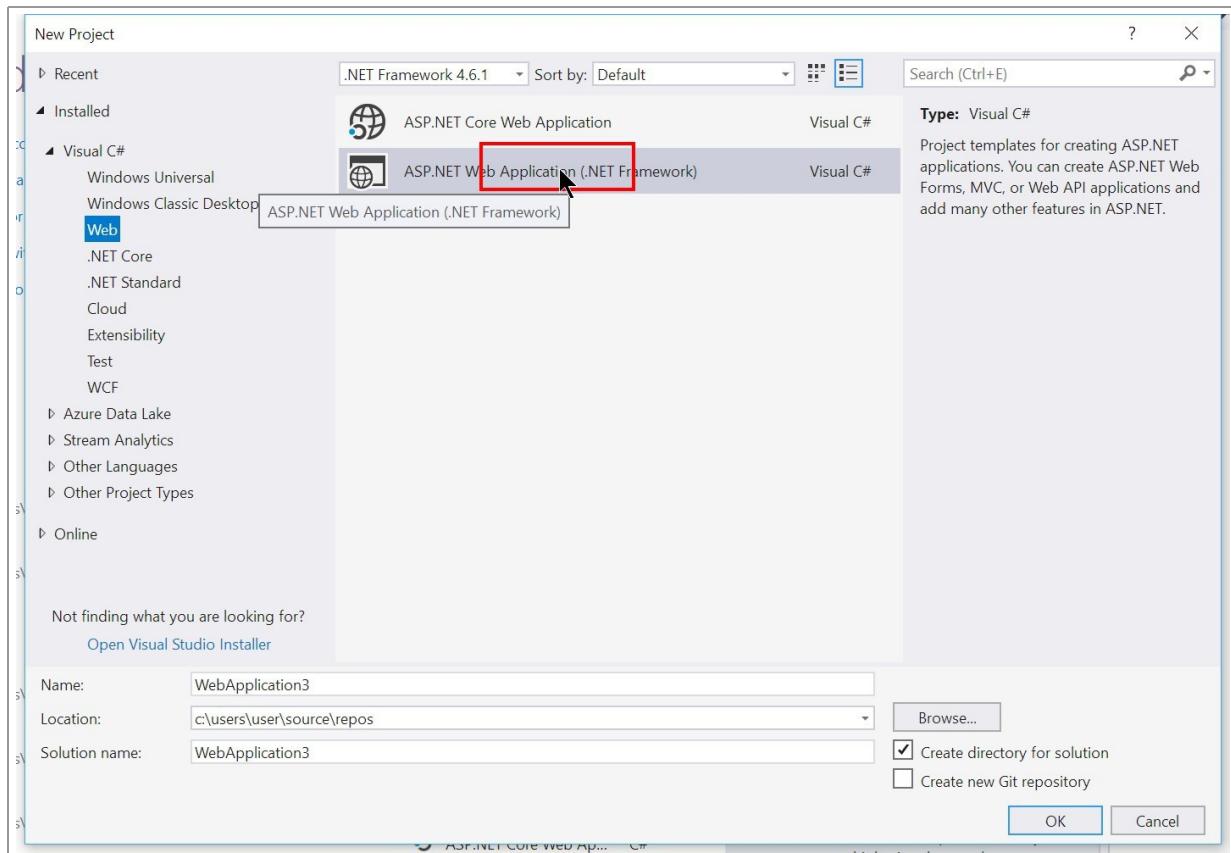


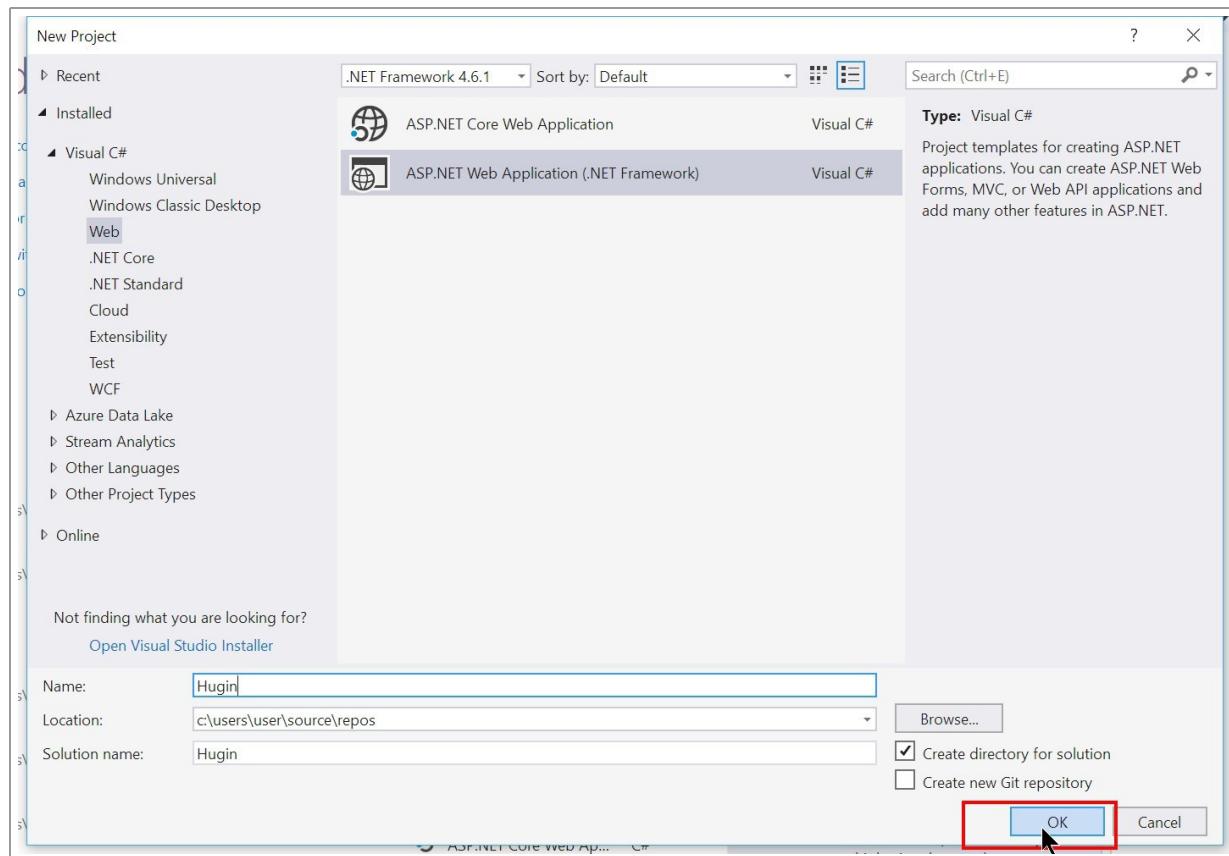
MVC Exercise 1

E1\User

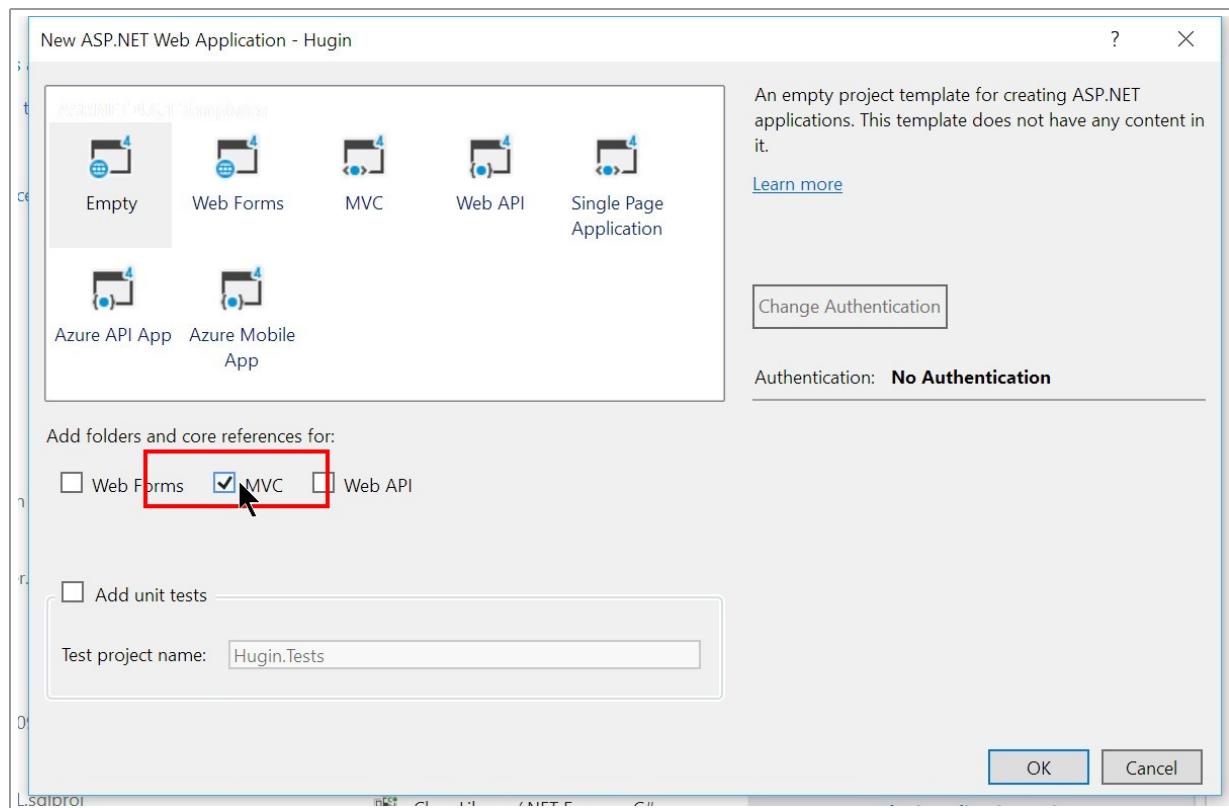
Default description



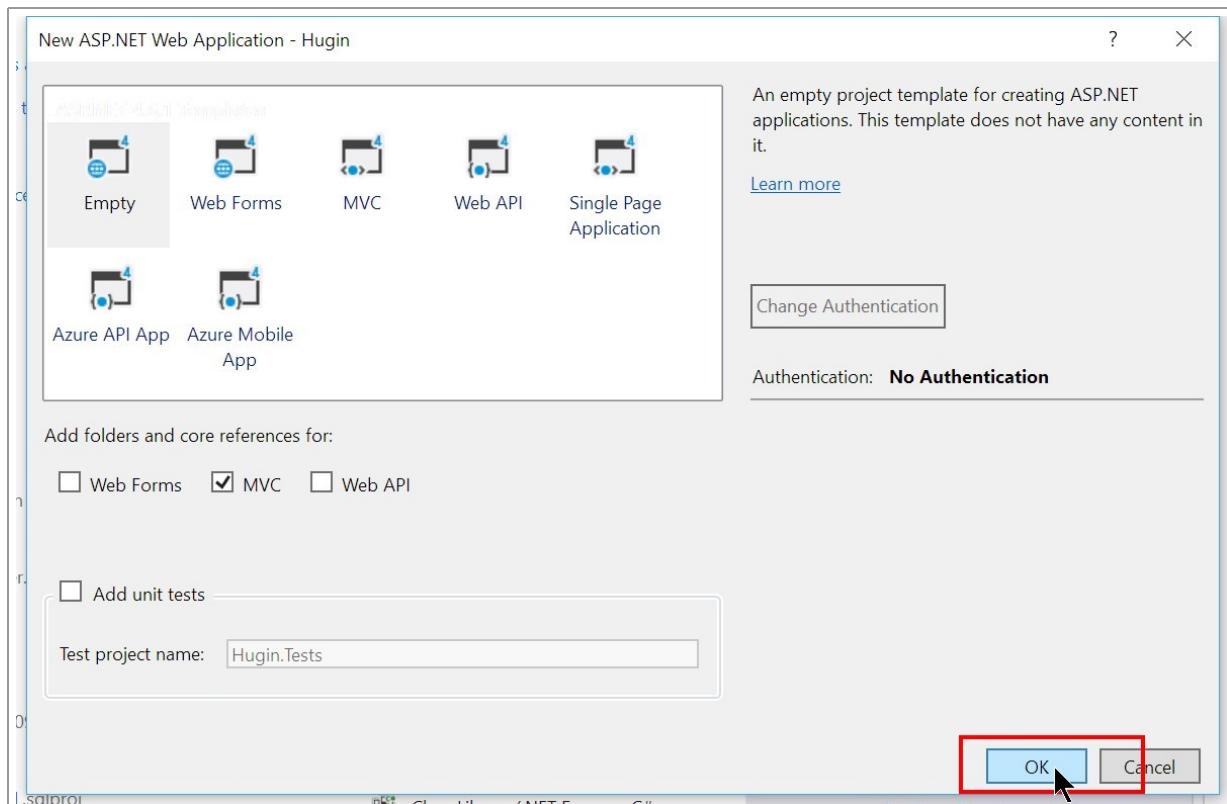
Click on "ASP.NET Web Application (.NET Framework)" text in "New Project"



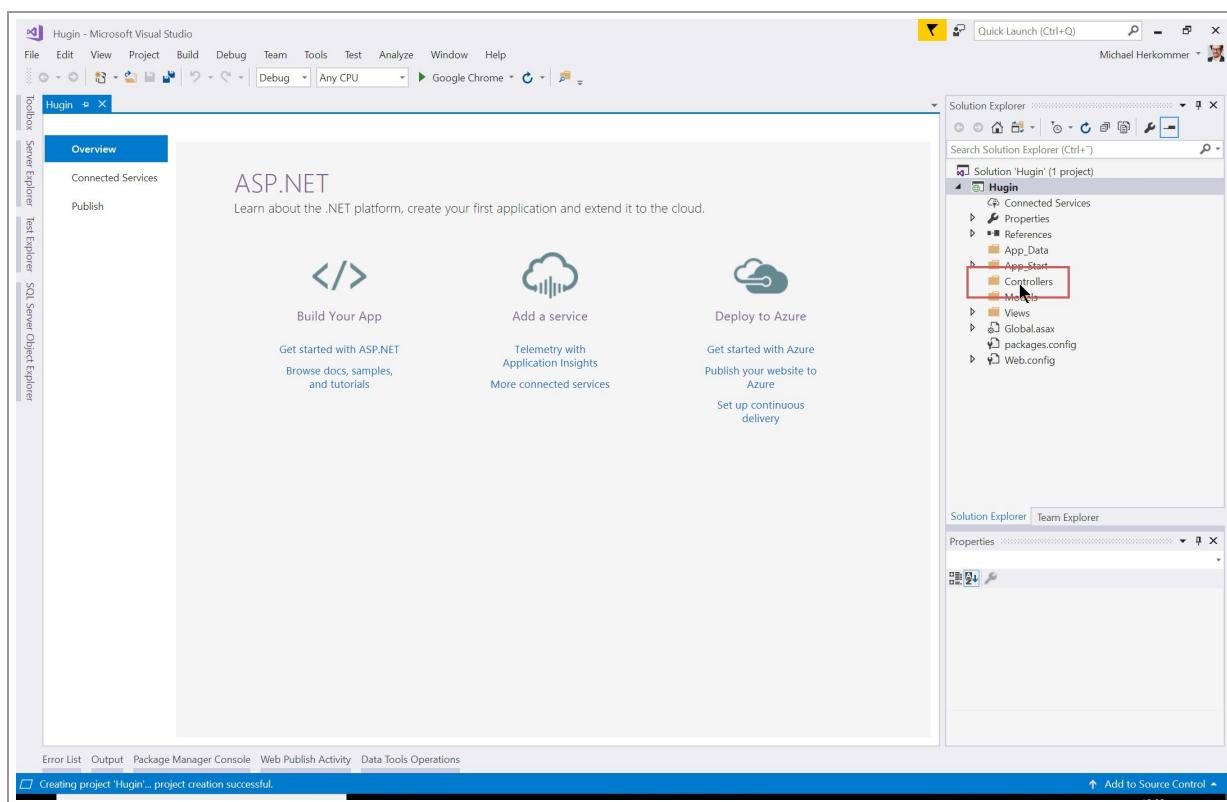
Click on "OK" button in "New Project"



