

XÂY DỰNG ỨNG DỤNG THEO MÔ HÌNH 3 LỚP

Ứng dụng được xây dựng là một Web site EBook trưng bày các thông tin về ebook do mọi người đưa lên. Hiện tại ứng dụng có hai nhóm chức năng chính gồm: thêm, xóa, sửa cho danh mục thể loại sách và thêm, xóa, sửa cho danh mục thông tin sách có trên Web site EBook. Ứng dụng được xây dựng theo mô hình 3 lớp (Data Layer, Business Logic Layer và Presentation Layer) có sử dụng Entity Framework để hỗ trợ việc đồng bộ dữ liệu giữa các lớp và áp dụng mô hình MVC để thuận tiện cho việc cài đặt các xử lý trong hai lớp Business Layer và Presentation Layer như sau:

I) LỚP DATA LAYER:

Database EBook được cài đặt trên SQLServer Express. Database gồm:

1) Các bảng:

a) Sach. Bảng này có khóa là Id, giá trị của Id tự động tăng khi chèn dòng mới (thêm thuộc tính IDENTITY(1,1) cho trường Id).

Update	Script File:	dbo.Table.sql*			
	Name	Data Type	Allow Nulls	Default	
	Id	int	<input type="checkbox"/>		
	TuaSach	nvarchar(50)	<input checked="" type="checkbox"/>		
	HinhMinhHoa	text	<input checked="" type="checkbox"/>		
	Noidung	nvarchar(MAX)	<input checked="" type="checkbox"/>		
	LoaiID	int	<input type="checkbox"/>		
			<input type="checkbox"/>		

b) LoaiSach. Bảng này có khóa là LoaiID, giá trị của LoaiID tự động tăng khi chèn dòng mới (thêm thuộc tính IDENTITY(1,1) cho trường LoaiID).

Update	Script File:	dbo.Table_1.sql			
	Name	Data Type	Allow Nulls	Default	
	LoaiID	int	<input type="checkbox"/>		
	TenTheLoai	nvarchar(50)	<input checked="" type="checkbox"/>		
			<input type="checkbox"/>		

2) Khóa ngoại:

gồm một khóa ngoại FK_Sach_To_LoaiSach từ bảng Sach đến bảng LoaiSach trên thuộc tính

LoaiID.

3) Triggers:

CapNhatHinhMinhHoa có nhiệm vụ lấy ID của Sach để tạo đường dẫn đến hình minh họa của một quyển sách. Đường dẫn đến hình có dạng: "images/ID.jpg" trong đó ID là mã của quyển sách của file ảnh được người sử dụng upload lên.

```
CREATE TRIGGER [dbo].[CapNhatHinhMinhHoa]
ON [dbo].[Sach]
FOR INSERT
AS
BEGIN
    SET NoCount ON

    DECLARE @SachID NVARCHAR(50)
    DECLARE TMP CURSOR FOR SELECT Id FROM INSERTED
    OPEN TMP;
    FETCH NEXT FROM TMP INTO @SachID;
    WHILE @@FETCH_STATUS = 0
    BEGIN
        UPDATE Sach
        SET HinhMinhHoa = CONCAT('/images/', Id, '.jpg')
        WHERE Id = CAST(@SachID AS INT)
        FETCH NEXT FROM TMP INTO @SachID
    END
END
```

4) Store Procedure:

- LietKeSach trả về tất cả các sách trong đó LoaiID được thay bằng TenTheLoai

```
CREATE PROCEDURE [dbo].[LietKeSach]
```

AS

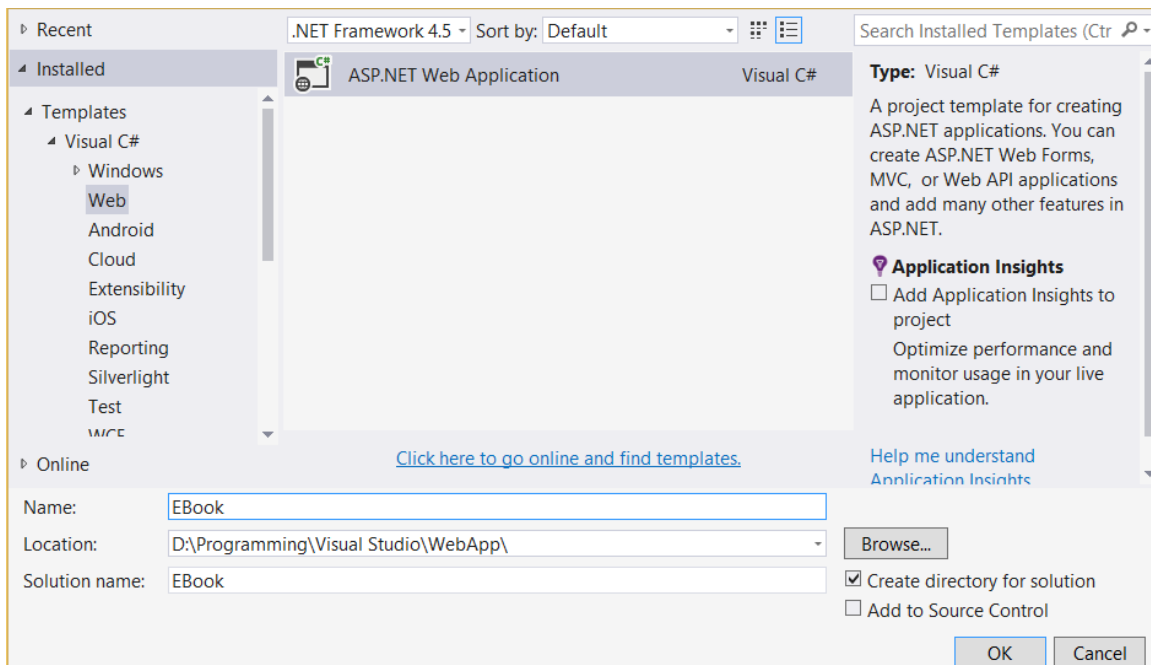
```
SELECT Id, TuaSach, HinhMinhHoa, Noidung, TenTheLoai
FROM Sach INNER JOIN LoaiSach ON Sach.LoaiID = LoaiSach.LoaiID
RETURN 0
```

II) LỚP BUSINESS LOGIC LAYER VÀ PRESENTATION LAYER:

