

Lab 06

Data Access

Mục tiêu

- Biết sử dụng các phương pháp truy cập cơ sở dữ liệu SQL Server sử dụng Entity Framework.
- Tích hợp CKEditor và CKFinder vào ứng dụng

Phần I Bài tập step by step

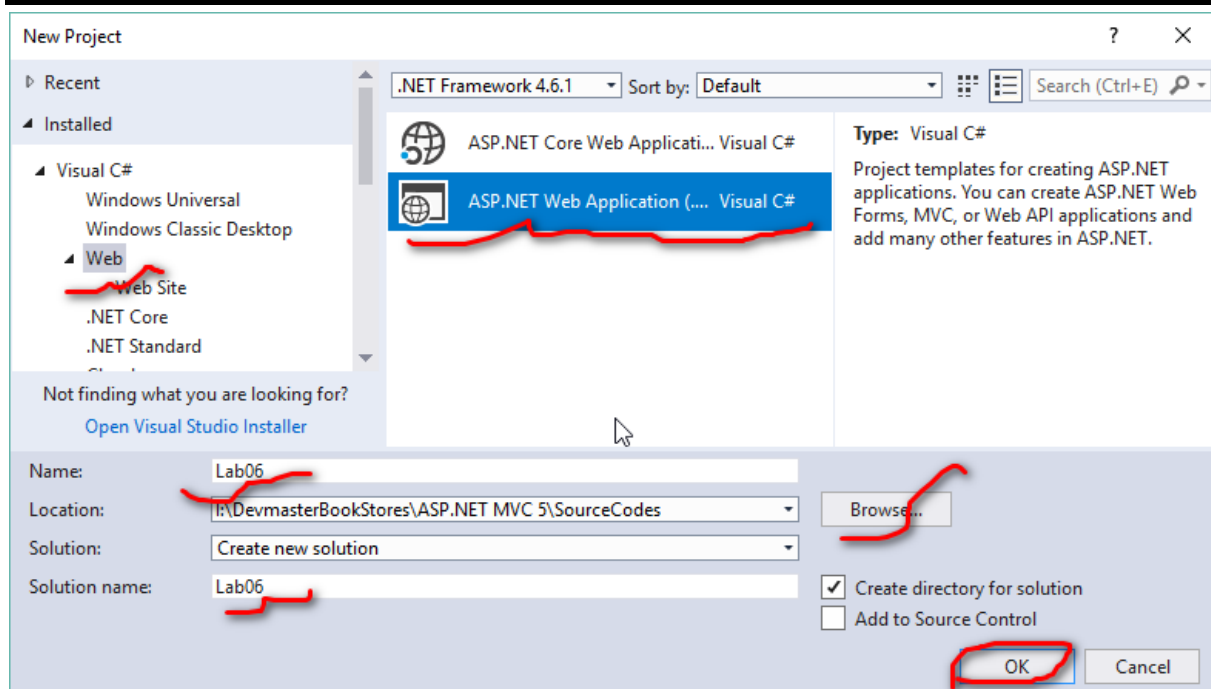
Bài 6.1

Tạo ứng dụng ASP.NET MVC5 để truy xuất cơ sở dữ liệu SQL Server sử dụng phương pháp Code First:

- Tạo lớp Product(ProductId, Name, Price) mô tả sản phẩm.
- Tạo lớp DataContext để quản lý việc thao tác với CSDL.
- Tạo Controller và View thực hiện việc hiển thị, thêm, xóa, sửa dữ liệu trong bảng Product.

Bước 1: Mở Visual Studio 2017 -> vào File -> New -> Projects -> Chọn Web -> ASP.NET Web Application -> Nhập tên project -> Chọn đường dẫn lưu trữ -> Nhập tên solution -> Nhấn OK (xem hình dưới).





Nhấn OK -> màn hình tiếp theo xuất hiện bạn chọn template là MVC sau đó chọn Change Authentication -> No Authentication -> OK -> OK.

Bước 2: Kích chuột phải vào thư mục Models chọn Add -> Class -> Nhập tên "Product" -> Add và code như sau:

```
public class Product
{
    public int ProductId { get; set; }
    public string ProductName { get; set; }
    public int Price { get; set; }
}
```

Bước 3: Kích chuột phải vào thư mục Models chọn Add -> Class -> Nhập tên "ShopContext" -> Add và code như sau:

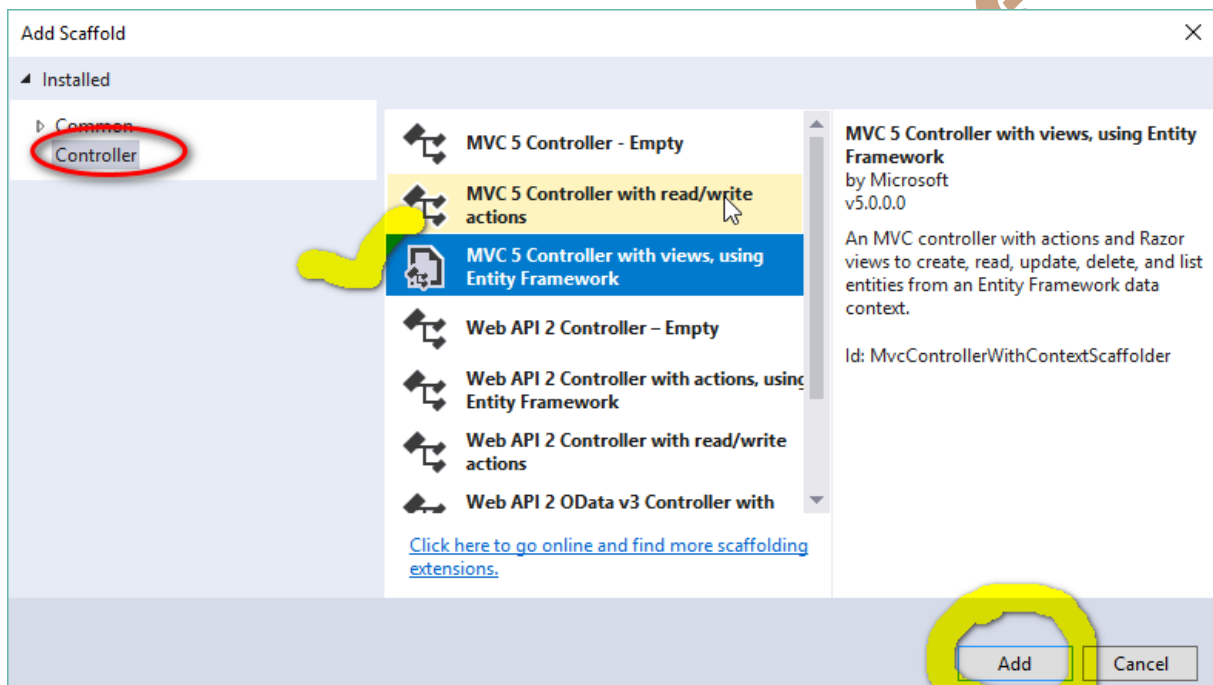
```
public class ShopContext:DbContext
{
    public ShopContext():base("name=ShopContext")
    {
        //khởi tạo những gì cần thiết
    }
    public DbSet<Product> Products { get; set; }
}
```