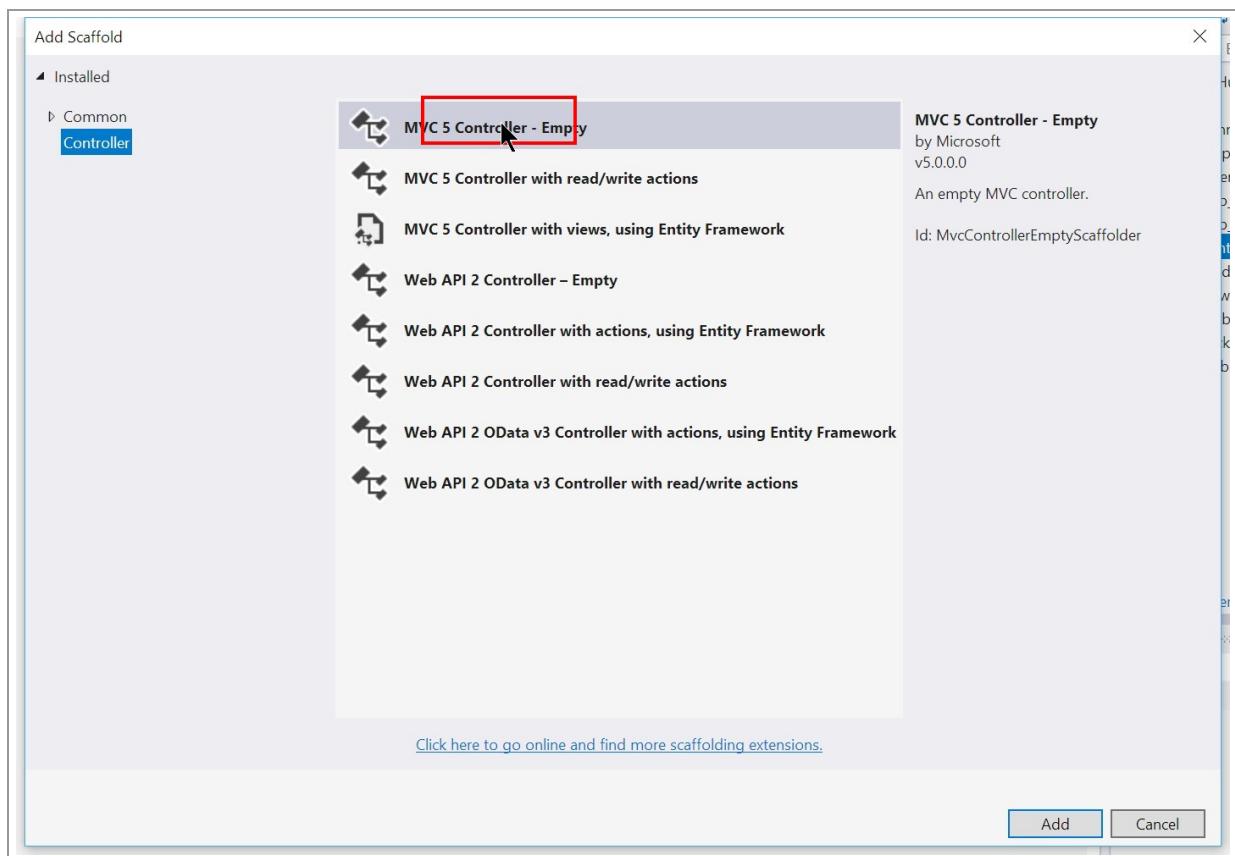
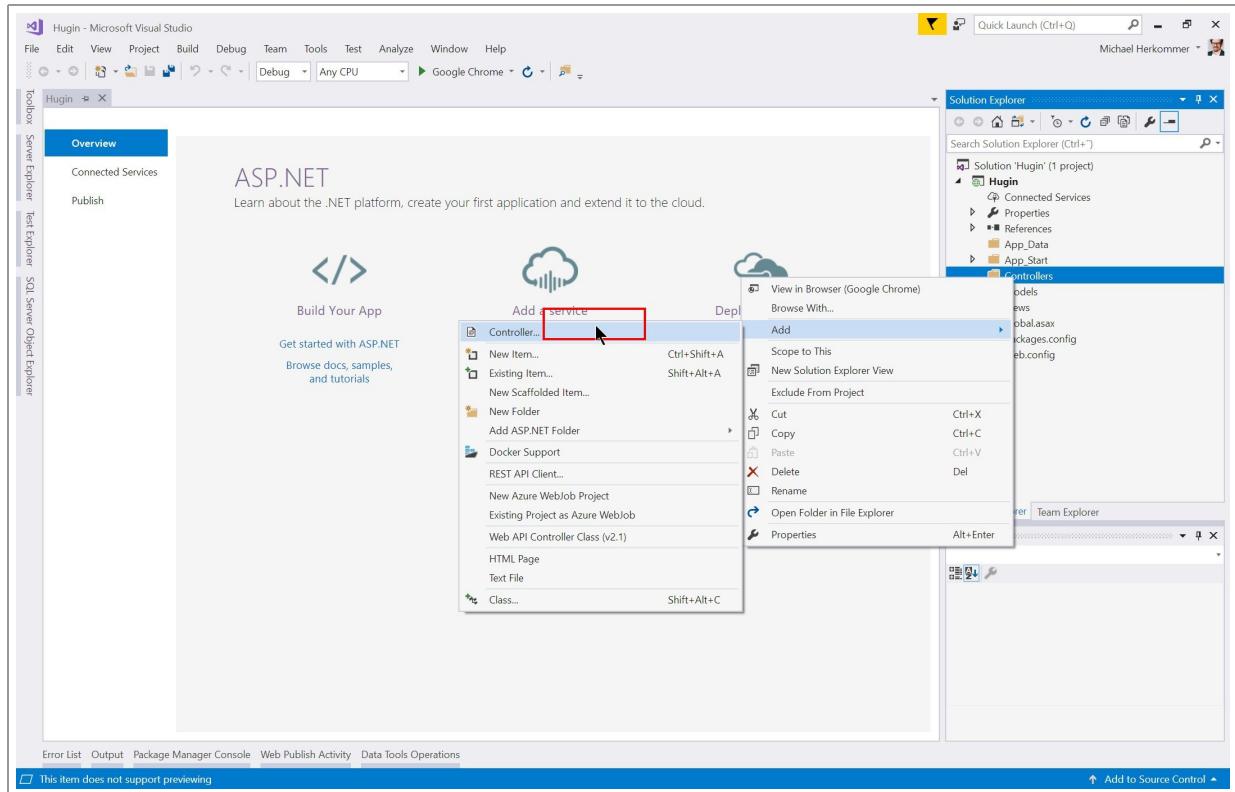
Check "MVC" in "New ASP.NET Web Application - Hugin"



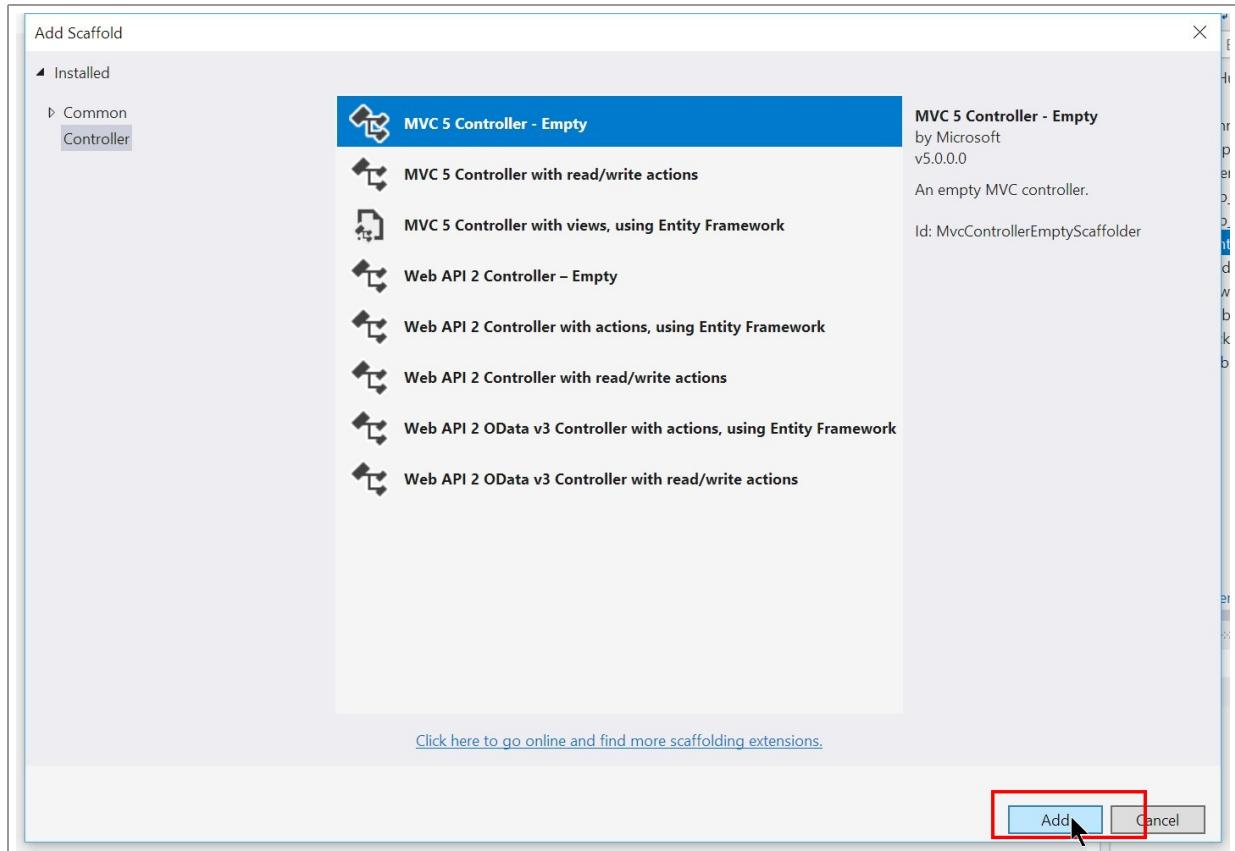
Click on "OK" button in "New ASP.NET Web Application - Hugin"



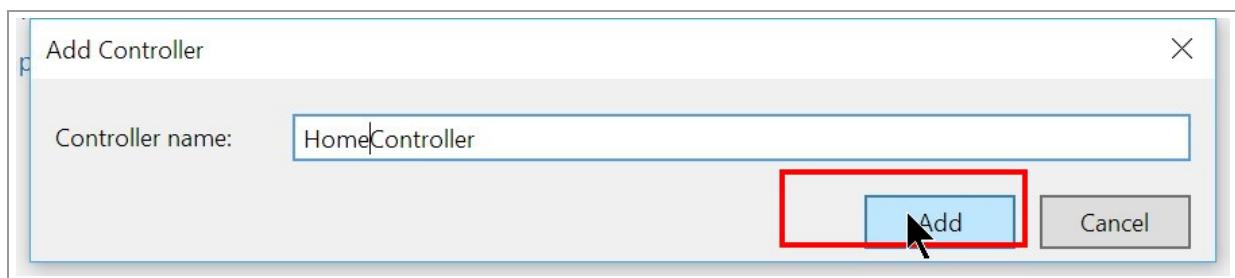
Right click on "Controllers" tree view item in "Solution Explorer"



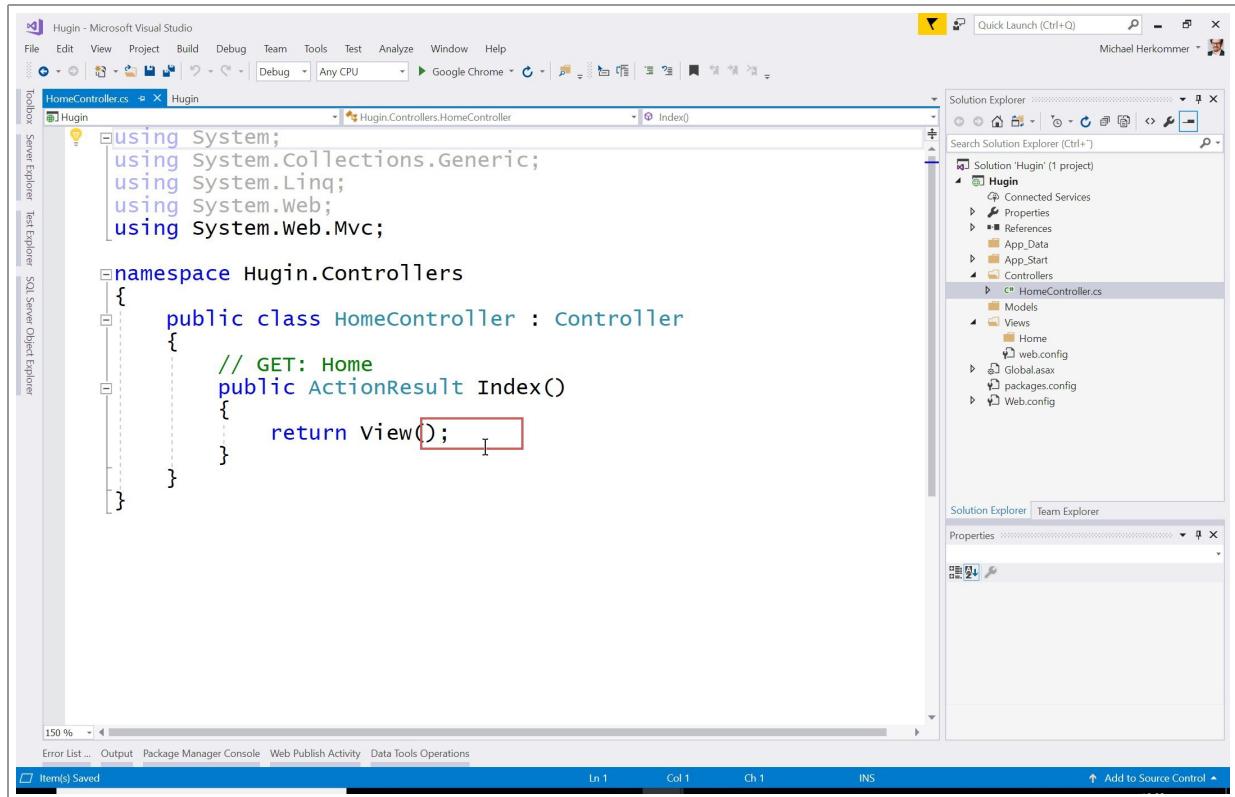
Click on "MVC 5 Controller - Empty" text in "Add Scaffold"