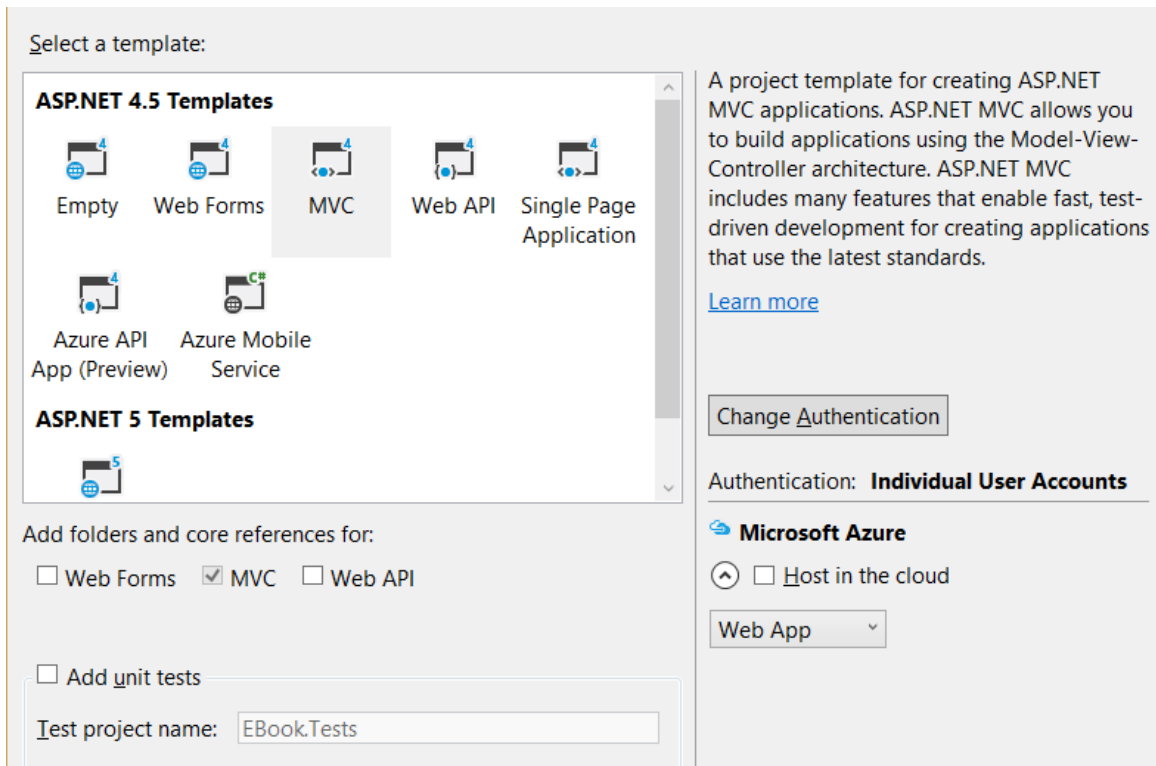
Trong ứng dụng MVC thì Controller được xử lý trên Web Server có vai trò là lớp Business Logic, View được hiển thị và tương tác trên Web Browser (trình duyệt) có vai trò là lớp Presentation. Khi cài đặt ứng dụng theo MVC sẽ thực hiện cùng lúc hai lớp trong mô hình 3 lớp này.

1) Cài đặt EntityFramework:

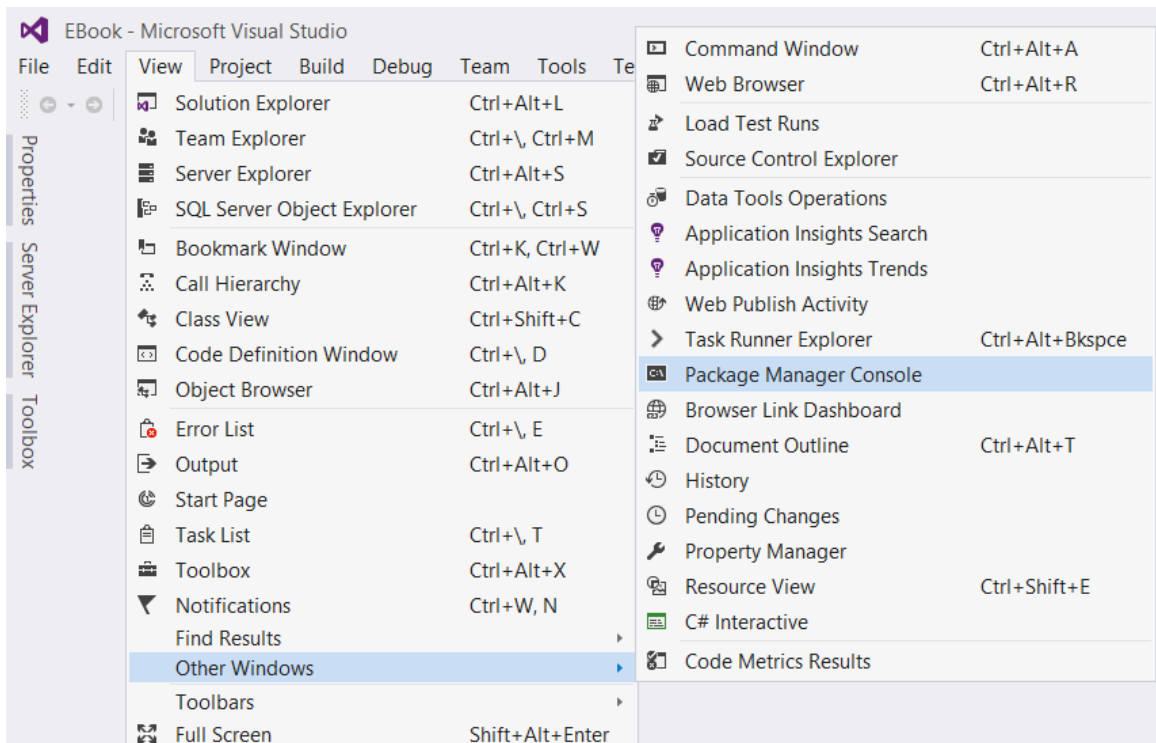
Tạo Project ASP.NET Web Application với .NET Framework 4.5, đặt tên là EBook.