Web Application Development (ASP.NET MVC 5)

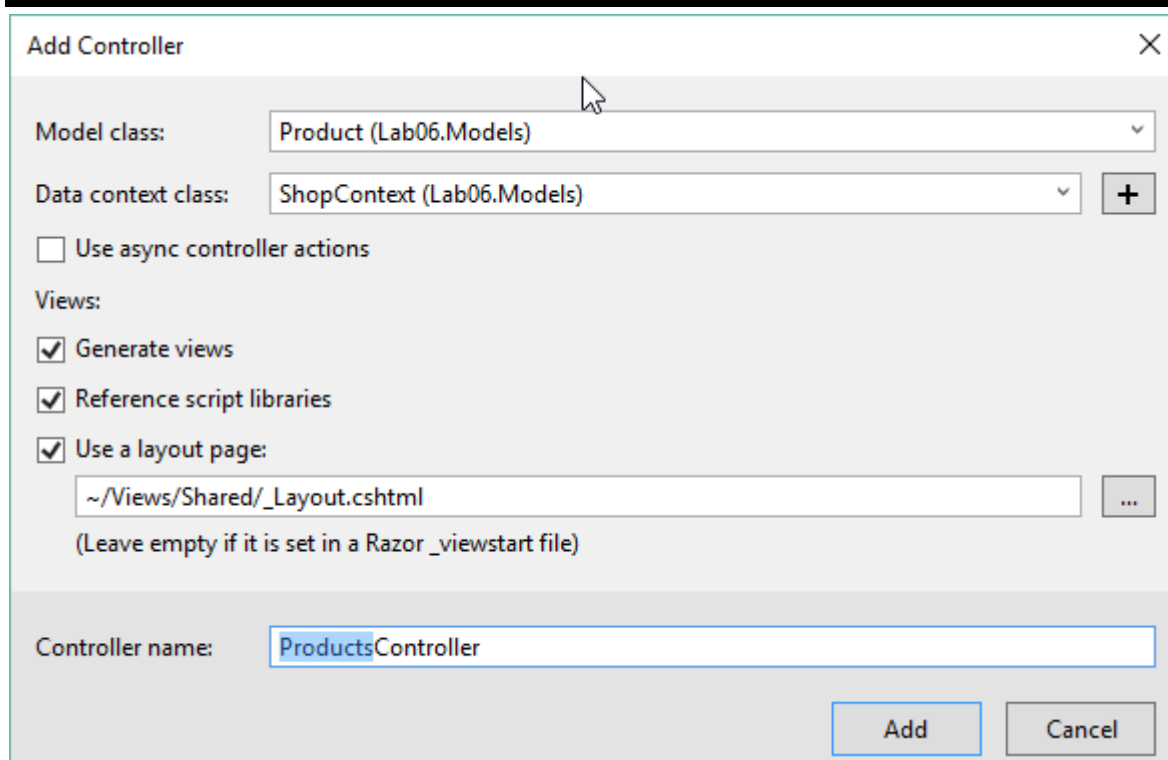
Bước 4: Mở tệp Web.config trong project -> bổ sung đoạn cấu hình kết nối sau vào trong thẻ **configuration**

```
<connectionStrings>  
  <add name="ShopContext"  
  connectionString="server=ChungTrinh\SQL2016;database=Lab06_1;uid=sa;pwd=12346  
5" providerName="System.Data.SqlClient"/>  
</connectionStrings>
```

Bước 5: Kích chuột phải vào thư mục Controllers chọn Add -> Controllers -> Chọn MVC 5 Controller with views, using Entity Framework -> chọn Add.



Nhập tên controller và chọn Model class, Data context class -> chọn Add



Add Controller

Model class: Product (Lab06.Models)

Data context class: ShopContext (Lab06.Models)

☐ Use async controller actions

Views:

☒ Generate views

☒ Reference script libraries

☒ Use a layout page:

~/Views/Shared/_Layout.cshtml

(Leave empty if it is set in a Razor _viewstart file)

Controller name: ProductsController

Add Cancel

Bước 6: Chạy và kiểm tra kết quả

- Trang sản phẩm

DEVMASTER.vn Home About Contact Product

Index

[Create New](#)

ProductName	Price
-------------	-------

© 2018 - ITRD- <http://www.devmaster.edu.vn>

- Trang tạo mới

Create

Product

ProductName

Price

[Back to List](#)© 2018 - ITRD- <http://www.devmaster.edu.vn>

- Kết quả

Index

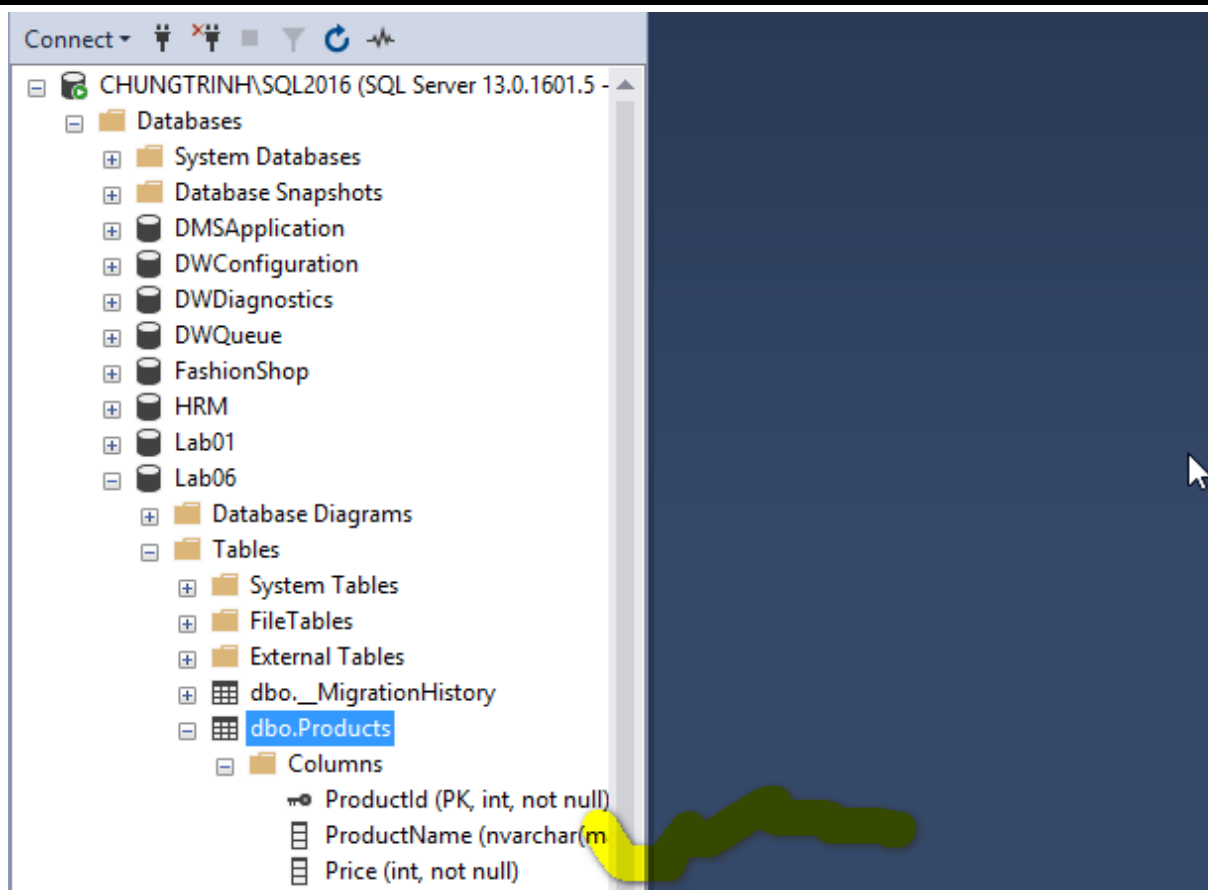
[Create New](#)

