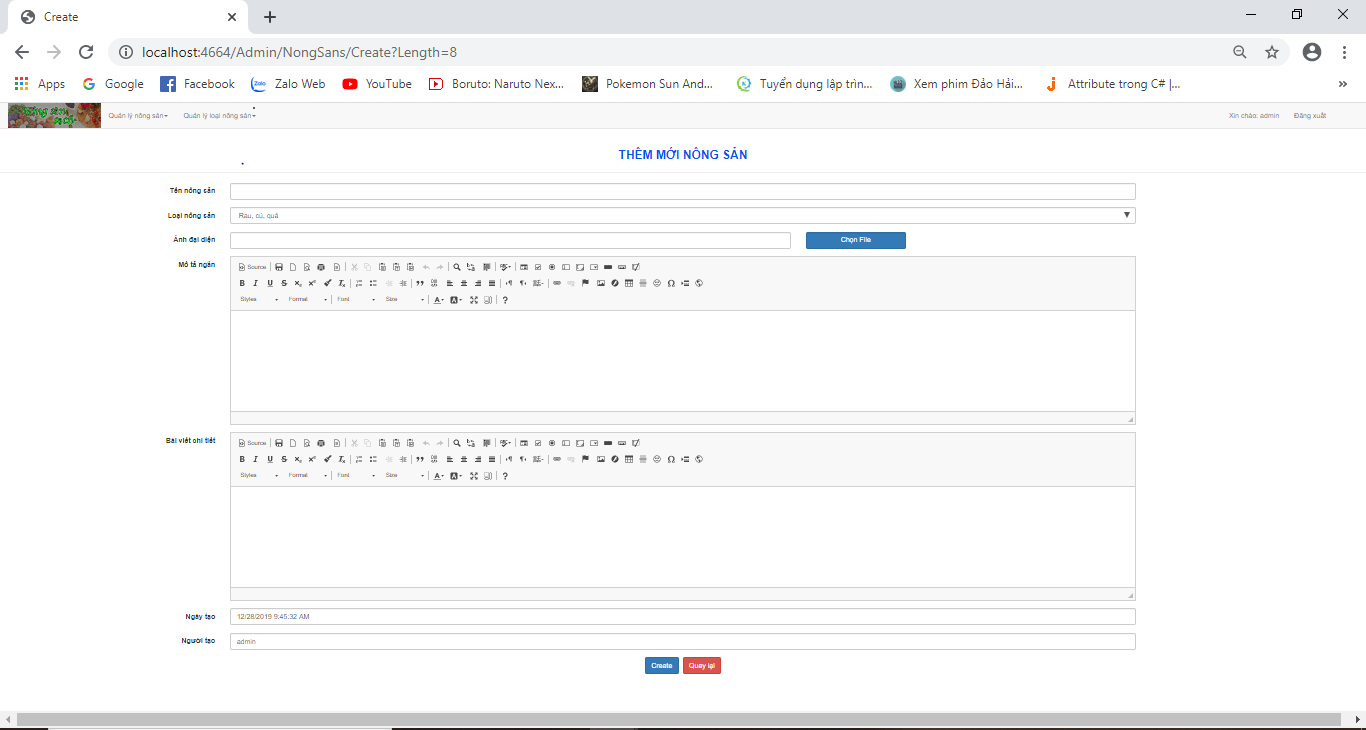
**a. Giao diện:**



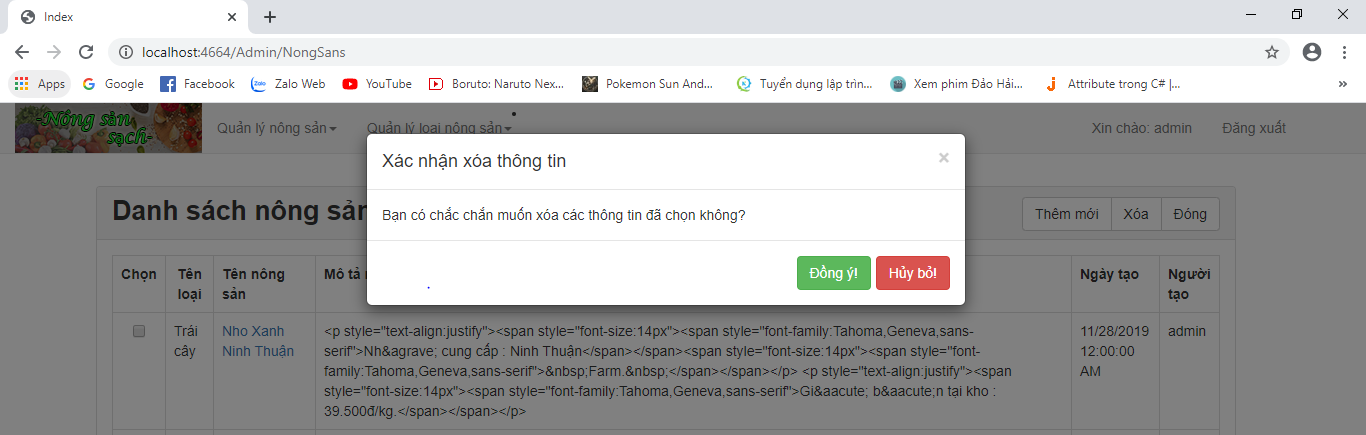
***Hình 4.1 Màn hình trang quản lý nông sản***