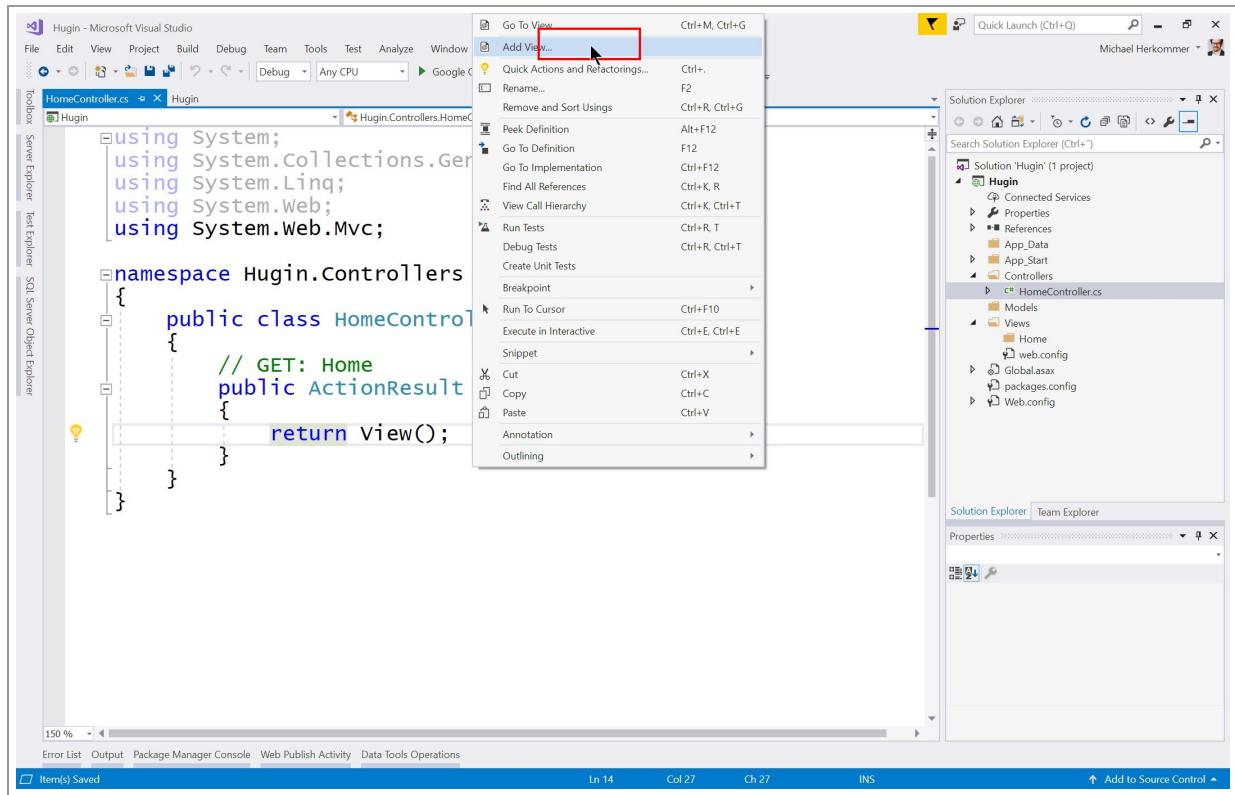
Click on "Add" text in "Add Scaffold"

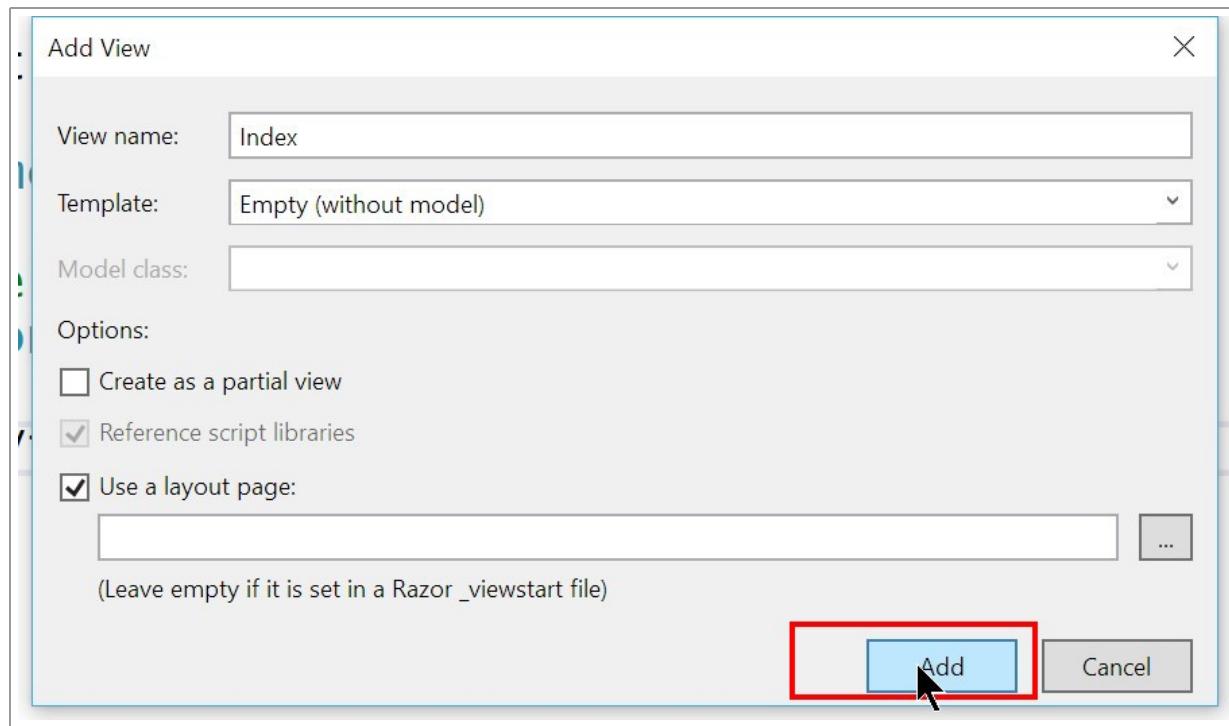


Click on "Add" button in "Add Controller"

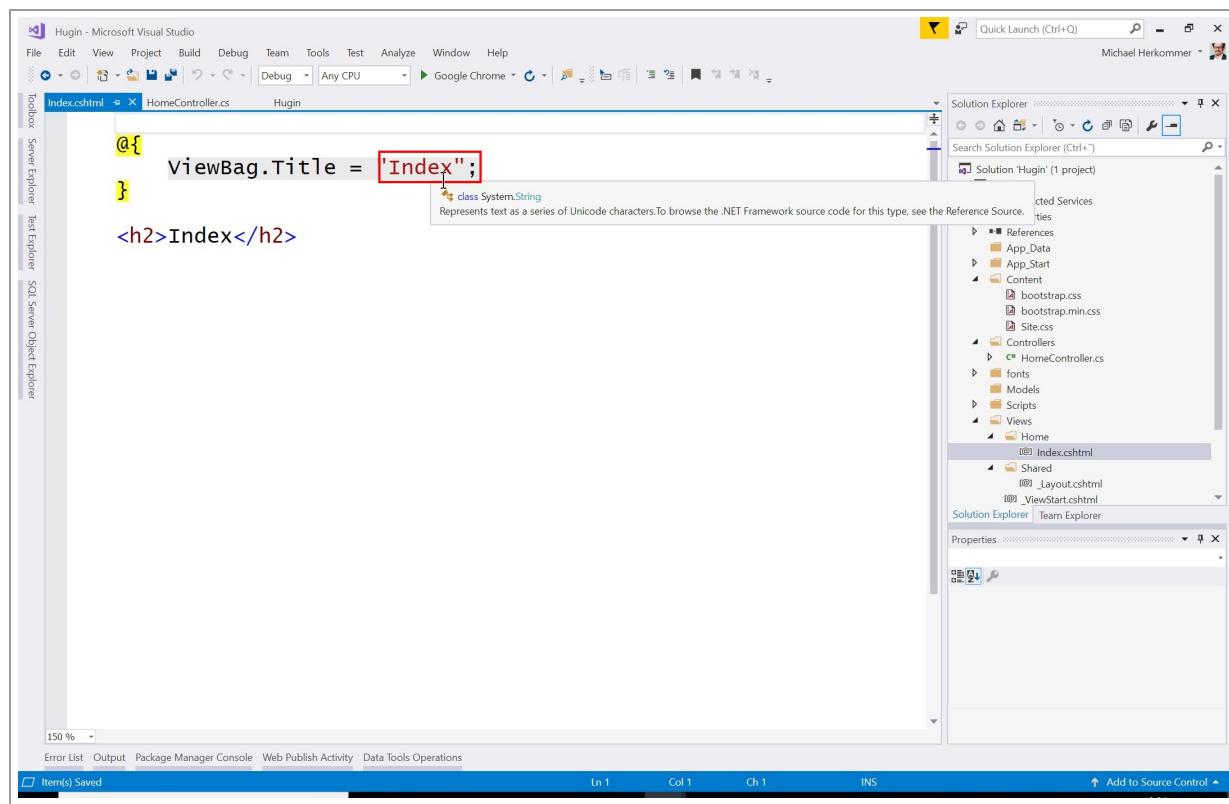


Right click on "Text Editor" edit in "HomeController.cs"

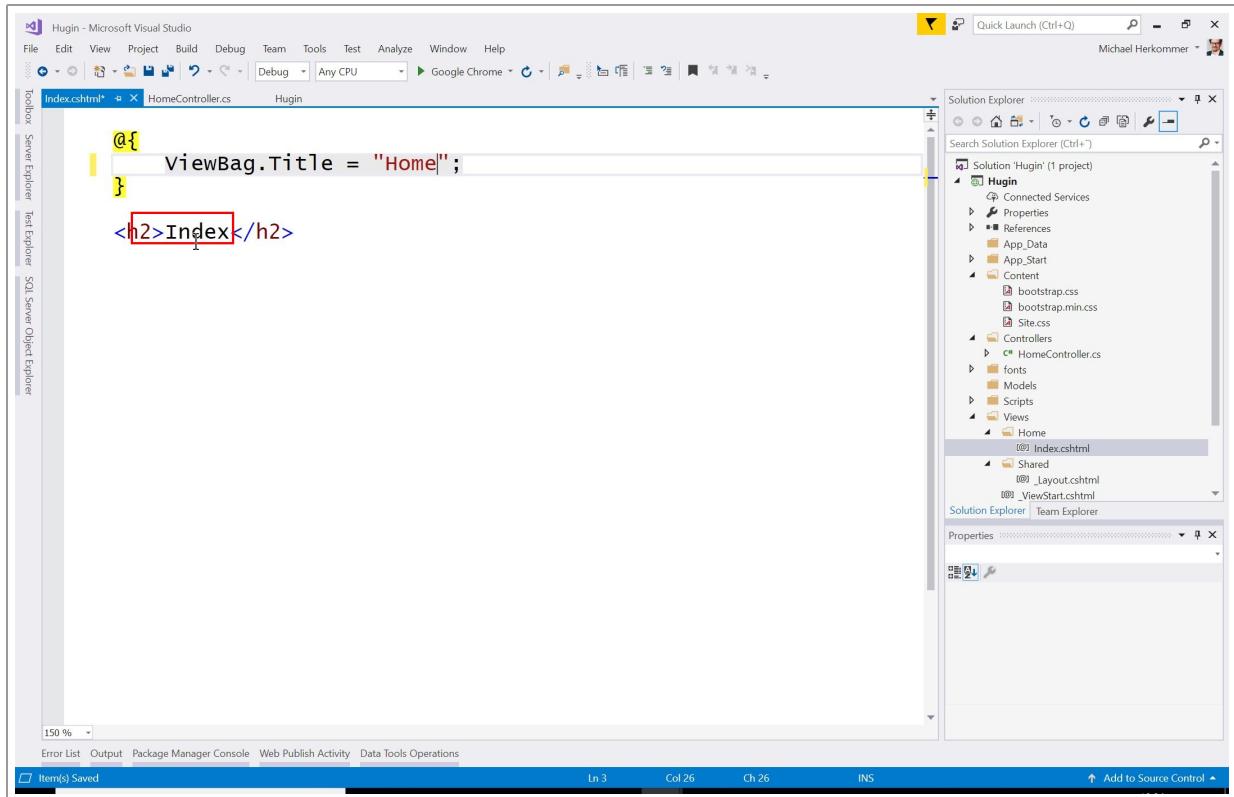




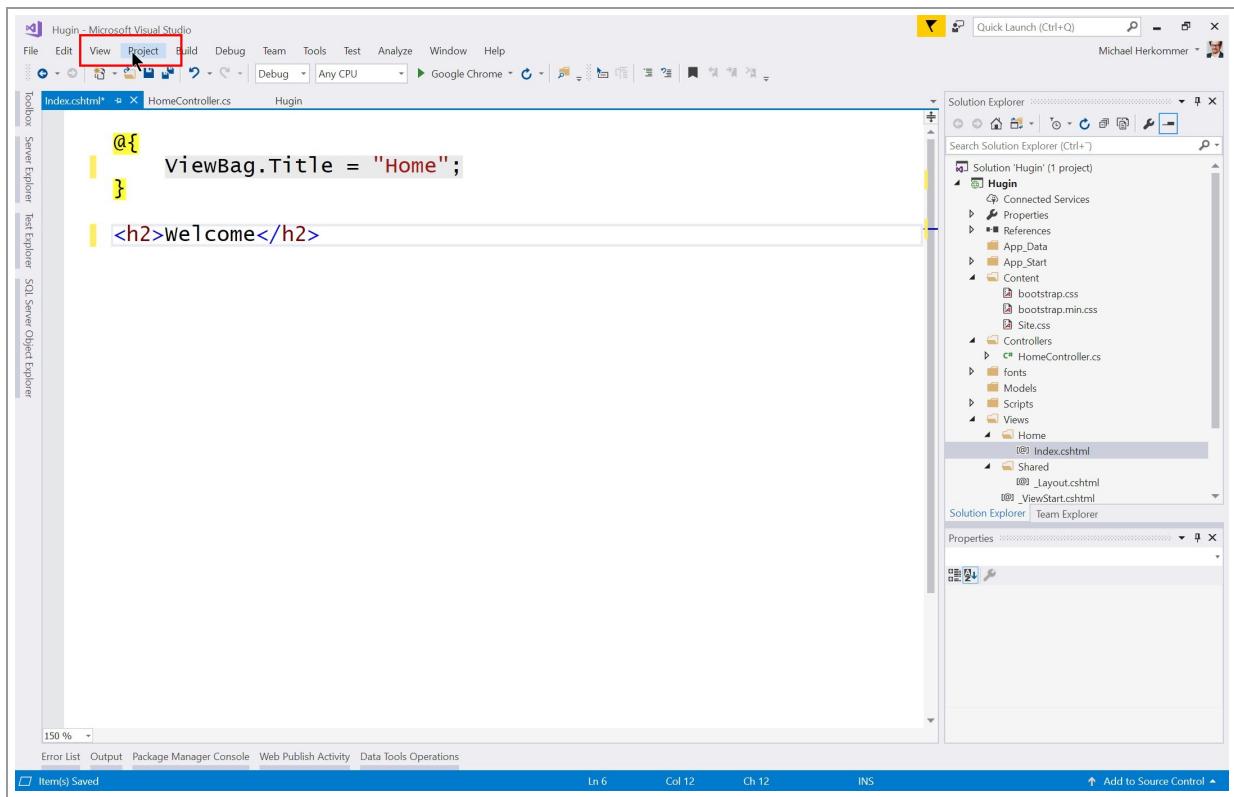
Click on "Add" button in "Add View"

