

Lab04 - Model in ASP.NET MVC

Mục tiêu:

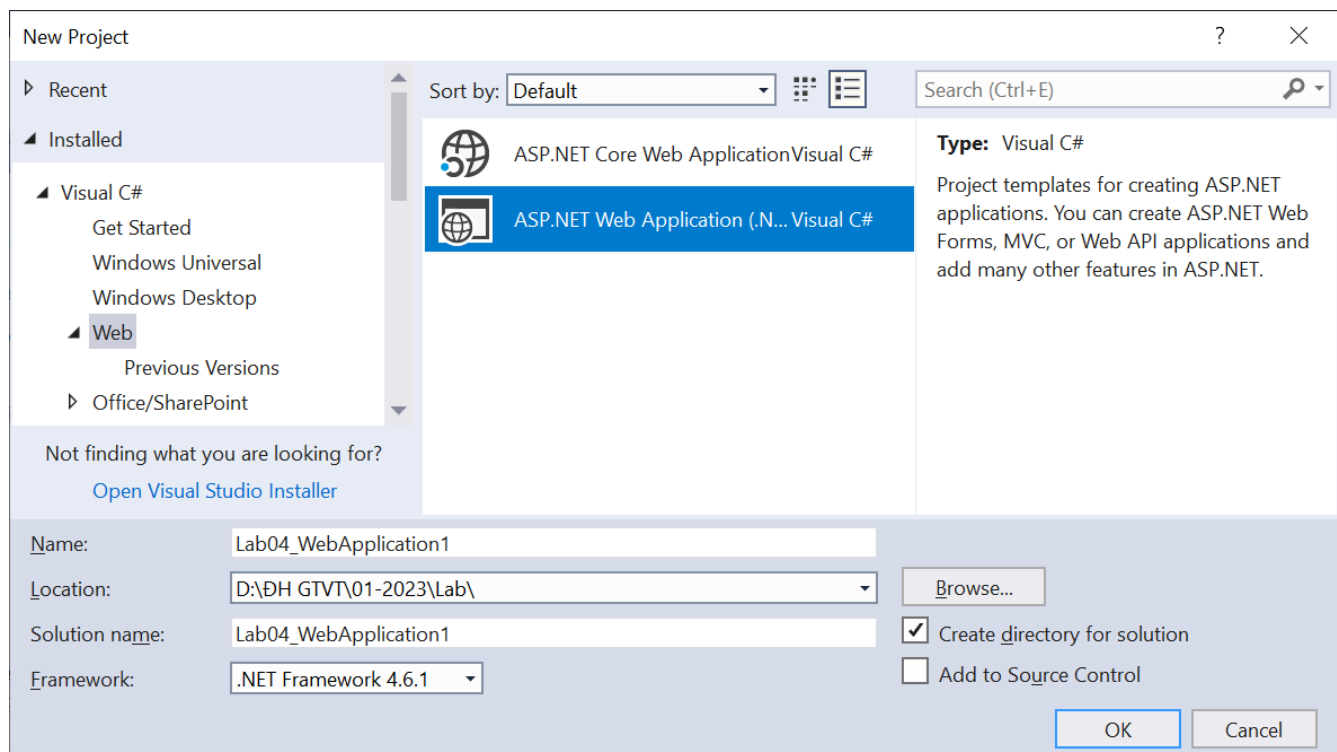
1. Biết cách tạo Model và tương tác trong MVC
2. Biết cách sinh view từ Model

Bài 1: Xây dựng ứng dụng ASP.NET MVC5 thực hiện những công việc sau:

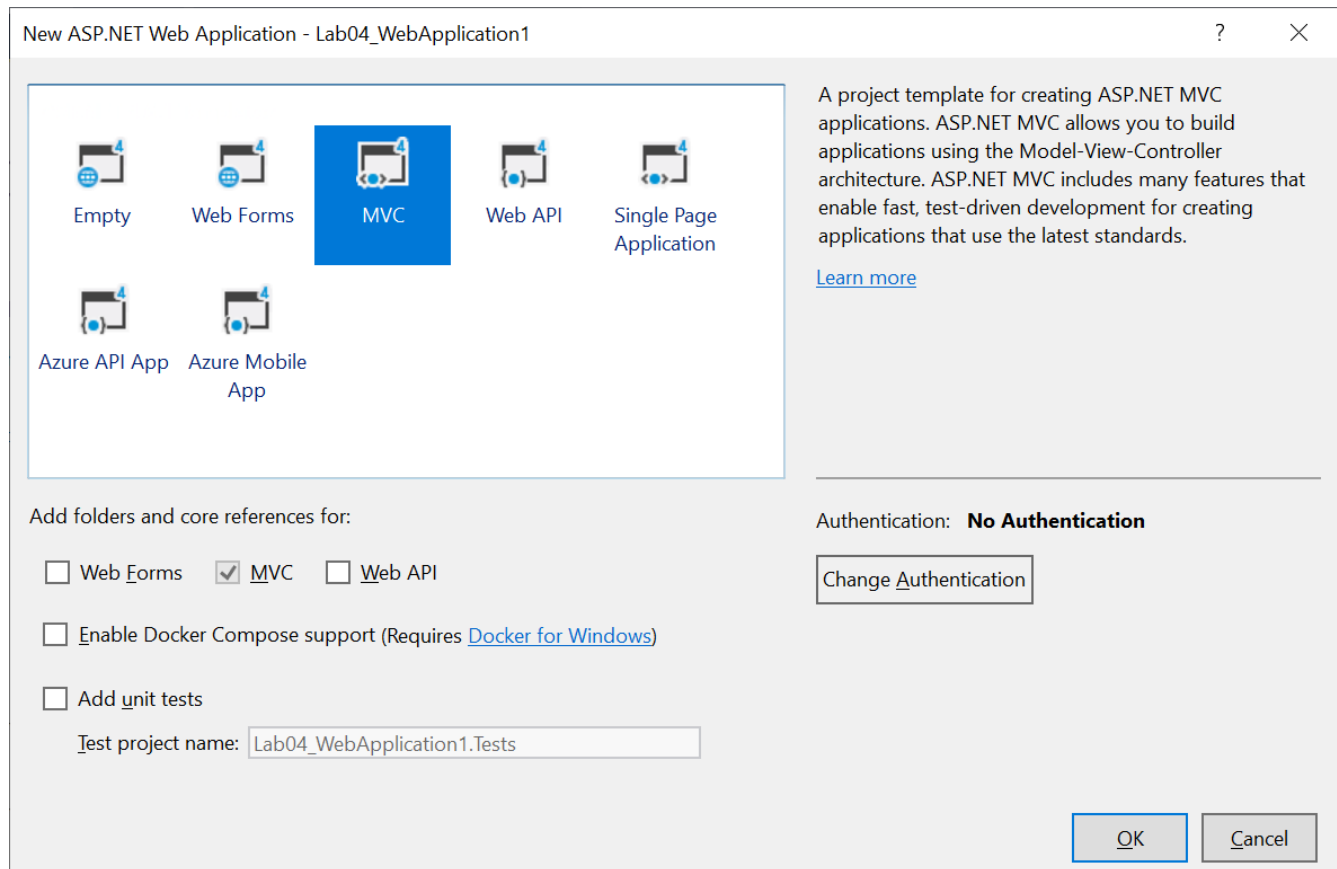
- **Hiển thị thông tin về đối tượng Customer, thông tin bao gồm: CustomerId, FullName, Address, Phone, Email, Balance (sử dụng ViewBag, Model...)**
- **Hiển thị 1 danh sách các Customer.**

Bước 1 : Mở Visual Studio 2017/2019 tạo project

Vào File -> New -> Projects -> Chọn Web -> ASP.NET Web Application -> Nhập tên project -> Chọn đường dẫn lưu trữ -> Nhập tên solution -> Nhấn OK -> Change Authentication -> No Authentication -> OK -> OK.



