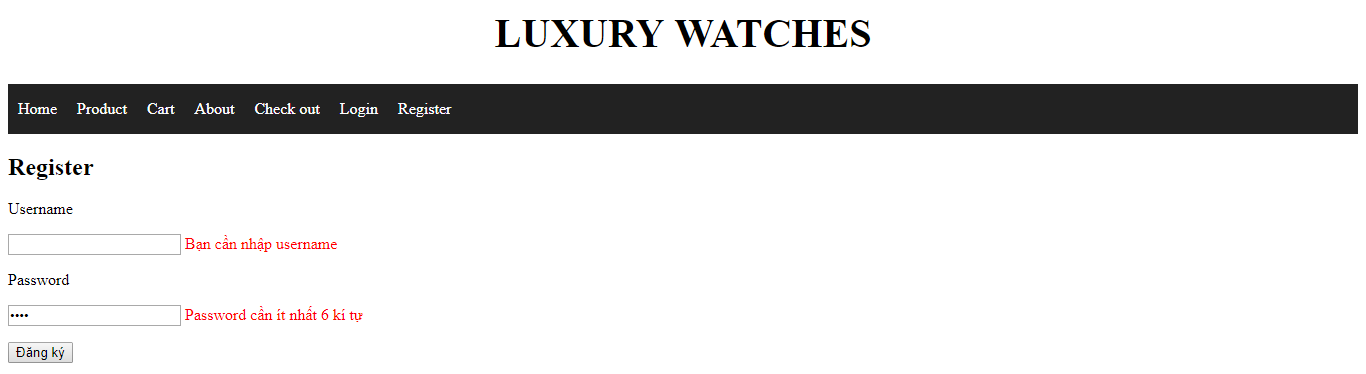
**HƯỚNG DẪN LẬP TRÌNH WEB VỚI ASP.NET MVC**

**( Phần 4 - Validation data và html helper trong trang đăng ký)**

1. Nhiệm vụ

Trong phần trước chúng ta đã làm quen với ASP.NET MVC và tạo ra trang chủ có kết nối với CSDL. Trong chương này chúng ta sẽ đi tiếp vào việc xây dựng một trang đăng ký để nắm bắt cách bắt validation data sử dụng Datannotation của C#. Demo sẽ trông như sau:



1. Mô tả cách làm.

* Đầu tiên ta cần vào masterpage thêm vào menu mục Register như sau:



Ta chỉ cần thêm một thẻ

<li class="menu-item"><a href="~/Account/Register">Register</a></li> là xong.

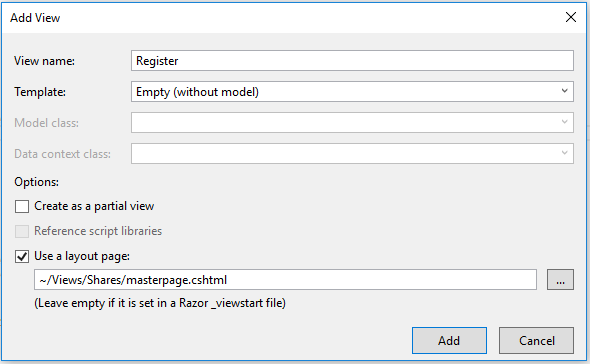
Tiếp theo ta tạo một action register method get trong Account controller. Như sau:

public ActionResult Register()

{

return View();

}

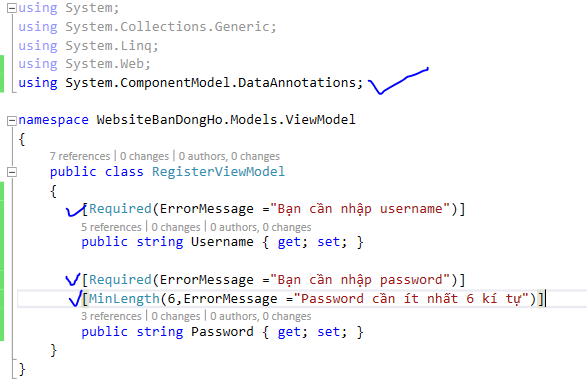
Chuột phải vào action Register và chọn add view và điền các thông tin trong hộp thoại Add View ( Nhớ chọn masterpage nha).