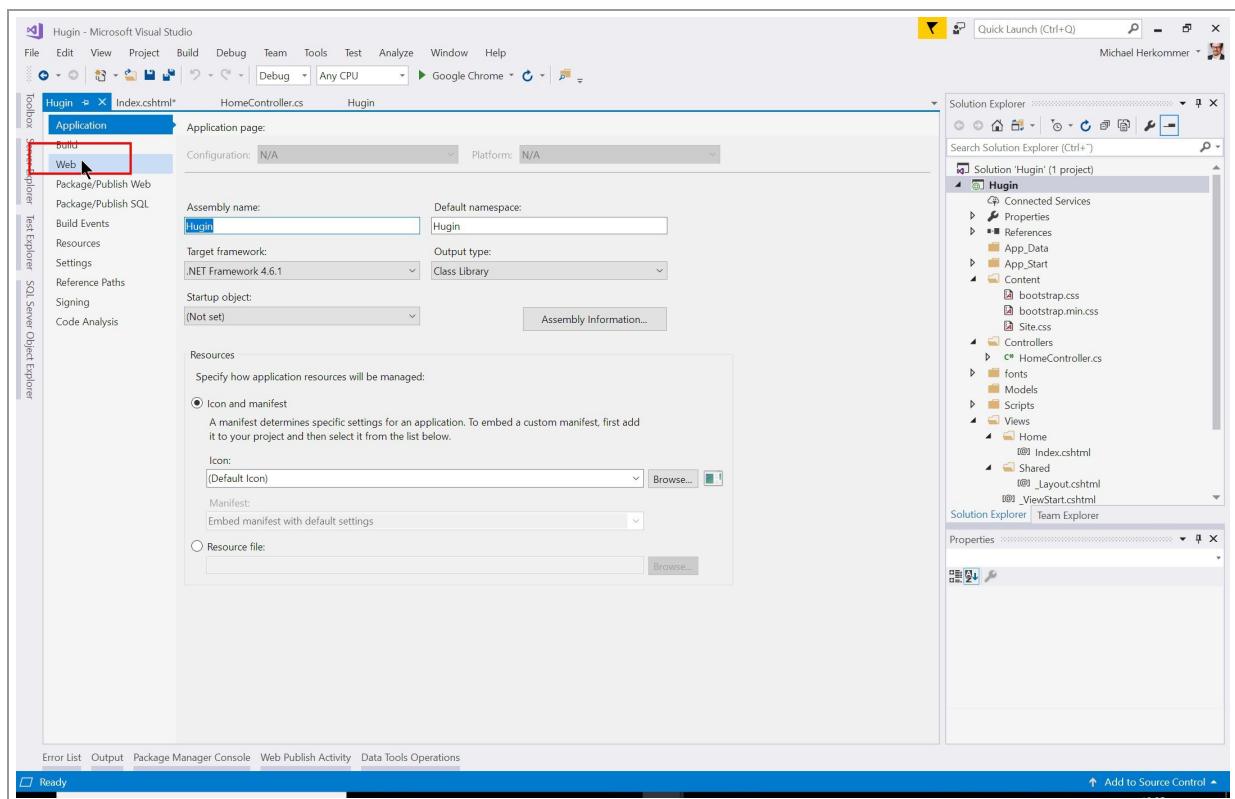
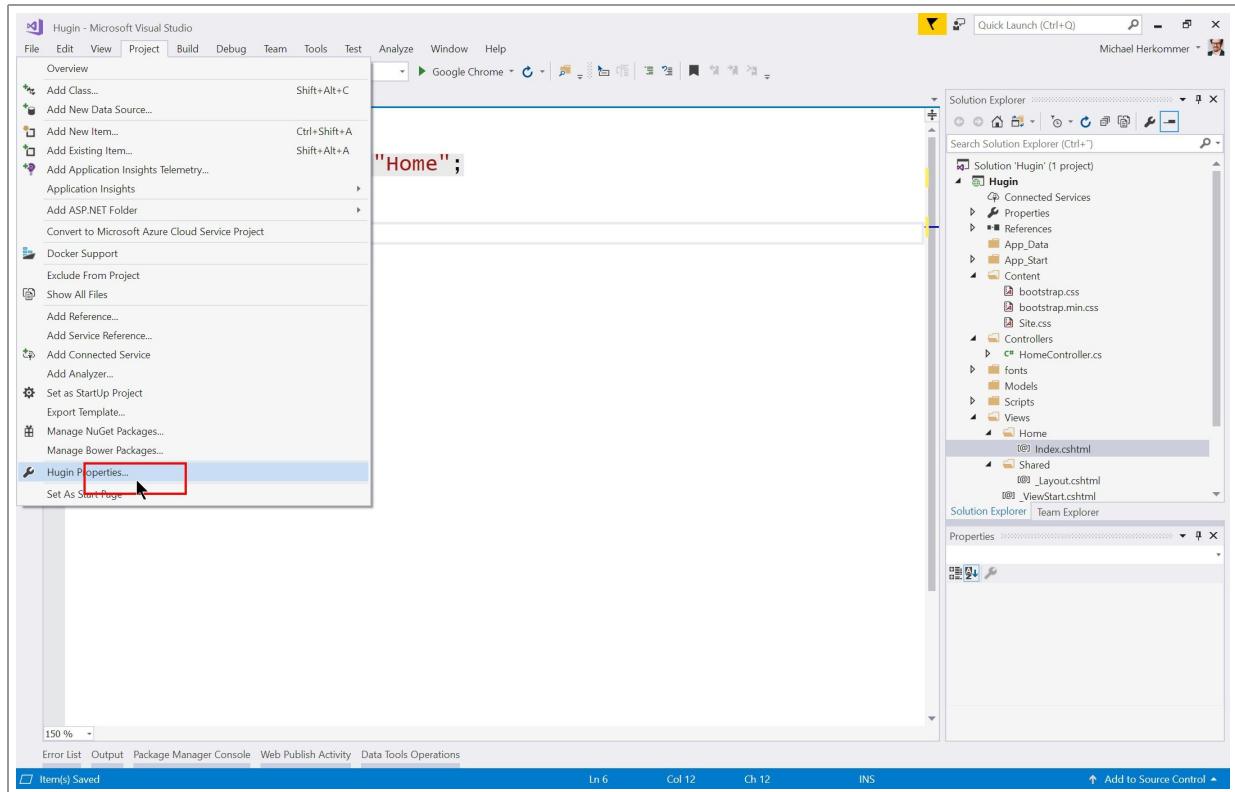


Click on "Text Editor" edit in "Index.cshtml"

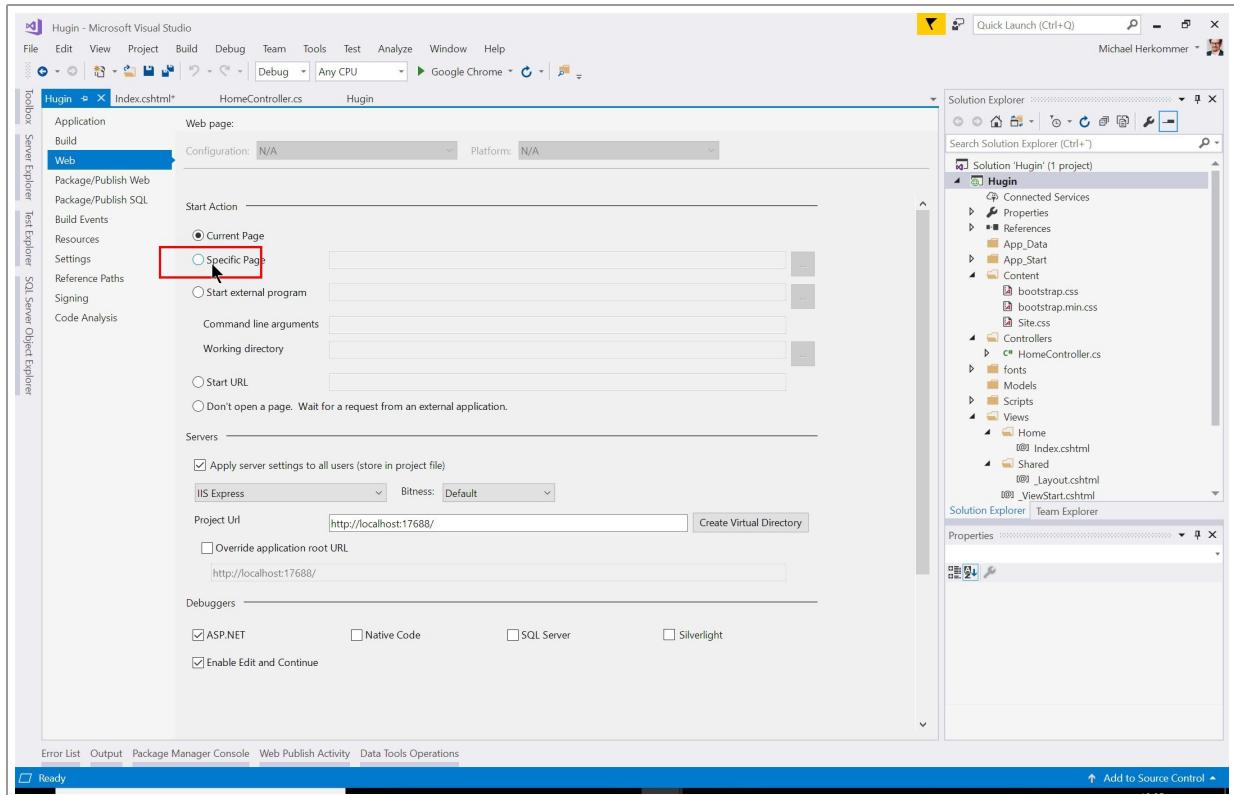


Click on "Text Editor" edit in "Index.cshtml"

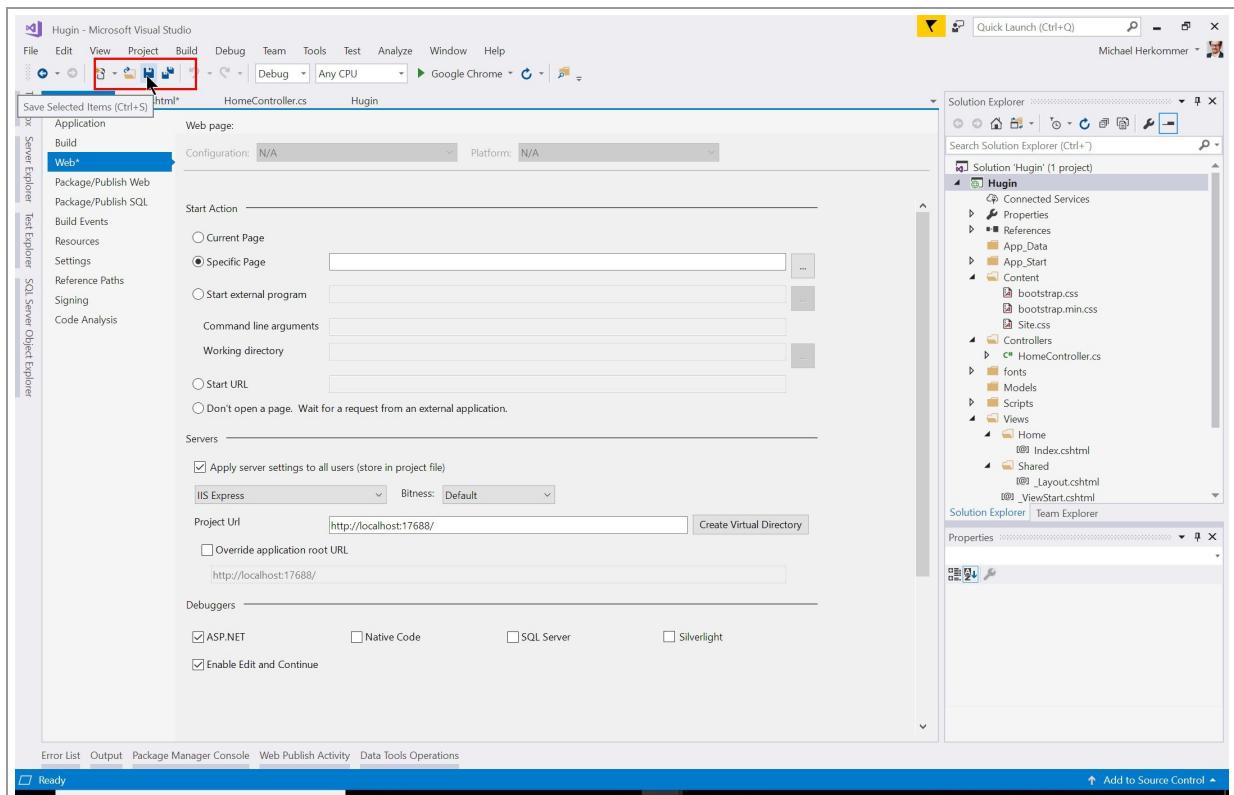




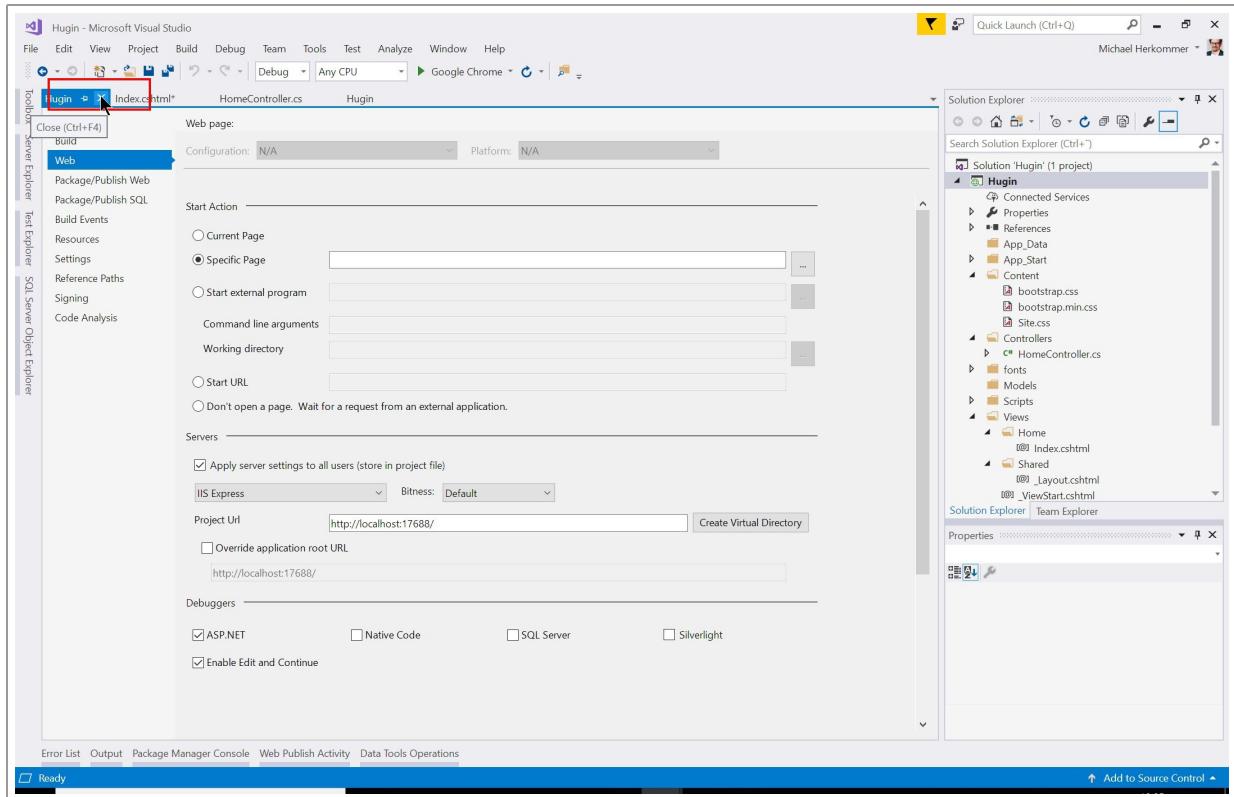
Click on "Web" pane in "Hugin"