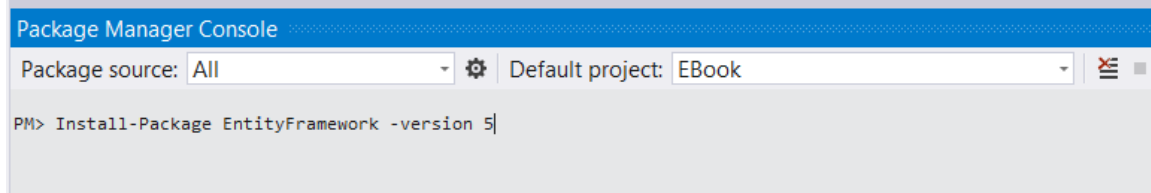
Chọn ASP.NET 4.5 Template là MVC, chọn Change Authentication là No Authentication



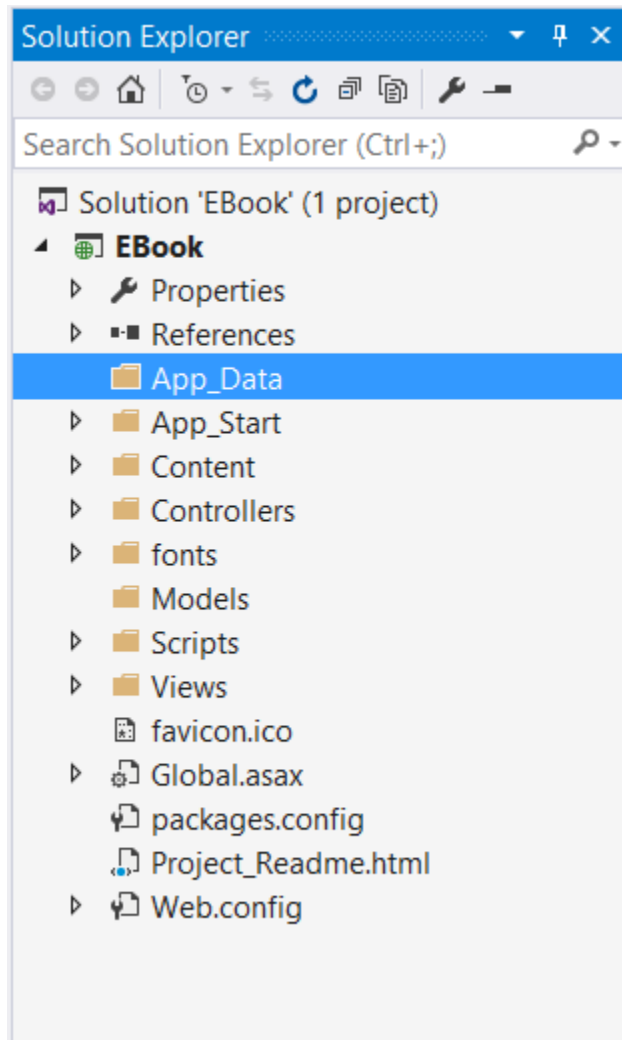
Cài đặt Entity Framework cho Project: Mở Package Manage Console



Gõ Install-Package EntityFramework -version 5

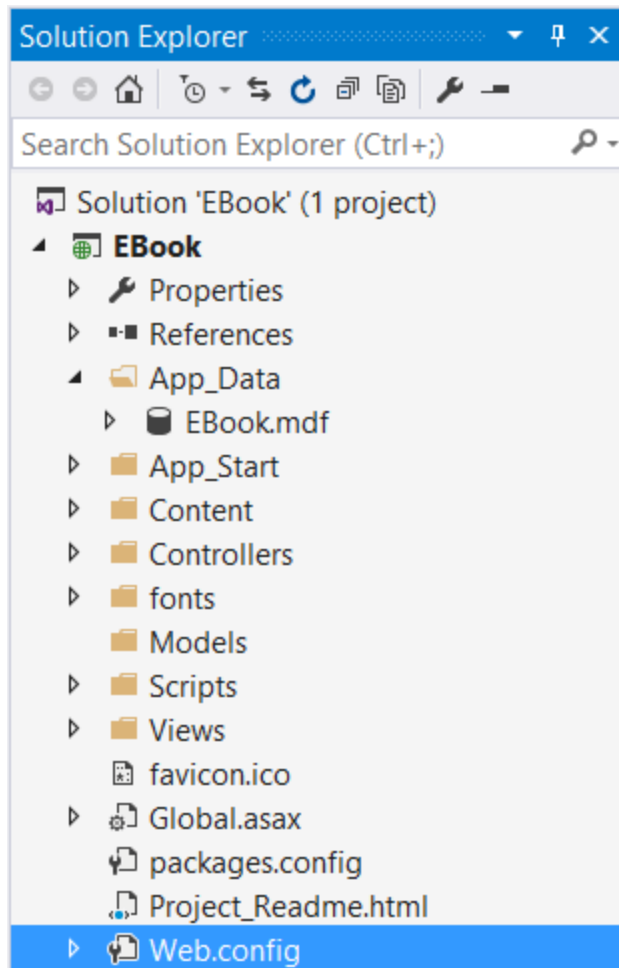


2) Đưa CSDL EBook vào Project:



Click phải vào App_Data, chọn Add > Existing Item sau đó chọn file EBook.mdf.

Thêm kết nối CSDL vào Project: Double click vào file Web.config



Thêm chuỗi kết nối vào như sau:

```
<connectionStrings>
```

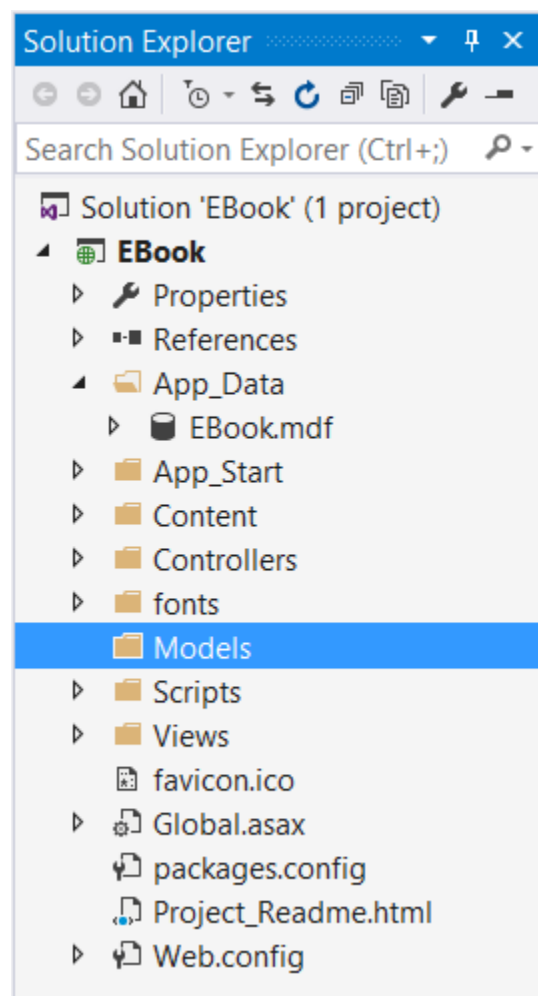
```
  <add name="EBookDB" connectionString="Data Source=(localdb)\MSSQLLocalDB; Initial  
Catalog=EBook; Integrated Security=True; MultipleActiveResultSets=True;  
AttachDbFilename=|DataDirectory|EBook.mdf" providerName="System.Data.SqlClient" />
```

```
</connectionStrings>
```

```
3  For more information on how to configure your ASP.NET application,
4  http://go.microsoft.com/fwlink/?LinkId=301880
5  -->
6  <configuration>
7  <configSections>
8      <!-- For more information on Entity Framework configuration, visit
9      <section name="entityFramework" type="System.Data.Entity.Internal
10     <!-- For more information on Entity Framework configuration, visit
11     </configSections>
12  <connectionStrings>
13     <add name="EBookDB" connectionString="Data Source=(localdb)\MSSQ
14  </connectionStrings>
15  <appSettings>
16     <add key="webpages:Version" value="3.0.0.0" />
17     <add key="webpages:Enabled" value="false" />
18     <add key="ClientValidationEnabled" value="true" />
```