Nhấn OK -> màn hình tiếp theo xuất hiện bạn chọn template là MVC sau đó chọn Change Authentication -> No Authentication -> OK -> OK (Xem hình bên dưới)



Bước 2: Kích chuột phải vào thư mục **Models** chọn **Add -> Class** -> Nhập tên “**Customer**” -> Add và code như sau:

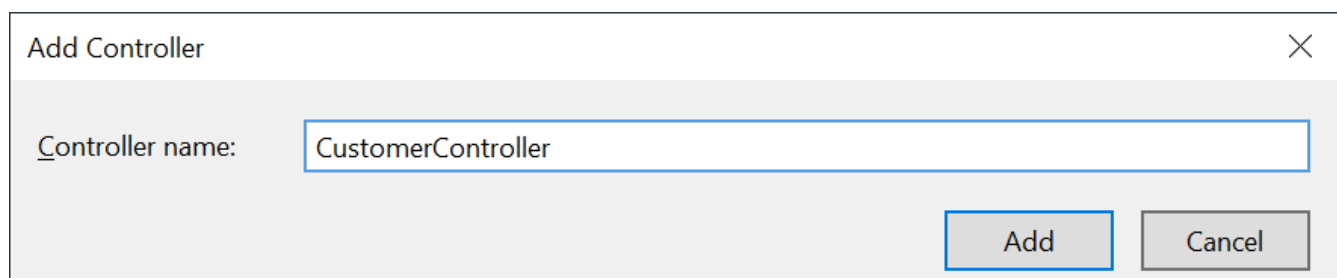
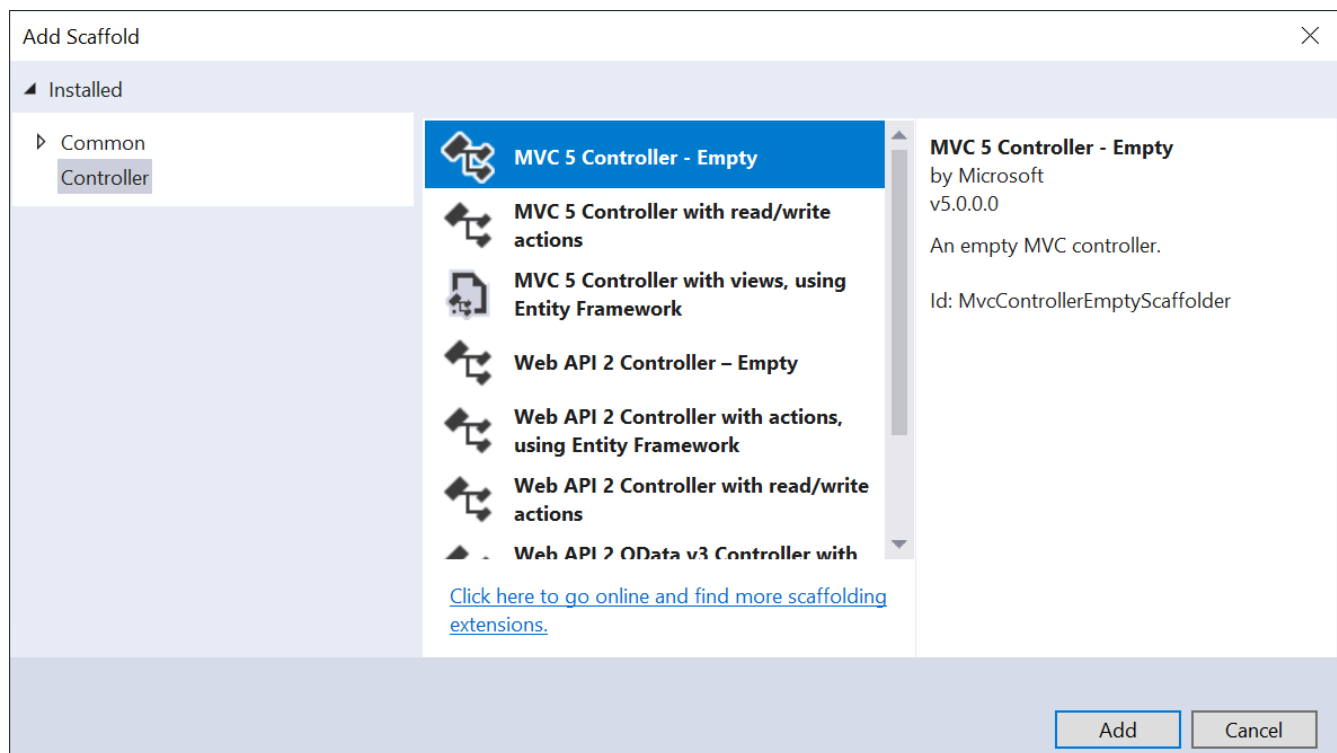
```
namespace Lab04_WebApplication1.Models
{
    public class Customer
    {
        public string CustomerId { get; set; }
        public string FullName { get; set; }
        public string Address { get; set; }
        public string Email { get; set; }
        public string Phone { get; set; }
        public int Balance { get; set; }
    }
}
```

Bước 3: Mở trang **_Layout.cshtml** trong đường dẫn **/View/Shared/_Layout.cshtml** và tìm đoạn code sau và thêm vào đúng vị trí:

```
<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>

        <li>@Html.ActionLink("Khách hàng", "CustomerDetail", "Customer")</li>
        <li>@Html.ActionLink("Danh sách khách hàng", "CustomerList", "Customer")</li>
    </ul>
</div>
```

Bước 4: Kích chuột phải vào thư mục Controllers chọn Add -> Controller -> Nhập tên controller -> Add