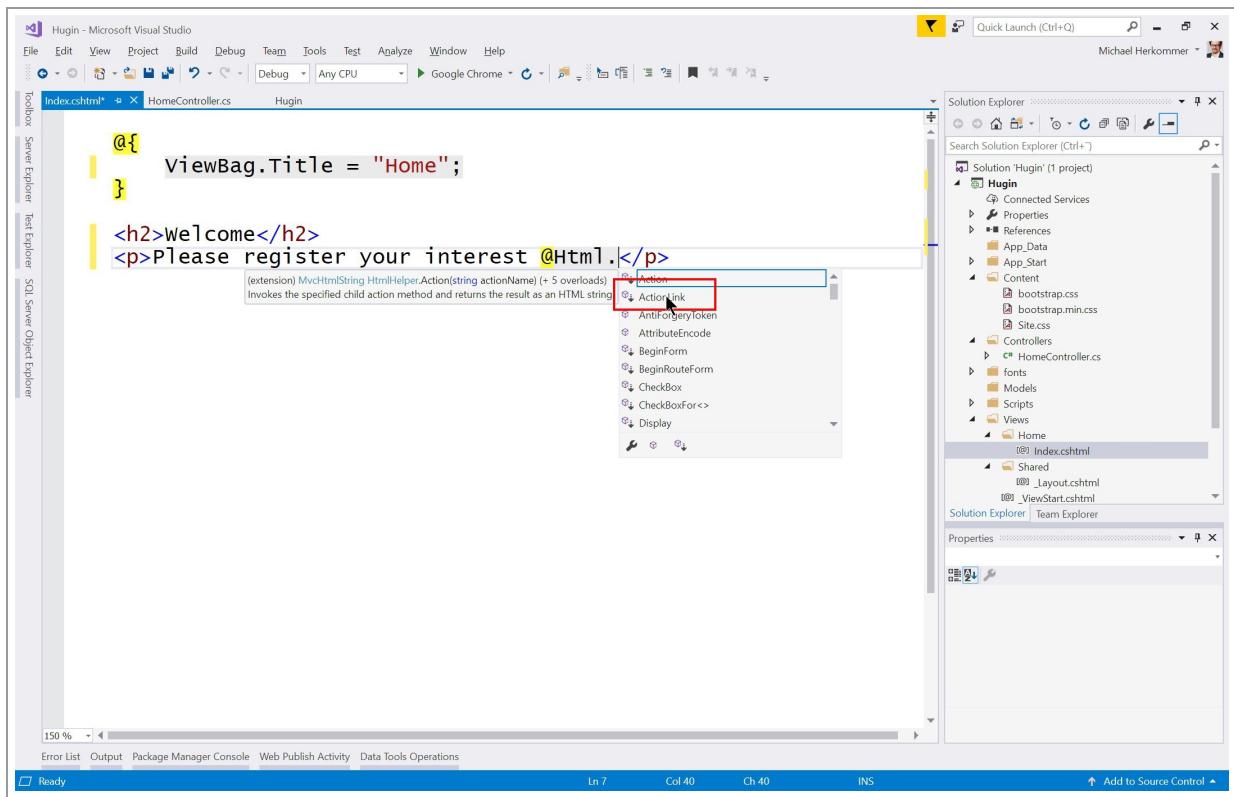
Choose "Specific Page" option in "Hugin"

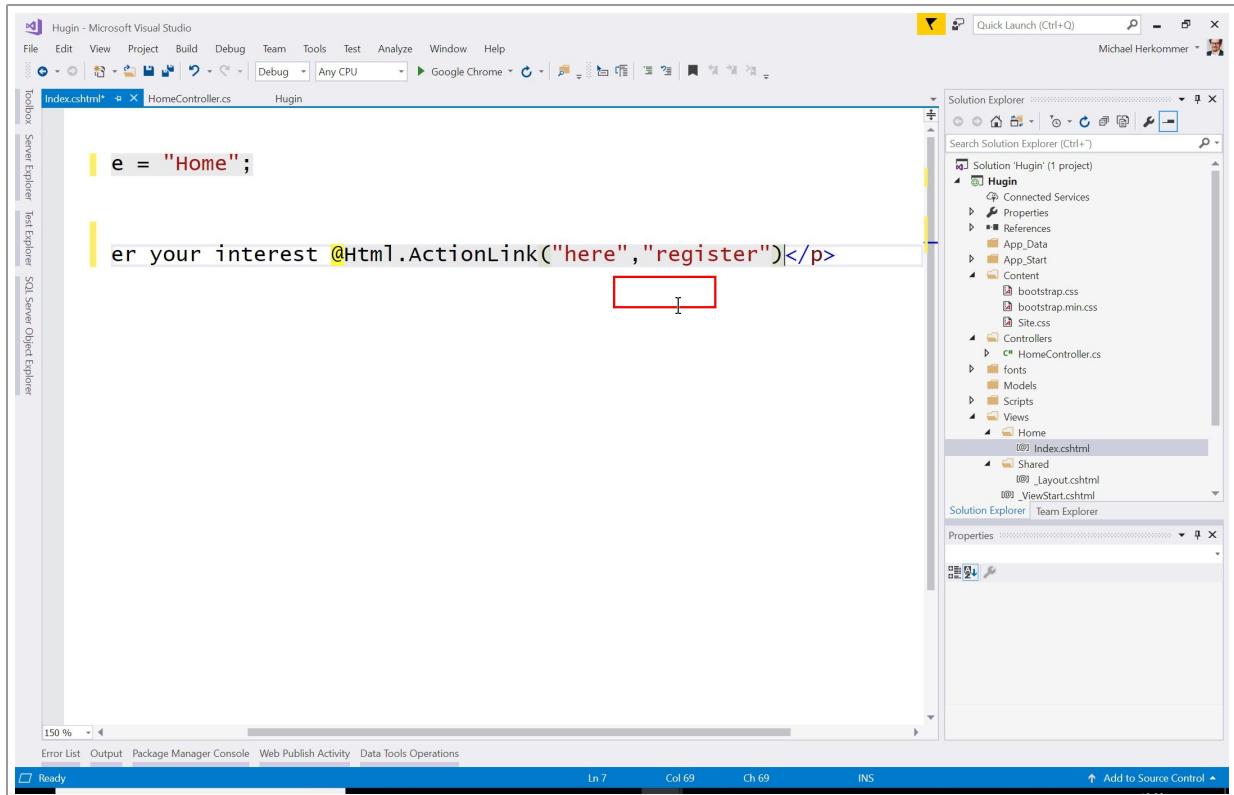


Click on "Save Selected Items" button in "Hugin - Microsoft Visual Studio "

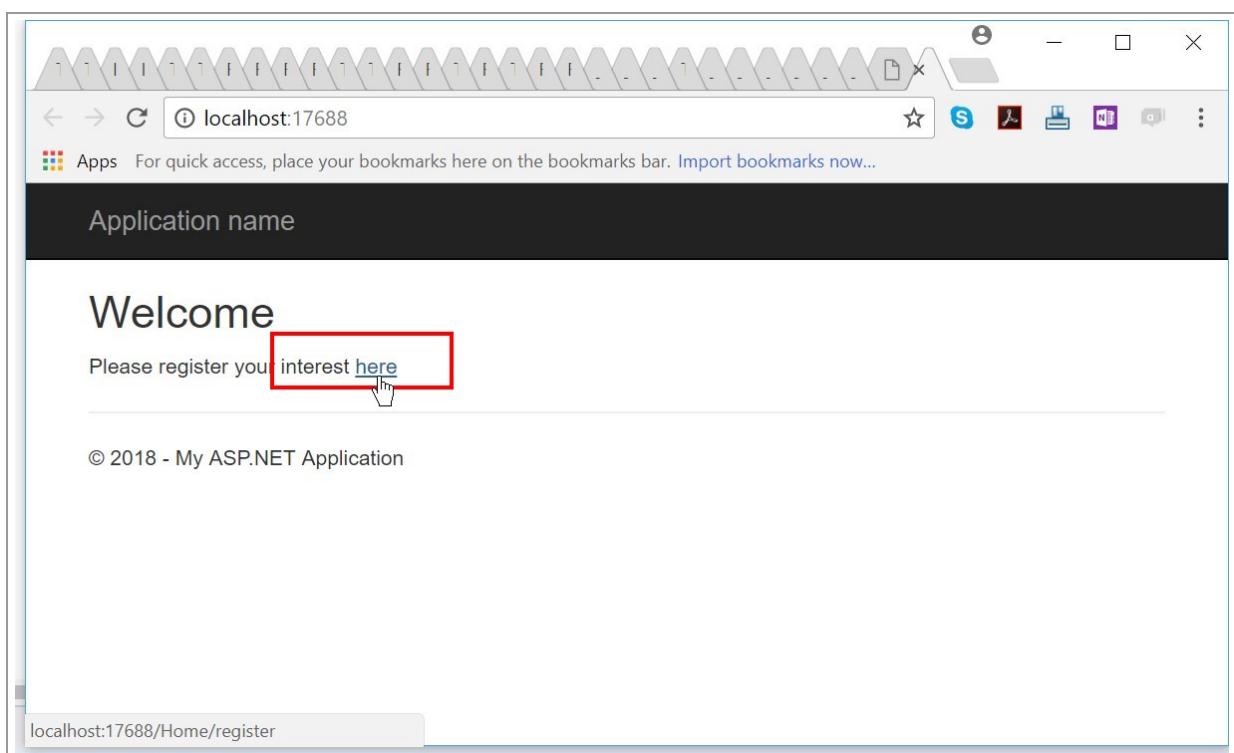


Click on "Close (Ctrl+F4)" button in "Hugin"

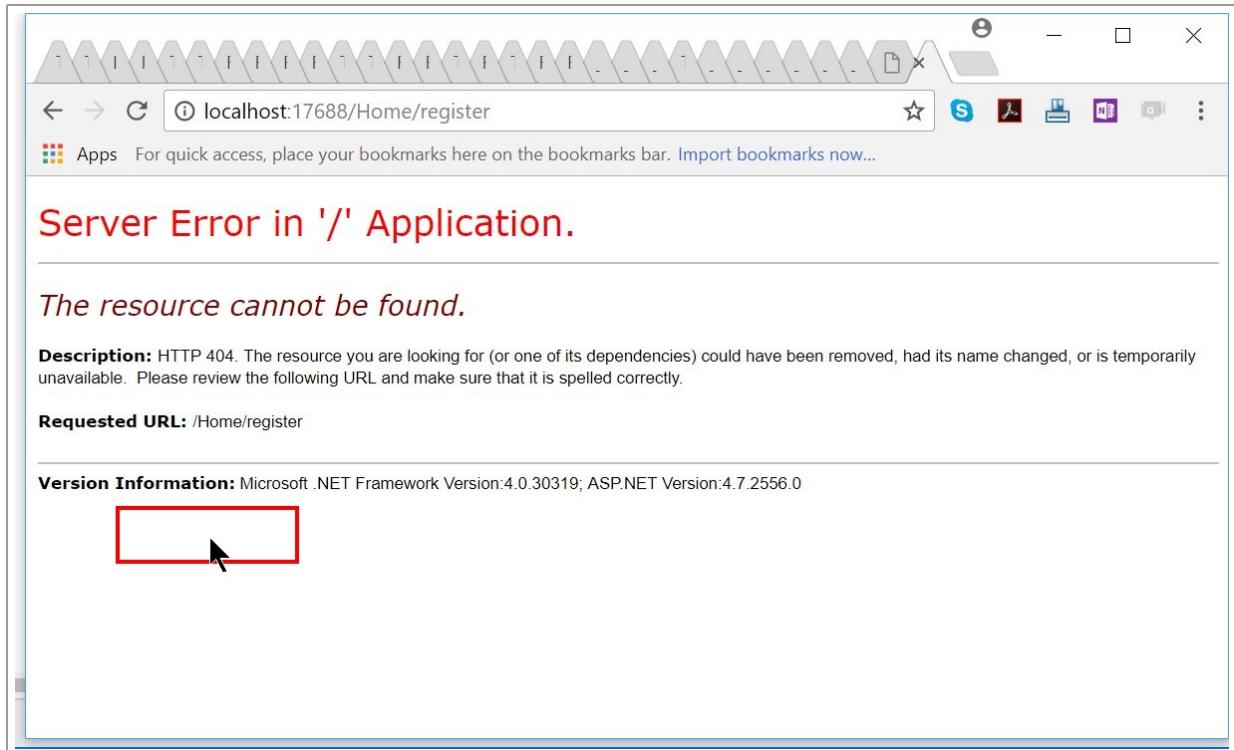




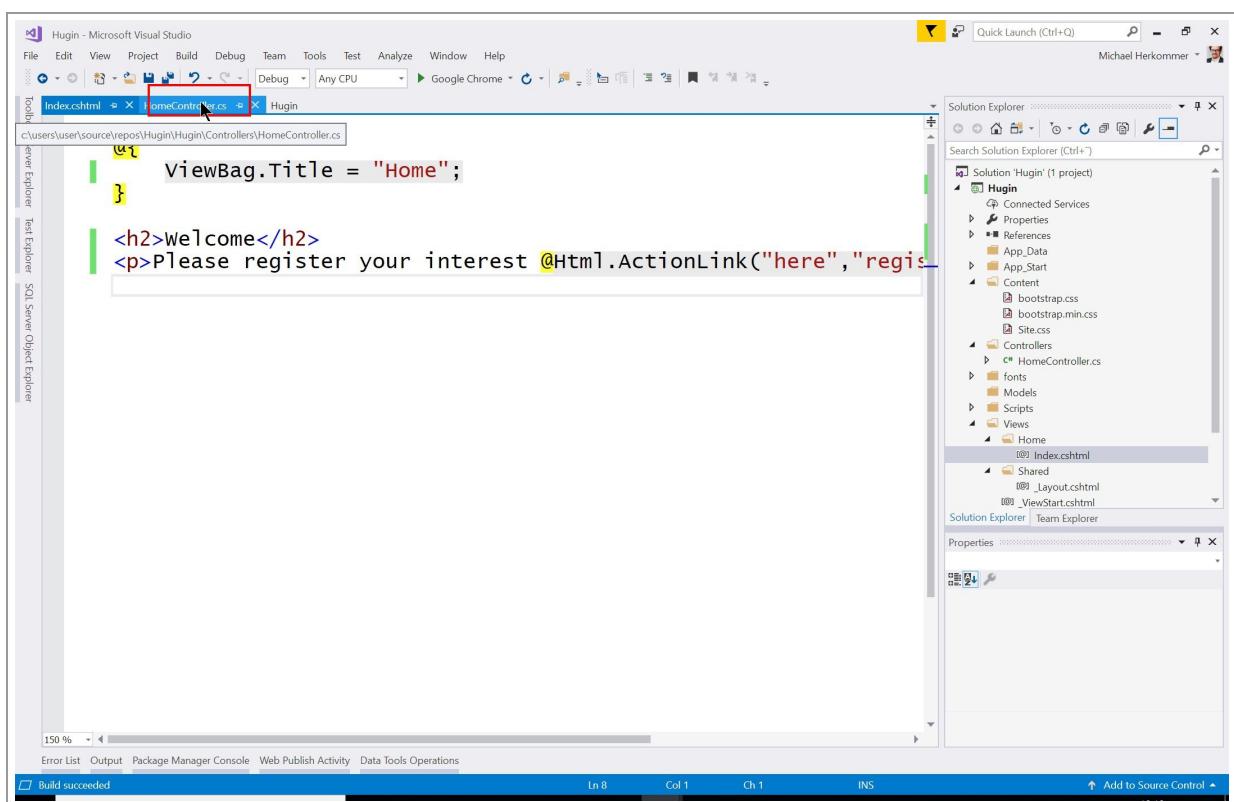
Click on "Text Editor" edit in "Index.cshtml"



Click on "Chrome Legacy Window" document in "Home - My ASP.NET Application"



Click on "Chrome Legacy Window" document in "The resource cannot be found."