3) Xây dựng các lớp Model để đồng bộ dữ liệu giữa các lớp:

Tạo các Model tương ứng với các bảng dữ liệu để đồng bộ dữ liệu từ Business Logic Layer với Data Layer



Click phải vào Models, chọn Add > Class, đặt tên lớp là LoaiSachModel tương ứng với bảng

LoaiSach. Các thuộc tính trong LoaiSachModel tương ứng với các trường dữ liệu trong bảng LoaiSach như sau:

```
[Table("LoaiSach")]

public class LoaiSachModel
{
    [Key]

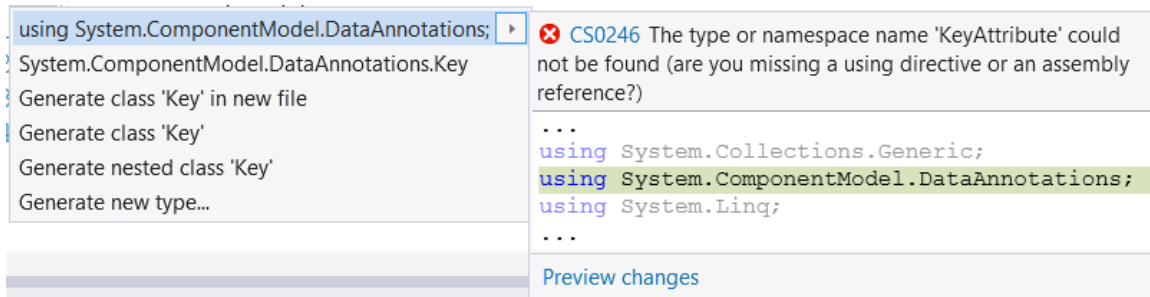
    public int LoaiID { get; set; }

    public String TenTheLoai { get; set; }
}
```

Đặt con trỏ tại vị trí [Key] Nhấn Ctrl+. và chọn

`using System.ComponentModel.DataAnnotations;`

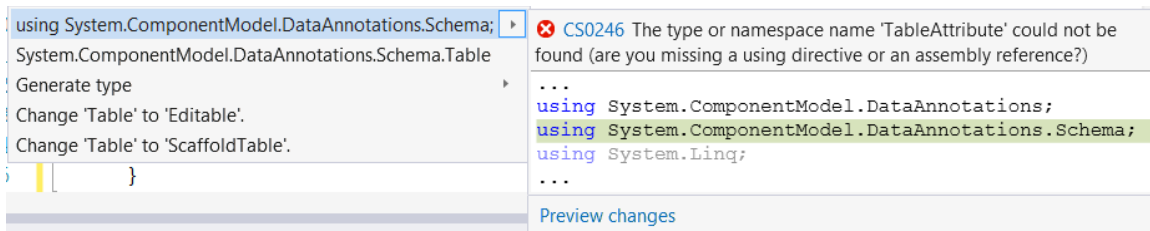
để import các lớp cần thiết.



Đặt con trỏ tại vị trí [Table("LoaiSach")] Nhấn Ctrl+. và chọn

`using System.ComponentModel.DataAnnotations.Schema;`

để import các lớp cần thiết.



Tạo lớp SachModel tương tự như trên với các thuộc tính như sau:

```
[Table("Sach")]
```



```

public class SachModel
{
    [Key]
    public int Id { get; set; }
    public String TuaSach { get; set; }
    public String HinhMinhHoa { get; set; }
    public String Noidung { get; set; }
    public int LoaiID { get; set; }
}

```

Tạo các Model để đồng bộ dữ liệu từ Business Logic Layer với Presentation Layer. Trong MVC, Business Logic Layer tương ứng với các Controller và Presentation Layer tương ứng với các View.

Tạo lớp SachViewModel tương tự với các thuộc tính sau:

```

public class SachViewModel
{
    public int Id { get; set; }
    public String TuaSach { get; set; }
    public String HinhMinhHoa { get; set; }
    public HttpPostedFile ImgFile { get; set; }
    public String Noidung { get; set; }
    public int LoaiID { get; set; }
    public String TenTheLoai { get; set; }
    public List<LoaiSachViewModel> DanhMuc { get; set; }
}

```

Trong đó có thêm thuộc tính DanhMuc để truyền danh sách các loại sách để chọn trong giao diện thêm mới sách, thuộc tính TenTheLoai để hiển thị tên thể loại thay cho mã số và thuộc tính ImgFile để upload file ảnh.

Tạo lớp EBookDB trong Models để gắn kết giữa CSDL với các Model tương tự như trên. Lớp

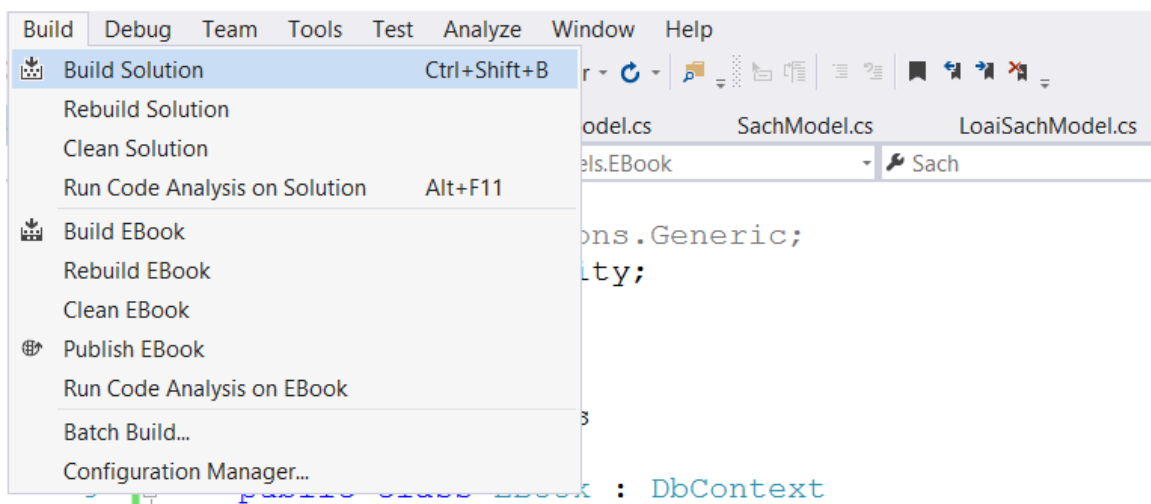
EBookDB được khai báo như sau:

```
using System.Data.Entity;
```