ProductName	Price	
Bình nước lọc Kanguru	150000	Edit Details Delete

© 2018 - ITRD- <http://www.devmaster.edu.vn>

- Mở SQL Server và kiểm tra





Bài 6.2

Đưa CKEditor và CKFinder vào ứng dụng ASP.NET MVC 5

Bước 1: Download bản mới nhất CKEditor và CKFinder từ trang

<http://ckeditor.com/download> và <http://cksource.com/ckfinder>

Bước 2: Kích chuột phải vào Solution Session6 và chọn Add -> New Project đặt tên "Lab6_2" -> chọn MVC và No Authentication.

Bước 3: Giải nén tệp **ckeditor_aspnet_3.6.6.2.zip** vừa download về tìm thư mục ckeditor trong thư mục **_Samples** copy vào thư mục Content của project.

Bước 4: Giải nén tệp **ckfinder_aspnet_2.4.2.zip** vừa download về tìm thư mục ckfinder trong thư mục giải nén copy vào thư mục Content của project, sau đó xóa thư mục **source** và **_Sample** đi.

Bước 5: Kích chuột phải vào mục References chọn Add References chọn tệp CKFinder.dll trong thư mục ckfinder/bin/Release vào project

Bước 6: Mở trang _Layout.cshtml trong thư mục Views/Shared/ và bổ sung các lệnh sau vào trong thẻ header:

```
@Scripts.Render("~/bundles/jquery") - (ở cuối trang có dòng này thì  
xóa đi)  
<script src("~/Content/ckeditor/ckeditor.js"></script>  
<script src("~/Content/ckfinder/ckfinder.js"></script>
```

Bước 7: Mở tệp tin ckfinder/config.ascx

Trong phương thức CheckAuthentication() sửa lại dòng sau:

```
return true;
```

Trong phương thức SetConfig() sửa lại 2 dòng sau:

```
// đường dẫn thư mục tương đối để CKFinder tìm đến.  
BaseUrl = "/Content/";  
// đường dẫn thư mục.  
BaseDir = HttpContext.Current.Server.MapPath("~/Content/");
```

Bước 8: Mở tệp tin ckeditor/config.js cấu hình lại đường dẫn cho editor gọi :

```
CKEDITOR.editorConfig = function( config )  
{  
    // Chỉ ra ngôn ngữ  
    config.language = 'en';  
    // config.uiColor = '#AADC6E';  
    // cấu hình đường dẫn các loại tệp tin khi finder  
    config.filebrowserBrowseUrl = "/Content/ckfinder/ckfinder.html";  
    config.filebrowserImageUrl =  
        "/Content/ckfinder/ckfinder.html?type=Images";  
    config.filebrowserFlashUrl =  
        "/Content/ckfinder/ckfinder.html?type=Flash";
```



```
config.filebrowserUploadUrl =  
    "/Content/ckfinder/core/connector/aspx/connector.aspx?command=QuickUpload&type=Files";  
config.filebrowserImageUploadUrl =  
    "/Content/ckfinder/core/connector/aspx/connector.aspx?command=QuickUpload&type=Images";  
config.filebrowserFlashUploadUrl =  
    "/Content/ckfinder/core/connector/aspx/connector.aspx?command=QuickUpload&type=Flash";  
};
```

Bước 9: Mở tệp Index.cshtml trong Views/Home -> xóa code mẫu trong đó và thay bằng đoạn code sau:

```
@{  
    ViewBag.Title = "Home Page";  
}  
<h3>Sử dụng CKFinder và CKEditor</h3>  
@using(Html.BeginForm())  
{  
    <div>  
        Hình ảnh: @Html.TextBox("Picture")  
        <input type="button" id="Browse" value="Duyệt ảnh" />  
    </div>  
    <div>  
        <h4>Mô tả chi tiết</h4>  
        @Html.TextArea("Detail")  
        @*đoạn code đáp ứng việc áp dụng Editor cho TextArea (phải đặt sau  
        TextArea)*@  
        <script type="text/javascript">  
            CKEDITOR.replace("Detail");  
        </script><br />  
        <input type="submit" value="Lưu"/>  
    </div>  
}
```

@*đoạn code đáp ứng sự kiện click của nút Browse (phải đặt sau nút)*@


```
<script>
    $("#Browse").click(function () {
        var finder = new CKFinder();
        finder.selectActionFunction = function (fileUrl) {
            //lấy file ảnh cuối cùng
            fileUrl = fileUrl.substring(fileUrl.lastIndexOf("/") + 1);
            $("#Picture").val(fileUrl);
        };
        finder.popup();
    });
</script>
```