Click on "HomeController.cs" text in "HomeController.cs"

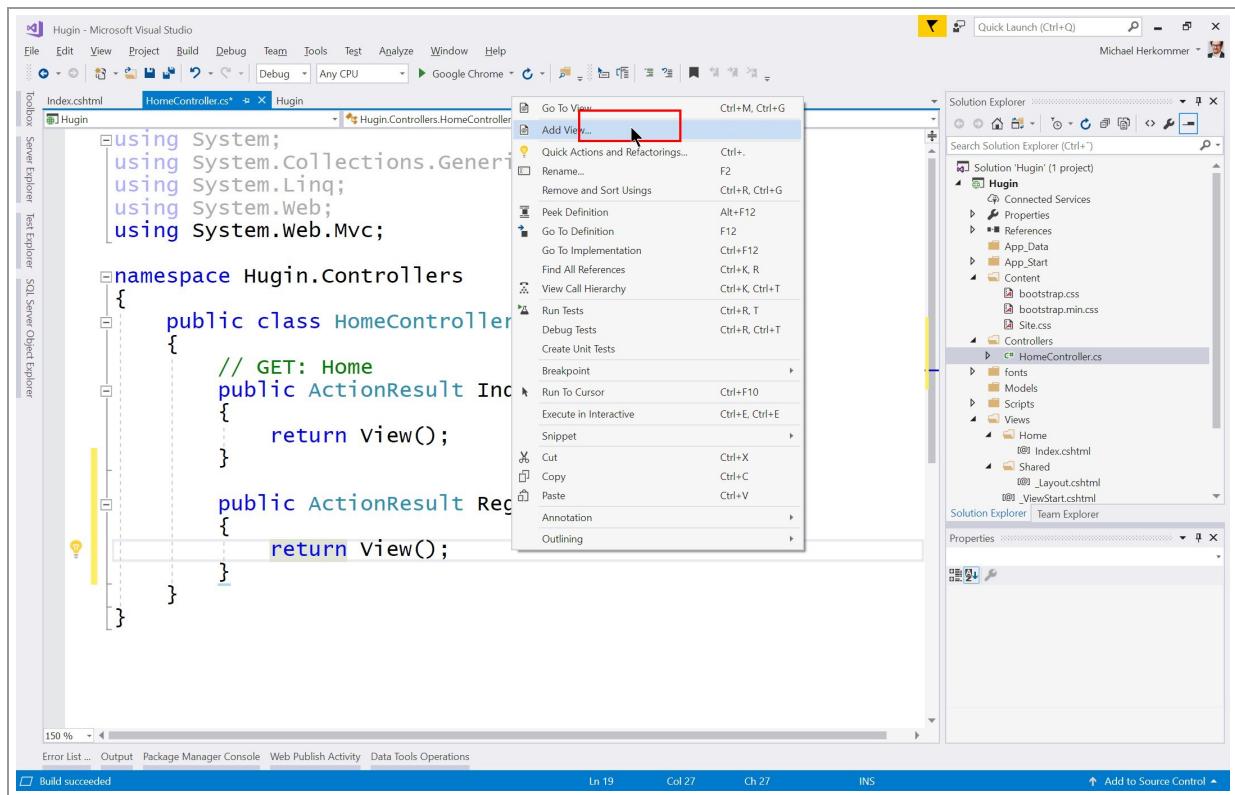
A screenshot of Microsoft Visual Studio showing the code editor for the `HomeController.cs` file. The code defines two actions: `Index()` and `Register()`, both returning `View()`. The `Index()` action is annotated with `// GET: Home`. The `Register()` action is currently selected. The Solution Explorer on the right shows the project structure for the `Hugin` solution.

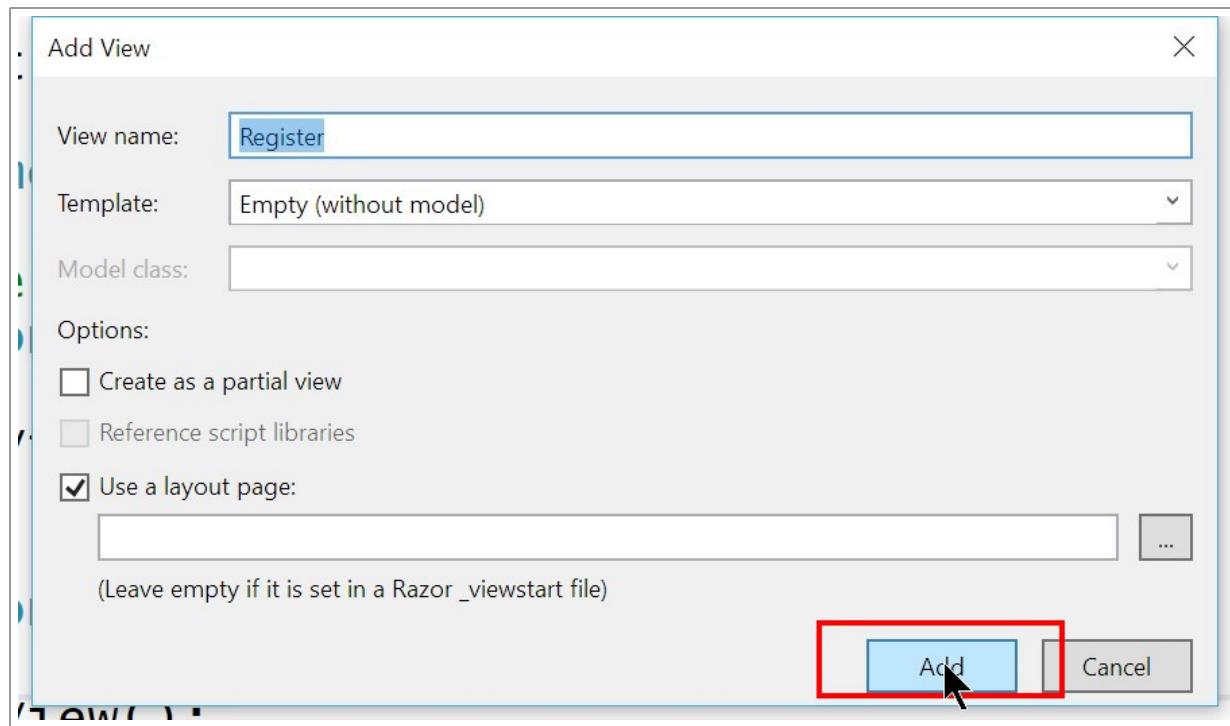
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;

namespace Hugin.Controllers
{
    public class HomeController : Controller
    {
        // GET: Home
        public ActionResult Index()
        {
            return View();
        }