Bước 5: Code cho CustomerController như sau:

```

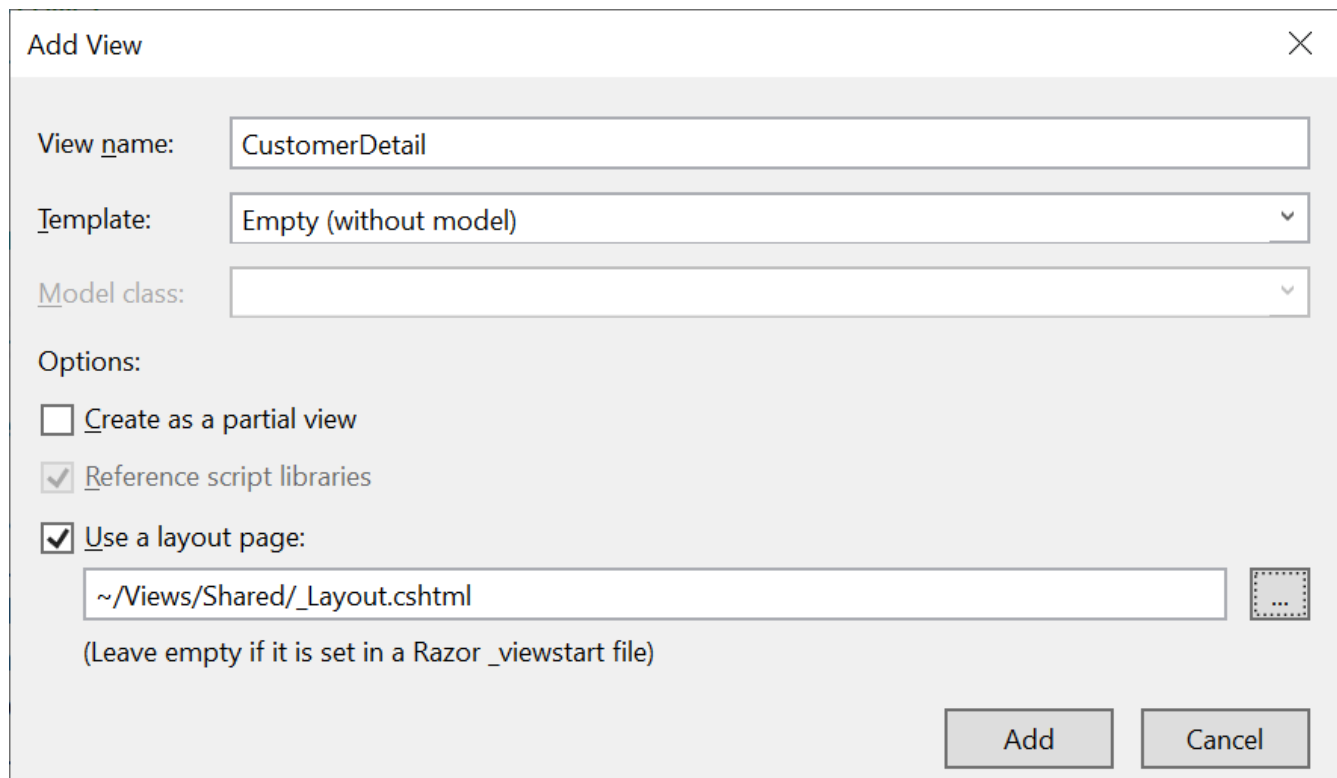
7  | using Lab04_WebApplication1.Models;

// GET: /Customer/CustomerDetail action trả về thông tin
//chi tiết 1 khách hàng cho view CustomerDetail
public ActionResult CustomerDetail()
{
    Customer cus = new Customer()
    {
        CustomerId = "KH001",
        FullName = "Trịnh Văn Chung",
        Address = "Tp.HCM",
        Email = "dungta.uit@gmail.com",
        Phone = "0988.611.889",
        Balance = 15200000
    };
    //cách 1 gán dữ liệu vào ViewBag để chuyển tới View
    //ViewBag.customer = cus;
    //return View();
    //có thể truyền dữ liệu qua đối tượng model để chuyển tới View
    return View(cus);
}

//GET /Product/CustomerList (action trả về danh sách khách hàng cho view CustomerList
public ActionResult CustomerList()
{
    //tạo một danh sách khách hàng
    List<Customer> listcustomer = new List<Customer>(){
        new Customer(){ CustomerId = "KH001",FullName="Trịnh Văn Chung",
            Address = "Tp.HCM", Email = "dungta.uit@gmail.com",
            Phone = "0988.611.889", Balance = 15200000},
        new Customer(){ CustomerId = "KH002", FullName = "Đỗ Thị Cúc",
            Address = "Hà Nội",Email = "cucdt@gmail.com",
            Phone = "0988.767.444",Balance = 22000000},
        new Customer(){ CustomerId="KH003", FullName="Nguyễn Tuấn Thắng",
            Address = "Nam Định",Email = "thangnt@gmail.com",
            Phone = "0924.656.542",Balance = 12000000},
        new Customer(){ CustomerId = "KH004", FullName = "Lê Ngọc Hải",
            Address = "Hà Nội",Email = "hailn@gmail.com",
            Phone = "0996.555.267",Balance = 6200000 }
    };
    //gán dữ liệu vào ViewBag để chuyển qua View
    ViewBag.listcustomer = listcustomer;
    return View();
}

```

Bước 6: Kích chuột phải vào tên phương thức **CustomerDetail** chọn **Add View -> nhấn Add**



Add View

View name: CustomerDetail

Template: Empty (without model)

Model class:

Options:

☐ Create as a partial view

☒ Reference script libraries

☒ Use a layout page:

~/Views/Shared/_Layout.cshtml

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

Add Cancel

Bước 7: Code cho View **CustomerDetail (CustomerDetail.cshtml)** như sau:

```
@model Lab04_WebApplication1.Models.Customer
@{
    ViewBag.Title = "CustomerDetail";
    Layout = "~/Views/Shared/_Layout.cshtml";
    //lấy dữ liệu truyền qua đối tượng theo cách 2
    //var customer = Model as Lab04_WebApplication1.Models.Customer;
}
<h2>Chi tiết khách hàng lấy theo cách 1 </h2>
<p>Mã số: @Model.CustomerId </p>
<p>Họ và tên: @Model.FullName </p>
<p>Địa chỉ: @Model.Address </p>
<p>Điện thoại: @Model.Phone </p>
<p>Email: @Model.Email </p>
<p>Số tiền dư: @Model.Balance </p>
```

@*

Chi tiết khách hàng lấy theo cách 1

Mã số: @customer.CustomerId

Họ và tên: @customer.FullName

Địa chỉ: @customer.Address

Điện thoại: @customer.Phone

Email: @customer.Email

Số tiền dư: @customer.Balance

*@

@*

Chi tiết khách hàng lấy qua ViewBag

Mã số: @ViewBag.customer.CustomerId

Họ và tên: @ViewBag.customer.FullName

Địa chỉ: @ViewBag.customer.Address

Điện thoại: @ViewBag.customer.Phone

Email: @ViewBag.customer.Email

Số tiền dư: @ViewBag.customer.Balance

*@

Bước 8: Mở **CustomerController** kích chuột phải vào tên phương thức **CustomerList** chọn Add View
-> nhấn Add

Add View

View name: CustomerList

Template: Empty (without model)

Model class:

Options:

☐ Create as a partial view

☐ Reference script libraries

☒ Use a layout page:

~/Views/Shared/_Layout.cshtml

...