***Hình 4.2 Màn hình trang quản lý nông sản***



***Hình 4.3 Màn hình trang quản lý nông sản***



***Hình 4.4 Màn hình trang quản lý nông sản***

**b. Xử lý load dữ liệu:**

@model NongSanVietNam.Models.LoaiN

@{

ViewBag.Title = "Details";

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

}

<h2>Details</h2>

<div>

<h4>LoaiN</h4>

<hr />

<dl class="dl-horizontal">

<dt>

@Html.DisplayNameFor(model => model.TenLoai)

</dt>

<dd>

@Html.DisplayFor(model => model.TenLoai)

</dd>

<dt>

@Html.DisplayNameFor(model => model.HinhAnh)

</dt>

<dd>

@Html.DisplayFor(model => model.HinhAnh)

</dd>

<dt>

@Html.DisplayNameFor(model => model.MotaNgan)

</dt>

<dd>

@Html.DisplayFor(model => model.MotaNgan)

</dd>

<dt>

@Html.DisplayNameFor(model => model.NguoiTao)

</dt>

<dd>

@Html.DisplayFor(model => model.NguoiTao)

</dd>

<dt>

@Html.DisplayNameFor(model => model.NgayTao)

</dt>

<dd>

@Html.DisplayFor(model => model.NgayTao)

</dd>

<dt>

@Html.DisplayNameFor(model => model.NguoiCapNhat)

</dt>

<dd>

@Html.DisplayFor(model => model.NguoiCapNhat)

</dd>

<dt>

@Html.DisplayNameFor(model => model.NgayCapNhat)

</dt>

<dd>

@Html.DisplayFor(model => model.NgayCapNhat)

</dd>

</dl>

</div>

<p>

@Html.ActionLink("Edit", "Edit", new { id = Model.IDLoaiNS }) |

@Html.ActionLink("Back to List", "Index")

</p>

public ActionResult Details(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

NongSan nongSan = db.NongSans.Find(id);

if (nongSan == null)

{

return HttpNotFound();

}

return View(nongSan);

}

**c. Xử lý thêm nông sản:**

@model NongSanVietNam.Models.NongSan

@{

ViewBag.Title = "Create";

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

}

<h2>&nbsp;</h2>

@using (Html.BeginForm())

{

@Html.AntiForgeryToken()

<div class="form-horizontal">

<h3 style="text-align:center; color:#0056f1;"><b>THÊM MỚI NÔNG SẢN</b></h3>

<hr />

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

<div class="form-group">

@Html.LabelFor(model => model.TenNS, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

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

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

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.IDLoaiNS, "Loại nông sản", htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

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

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

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.HinhAnhDaiDien, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-5">

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

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

</div>

<div class="col-md-1">

<input type="button" class="btn-primary form-control" id="browse" value="Chọn File" />

</div>

<script>

$("#browse").click(function () {

var finder = new CKFinder();

finder.selectActionFunction = function (fileUrl) {

//fileUrl = fileUrl.substring(fileUrl.lastIndexOf("/") + 1);

$("#HinhAnhDaiDien").val(fileUrl);

}

finder.popup();

});

</script>

</div>

<div class="form-group">

@Html.LabelFor(model => model.MoTaNgan, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

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

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

</div>

<script type="text/javascript">

CKEDITOR.replace("MoTaNgan");

</script>

</div>

<div class="form-group">

@Html.LabelFor(model => model.MotaChiTiet, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

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

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

</div>

<script type="text/javascript">

CKEDITOR.replace("MotaChiTiet");