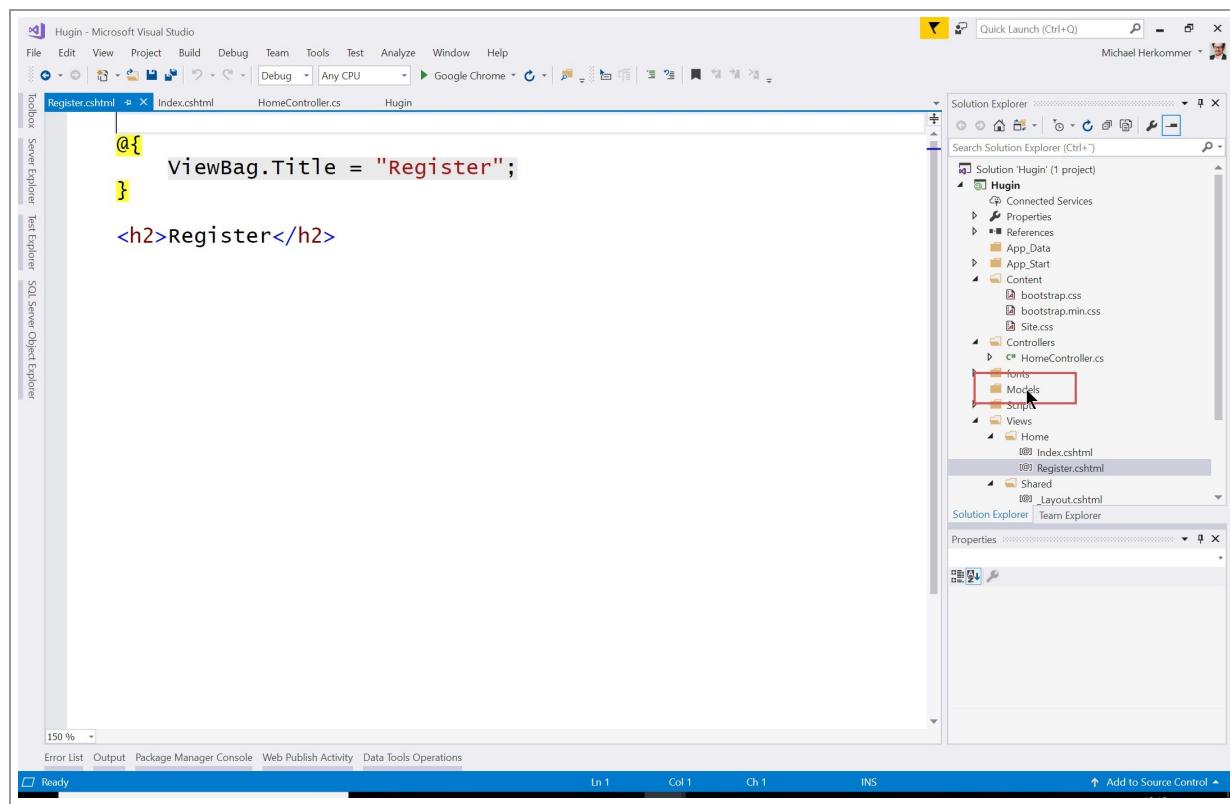
        public ActionResult Register()
        {
            return View();
        }
    }
}
```

Right click on "Text Editor" edit in "HomeController.cs"

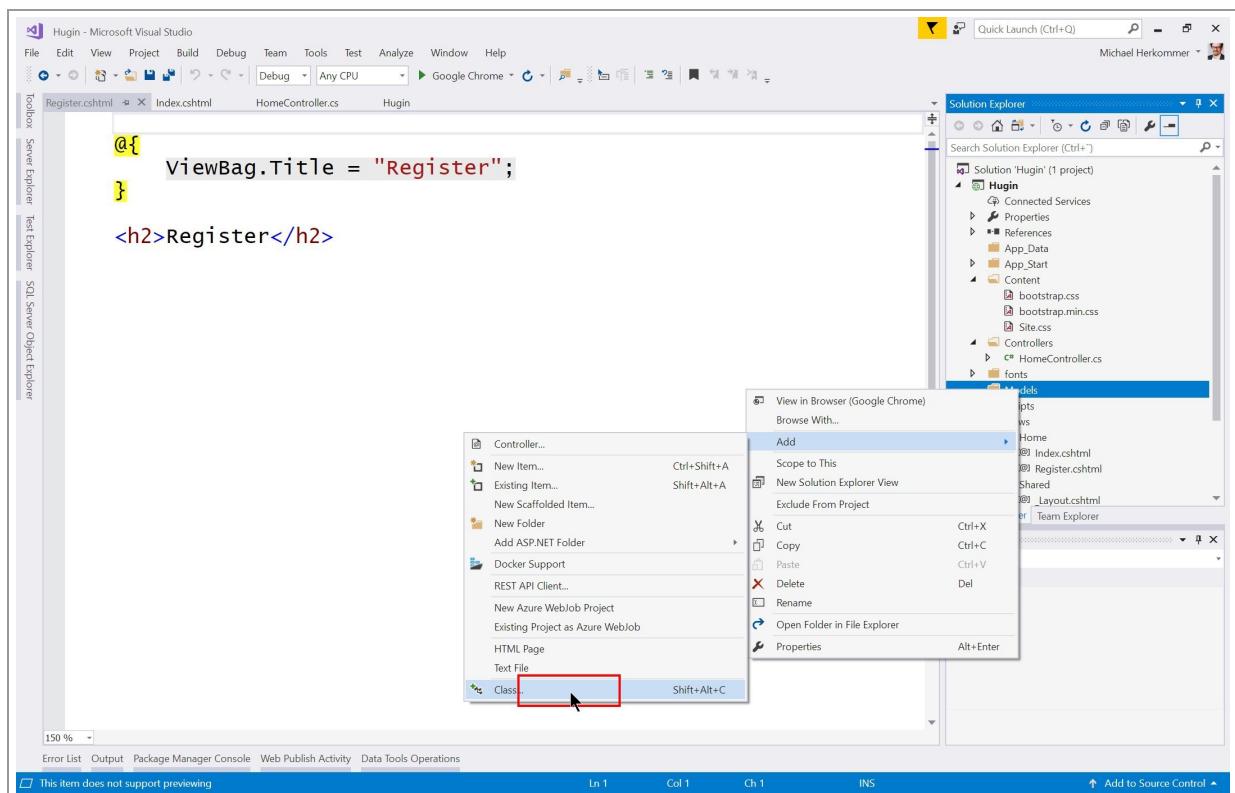
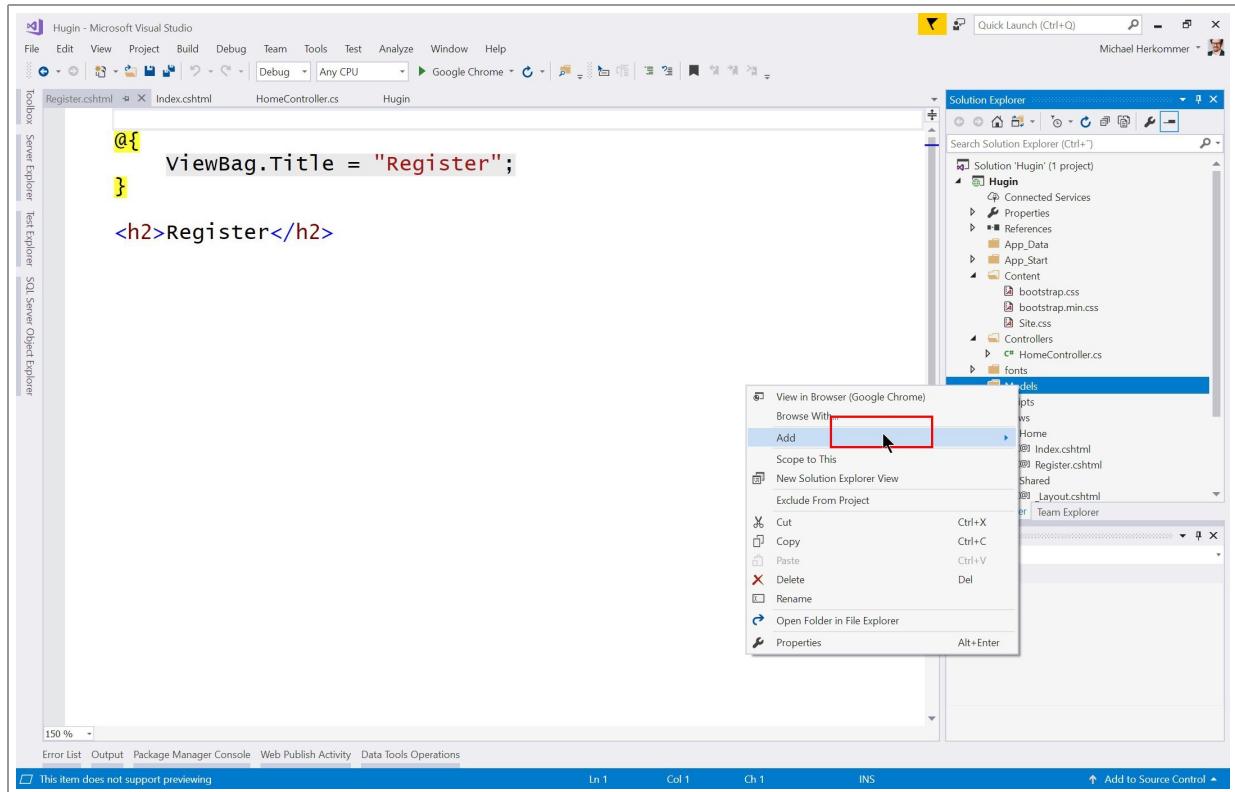


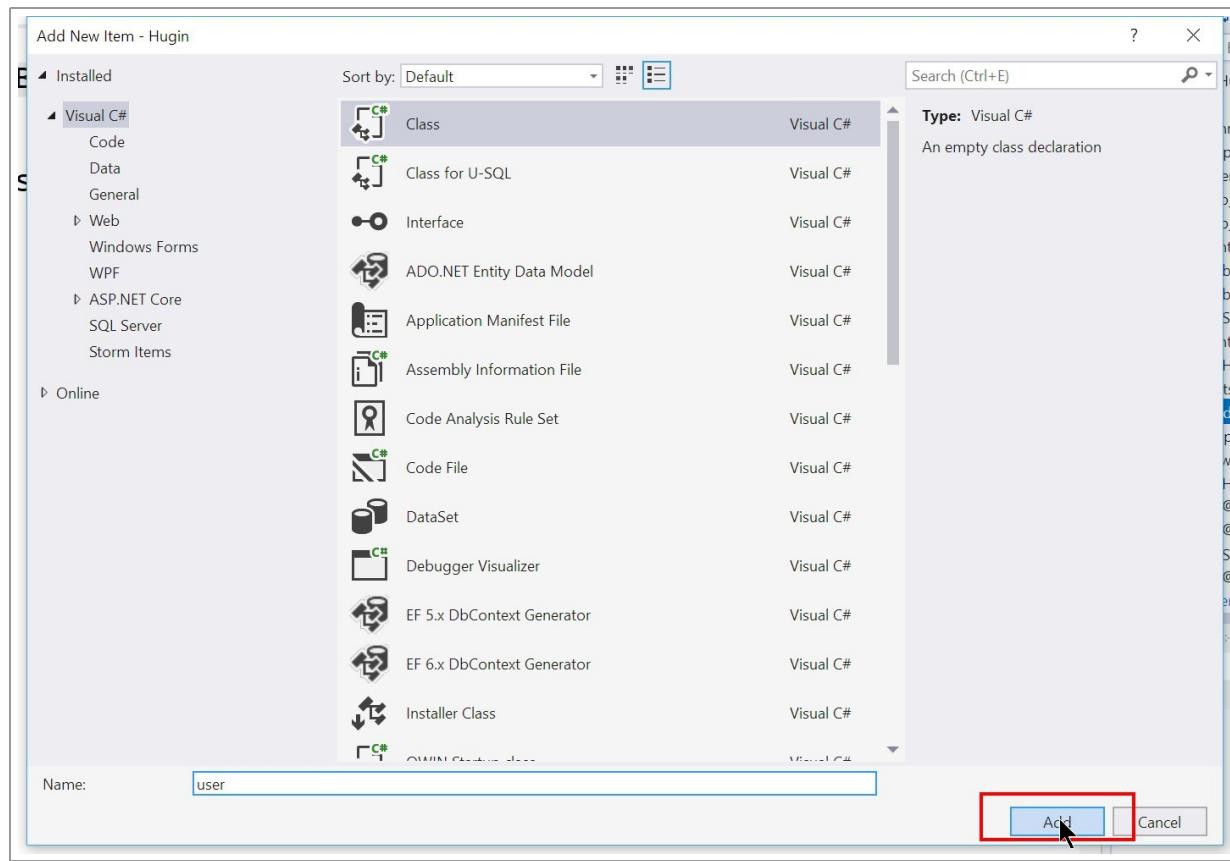


Click on "Add" text in "Add View"

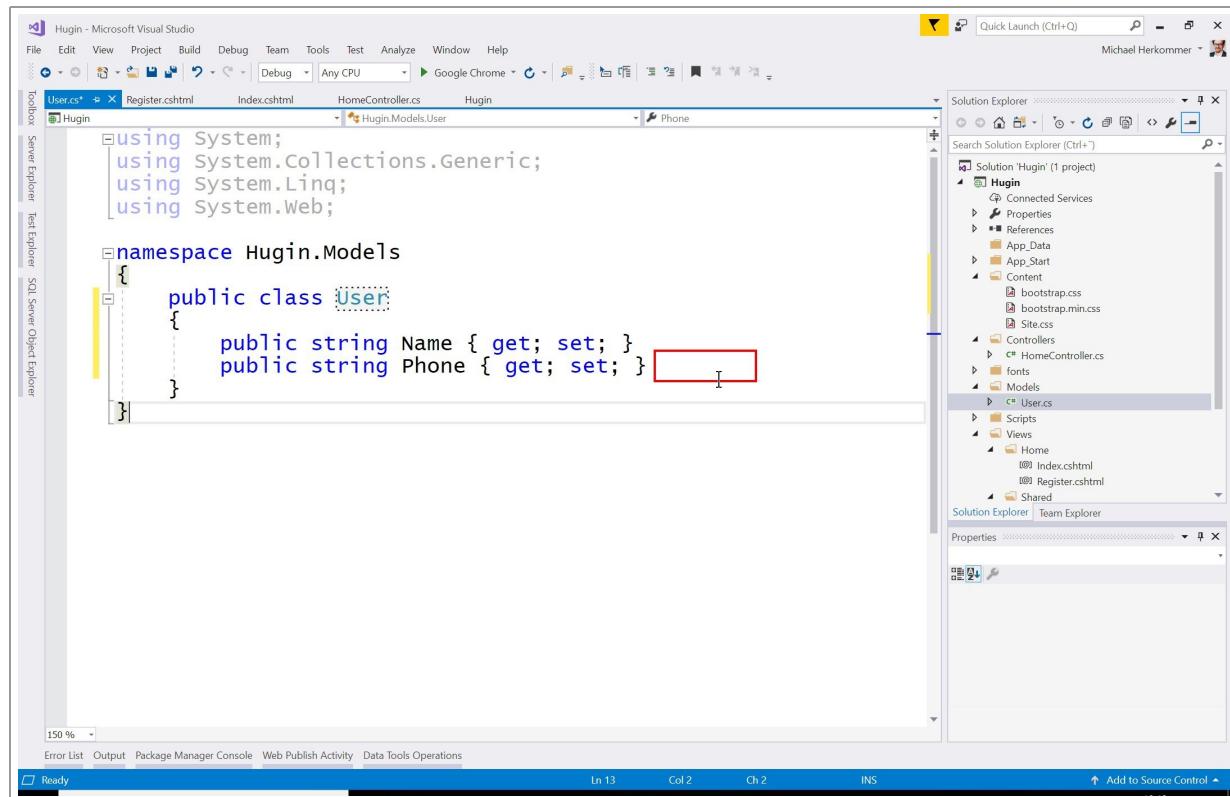


Right click on "Models" tree view item in "Solution Explorer"

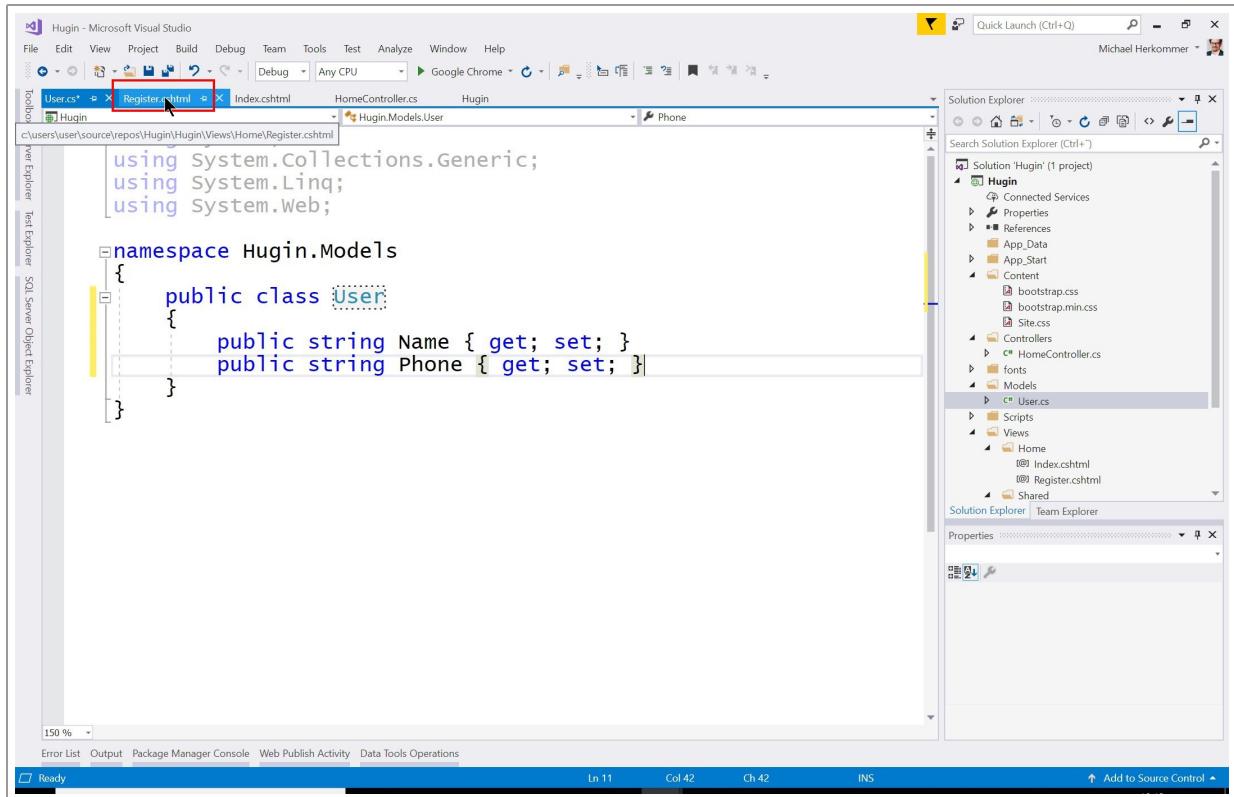




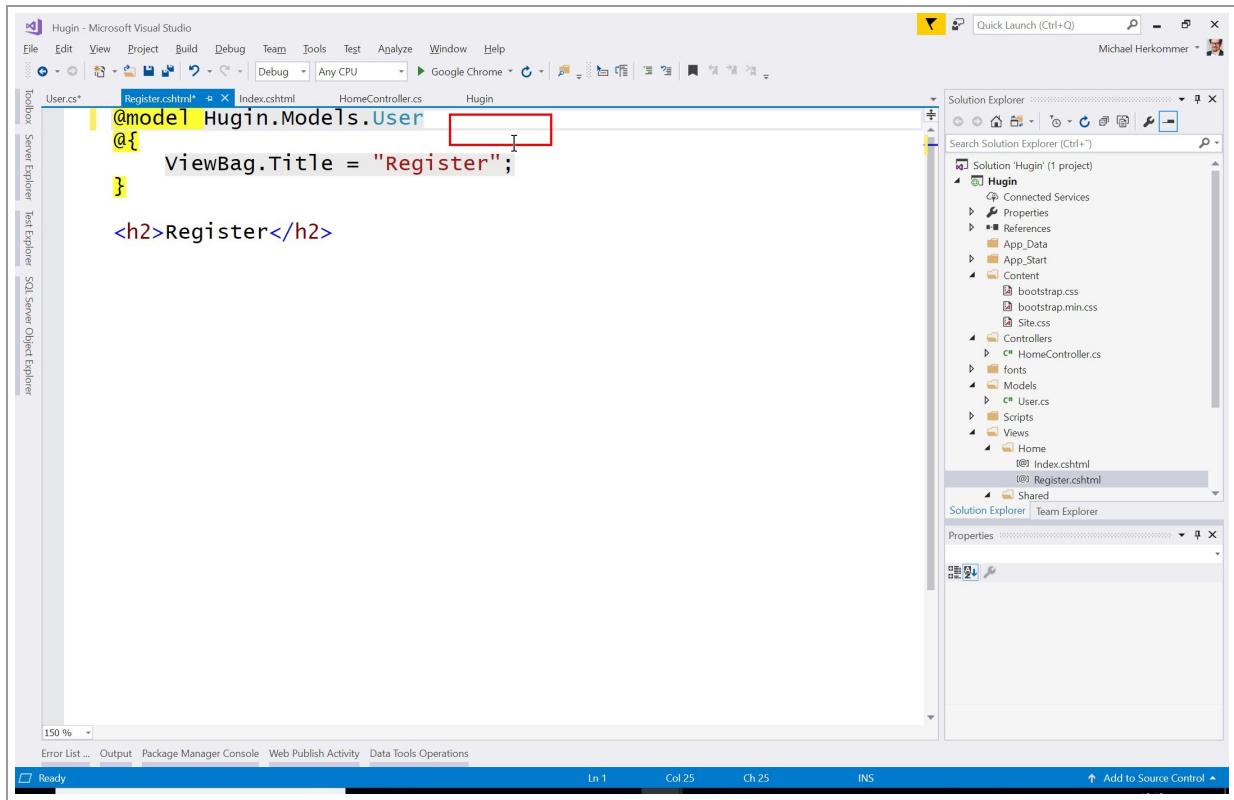