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

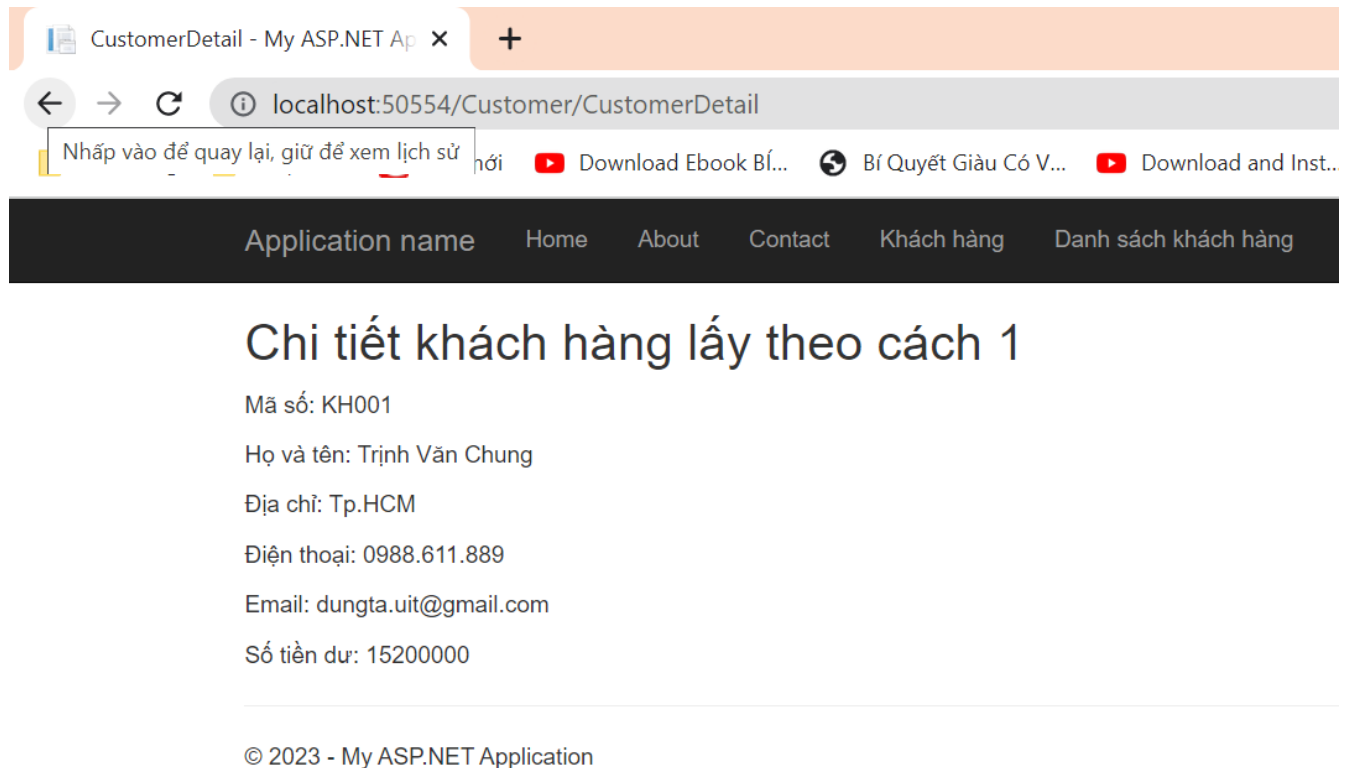
Add

Cancel

Bước 9: Code cho View **CustomerList (CustomerList.cshtml)** như sau:

```
@model Lab04_WebApplication1.Models.Customer
@{
    ViewBag.Title = "Danh sách khách hàng";
    //lấy danh sách khách hàng và convert data
    var listcustomer = ViewBag.listcustomer as
        List<Lab04_WebApplication1.Models.Customer>;
}
<h2>Danh sách khách hàng</h2>
<table class="table-bordered">
    <thead>
        <tr>
            <th>Mã KH</th>
            <th>Họ và tên</th>
            <th>Địa chỉ</th>
            <th>Email</th>
            <th>Điện thoại</th>
            <th>Số dư</th>
        </tr>
    </thead>
    <tbody>
        @*Hiển thị danh sách khách hàng*@
        @foreach (var c in listcustomer)
        {
            <tr>
                <td>@c.CustomerId</td>
                <td>@c.FullName</td>
                <td>@c.Address</td>
                <td>@c.Email</td>
                <td>@c.Phone</td>
                <td>@c.Balance</td>
            </tr>
        }
    </tbody>
</table>
```

Bước 10: Cấu hình **RouteConfig.cs** để chạy controller **Customer** trước. Nhấn F5 để chạy và kích vào link “**Khách hàng**”, “**Danh sách khách hàng**” kiểm tra kết quả.



CustomerDetail - My ASP.NET Ap x +

localhost:50554/Customer/CustomerDetail

Nhấp vào để quay lại, giữ để xem lịch sử

Download Ebook Bấ... Bí Quyết Giàu Có V... Download and Inst..