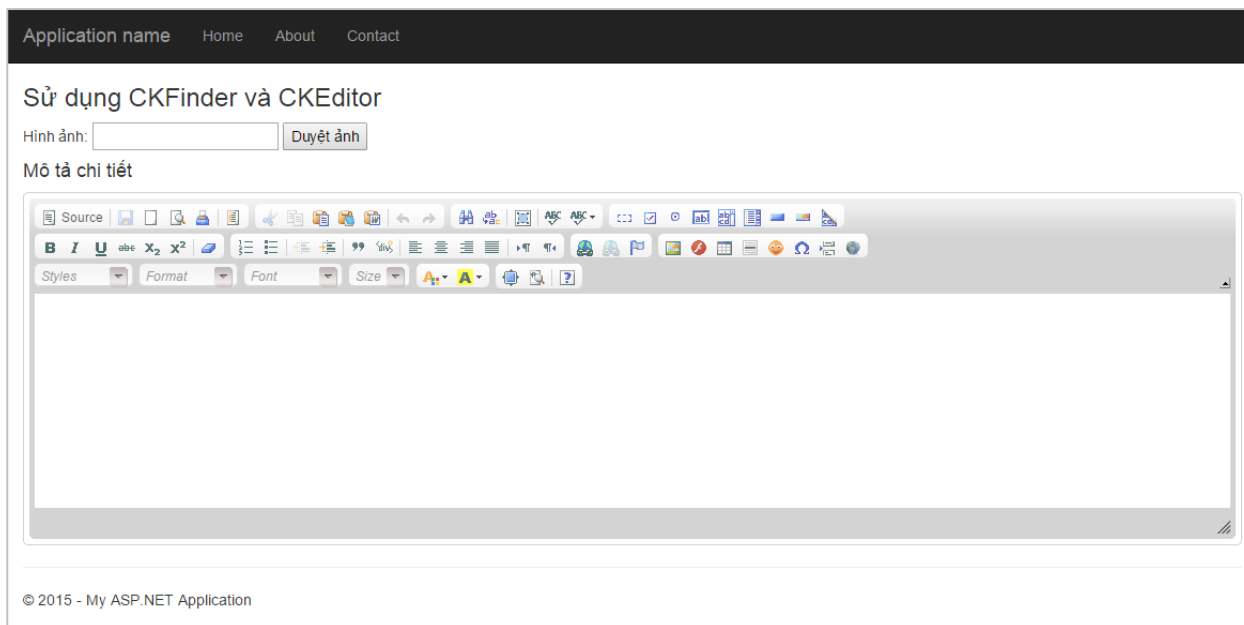
Bước 10: Mở HomeController và thêm code sau:

```
//đáp ứng hành động Post trên form
[HttpPost]
//không validate đầu vào
[ValidateInput(false)]
public ActionResult Index(string picture, string detail)
{
    ViewBag.picture = picture;
    ViewBag.detail = detail;
    //chuyển tới view Detail để hiển thị
    return View("Detail");
}
public ActionResult Detail()
{
    return View();
}
```

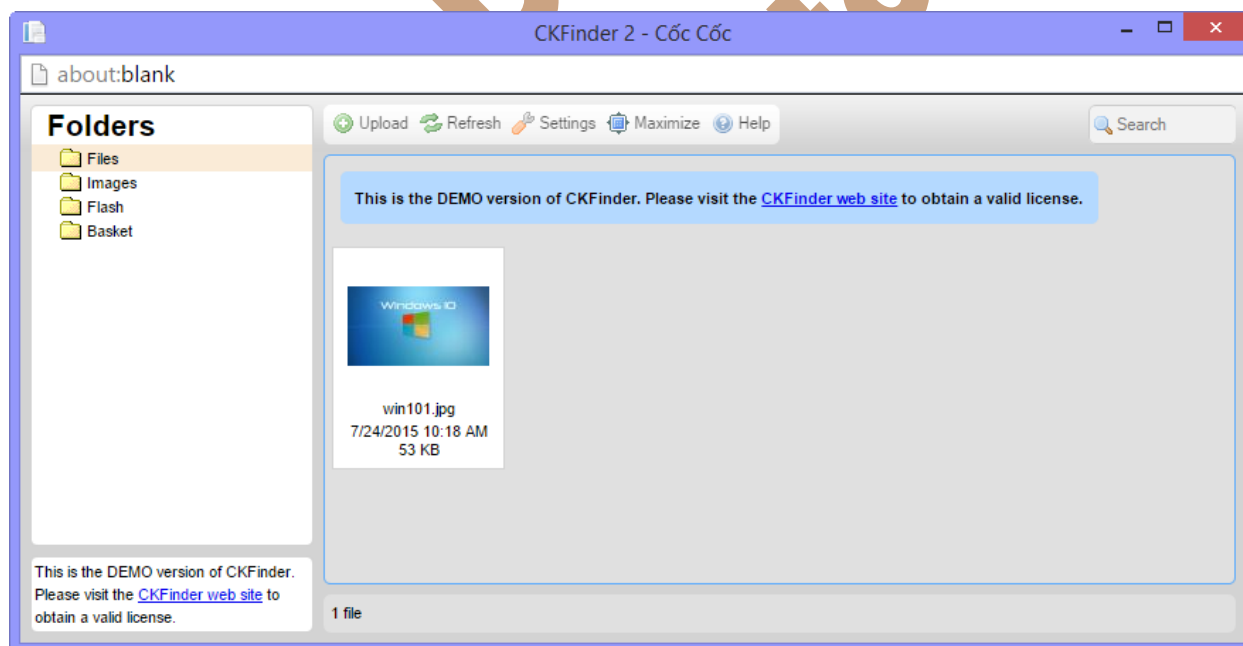
Bước 11: Chuột phải vào phương thức Detail tạo view và nhập code sau cho view

```
@{
    ViewBag.Title = "Detail";
}
<h3>Thông tin submit</h3>
Ảnh: @ViewBag.picture<br />
Chi tiết<br />
@Html.Raw(ViewBag.detail)
```



Bước 12: Chạy và xem kết quả

- Khi kích vào nút “Duyệt ảnh” màn hình xuất hiện như hình dưới, tại đây bạn có thể upload ảnh và chọn ảnh.

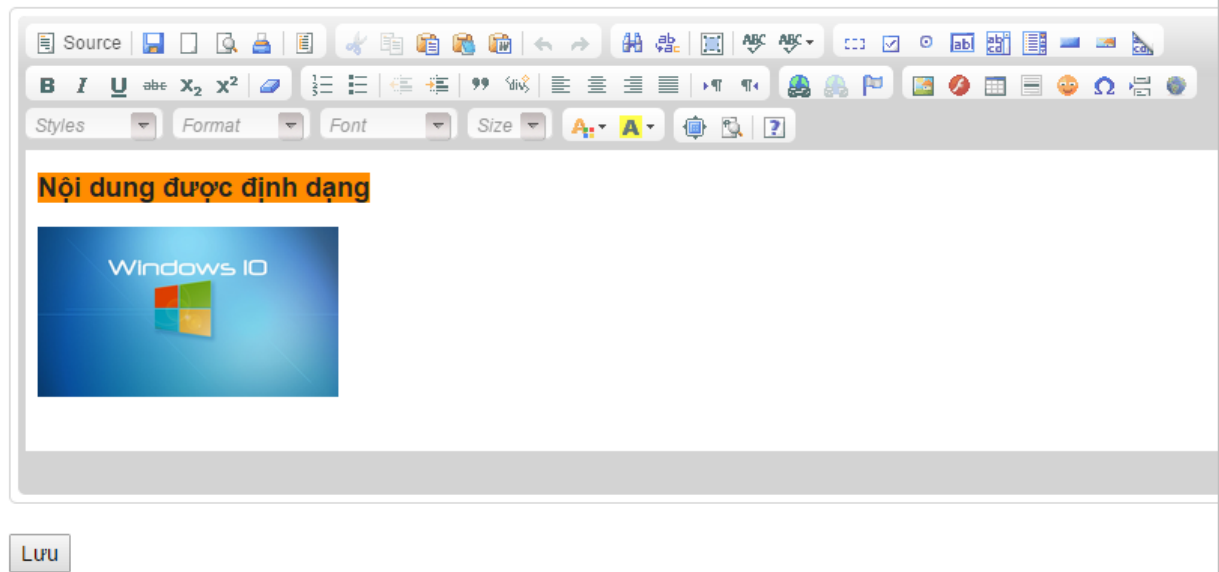


- Nhập dữ liệu để submit

Sử dụng CKFinder và CKEditor

Hình ảnh:

Mô tả chi tiết



- Kết quả khi submit

Thông tin submit

Ảnh: win101.jpg

Chi tiết

Nội dung được định dạng

© 2015 - My ASP.NET Application



Bài 6.3

Tạo ứng dụng ASP.NET MVC 5 với các yêu cầu sau:

- Hiển thị danh sách món ăn theo mẫu có phân trang dữ liệu
- Tìm kiếm món ăn theo tên, lọc món ăn theo nhóm
- Thêm sản phẩm (có tích hợp CKEditor và CKFinder)
- Sử dụng Entity Framework để truy xuất dữ liệu
- Cơ sở dữ liệu được cung cấp sẵn kèm theo ảnh

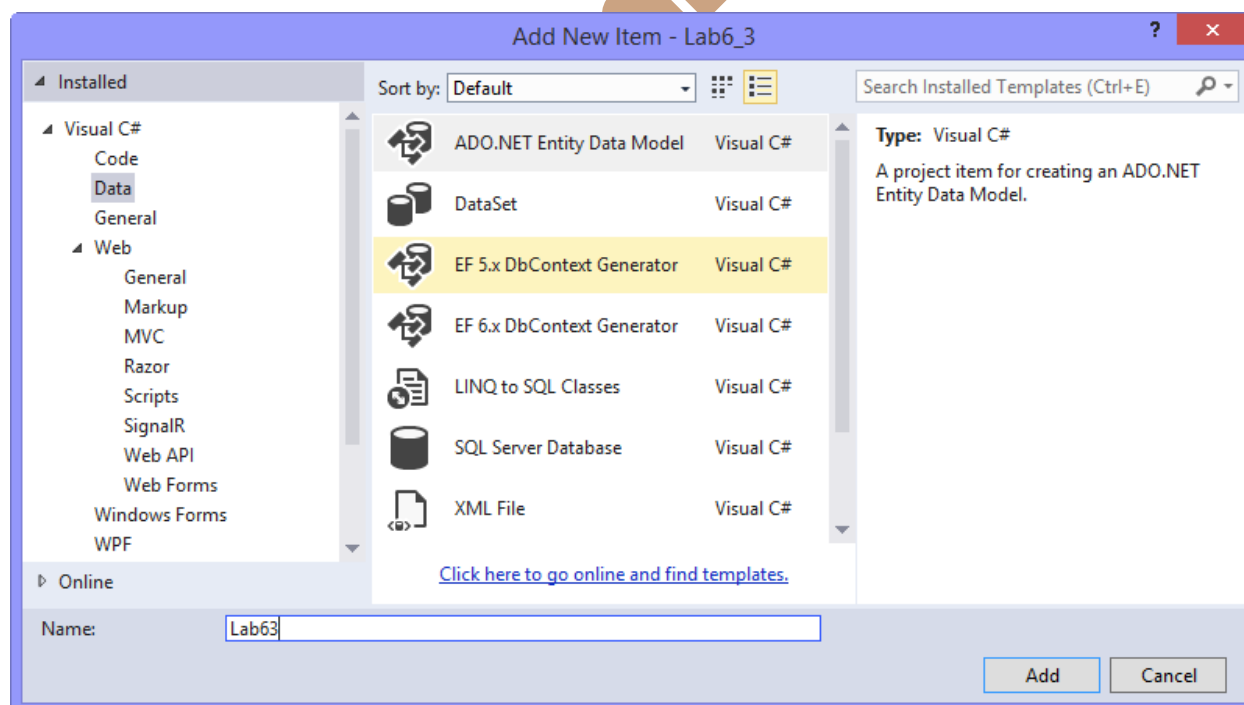
Bước 1: Mở SQL Server và chạy tệp Script được cung cấp

Bước 2: Kích chuột phải vào Solution Session6 và chọn Add -> New Project đặt tên "Lab6_3" -> chọn MVC và No Authentication -> OK.

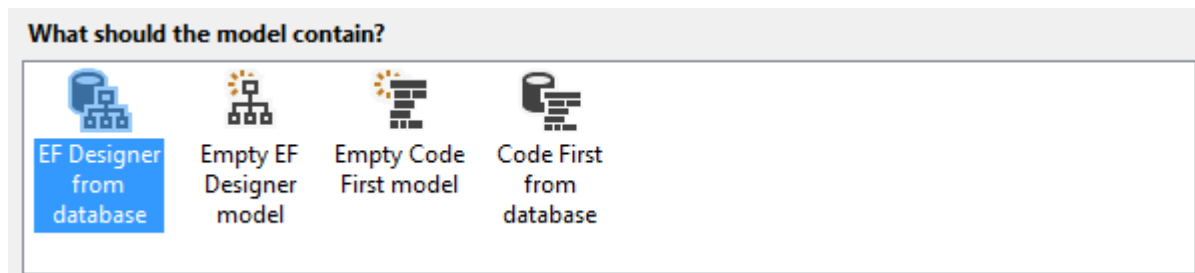
Bước 3: Copy thư mục images món ăn được cung cấp vào thư mục Content của project.

Bước 4: Đưa CKEditor và CKFinder vào thư mục Content và chỉnh sửa cấu hình như bài Lab6_2

Bước 5: Kích chuột phải vào Models chọn Add -> New Item -> ADO.NET Entity Data Model -> đặt tên Lab63