</script>

</div>

<div class="form-group">

@Html.LabelFor(model => model.NgayTao, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

@Html.EditorFor(model => model.NgayTao, new { htmlAttributes = new { @class = "form-control", @Value = DateTime.Now.ToString() } })

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

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.NguoiTao, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

@Html.EditorFor(model => model.NguoiTao, new { htmlAttributes = new { @class = "form-control", @Value = Session["ten"].ToString() } })

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

</div>

</div>

<div class="form-group" style="text-align:center">

<div class="col-md-offset-2 col-md-8">

<input type="submit" value="Create" class="btn btn-primary" />&nbsp;

@Html.ActionLink("Quay lại", "Index", "NongSans", new { @class = "btn btn-danger" })

</div>

</div>

</div>

}

public ActionResult Create()

{

ViewBag.IDLoaiNS = new SelectList(db.LoaiNS, "IDLoaiNS", "TenLoai");

return View();

}

[HttpPost]

[ValidateInput(false)]

public ActionResult Create([Bind(Include = "ID,TenNS,IDLoaiNS,MoTaNgan,MotaChiTiet,NgayTao,NguoiTao,NgayCapNhat,NguoiCapNhat,HinhAnhDaiDien")] NongSan nongSan)

{

if (ModelState.IsValid)

{

db.NongSans.Add(nongSan);

db.SaveChanges();

return RedirectToAction("Index");

}

ViewBag.IDLoaiNS = new SelectList(db.LoaiNS, "IDLoaiNS", "TenLoai", nongSan.IDLoaiNS);

return View(nongSan);

}

**d. Xử lý cập nhập nông sản:**

@model NongSanVietNam.Models.NongSan

@{

ViewBag.Title = "Edit";

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

}

<h2>&nbsp;</h2>

@using (Html.BeginForm())

{

@Html.AntiForgeryToken()

<div class="form-horizontal">

<h3 style="text-align:center; color:#0056f1;"><b>CẬP NHẬT NÔNG SẢN</b></h3>

<hr />

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

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

<div class="form-group">

@Html.LabelFor(model => model.TenNS, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

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

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

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.IDLoaiNS, "Loại nông sản", htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

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

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

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.MoTaNgan, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

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

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

</div>

<script type="text/javascript">

CKEDITOR.replace("MoTaNgan");

</script>

</div>

<div class="form-group">

@Html.LabelFor(model => model.MotaChiTiet, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

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

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

</div>

<script type="text/javascript">

CKEDITOR.replace("MotaChiTiet");

</script>

</div>

<div class="form-group">

@Html.LabelFor(model => model.NgayTao, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

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

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

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.NguoiTao, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

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

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

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.NgayCapNhat, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

@Html.EditorFor(model => model.NgayCapNhat, new { htmlAttributes = new { @class = "form-control", @Value = DateTime.Now.ToString() } })

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

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.NguoiCapNhat, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

@Html.EditorFor(model => model.NguoiCapNhat, new { htmlAttributes = new { @class = "form-control", @Value = Session["ten"].ToString() } })

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

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.HinhAnhDaiDien, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-5">

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

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

</div>

<div class="col-md-1">

<input type="button" class="btn-primary form-control" id="browse" value="Chọn File" />

</div>

<script>

$("#browse").click(function () {

var finder = new CKFinder();

finder.selectActionFunction = function (fileUrl) {

$("#HinhAnhDaiDien").val(fileUrl);

}

finder.popup();

});

</script>

</div>

<div class="form-group" style="text-align:center">

<div class="col-md-offset-2 col-md-8">

<input type="submit" value="Lưu lại" class="btn btn-primary" />

@Html.ActionLink("Hủy bỏ", "Index", "NongSans", new { @class = "btn btn-danger" })

</div>

</div>

</div>

}

public ActionResult Edit(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

NongSan nongSan = db.NongSans.Find(id);

if (nongSan == null)

{

return HttpNotFound();

}

ViewBag.IDLoaiNS = new SelectList(db.LoaiNS, "IDLoaiNS", "TenLoai", nongSan.IDLoaiNS);

return View(nongSan);

}

[HttpPost]

[ValidateInput(false)]

public ActionResult Edit([Bind(Include = "ID,TenNS,IDLoaiNS,MoTaNgan,MotaChiTiet,NgayTao,NguoiTao,NgayCapNhat,NguoiCapNhat,HinhAnhDaiDien")] NongSan nongSan)