Thêm hàm OnModelCreating để có kết quả:

```
public class EBookDB : DbContext
{
    public DbSet<LoaiSachModel> LoaiSach { get; set; }
    public DbSet<SachModel> Sach { get; set; }
    protected override void OnModelCreating(DbModelBuilder modelBuilder)
    {
        modelBuilder.Entity<LoaiSachModel>().ToTable("LoaiSach");
        modelBuilder.Entity<SachModel>().ToTable("Sach");
    }
}
```

Trước khi cài đặt các xử lý, cần Build Project để VS cài đặt các lớp cần thiết cho kết nối CSDL. Không có bước Build này sẽ không thể thực hiện việc thêm tự động các Controller.

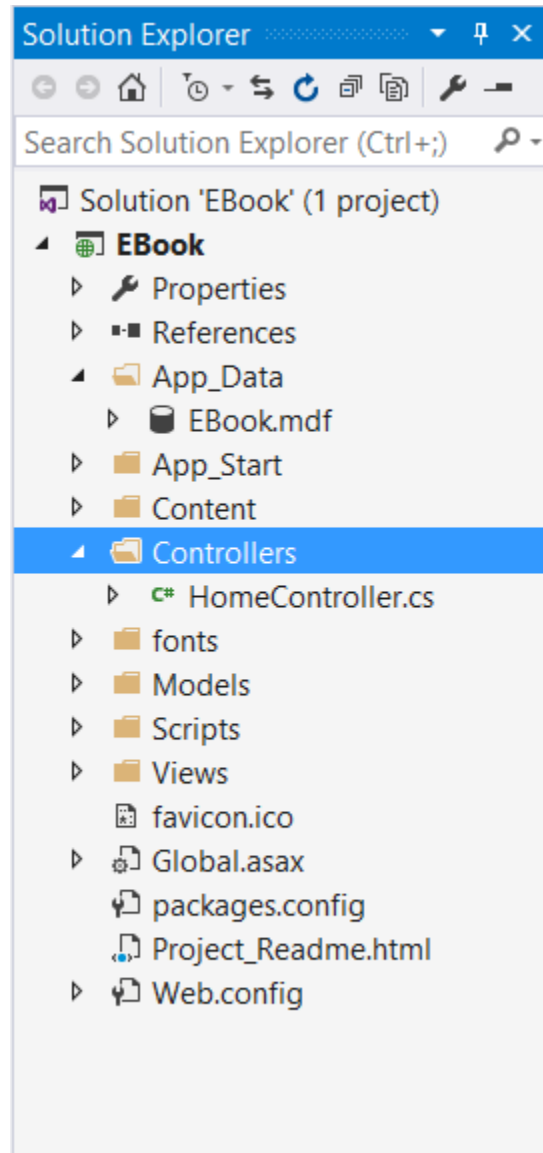


4) Xây dựng các lớp xử lý (Controller) và các trang giao diện (View)

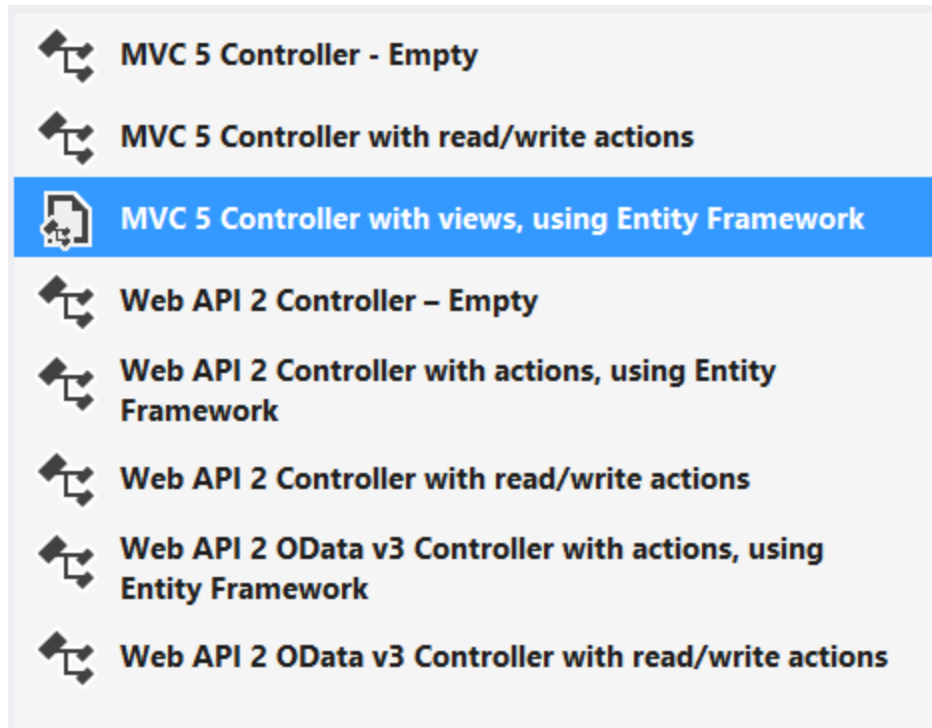
a) Tạo Controller *LoaiSachController*

Tạo Controller để Thêm, Sửa, Xóa dữ liệu trong bảng LoaiSach:

Click phải vào Controllers



Chọn Add > Controller



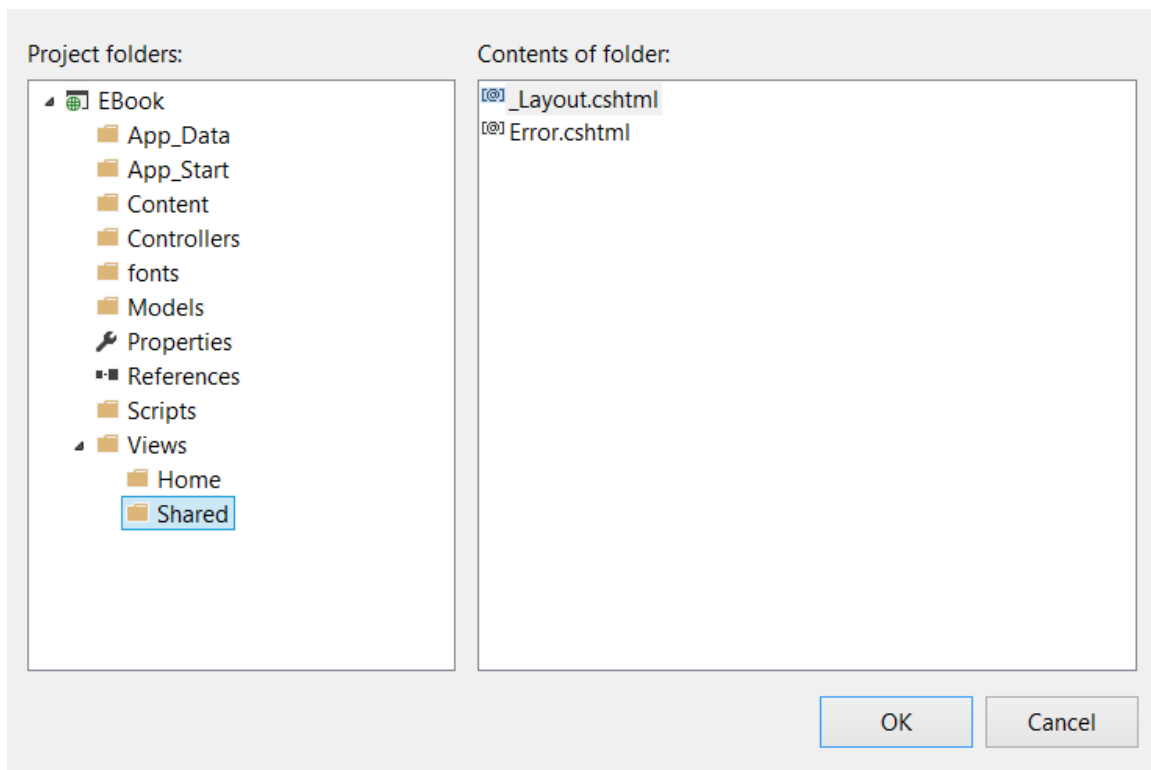
Chọn MVC 5 Controller with views, using Entity Framework

A screenshot of the 'Add MVC Controller' dialog box in Visual Studio. The dialog contains the following fields and options:

- Model class:** A dropdown menu showing 'LoaiSachModel (EBook.Models)'.
- Data context class:** A dropdown menu showing 'EBookDB (EBook.Models)' with a '+' button to the right.
- ☐ Use async controller actions
- Views:**
 - ☒ Generate views
 - ☒ Reference script libraries
 - ☒ Use a layout page:
 - A text box for the layout page name.
 - A button with three dots '...' to the right of the text box.
- (Leave empty if it is set in a Razor _viewstart file)
- Controller name:** A text box containing 'LoaiSachModelsController'.
- At the bottom right, there are two buttons: 'Add' and 'Cancel'.

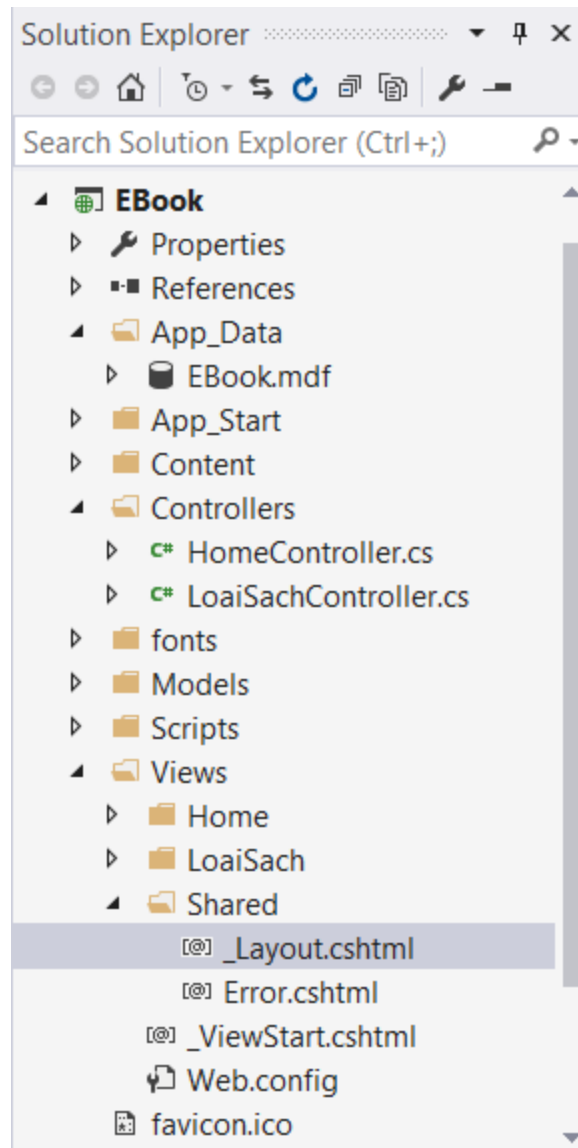
Chọn Model class là LoaiSachModel, Data context class là EBookDB

Click vào nút chọn Use a layout page, chọn _Layout.cshtml



Đặt tên Controller là LoaiSachController (VS tự động đặt là LoaiSachModelsController)

Vào trang _Layout.cshtml



Sửa dòng:

```
@Html.ActionLink("Application name", "Index", "Home", new { area = "" }, new { @class = "navbar-brand" })
```

Thành

```
@Html.ActionLink("Thông tin EBook", "Index", "Home", new { area = "" }, new { @class = "navbar-brand" })
```

Sửa:

```
<div class="navbar-collapse collapse">
  <ul class="nav navbar-nav">
    <li>@Html.ActionLink("Home", "Index", "Home")</li>
    <li>@Html.ActionLink("About", "About", "Home")</li>
    <li>@Html.ActionLink("Contact", "Contact", "Home")</li>
```

```
</ul>
</div>
```