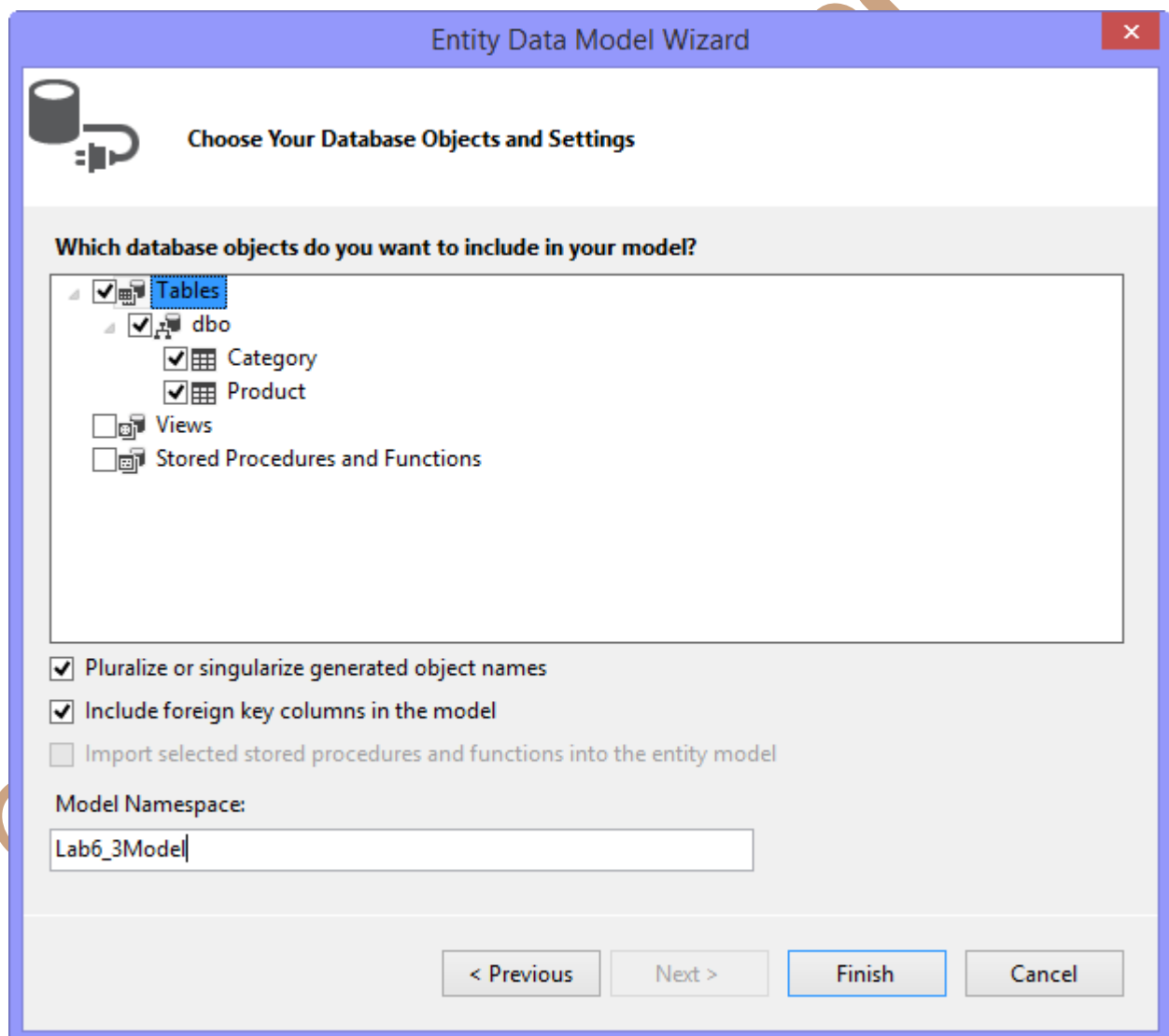


Chọn Add



Chọn Next

Chọn Next -> Next



Kích Finish để kết thúc

Bước 6: Kích chuột phải vào Controllers chọn Add -> New Item -> Controller và đặt tên “ProductController” sau đó code cho controller đó theo gợi ý sau:

```
public class ProductController : Controller
{
    //khởi tạo Entity Model
    Lab6_3Entities lab = new Lab6_3Entities();
    // GET: /Product/ShowProducts/pageno
    public ActionResult ShowProducts(int? pageno)
    {
        //thiết lập số trang
        int pagesize = 12;
        //lấy tất cả thực đơn
        var products = lab.Products.AsEnumerable();
        //tính tổng số thực đơn
        var count = products.Count();
        //tính số trang
        int pagecount = (count % pagesize == 0) ? count / pagesize : (count /
        pagesize) + 1;
        //nếu trang không được kích thì gán trang hiện tại là 1
        if (pageno == null)
        {
            pageno = 1;
        }
        //lấy ra giá trị tổng số trang để chuyển qua view
        ViewBag.pagecount = pagecount;
        //lấy số trang được chọn để chuyển qua view
        ViewBag.pageno = pageno;
        //lấy ra các bản ghi đúng trang mà người dùng chọn
        var result = lab.Products.OrderBy(p =>
        p.ProductName).Skip((pageno.Value - 1) * pagesize).Take(pagesize);
        //đưa kết quả ra view
        return View(result);
    }
    //GET: Product/Search
    public ActionResult Search()
    {
```

```
//lấy danh sách loại
 ViewBag.listcate = lab.Categories.ToList();
 return View();
}

//POST: Product/Search
[HttpPost]
public ActionResult Search(string name, int cateid)
{
    //lấy danh sách thực đơn theo loại và tên bắt đầu bằng chuỗi nhập vào
    var result = (from p in lab.Products where p.CateId == cateid &&
        p.ProductName.ToLower().StartsWith(name.ToLower()) select
        p).AsEnumerable();
    //lấy danh sách loại
    ViewBag.listcate = lab.Categories.ToList();
    //lấy loại thực đơn được chọn
    ViewBag.cateid = cateid;
    //chuyển kết quả tìm kiếm tới View
    return View(result);
}

//GET: Product/Create
public ActionResult Create()
{
    //lấy danh sách loại
    ViewBag.listcate = lab.Categories.ToList();
    return View();
}

//POST: /Product/Create
[HttpPost]
[ValidateInput(false)] //không validate dữ liệu đầu vào
public ActionResult Create(string productname, int? price, string unit,
    string Picture, int cateid, bool? check1, bool? check2, bool? check3,
    string keyword, string description, string detail)
{
    //tạo sản phẩm mới
    Product p = new Product();
    p.ProductName = productname;
    p.Price = price;
```



```
p.unit = unit;
p.CateId = cateid;
p.Picture = Picture;
p.Check1 = check1;
p.Check2 = check2;
p.Check3 = check3;
p.keyword = keyword;
p.description = description;
p.Detail = detail;
//thêm
lab.Products.Add(p);
//ghi
lab.SaveChanges();
//chuyển đến view hiển thị sản phẩm
return RedirectToAction("ShowProducts");
}
}
```

Bước 7: Mở tệp Site.css trong thư mục Content ra và bổ sung code css như sau:

```
/* Phần định nghĩa css bố cục sản phẩm */
.products {
    margin: auto;
    width: 100%;
    display: inline-block;
    background: #f6eeee;
}

.product {
    float: left;
    margin: 5px;
    width: 180px;
    height: 190px;
    text-align: center;
    background-color: #b94a48;
    padding: 10px;
}

.productname
{
    padding-top: 10px;
    color: white;
    text-wrap: normal;
}

/*hết phần định nghĩa CSS*/
```


Bước 8: Kích chuột phải vào phương thức ShowProducts để tạo view và code Razor theo gợi ý sau

```
@model IEnumerable<Lab6_3.Models.Product>
@{
    ViewBag.Title = "ShowProducts";
    //lấy tổng số trang và số trang hiện tại
    int pagecount = ViewBag.pagecount;
    int pageno = ViewBag.pageno;
}
<h2>Dách sách sản phẩm</h2>
<hr />
<div class="products">
    @foreach (var p in Model)
    {
        <div class="product">
            <img width="160" height="140"
src=@Url.Content("~/Content/images/")@p.Picture' title="@p.ProductName"/>
            <div class="productname">@p.ProductName</div>
        </div>
    }
</div>
@*Tạo số trang*@
<hr />Trang:
@for (int i = 0; i < pagecount; i++)
{
    if(i+1==pageno)
    {
        @Html.DisplayName(" " + (i + 1).ToString() + " ");
    }
    else
    {
        @Html.ActionLink("[ " + (i + 1).ToString() + "]", "ShowProducts", new
{ pageno = i + 1 });
    }
}
```