{

if (ModelState.IsValid)

{

db.Entry(nongSan).State = EntityState.Modified;

db.SaveChanges();

return RedirectToAction("Index");

}

ViewBag.IDLoaiNS = new SelectList(db.LoaiNS, "IDLoaiNS", "TenLoai", nongSan.IDLoaiNS);

return View(nongSan);

}

**e. Xử lý xóa nông sản:**

<script>

$(function () {

$(".popup\_button\_delete").click(function (e) {

e.preventDefault();

bootbox.dialog({

message: "Bạn có chắc chắn muốn xóa các thông tin đã chọn không?",

title: "Xác nhận xóa thông tin",

buttons: {

success: {

label: "Đồng ý!",

className: "btn-success",

callback: function () {

$('form').submit();

//\_\_doPostBack('btnDelete', 'OnClick');

}

},

danger: {

label: "Hủy bỏ!",

className: "btn-danger",

callback: function () {

//Example.show("Xem lại");

}

}

}

});

return false;

});

});

</script>

public ActionResult Delete(int[] id)

{

foreach (int i in id)

{

NongSan ns = db.NongSans.Find(i);

db.NongSans.Remove(ns);

db.SaveChanges();

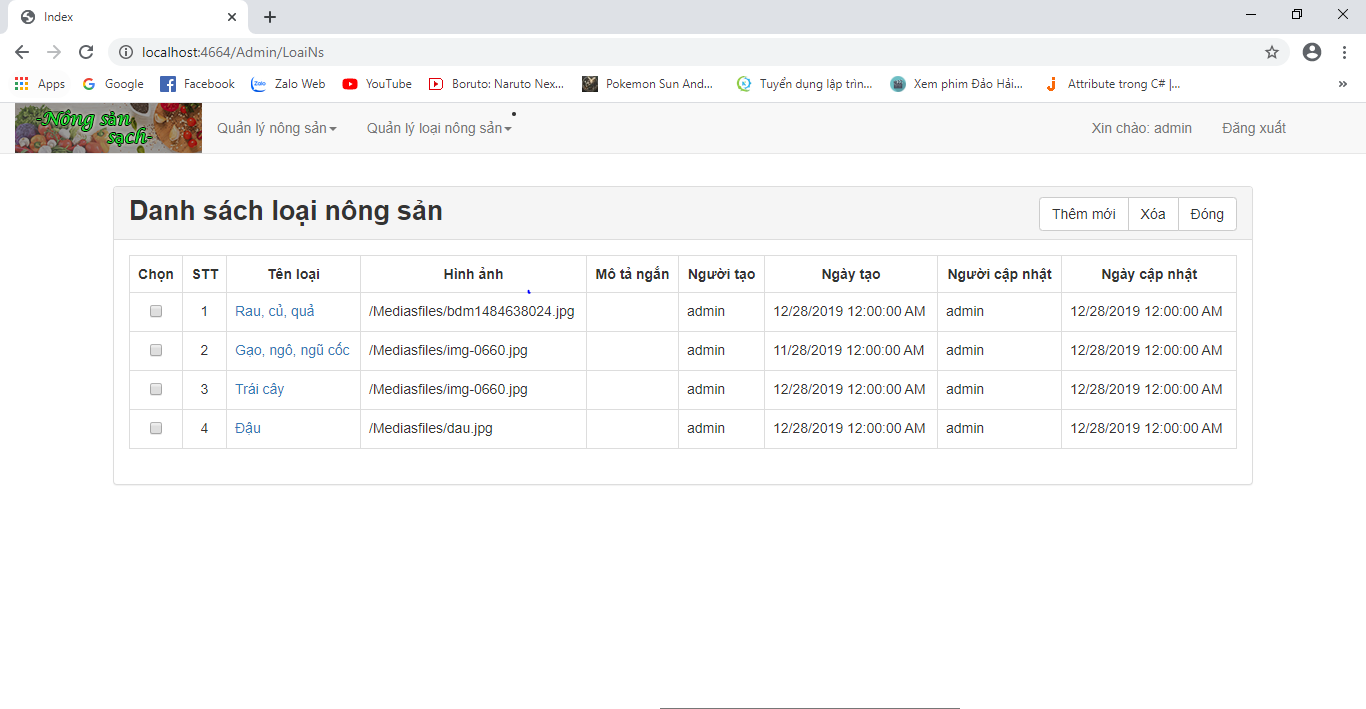
}

return RedirectToAction("Index");