Ta sẽ có được view rỗng của trang đăng ký.

Ta sẽ viết xử lý cho action và view Register này sau.

* 1. Xây dựng Model

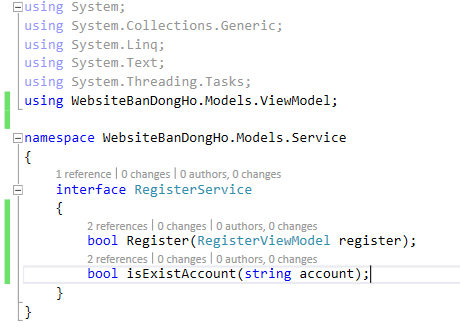
Trang đăng ký chúng ta sẽ cho người nhập 2 thông tin là username và password vì vậy ta cần tạo một RegisterViewModel chứa 2 thông tin này. Trong C# có hỗ trợ cơ chế bắt validation cho các thuộc tính của class sử dụng Datannotation. Phần ErrorMessage sẽ được hiển thị lên view. Cách để hiển thị các lỗi lên view sẽ được trình bày trong phần viết code view Register.



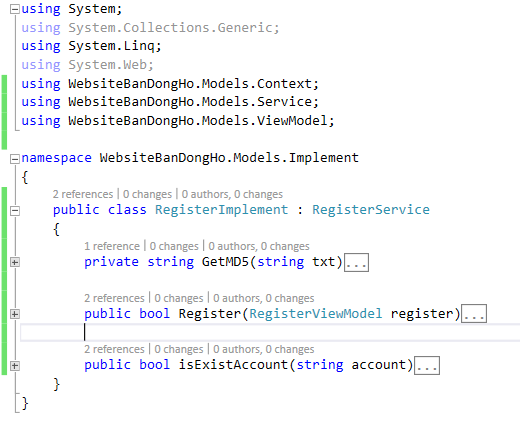
Tiếp theo cần viết các xử lý với cơ sở dữ liệu cho action đăng ký. Ta sẽ tạo interface RegisterService và class RegisterImplement để viết các xử lý này.

Dự định sẽ có 2 hàm chính, một là kiểm tra tài khoản có tồn tại trên hệ thống hay chưa và 2 là hàm thêm tài khoản và cơ sở dữ liệu.

Do đó interface RegisterService sẽ trông như sau:



Và class RegisterImplement sẽ như sau:



Ta cần chú ý là tham số cho hàm Register là một RegisterViewModel chứ không phải là một class TAIKHOAN. Mục đính chính là một ViewModel giúp ta dễ dàng làm theo ý mình hơn. Class TAIKHOAN có quá nhiều thông tin không cần hiển thị cho người dùng biết như NGAYDANGKY, TRANGTHAI, MALOAITK. Trong khi ta chỉ cần hiển thị 2 thông tin là Username và Password, việc sử dụng ViewModel sẽ hợp lý và dễ dàng hơn rất nhiều. Nếu ta viết các Dataanotaton cho class TAIKHOAN thì vẫn ok. Nhưng như thế sẽ gây rối code và không tiện bằng việc tạo một ViewModel.

Chi tiết các hàm:

public class RegisterImplement : RegisterService

{

private string GetMD5(string txt)

{

String str = "";

Byte[] buffer = System.Text.Encoding.UTF8.GetBytes(txt);

System.Security.Cryptography.MD5CryptoServiceProvider md5 =

new System.Security.Cryptography.MD5CryptoServiceProvider();

buffer = md5.ComputeHash(buffer);

foreach (Byte b in buffer)

{

str += b.ToString("X2");

}

return str;

}

public bool Register(RegisterViewModel register)

{

using (var db = new BANDONGHOEntities())