Bước 9: Kích chuột phải vào phương thức Search để tạo view và code Razor theo gợi ý sau

```
@model IEnumerable<Lab6_3.Models.Product>
@{
    ViewBag.Title = "Tìm kiếm thực đơn";
}
@*Tạo form tìm kiếm*@
<h2>Tìm kiếm thực đơn</h2>
@using (Html.BeginForm())
{
    <span>Chọn chọn loại</span>
    if (ViewBag.cateid != null)
    {
        @Html.DropDownList("cateid", new SelectList(ViewBag.listcate,
"CateId", "Name", ViewBag.cateid))
    }
    else
    {
        @Html.DropDownList("cateid", new SelectList(ViewBag.listcate,
"CateId", "Name"))
    }
    <span>Nhập tên</span>
    @Html.TextBox("name")
    <input type="submit" value="Tìm" />
}
<hr />
@*Hiển thị sản phẩm nếu Model có dữ liệu*@
<div class="products">
    @if (Model != null)
    {
        foreach (var p in Model)
        {
            <div class="product">
                
                <div class="productname">@p.ProductName</div>
            </div>
        }
    }
}
```

```

        </div>
    }
}
</div>

```

Bước 10: Kích chuột phải vào phương thức Create để tạo view và code Razor theo gợi ý sau (lưu ý chọn template Create và Model class nhé)

```
@model Lab6_3.Models.Product
```

```
@{
```

```
    ViewBag.Title = "Thêm sản phẩm";
```

```
}
```

```
<h3>Thêm sản phẩm mới</h3>
```

```
@using (Html.BeginForm())
```

```
{
```

```
    @Html.AntiForgeryToken()
```

```
    <div class="form-horizontal">
```

```
        <hr />
```

```
        @Html.ValidationSummary(true)
```

```
        <div class="form-group">
```

```
            @Html.Label("Tên sản phẩm", new { @class = "control-label col-md-2" })
```

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

```
                @Html.EditorFor(model => model.ProductName)
```

```
                @Html.ValidationMessageFor(model => model.ProductName)
```

```
            </div>
```

```
        </div>
```

```
        <div class="form-group">
```

```
            @Html.Label("Giá", new { @class = "control-label col-md-2" })
```

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

```
                @Html.EditorFor(model => model.Price)
```

```
                @Html.ValidationMessageFor(model => model.Price)
```

```
            </div>
```

```
        </div>
```

```
        <div class="form-group">
```



```
@Html.Label("ĐVT", new { @class = "control-label col-md-2" })
<div class="col-md-10">
    @Html.EditorFor(model => model.unit)
    @Html.ValidationMessageFor(model => model.unit)
</div>
</div>
<div class="form-group">
    @Html.Label("Hình ảnh", new { @class = "control-label col-md-2" })

    <div class="col-md-10">
        @Html.EditorFor(model => model.Picture)
        <input type="button" id="Browse" value="Duyệt ảnh" />
    </div>
</div>
<div class="form-group">
    @Html.Label("Loại", new { @class = "control-label col-md-2" })
    <div class="col-md-10">
        @Html.DropDownListFor(model => model.CateId, new
SelectList(@ViewBag.listcate, "CateId", "Name"))
    </div>
</div>
<div class="form-group">
    @Html.Label("Món mới", new { @class = "control-label col-md-2" })
    <div class="col-md-10">
        @Html.CheckBox("Check1")
    </div>
</div>
<div class="form-group">
    @Html.Label("Hôm nay có", new { @class = "control-label col-md-2" })
    <div class="col-md-10">
        @Html.CheckBox("Check2")
    </div>
</div>
<div class="form-group">
    @Html.Label("Trạng thái", new { @class = "control-label col-md-2" })
    <div class="col-md-10">
        @Html.EditorFor(model => model.Status)
    </div>
</div>
</div>
</div>
```

```

        <div class="col-md-10">
            @Html.CheckBox("Check3")
        </div>
    </div>
    <div class="form-group">
        @Html.Label("Từ khóa SEO", new { @class = "control-label col-md-
2" })

        <div class="col-md-10">
            @Html.EditorFor(model => model.keyword)
        </div>
    </div>
    <div class="form-group">
        @Html.Label("Mô tả SEO", new { @class = "control-label col-md-2"
    })

        <div class="col-md-10">
            @Html.EditorFor(model => model.description)
        </div>
    </div>
    <div class="form-group">
        @Html.Label("Chi tiết", new { @class = "control-label col-md-2"
    })

        <div class="col-md-10">
            @Html.EditorFor(model => model.Detail)
        </div>
        @*đoạn code đáp ứng việc áp dụng Editor cho TextArea (phải đặt
sau TextArea)*@
        <script type="text/javascript">
            CKEDITOR.replace("Detail");
        </script>
    </div>
    <div class="form-group">
        <div class="col-md-offset-2 col-md-10">
            <input type="submit" value=" Lưu " class="btn btn-default" />
        </div>
    </div>
</div>
}

```

```
<div>
    @Html.ActionLink("Quay lại", "ShowProducts")
</div>
@*đoạn code đáp ứng sự kiện click của nút Browse (phải đặt sau nút)*@
<script>
    $("#Browse").click(function () {
        var finder = new CKFinder();
        finder.selectActionFunction = function (fileUrl) {
            //lấy file ảnh cuối cùng
            fileUrl = fileUrl.substring(fileUrl.lastIndexOf("/") + 1);
            $("#Picture").val(fileUrl);
        };
        finder.popup();
    });
</script>
```

Bước 11: Trong trang View/Shared/_Layout.cshtml bổ sung đoạn script sau:

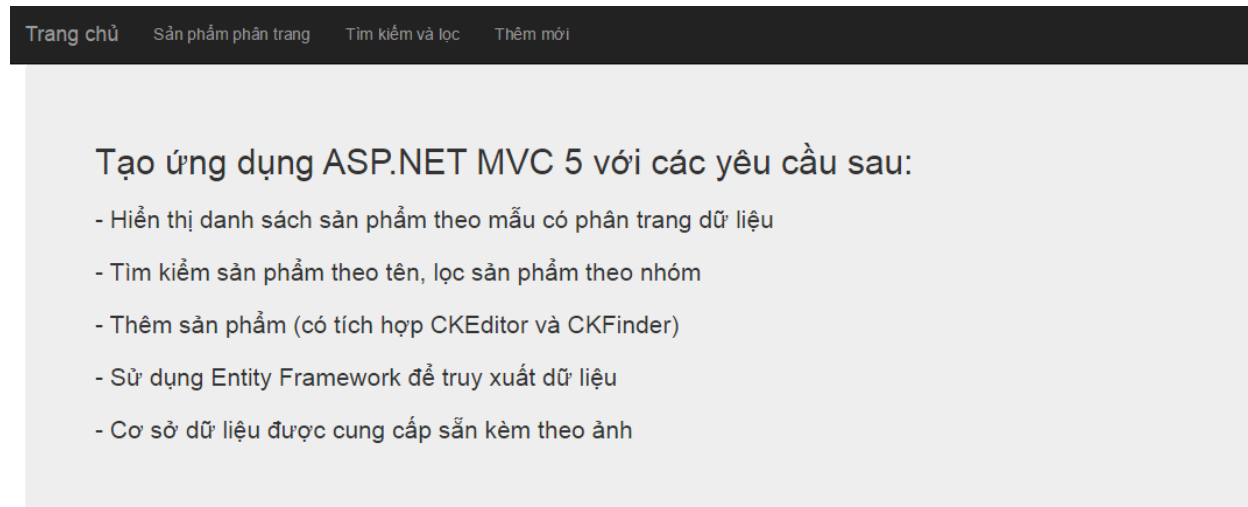
```
@Scripts.Render("~/bundles/jquery")
<script src("~/Content/ckeditor/ckeditor.js")></script>
<script src("~/Content/ckfinder/ckfinder.js")></script>
```

Và thêm sửa lại Link trên trang như sau:

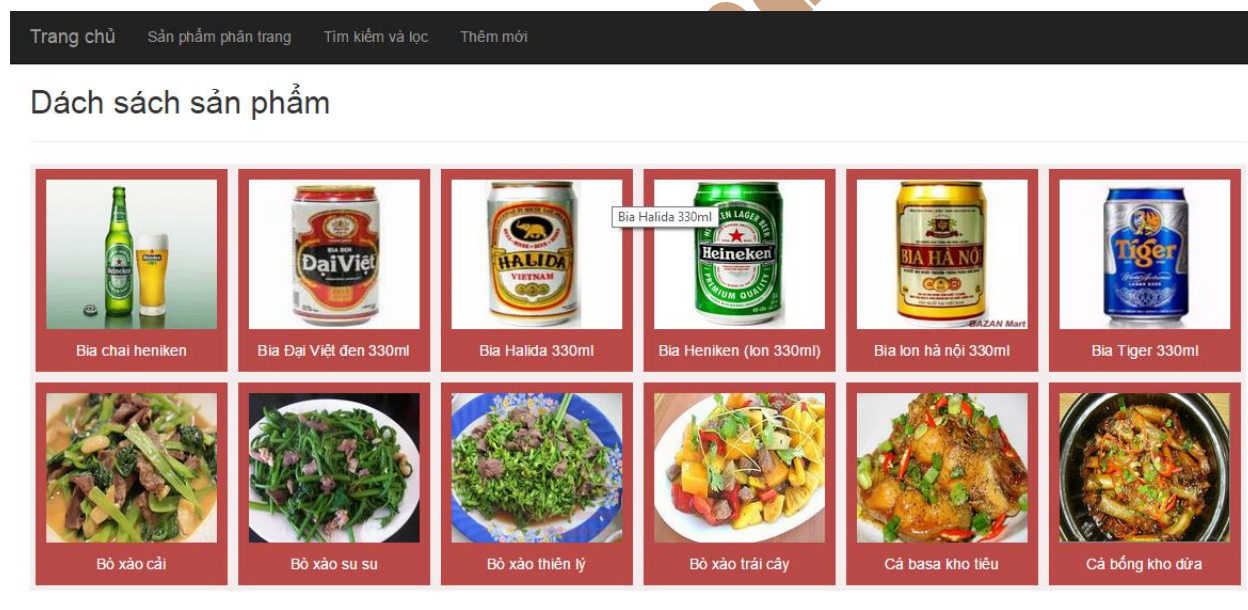
```
<ul class="nav navbar-nav">
    <li>@Html.ActionLink("Sản phẩm phân trang", "ShowProducts",
        "Product")</li>
    <li>@Html.ActionLink("Tìm kiếm và lọc", "Search", "Product")</li>
    <li>@Html.ActionLink("Thêm mới", "Create", "Product")</li>
</ul>
```

Bước 12: Chạy để xem kết quả

- Trang chủ



- Hiển thị sản phẩm có phân trang



Trang chủ Sản phẩm phân trang Tìm kiếm và lọc Thêm mới

Tìm kiếm thực đơn

Chọn chọn loại Thịt Dê Nhập tên dê Tìm



© 2015 - Phòng chuyên môn - Bách Khoa Aptech

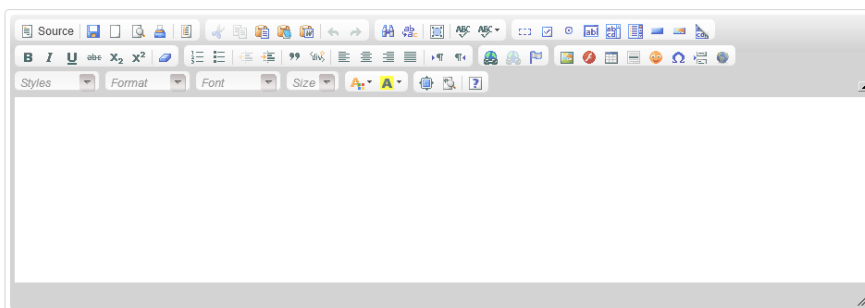
- Trang thêm san phẩm

Trang chủ Sản phẩm phân trang Tìm kiếm và lọc Thêm mới

Thêm sản phẩm mới

Tên sản phẩm
 Giá
 DVT
 Hình ảnh Duyệt ảnh
 Loại
 Món mới ☐
 Hôm nay có ☐
 Trạng thái ☐
 Từ khóa SEO
 Mô tả SEO

Chi tiết



Lưu

[Quay lại](#)


© 2015 - Phòng chuyên môn - Bách Khoa Aptech

Phần II Bài tập tự làm


Bài 1: Tạo ứng dụng ASP.NET MVC cho phép hiển thị các sản phẩm quần áo theo nhóm và phân trang như hình minh họa bên dưới, dữ liệu phân tích và tìm kiếm.

SẢN PHẨM


- ☒ Thời trang nữ
 - Áo Nữ
 - Áo Khoác Nữ
 - Váy Đầm
 - Giày Dép Nữ
 - Phụ Kiện Nữ
- ☐ Thời Trang Nam
 - Áo Nam
 - Áo Khoác Nam
 - Quần Nam




[DShop]
Đầm Xoè Nơ Vai
249,000 đ Mã: 04078




[DShop]
Áo Thun Cổ Trụ
150,000 đ Mã: 04077




[DShop]
Áo Thun Cổ Trụ
150,000 đ Mã: 04077



[DShop]
Đầm Đuối Cá 3 Tầng
239,000 đ Mã: 03993



[DShop]
Chân Váy Xoè Vintage
185,000 đ Mã: 04076



[DShop]
Áo Thun Tay Dài Phối Màu
150,000 đ Mã: 04074

HẾT