Thành

```
<div class="navbar-collapse collapse">
  <ul class="nav navbar-nav">
    <li>@Html.ActionLink("Trang Chủ", "Index", "Home")</li>
    <li>@Html.ActionLink("Thẻ loại sách", "Index", "LoaiSach")</li>
    <li>@Html.ActionLink("Giới thiệu", "About", "Home")</li>
    <li>@Html.ActionLink("Liên hệ", "Contact", "Home")</li>
  </ul>
</div>
```

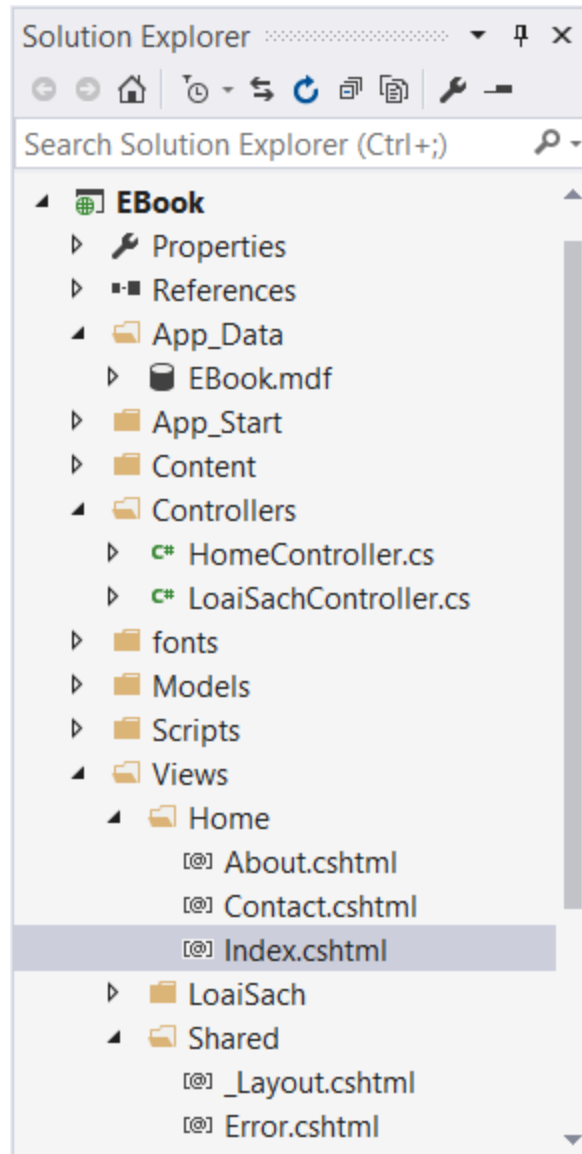
Sửa

```
<footer>
  <p>&copy; @DateTime.Now.Year - My ASP.NET Application</p>
</footer>
```

Thành

```
<footer>
  <p>&copy; @DateTime.Now.Year - Demo Xây dựng hệ thống theo mô hình 3 lớp</p>
</footer>
```

Vào Trang Home > Index.cshtml



Xóa nội dung:

```
<div class="row">
```

```
  <div class="col-md-4">
```

```
    <h2>Getting started</h2>
```

```
    <p>
```

ASP.NET MVC gives you a powerful, patterns-based way to build dynamic websites that enables a clean separation of concerns and gives you full control over markup for enjoyable, agile development.

```
  </p>
```

```
  <p><a class="btn btn-default" href="http://go.microsoft.com/fwlink/?LinkId=301865">Learn more &raquo;</a></p>
```


</div>

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

<h2>Get more libraries</h2>

<p>NuGet is a free Visual Studio extension that makes it easy to add, remove, and update libraries and tools in Visual Studio projects.</p>

<p>Learn more »</p>

</div>

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

<h2>Web Hosting</h2>

<p>You can easily find a web hosting company that offers the right mix of features and price for your applications.</p>

<p>Learn more »</p>

</div>

</div>

Sửa

<div class="jumbotron">

<h1>ASP.NET</h1>

<p class="lead">ASP.NET is a free web framework for building great Web sites and Web applications using HTML, CSS and JavaScript.</p>

<p>Learn more »</p>

</div>

Thành

<div class="jumbotron">

<h1>Thông tin EBook</h1>

<p class="lead">Thông tin EBook là trang web giới thiệu các sách điện tử của mỗi người. Bất kỳ ai cũng có thể giới thiệu sách điện tử mà mình đã đọc

Mọi người cùng chia sẻ để cùng mở rộng tầm hiểu biết chung của chúng ta.</p>

</div>

Mở trang Home > About.cshtml tương tự như trang Index.cshtml

Sửa

<p>Use this area to provide additional information.</p>

Thành

<p>Nhóm thực hiện:

1. Thành viên thứ nhất

2. Thành viên thứ hai

3. Thành viên thứ ba

</p>

Mở trang Home > Contact.cshtml tương tự như trang Index.cshtml

Sửa

<address>

One Microsoft Way

Redmond, WA 98052-6399

<abbr title="Phone">P:</abbr>

425.555.0100

</address>

Thành

<address>

Khoa của sinh viên

Trường đại học Công nghệ thông tin

<abbr title="Phone">P:</abbr>

425.555.0100

</address>

Sửa

<address>

Support: Support@example.com

Marketing: Marketing@example.com

</address>

Thành

<address>

Liên hệ: email@uit.edu.vn

</address>

Sửa các thông báo trong lớp HomeController trong Controller

Sửa

Your application description page.

Thành

Đồ án môn học Nhập môn công nghệ phần mềm

Sửa

Your contact page.

Thành

Thông tin liên hệ

Chạy thử: Nhấn CTR+F5.

b) Tạo Controller SachController

Tạo Controller tên SachController để xử lý việc thêm, sửa, xóa dữ liệu trên bảng Sach tương tự như tạo LoaiSachController

Mở SachController,

Sửa hàm xử lý Index trong SachController:

```
public ActionResult Index()
{
    return View(db.Sach.ToList());
}
```

Thành

```
public ActionResult Index()
{
    List<LoaiSachModel> loaisach = db.LoaiSach.ToList<LoaiSachModel>();
    SachViewModel model = new SachViewModel();
    model.DanhMuc = loaisach;
    return View(model);
}
```

Mở file Views > Sach > Index.cshtml, sửa

```
<td>
```

```
@Html.DisplayFor(modelItem => item.HinhMinhHoa)

</td>
```

Thành

```
<td>



</td>
```

Sửa

```
<td>

@Html.DisplayFor(modelItem => item.LoaiID)

</td>
```

Thành

```
<td>

@Html.DisplayFor(modelItem => item.TenTheLoai)

</td>
```

Sửa hàm xử lý Create trong SachController

```
public ActionResult Create()

{

    return View();

}
```