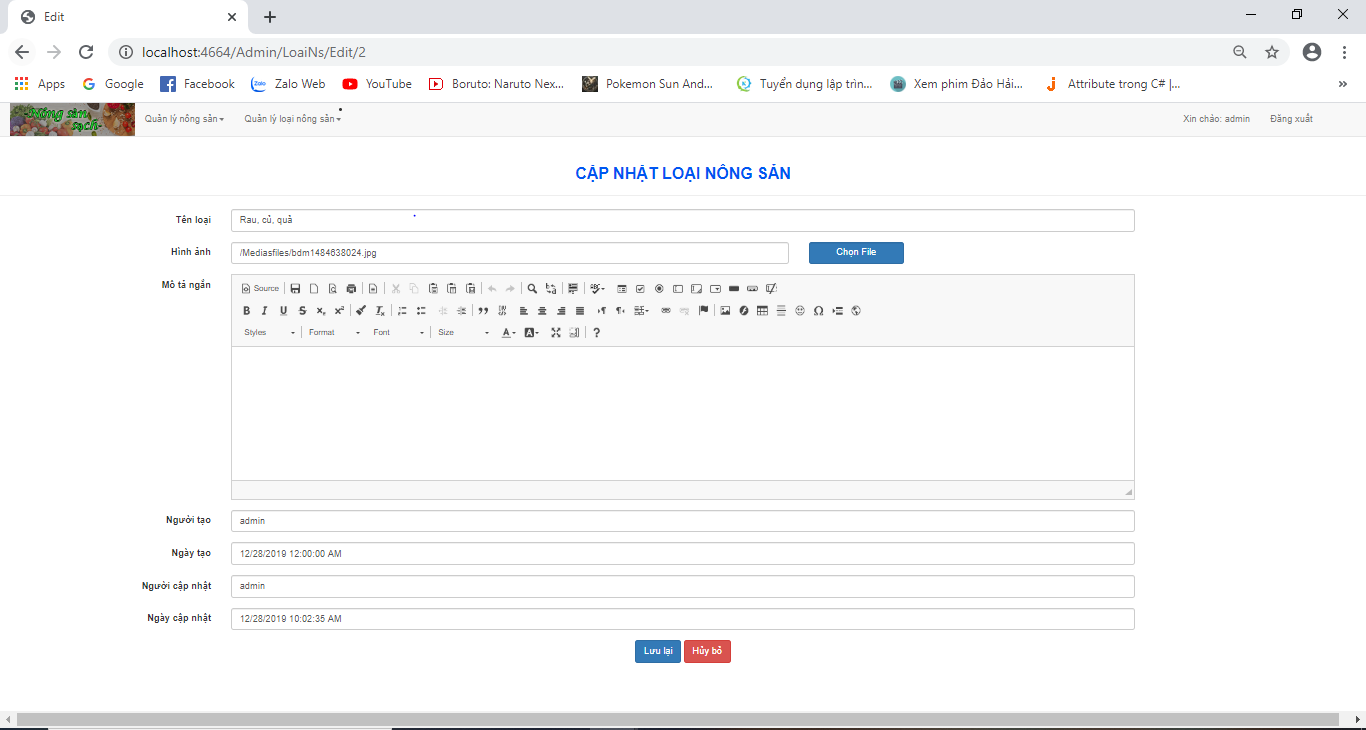
}

**3.2 Quản lý loại nông sản:**

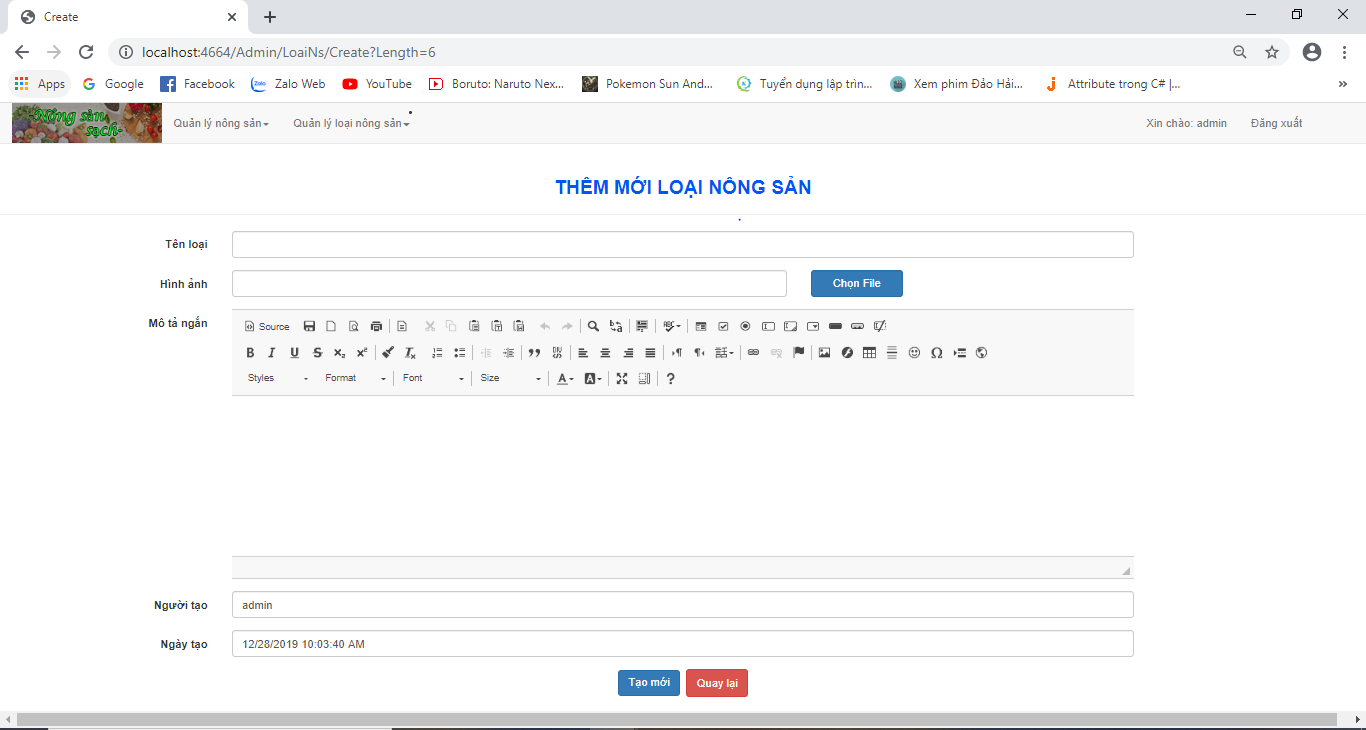
**a. Giao diện:**



***Hình 3.4Màn hình trang quản lý loại thực phẩm***



***Hình 3.5 Màn hình trang quản lý loại nông sản***



***Hình 3.6 Màn hình trang quản lý loại nông sản***

**b. Xử lý load dữ liệu:**

@model NongSanVietNam.Models.LoaiN

@{

ViewBag.Title = "Delete";

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

}

<h2>Delete</h2>

<h3>Are you sure you want to delete this?</h3>

<div>

<h4>LoaiN</h4>

<hr />

<dl class="dl-horizontal">

<dt>

@Html.DisplayNameFor(model => model.TenLoai)

</dt>

<dd>

@Html.DisplayFor(model => model.TenLoai)

</dd>

<dt>

@Html.DisplayNameFor(model => model.HinhAnh)

</dt>

<dd>

@Html.DisplayFor(model => model.HinhAnh)

</dd>

<dt>

@Html.DisplayNameFor(model => model.MotaNgan)

</dt>

<dd>

@Html.DisplayFor(model => model.MotaNgan)

</dd>

<dt>

@Html.DisplayNameFor(model => model.NguoiTao)

</dt>

<dd>

@Html.DisplayFor(model => model.NguoiTao)

</dd>

<dt>

@Html.DisplayNameFor(model => model.NgayTao)

</dt>

<dd>

@Html.DisplayFor(model => model.NgayTao)

</dd>

<dt>

@Html.DisplayNameFor(model => model.NguoiCapNhat)

</dt>

<dd>

@Html.DisplayFor(model => model.NguoiCapNhat)

</dd>

<dt>

@Html.DisplayNameFor(model => model.NgayCapNhat)

</dt>

<dd>

@Html.DisplayFor(model => model.NgayCapNhat)

</dd>

</dl>

@using (Html.BeginForm()) {

@Html.AntiForgeryToken()

<div class="form-actions no-color">

<input type="submit" value="Delete" class="btn btn-default" /> |

@Html.ActionLink("Back to List", "Index")

</div>

}

</div>

public ActionResult Details(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

LoaiN loaiN = db.LoaiNS.Find(id);

if (loaiN == null)

{

return HttpNotFound();