Application name Home About Contact Khách hàng Danh sách khách hàng

Chi tiết khách hàng lấy theo cách 1

Mã số: KH001

Họ và tên: Trịnh Văn Chung

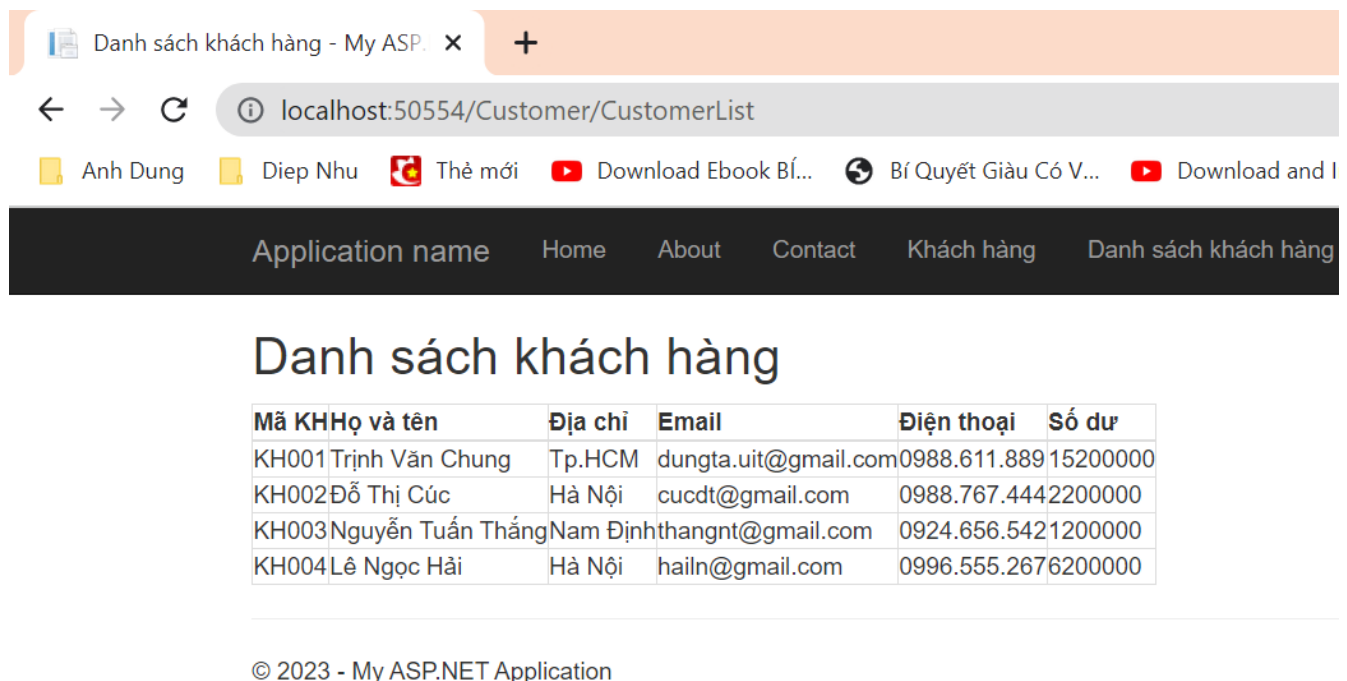
Địa chỉ: Tp.HCM

Điện thoại: 0988.611.889

Email: dungta.uit@gmail.com

Số tiền dư: 15200000

© 2023 - My ASP.NET Application



Danh sách khách hàng - My ASP. x +

localhost:50554/Customer/CustomerList

Anh Dung Diệp Nhu Thẻ mới Download Ebook Bấ... Bí Quyết Giàu Có V... Download and I

Application name Home About Contact Khách hàng Danh sách khách hàng

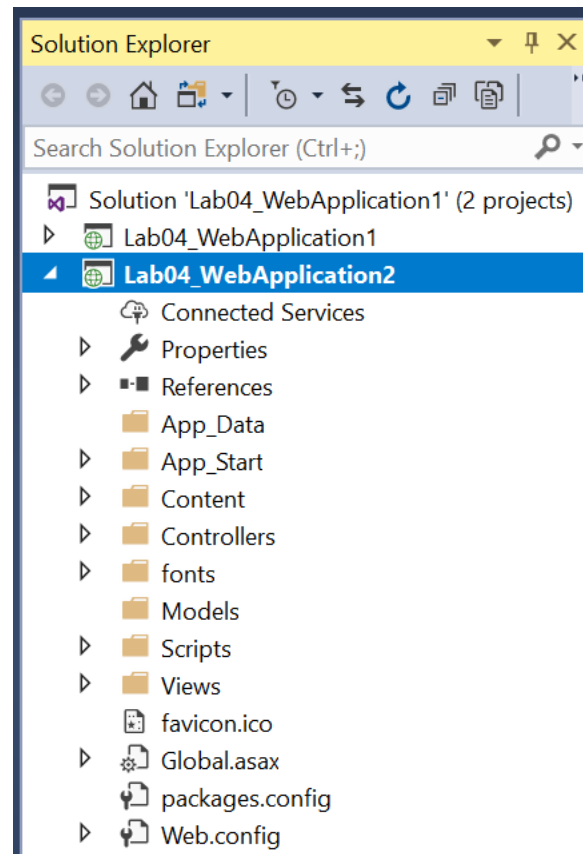
Danh sách khách hàng

Mã KH	Họ và tên	Địa chỉ	Email	Điện thoại	Số dư
KH001	Trịnh Văn Chung	Tp.HCM	dungta.uit@gmail.com	0988.611.889	15200000
KH002	Đỗ Thị Cúc	Hà Nội	cucdt@gmail.com	0988.767.444	2200000
KH003	Nguyễn Tuấn Thắng	Nam Định	thangnt@gmail.com	0924.656.542	1200000
KH004	Lê Ngọc Hải	Hà Nội	hailn@gmail.com	0996.555.267	6200000

© 2023 - My ASP.NET Application

Bài 2: Xây dựng ứng dụng ASP.NET MVC 5 và sử dụng tính năng Scaffolding của Visual Studio để sinh View dựa trên model class “Customer”, thông tin Customer như Bài 1. Các thao tác cần thực hiện là hiển thị, tìm kiếm theo tên, thêm, xóa, sửa, xem chi tiết.

Bước 1: Kích chuột phải vào Solution “**Lab04_WebApplication1**” -> chọn Add -> Chọn **New Project** -> Chọn **ASP.NET Web Application** và đặt tên project “**Lab04_WebApplication2**” -> OK -> Chọn **MVC và No Authentication** -> OK.



Bước 2: Mở trang **_Layout.cshtml** trong đường dẫn **/View/Shared/_Layout.cshtml** và tìm đoạn code sau và thêm vào đúng vị trí

```
<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>
    <li>@Html.ActionLink("Quản lý khách hàng", "GetCustomers", "Customer")</li>
  </ul>
</div>
```

Bước 3: Tạo lớp “**Customer**” trong thư mục Models với code gợi ý như sau:

```
public class Customer
{
    public string CustomerId { get; set; }

    public string FullName { get; set; }

    public string Address { get; set; }

    public string Email { get; set; }

    public string Phone { get; set; }

    public int Balance { get; set; }
}
```

Bước 4: Định nghĩa Interface “**ICustomerRepository**” trong thư mục **Modes** với code gợi ý sau:

```
public interface ICustomerRepository
{
    //khai báo phương thức lấy danh sách khách hàng
    IList<Customer> GetCustomers();

    //khai báo phương thức lấy khách hàng
    Customer GetCustomer(string customerId);

    //khai báo phương thức thêm khách hàng
    void AddCustomer(Customer cus);

    //khai báo phương thức cập nhật khách hàng
    void UpdateCustomer(Customer cus);

    //khai báo phương thức tìm kiếm khách hàng
    IList<Customer> SearchCustomer(string name);

    //khai báo phương thức xóa khách hàng
    void DeleteCustomer(Customer cus);
}
```

Bước 5: Định nghĩa lớp “**CustomerRepository**” trong thư mục **Modes** thực thi từ giao diện **ICustomerRepository** với code gợi ý sau:

```

public class CustomerRepository : ICustomerRepository
{
    //khởi tạo danh sách khách hàng
    static readonly List<Customer> data = new List<Customer>()
    {
        new Customer(){ CustomerId = "KH001",
            FullName = "Trịnh Văn Chung",Address = "Hà Nội",
            Email = "devmaster.founder@gmail.com",
            Phone = "0978.611.889",Balance = 15200000},
        new Customer(){ CustomerId = "KH002",
            FullName = "Đỗ Thị Cúc",Address = "Hà Nội",
            Email = "cucdt@gmail.com",Phone = "0986.767.444",
            Balance = 2200000},
        new Customer(){ CustomerId = "KH003",
            FullName = "Nguyễn Tuấn Thắng",Address = "Nam Định",
            Email = "thangnt@gmail.com",Phone = "0924.656.542",
            Balance = 1200000},
        new Customer(){ CustomerId = "KH004",
            FullName = "Lê Ngọc Hải",Address = "Hà Nội",
            Email = "hailn@gmail.com",Phone = "0996.555.267",
            Balance = 6200000}
    };
    //thực thi phương thức lấy danh sách khách hàng
    public IList<Customer> GetCustomers()
    {
        return data;
    }
    //thực thi phương thức tìm khách hàng theo tên
    public IList<Customer> SearchCustomer(string name)
    {
        return data.Where(c => c.FullName.EndsWith(name)).ToList();
    }
    //thực thi phương thức lấy khách hàng theo Id
    public Customer GetCustomer(string customerId)
    {
        return data.FirstOrDefault(c => c.CustomerId.Equals(customerId));
    }
    //thực thi phương thức thêm khách hàng