{

try

{

TAIKHOAN tk = new TAIKHOAN { TENDN = register.Username, MATKHAU = GetMD5(register.Password), NGAYDANGKY = DateTime.Now, TRANGTHAI = true, MALOAITK = "LK00002" };

db.TAIKHOANs.Add(tk);

db.SaveChanges();

return true;

}

catch

{

return false;

}

}

}

public bool isExistAccount(string account)

{

using (var db = new BANDONGHOEntities())

{

TAIKHOAN taikhoan = (from tk in db.TAIKHOANs

where tk.TENDN.Equals(account)

select tk).SingleOrDefault();

if (taikhoan != null)

{

return true;

}

return false;

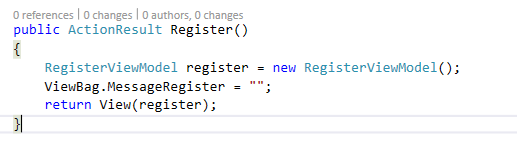
}

}

}

* 1. Xây dựng action Register trong AccountController

Ta cần trả cho view một RegisterViewModel để thực hiện việc bắt lỗi bằng ViewModel. ViewBag để chứa thông báo đăng ký thành công hay thất bại.



Tạo action Register method Post để nhận dữ liệu gửi từ form và xử lý request đăng ký từ người dùng. Ta chú ý dòng lệnh if( ModelState.IsValid) chính là lệnh kiểm tra các field dữ liệu có đúng với định dạng đã định trước trong RegisterViewModdel hay không. Nếu có lỗi thì trả lại cho View với các lỗi được chứa trong biến kiểu RegisterViewModel là register.



* 1. Xây dựng View Register.

View của chúng ta sẽ chứa một form để submit dữ liệu. Cần chú ý là các field input cần bọc trong thẻ form.



Ta cần chú ý các chỗ có dấu check.

Dấu check đầu là khai báo một RegisterViewModel để chứa các lỗi bắt được trả về từ Controller.

Các lỗi sẽ được hiển thị bằng cách sử dụng html helper của c#. Code sẽ như sau:

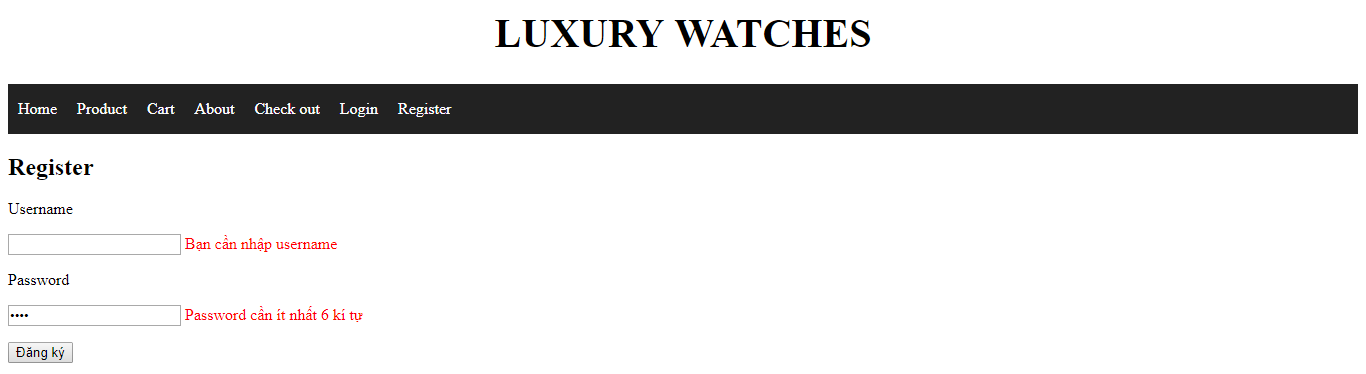
@Html.ValidationMessageFor(model => model.Username, "", new { @class = "text-danger", @style = "color: red" })

* 1. Kết quả demo

Kết quả khi không điền thông tin.



Kết quả khi mật khẩu không đủ 6 kí tự.



Đăng ký thành công