}

return View(loaiN);

}

**c. Xử lý thêm nông sản:**

@model NongSanVietNam.Models.LoaiN

@{

ViewBag.Title = "Create";

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

}

<h2>&nbsp;</h2>

@using (Html.BeginForm())

{

@Html.AntiForgeryToken()

<div class="form-horizontal">

<h3 style="text-align:center; color:#0056f1;"><b>THÊM MỚI LOẠI NÔNG SẢN</b></h3>

<hr />

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

<div class="form-group">

@Html.LabelFor(model => model.TenLoai, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

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

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

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.HinhAnh, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-5">

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

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

</div>

<div class="col-md-1">

<input type="button" class="btn-primary form-control" id="browse" value="Chọn File" />

</div>

<script>

$("#browse").click(function () {

var finder = new CKFinder();

finder.selectActionFunction = function (fileUrl) {

//fileUrl = fileUrl.substring(fileUrl.lastIndexOf("/") + 1);

$("#HinhAnh").val(fileUrl);

}

finder.popup();

});

</script>

</div>

<div class="form-group">

@Html.LabelFor(model => model.MotaNgan, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

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

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

</div>

<script type="text/javascript">

CKEDITOR.replace("MotaNgan");

</script>

</div>

<div class="form-group">

@Html.LabelFor(model => model.NguoiTao, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

@Html.EditorFor(model => model.NguoiTao, new { htmlAttributes = new { @class = "form-control", @Value = Session["ten"].ToString()} })

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

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.NgayTao, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

@Html.EditorFor(model => model.NgayTao, new { htmlAttributes = new { @class = "form-control", @Value = DateTime.Now.ToString() } })

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

</div>

</div>

<div class="form-group" style="text-align:center">

<div class="col-md-offset-2 col-md-8">

<input type="submit" value="Tạo mới" class="btn btn-primary" />&nbsp;

@Html.ActionLink("Quay lại", "Index","LoaiNs", new { @class = "btn btn-danger" })

</div>

</div>

</div>

}

public ActionResult Create()

{

return View();

}

[HttpPost]

[ValidateInput(false)]

public ActionResult Create([Bind(Include = "IDLoaiNS,TenLoai,HinhAnh,MotaNgan,NguoiTao,NgayTao,NguoiCapNhat,NgayCapNhat")] LoaiN loaiN)

{

if (ModelState.IsValid)

{

db.LoaiNS.Add(loaiN);

db.SaveChanges();

return RedirectToAction("Index");

}

return View(loaiN);

}

**d. Xử lý cập nhập nông sản:**

@model NongSanVietNam.Models.LoaiN

@{

ViewBag.Title = "Edit";

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

}

<h2>&nbsp;</h2>

@using (Html.BeginForm())

{

@Html.AntiForgeryToken()

<div class="form-horizontal">

<h3 style="text-align:center; color:#0056f1;"><b>CẬP NHẬT LOẠI NÔNG SẢN</b></h3>

<hr />

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

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

<div class="form-group">

@Html.LabelFor(model => model.TenLoai, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

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

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

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.HinhAnh, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-5">

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

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

</div>

<div class="col-md-1">

<input type="button" class="btn-primary form-control" id="browse" value="Chọn File" />

</div>

<script>

$("#browse").click(function () {

var finder = new CKFinder();

finder.selectActionFunction = function (fileUrl) {

$("#HinhAnh").val(fileUrl);

}

finder.popup();

});

</script>

</div>

<div class="form-group">

@Html.LabelFor(model => model.MotaNgan, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

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

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

</div>

<script type="text/javascript">

CKEDITOR.replace("MotaNgan");

</script>

</div>

<div class="form-group">

@Html.LabelFor(model => model.NguoiTao, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

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

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

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.NgayTao, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

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

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

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.NguoiCapNhat, htmlAttributes: new { @class = "control-label col-md-2"})

<div class="col-md-8">

@Html.EditorFor(model => model.NguoiCapNhat, new { htmlAttributes = new { @class = "form-control", @Value = Session["ten"].ToString() } })

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

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.NgayCapNhat, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-8">

@Html.EditorFor(model => model.NgayCapNhat, new { htmlAttributes = new { @class = "form-control", @Value = DateTime.Now.ToString() } })

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

</div>

</div>

<div class="form-group" style="text-align:center">

<div class="col-md-offset-2 col-md-8">

<input type="submit" value="Lưu lại" class="btn btn-primary" />

@Html.ActionLink("Hủy bỏ", "Index", "LoaiNs", new { @class = "btn btn-danger" })