```

```

public void AddCustomer(Customer cus)
{
    data.Add(cus);
}
//thực thi phương thức cập nhật khách hàng
public void UpdateCustomer(Customer cus)
{
    //lấy khách hàng theo id
    var customer=data.FirstOrDefault(c=>c.CustomerId.Equals(cus.CustomerId));
    //nếu có thì sửa thông tin
    if (customer != null)
    {
        customer.FullName = cus.FullName;
        customer.Address = cus.Address;
        customer.Email = cus.Email;
        customer.Phone = cus.Phone;
        customer.Balance = cus.Balance;
    }
}
//thực thi phương thức xóa khách hàng
public void DeleteCustomer(Customer cus)
{
    data.Remove(cus);
}
}
    
```

Bước 6: Tạo Controller “**CustomerController**” trong thư mục **Controllers**

```

public class CustomerController : Controller
{
    //khai báo biến CustomerRepository
    static CustomerRepository listCustomer;
    public CustomerController()
    {
        //khởi tạo CustomerRepository
        listCustomer = new CustomerRepository();
    }

    // GET: /Customer/GetCustomers
    
```

```

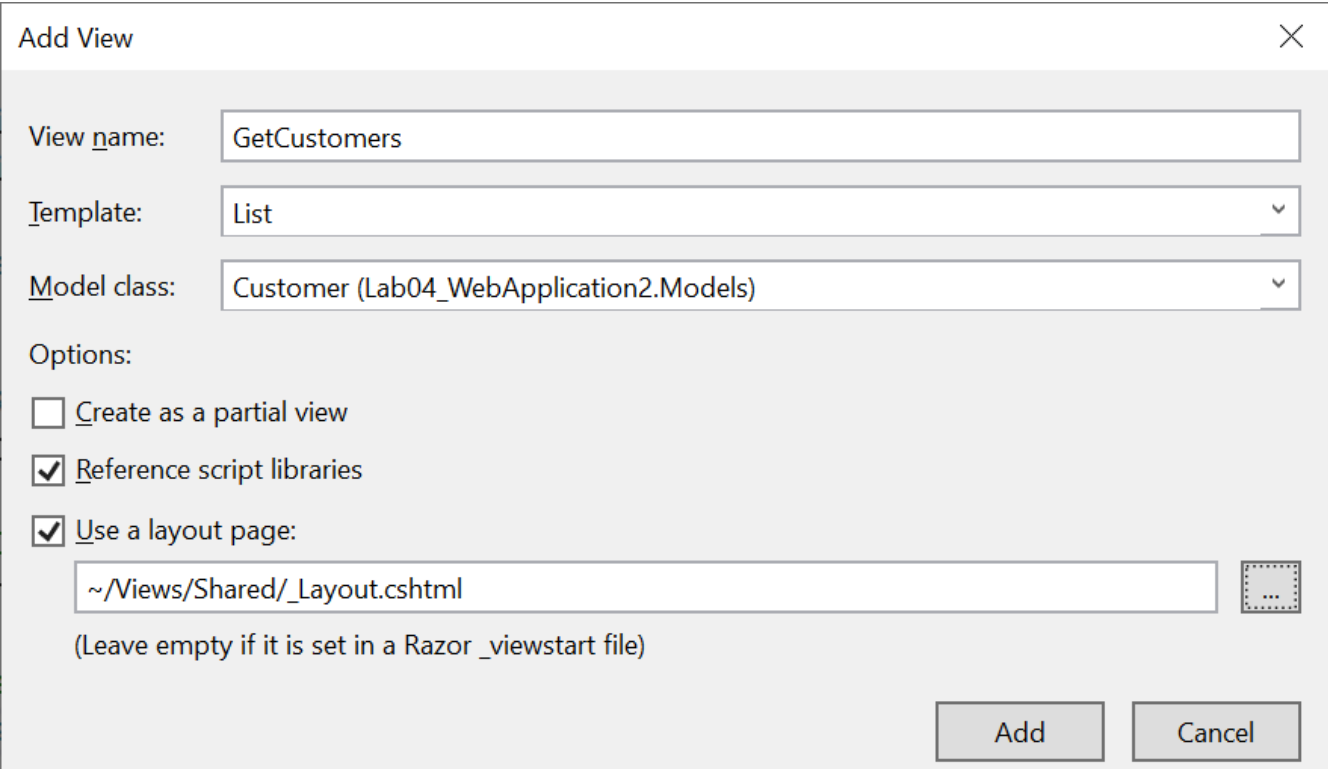
public ActionResult GetCustomers()
{
    return View(listCustomer.GetCustomers());
}
//POST: /Customer/GetCustomers
[HttpPost]
public ActionResult GetCustomers(string name)
{
    return View(listCustomer.SearchCustomer(name));
}
// GET: /Customer/Details/5
public ActionResult Details(string id)
{
    return View(listCustomer.GetCustomer(id));
}
// GET: /Customer/Create
public ActionResult Create()
{
    return View();
}
// POST: /Customer/Create
[HttpPost]
public ActionResult Create(Customer cus)
{
    listCustomer.AddCustomer(cus);
    return RedirectToAction("GetCustomers");
}
// GET: /Customer/Edit/5
public ActionResult Edit(string id)
{
    return View(listCustomer.GetCustomer(id));
}
// POST: /Customer/Edit
[HttpPost]
public ActionResult Edit(Customer cus)

```

```
{
    listCustomer.UpdateCustomer(cus);
    return RedirectToAction("GetCustomers");
}
// GET: /Customer/Delete/5
public ActionResult Delete(string id)
{
    listCustomer.DeleteCustomer(listCustomer.GetCustomer(id));
    return RedirectToAction("GetCustomers");
}
}
```

Bước 7: Tiếp theo chúng ta tạo các **View** tương ứng với các **Action** trong **Controller**

Kích chuột phải vào Action “**GetCustomers**” và chọn Add View.. -> Chọn template và Model class như hình sau:



Add View

View name: GetCustomers

Template: List

Model class: Customer (Lab04_WebApplication2.Models)

Options:

☐ Create as a partial view

☒ Reference script libraries

☒ Use a layout page:

~/Views/Shared/_Layout.cshtml

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

Add Cancel

Sau đó các bạn chỉnh sửa lại code Razor theo gợi ý sau:

```
@model IEnumerable<Lab04_WebApplication2.Models.Customer>
@{
    ViewBag.Title = "GetCustomers";
    Layout = "~/Views/Shared/_Layout.cshtml";
}
<h2>Danh sách khách hàng</h2>
@using (Html.BeginForm())
{
    <b>Nhập tên cần tìm </b> @Html.TextBox("name")
    <input type="submit" value=" Tìm " />
}
<p>
    @Html.ActionLink("Thêm mới", "Create")
</p>
<table class="table table-bordered">
    <tr>
        <th>Mã khách hàng</th>
        <th>Họ và tên</th>
        <th>Địa chỉ</th>
        <th>Email</th>
        <th>Điện thoại</th>
        <th>Số dư</th>
        <th>Thao tác</th>
    </tr>
    @foreach (var item in Model)
    {
        <tr>
            <td>@Html.DisplayFor(modelItem => item.CustomerId)</td>
            <td>@Html.DisplayFor(modelItem => item.FullName)</td>
            <td>@Html.DisplayFor(modelItem => item.Address)</td>
            <td>@Html.DisplayFor(modelItem => item.Email)</td>
            <td>@Html.DisplayFor(modelItem => item.Phone)</td>
            <td>@Html.DisplayFor(modelItem => item.Balance)</td>
            <td>
                @Html.ActionLink("Sửa", "Edit", new { id = item.CustomerId }) |
                @Html.ActionLink("Chi tiết", "Details", new { id = item.CustomerId }) |
                @Html.ActionLink("Xóa", "Delete", new { id = item.CustomerId },
                new { onclick = "return confirm('Bạn có muốn xóa khách hàng " +
                item.CustomerId + " không?')" })
            </td>
        </tr>
    }
</table>
```

Kích chuột phải vào Action “**Details**” và chọn Add View.. -> Chọn template và Model class như hình.

Add View

View name: Details

Template: Details

Model class: Customer (Lab04_WebApplication2.Models)

Options:

☐ Create as a partial view

☒ Reference script libraries

☒ Use a layout page:

~/Views/Shared/_Layout.cshtml

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

Add Cancel

Sau đó chỉnh lại code Razor theo gợi ý sau:

```
@model Lab04_WebApplication2.Models.Customer
@{
    ViewBag.Title = "Details";
    Layout = "~/Views/Shared/_Layout.cshtml";
}

<div>
    <h3>Thông tin chi tiết</h3>
    <hr />
    <dl class="dl-horizontal">
        <dt>Mã khách hàng</dt>
        <dd>@Html.DisplayFor(model => model.CustomerId)</dd>
        <dt>Họ và tên</dt>
        <dd>@Html.DisplayFor(model => model.FullName)</dd>
        <dt>Địa chỉ</dt>
    </dl>
</div>
```



```

        <dd>@Html.DisplayFor(model => model.Address)</dd>
        <dt>Email</dt>
        <dd>@Html.DisplayFor(model => model.Email)</dd>
        <dt>Điện thoại</dt>
        <dd>@Html.DisplayFor(model => model.Phone)</dd>
        <dt>Số dư</dt>
        <dd>@Html.DisplayFor(model => model.Balance)</dd>
    </dl>
</div>
<p>
    @Html.ActionLink("Sửa", "Edit", new { id = Model.CustomerId })|
    @Html.ActionLink("Quay lại", "GetCustomers")
</p>

```

Tiếp theo, kích chuột phải vào Action “**Create**” và chọn Add View.. -> Chọn template và Model class như hình.

Add View

View name: Create

Template: Create

Model class: Customer (Lab04_WebApplication2.Models)

Options:

☐ Create as a partial view

☒ Reference script libraries

☒ Use a layout page:

~/Views/Shared/_Layout.cshtml

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

Add Cancel

Sau đó chỉnh sửa lại code Razor theo gợi ý như sau:

```

@model Lab04_WebApplication2.Models.Customer
@{
    ViewBag.Title = "Create";
    Layout = "~/Views/Shared/_Layout.cshtml";
}
@using (Html.BeginForm())
{
    @Html.AntiForgeryToken()
    <div class="form-horizontal">
        <h3>Thêm mới khách hàng</h3>
        <hr />
        @Html.ValidationSummary(true)
        <div class="form-group">
            @Html.Label("Mã khách hàng", new { @class="control-label col-md-2" })
            <div class="col-md-10">
                @Html.EditorFor(model => model.CustomerId)
                @Html.ValidationMessageFor(model => model.CustomerId)
            </div>
        </div>
        <div class="form-group">
            @Html.Label("Họ và tên", new { @class = "control-label col-md-2" })
            <div class="col-md-10">
                @Html.EditorFor(model => model.FullName)
                @Html.ValidationMessageFor(model => model.FullName)
            </div>
        </div>
        <div class="form-group">
            @Html.Label("Địa chỉ", new { @class = "control-label col-md-2" })
            <div class="col-md-10">
                @Html.EditorFor(model => model.Address)
                @Html.ValidationMessageFor(model => model.Address)
            </div>
        </div>
        <div class="form-group">

```

```

        @Html.Label("Email", new { @class = "control-label col-md-2" })
        <div class="col-md-10">
            @Html.EditorFor(model => model.Email)
            @Html.ValidationMessageFor(model => model.Email)
        </div>
    </div>
    <div class="form-group">
        @Html.Label("Điện thoại", new { @class="control-label col-md-2" })
        <div class="col-md-10">
            @Html.EditorFor(model => model.Phone)
            @Html.ValidationMessageFor(model => model.Phone)
        </div>
    </div>
    <div class="form-group">
        @Html.Label("Số dư", new { @class = "control-label col-md-2" })
        <div class="col-md-10">
            @Html.EditorFor(model => model.Balance)
            @Html.ValidationMessageFor(model => model.Balance)
        </div>
    </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", "GetCustomers")
</div>

```

Tiếp theo, tạo View cho Action “**Edit**”

Add View

View name: Edit

Template: Edit

Model class: Customer (Lab04_WebApplication2.Models)

Options:

☐ Create as a partial view

☒ Reference script libraries

☒ Use a layout page:

~/Views/Shared/_Layout.cshtml

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

Add Cancel

Sau đó chỉnh sửa code Razor cho **View Edit** như sau:

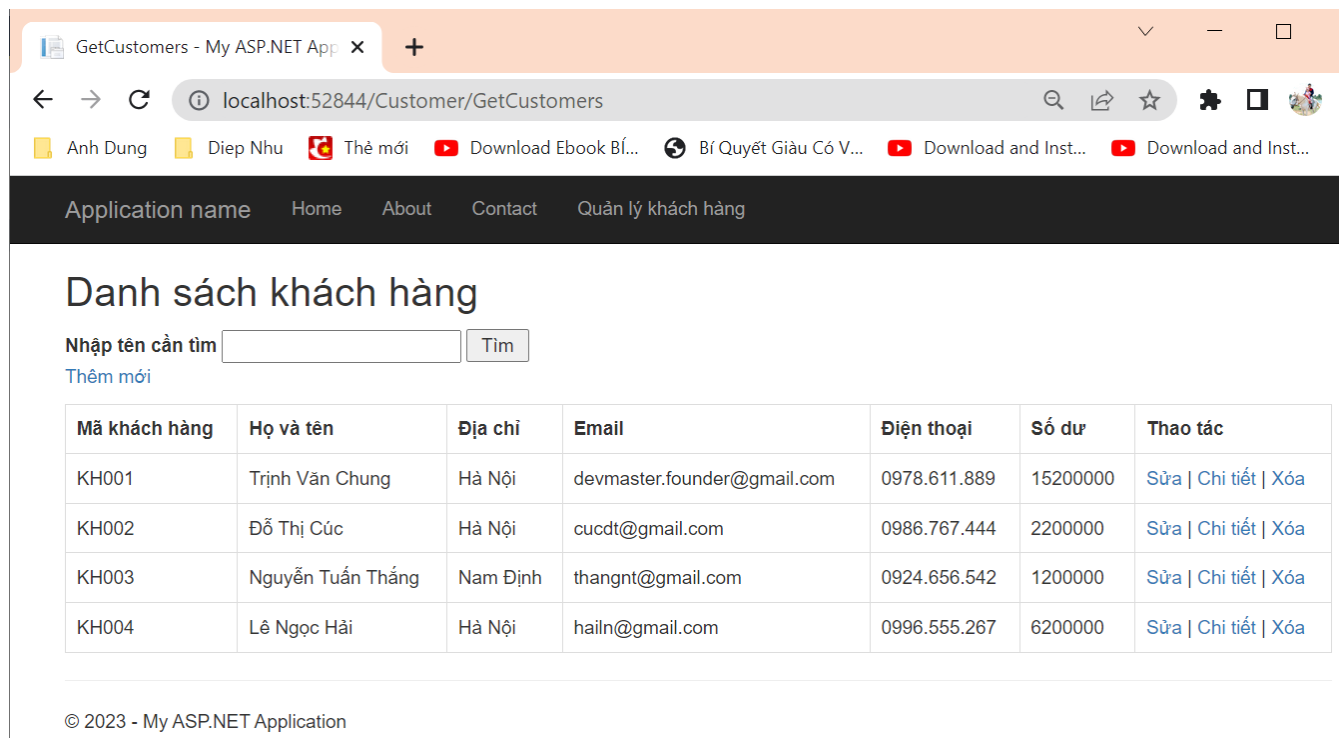
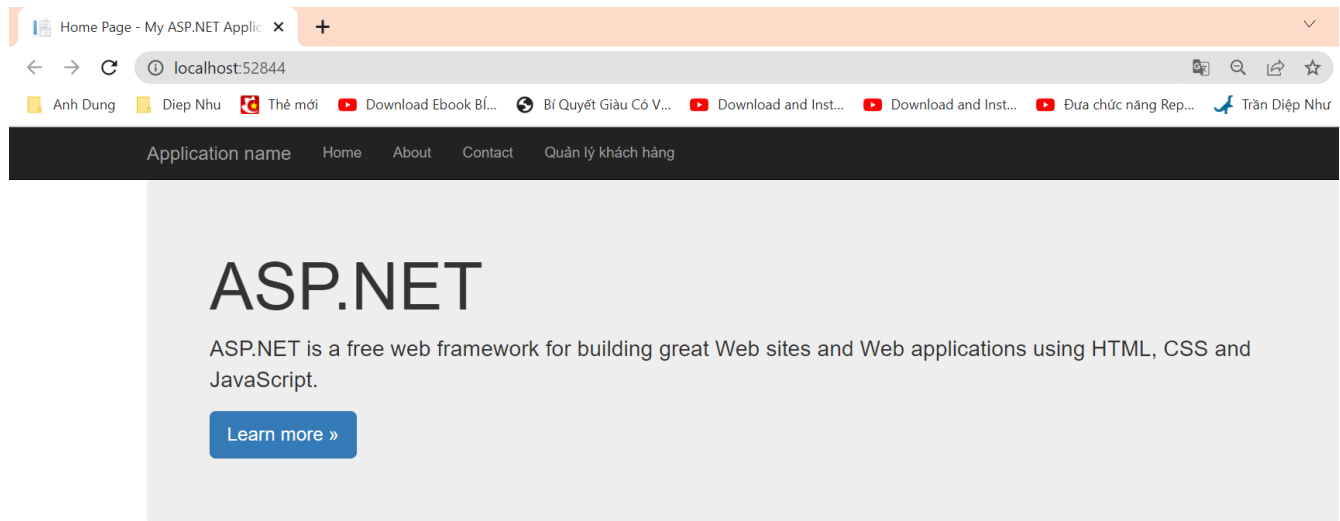
```
@model Lab04_WebApplication2.Models.Customer
@{
    ViewBag.Title = "Edit";
    Layout = "~/Views/Shared/_Layout.cshtml";
}
@using (Html.BeginForm())
{
    @Html.AntiForgeryToken()
    <div class="form-horizontal">
        <h3>Chỉnh sửa khách hàng</h3>
        <hr />
        @Html.ValidationSummary(true)
        @Html.HiddenFor(model => model.CustomerId)
        <div class="form-group">
            @Html.Label("Họ và tên", new { @class = "control-label col-md-2" })
            <div class="col-md-10">
                @Html.EditorFor(model => model.FullName)
                @Html.ValidationMessageFor(model => model.FullName)
            </div>
        </div>
    </div>
```

```

<div class="form-group">
    @Html.Label("Địa chỉ", new { @class = "control-label col-md-2" })
    <div class="col-md-10">
        @Html.EditorFor(model => model.Address)
        @Html.ValidationMessageFor(model => model.Address)
    </div>
</div>
<div class="form-group">
    @Html.Label("Email", new { @class = "control-label col-md-2" })
    <div class="col-md-10">
        @Html.EditorFor(model => model.Email)
        @Html.ValidationMessageFor(model => model.Email)
    </div>
</div>
<div class="form-group">
    @Html.Label("Điện thoại", new { @class = "control-label col-md-2" })
    <div class="col-md-10">
        @Html.EditorFor(model => model.Phone)
        @Html.ValidationMessageFor(model => model.Phone)
    </div>
</div>
<div class="form-group">
    @Html.Label("Số dư", new { @class = "control-label col-md-2" })
    <div class="col-md-10">
        @Html.EditorFor(model => model.Balance)
        @Html.ValidationMessageFor(model => model.Balance)
    </div>
</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", "GetCustomers")
</div>

```

Bước 8: Browse trang **Index.cshtml** và kích vào các link để kiểm tra kết quả



Edit - My ASP.NET Application
+

localhost:52844/Customer/Edit/KH001

Anh Dung
Diep Nhu
Thẻ mới
Download Ebook Bấ...
Bí Quyết Giàu Có V...
Download and Inst...

Application name
Home
About
Contact
Quản lý khách hàng

Chỉnh sửa khách hàng

Họ và tên

Trịnh Văn Chung

Địa chỉ

Hà Nội

Email

devmaster.founder@gmail.c

Điện thoại

0978.611.889

Số dư

15200000

Lưu

[Quay lại](#)

© 2023 - My ASP.NET Application

Details - My ASP.NET Application
+

localhost:52844/Customer/Details/KH001

Anh Dung
Diep Nhu
Thẻ mới
Download Ebook Bấ...
Bí Quyết Giàu Có V...
Download and Inst...

Application name
Home
About
Contact
Quản lý khách hàng

Thông tin chi tiết

Mã khách hàng

KH001

Họ và tên

Trịnh Văn Chung

Địa chỉ

Hà Nội

Email

devmaster.founder@gmail.com

Điện thoại

0978.611.889

Số dư

15200000

Sửa| Quay lại

© 2023 - My ASP.NET Application