Thành

```
public ActionResult Create()

{

    List<LoaiSachModel> loaisach = db.LoaiSach.ToList<LoaiSachModel>();

    SachViewModel model = new SachViewModel();

    model.DanhMuc = loaisach;

    return View(model);

}
```

Sửa hàm xử lý Create trong SachController

```
public ActionResult Create([Bind(Include = "Id,TuaSach,HinhMinhHoa,Noidung,LoaiID")] SachModel sachModel)

{
```

```

        if (ModelState.IsValid)
        {
            db.Sach.Add(sachModel);
            db.SaveChanges();

            return RedirectToAction("Index");
        }

        return View(sachModel);
    }
}

```

Thành

```

public ActionResult Create([Bind(Include = "TuaSach,ImgFile,Noidung,LoaiID")] SachViewModel sachModel)
{
    SachModel sach = new Models.SachModel();
    sach.LoaiID = sachModel.LoaiID;
    sach.TuaSach = sachModel.TuaSach;
    sach.Noidung = sachModel.Noidung;
    if (ModelState.IsValid)
    {
        db.Sach.Add(sach);
        db.SaveChanges();
        if (sachModel.ImgFile != null)
        {
            sachModel.ImgFile.SaveAs(Server.MapPath("/images") + "/" + sach.Id + ".jpg");
            return RedirectToAction("Index");
        }
        sachModel.DanhMuc = db.LoaiSach.ToList<LoaiSachModel>();
        return View(sachModel);
    }
}

```

Mở file Create.cshtml trong Views > Sach

Sửa

@model EBook.Models.SachModel

Thành

```
@using EBook.Models
```

```
@model EBook.Models.SachViewModel
```

Sửa

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

Thành

```
@using (Html.BeginForm("Create", "Sach", FormMethod.Post, new { enctype = "multipart/form-data", @class = "form-horizontal", role = "form" }))
```

Sửa

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

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

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

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

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

    </div>

</div>
```

Thành

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

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

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

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

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

    </div>

</div>
```

Sửa

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

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

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

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

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

    </div>

</div>
```

```
</div>
</div>
```

Thành

```
<div class="form-group">
    @Html.LabelFor(model => model.LoaiID, htmlAttributes: new { @class = "control-label col-md-2" })
    <div class="col-md-10">
        @Html.DropDownListFor(model => model.LoaiID, new SelectList(Model.DanhMuc,"LoaiID", "TenTheLoai", -1),
        "Chọn Thể Loại", new { @class = "form-control" })
        @Html.ValidationMessageFor(model => model.LoaiID, "", new { @class = "text-danger" })
    </div>
</div>
```

Mở file Delete.cshtml trong View > Sach

Sửa

```
<dd>
    @Html.DisplayFor(model => model.HinhMinhHoa)
</dd>
```

Thành

```
<dd>
    
</dd>
```

Sửa hàm xử lý Edit trong SachController

Thành

```
public ActionResult Edit(int? id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    SachModel sach = db.Sach.Find(id);
    if (sach == null)
```

```

{
    return HttpNotFound();
}

SachViewModel sachModel = new SachViewModel();

sachModel.Id = sach.Id;

sachModel.TuaSach = sach.TuaSach;

sachModel.HinhMinhHoa = sach.HinhMinhHoa;

sachModel.Noidung = sach.Noidung;

sachModel.LoaiID = sach.LoaiID;

sachModel.DanhMuc = db.LoaiSach.ToList<LoaiSachModel>();

return View(sachModel);
}

```

Và

```

public ActionResult Edit([Bind(Include = "Id,TuaSach,ImgFile,HinhMinhHoa,Noidung,LoaiID")] SachViewModel sachModel)
{
    SachModel sach = new Models.SachModel();

    sach.Id = sachModel.Id;

    sach.TuaSach = sachModel.TuaSach;

    sach.HinhMinhHoa = sachModel.HinhMinhHoa;

    sach.Noidung = sachModel.Noidung;

    sach.LoaiID = sachModel.LoaiID;

    if (ModelState.IsValid)
    {
        db.Entry(sach).State = EntityState.Modified;

        db.SaveChanges();

        if (sachModel.ImgFile != null)
        {
            sachModel.ImgFile.SaveAs(Server.MapPath("/images") + "/" + sach.Id + ".jpg");

            return RedirectToAction("Index");
        }

        sachModel.DanhMuc = db.LoaiSach.ToList<LoaiSachModel>();

        return View(sachModel);
    }
}

```


Mở file Edit.cshtml trong View > Sach

Sửa

```
@model EBook.Models.SachModel
```

Thành

```
@using EBook.Models
```

```
@model EBook.Models.SachViewModel
```

Sửa

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

Thành

```
@using (Html.BeginForm("Edit", "Sach", FormMethod.Post, new { enctype = "multipart/form-data", @class = "form-horizontal", role = "form" }))
```

Sửa

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

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

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

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

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

    </div>

</div>
```

Thành

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

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

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

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

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

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

    </div>

</div>
```

Sửa

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

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

```

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

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

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

</div>

</div>

```

Thành

```

<div class="form-group">

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

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

        @Html.DropDownListFor(model => model.LoaiID, new SelectList(Model.DanhMuc,"LoaiID", "TenTheLoai", -1),
        "Chọn Thể Loại", new { @class = "form-control" })

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

    </div>

</div>

```

Mở file _Layout.cshtml trong View > Share

Sửa

```

<div class="navbar-collapse collapse">

    <ul class="nav navbar-nav">

        <li>@Html.ActionLink("Trang Chủ", "Index", "Home")</li>

        <li>@Html.ActionLink("Thể loại sách", "Index", "LoaiSach")</li>

        <li>@Html.ActionLink("Giới thiệu", "About", "Home")</li>

        <li>@Html.ActionLink("Liên hệ", "Contact", "Home")</li>

    </ul>

</div>

```

Thành

```

<div class="navbar-collapse collapse">

    <ul class="nav navbar-nav">

        <li>@Html.ActionLink("Trang Chủ", "Index", "Home")</li>

        <li>@Html.ActionLink("Thể loại sách", "Index", "LoaiSach")</li>

        <li>@Html.ActionLink("Danh mục sách", "Index", "Sach")</li>

        <li>@Html.ActionLink("Giới thiệu", "About", "Home")</li>

    </ul>

</div>

```

```
</li>@Html.ActionLink("Liên hệ", "Contact", "Home")</li>
</ul>
</div>
```

Chạy thử bằng cách nhấn Ctrl+.

Phần chỉnh sửa hàm xử lý Detail và việc thêm các thao tác tìm kiếm là bài tập cho phần thực hành này.