</div>

</div>

</div>

}

public ActionResult Edit(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

LoaiN loaiN = db.LoaiNS.Find(id);

if (loaiN == null)

{

return HttpNotFound();

}

return View(loaiN);

}

[HttpPost]

[ValidateInput(false)]

public ActionResult Edit([Bind(Include = "IDLoaiNS,TenLoai,HinhAnh,MotaNgan,NguoiTao,NgayTao,NguoiCapNhat,NgayCapNhat")] LoaiN loaiN)

{

if (ModelState.IsValid)

{

db.Entry(loaiN).State = EntityState.Modified;

db.SaveChanges();

return RedirectToAction("Index");

}

return View(loaiN);

}

**eXử lý xóa nông sản:**

@model NongSanVietNam.Models.LoaiN

@{

ViewBag.Title = "Delete";

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

}

<h2>Delete</h2>

<h3>Are you sure you want to delete this?</h3>

<div>

<h4>LoaiN</h4>

<hr />

<dl class="dl-horizontal">

<dt>

@Html.DisplayNameFor(model => model.TenLoai)

</dt>

<dd>

@Html.DisplayFor(model => model.TenLoai)

</dd>

<dt>

@Html.DisplayNameFor(model => model.HinhAnh)

</dt>

<dd>

@Html.DisplayFor(model => model.HinhAnh)

</dd>

<dt>

@Html.DisplayNameFor(model => model.MotaNgan)

</dt>

<dd>

@Html.DisplayFor(model => model.MotaNgan)

</dd>

<dt>

@Html.DisplayNameFor(model => model.NguoiTao)

</dt>

<dd>

@Html.DisplayFor(model => model.NguoiTao)

</dd>

<dt>

@Html.DisplayNameFor(model => model.NgayTao)

</dt>

<dd>

@Html.DisplayFor(model => model.NgayTao)

</dd>

<dt>

@Html.DisplayNameFor(model => model.NguoiCapNhat)

</dt>

<dd>

@Html.DisplayFor(model => model.NguoiCapNhat)

</dd>

<dt>

@Html.DisplayNameFor(model => model.NgayCapNhat)

</dt>

<dd>

@Html.DisplayFor(model => model.NgayCapNhat)

</dd>

</dl>

@using (Html.BeginForm()) {

@Html.AntiForgeryToken()

<div class="form-actions no-color">

<input type="submit" value="Delete" class="btn btn-default" /> |

@Html.ActionLink("Back to List", "Index")

</div>

}

</div>

public ActionResult Delete(int[] id)

{

foreach(int i in id)

{

LoaiN lns = db.LoaiNS.Find(i);

var dsns = db.NongSans.Where(s => s.IDLoaiNS == i).ToList();

if(dsns.Count > 0)

{

foreach(var ns in dsns)

{

db.NongSans.Remove(ns);

db.SaveChanges();

}

}

db.LoaiNS.Remove(lns);

db.SaveChanges();

}

return RedirectToAction("Index");

}

1. **KẾT LUẬN**

***\* Ưu điểm :***

*-Đối với người dùng:*

• Người tiêu dùng có thể trực tiếp tìm hiểu, thu thập thông tin về nông sản một cách đầy đủ và chi tiết. Qua đó, hạn chế mua phải hàng kém chất lượng, hàng giả, hàng nhái, đặc biệt là những thực phẩm ảnh hưởng trực tiếp đến sức khỏe của người tiêu dùng.

• Giải pháp giúp nâng tầm giá trị nông sản, bảo vệ thương hiệu của nhà cung cấp, giúp tăng cường hiệu quả truyền thông, tạo được niềm tin cho khách hàng đối với những nông phẩm của mình.

*- Đối với người lập trình:*

• Nâng cao được khả năng lập trình font-end , back-end về asp.net và Sql.

• Học thêm được 1 số kiến thức nâng cao như mô hình MVC, một số giao diện upload dữ liệu tốt.

***\* Hạn chế :***

- Thao tác vẫn còn chậm khi lập trình.

- Chưa tư duy cao về các vần đề mới còn phải tham khảo tài liệu trên mạng nhiều.

- 1 số chức năng của ứng dụng chưa được tối ưu hóa, vẫn còn thô sơ về mặt hình thức giao diện.

- Giao diện chưa thu hút đặc biệt với người dùng, chưa có ý tưởng sáng tạo.