*****NOTE> Add User class and spell it with a CAPITAL U



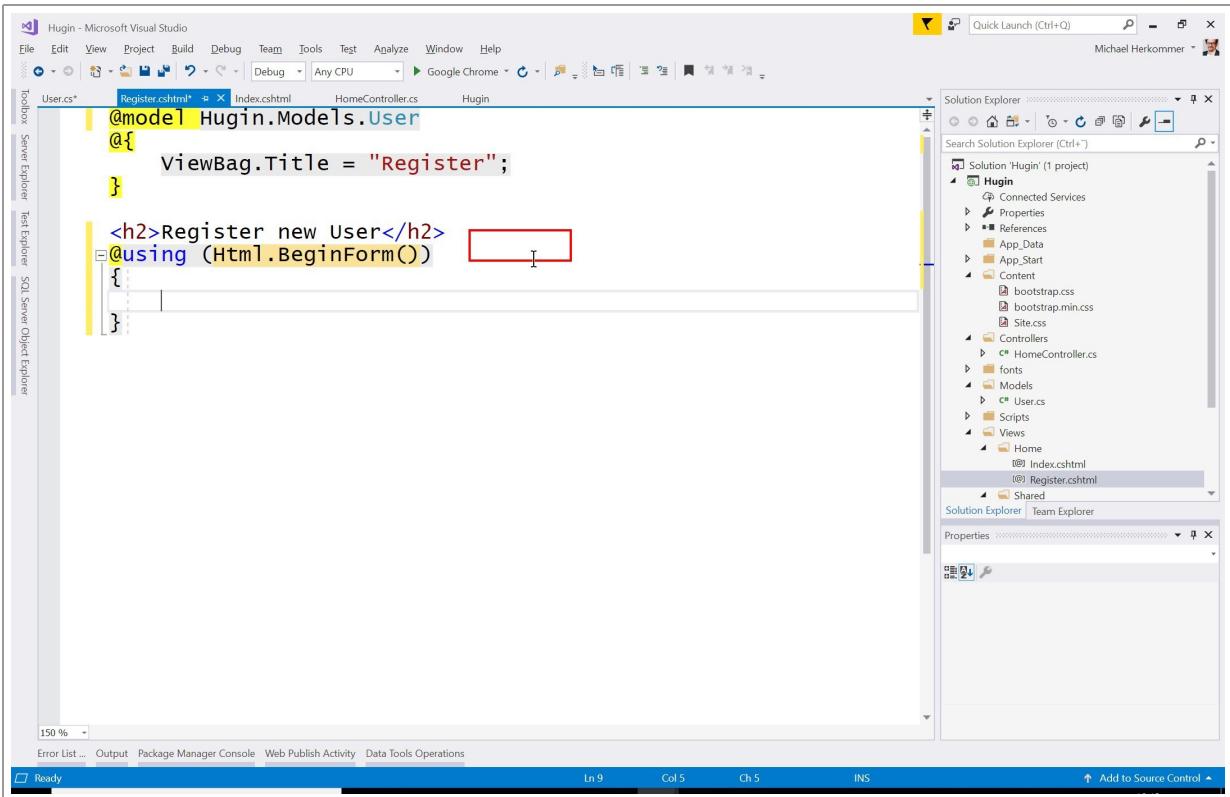
Click on "Text Editor" edit in "User.cs"



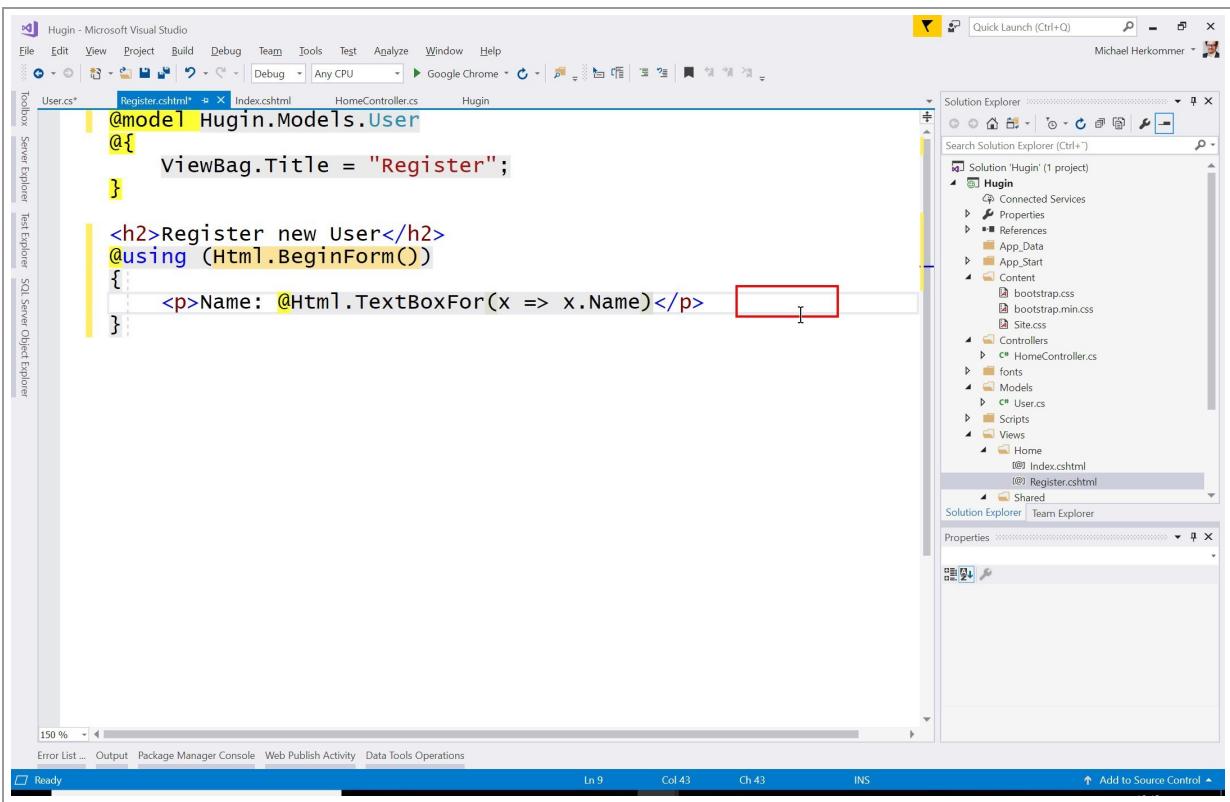
Click on "Register.cshtml" text in "Register.cshtml"



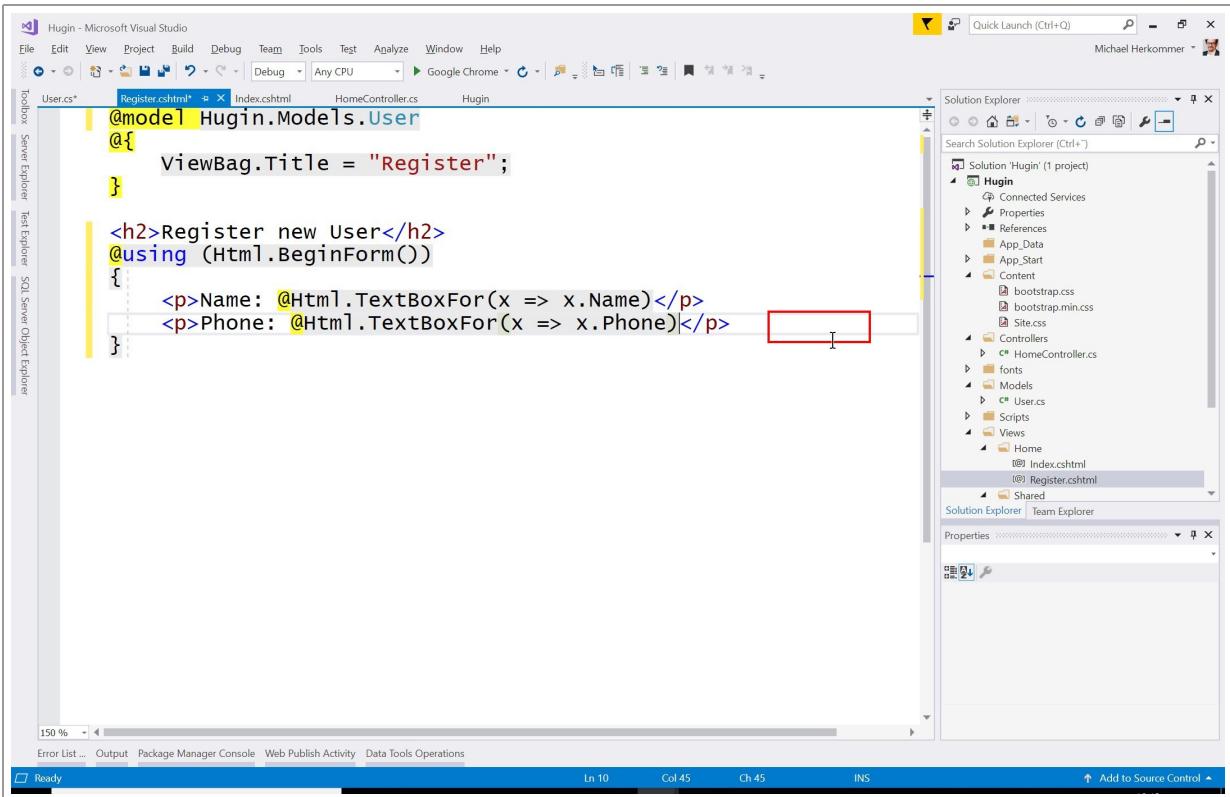
Click on "Text Editor" edit in "Register.cshtml*"



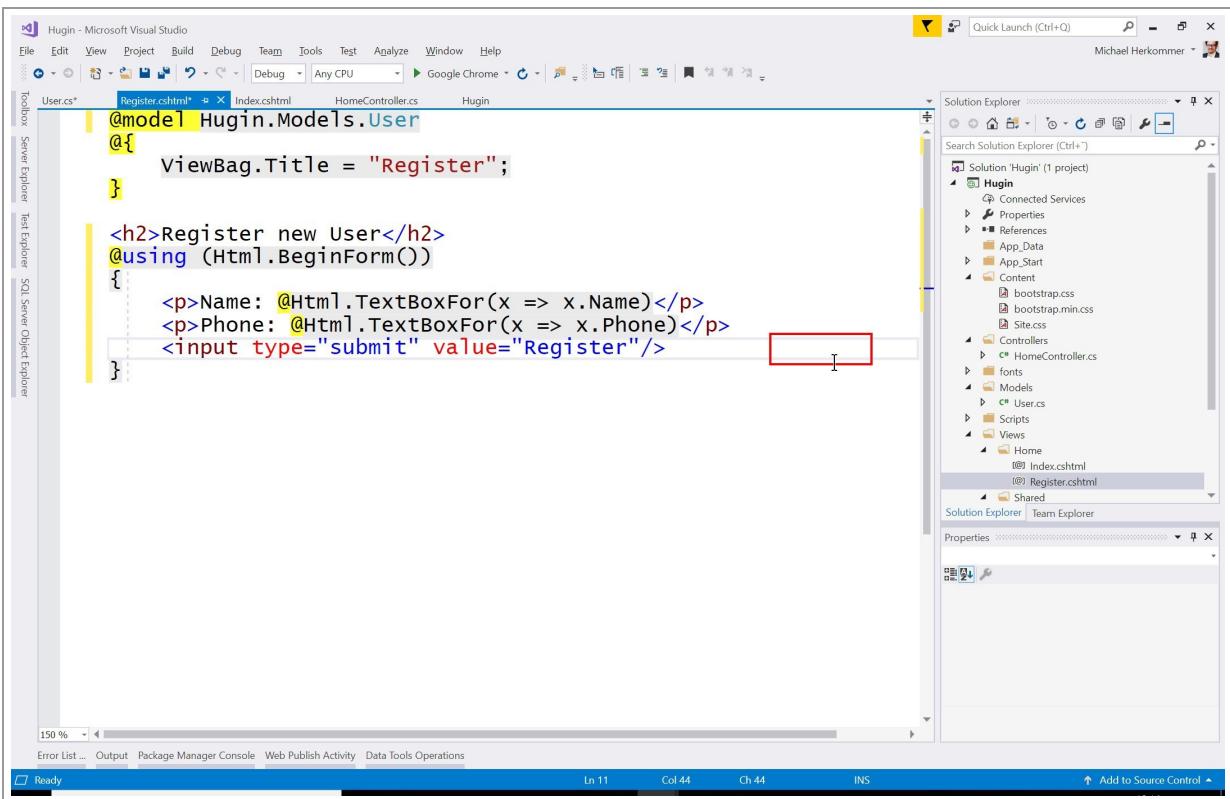
Click on "Text Editor" edit in "Register.cshtml*"



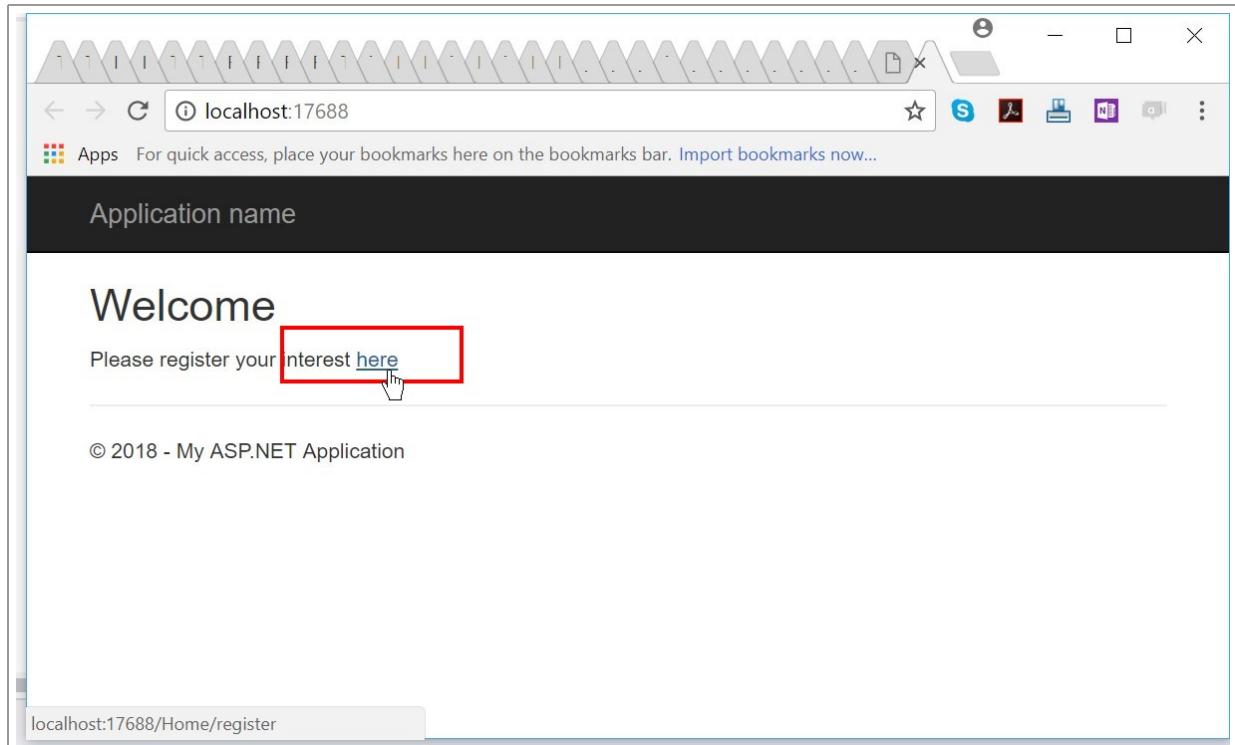
Click on "Text Editor" edit in "Register.cshtml*"



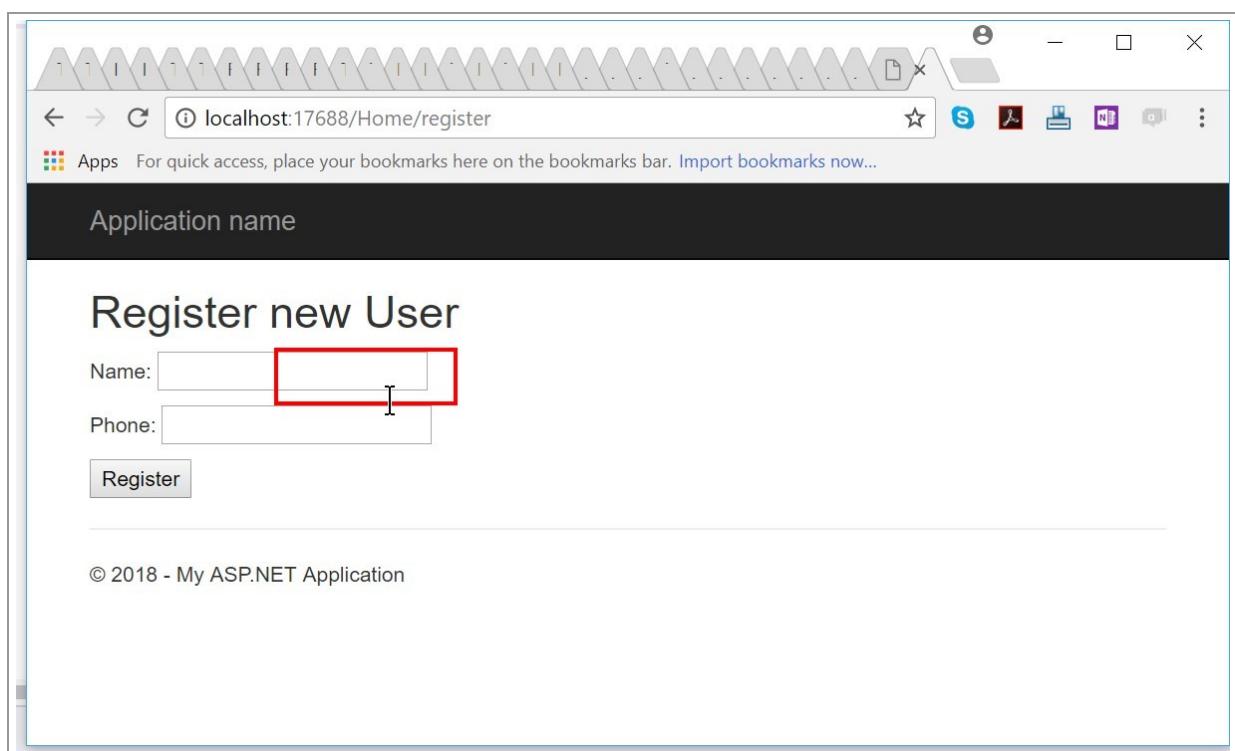
Click on "Text Editor" edit in "Register.cshtml*"



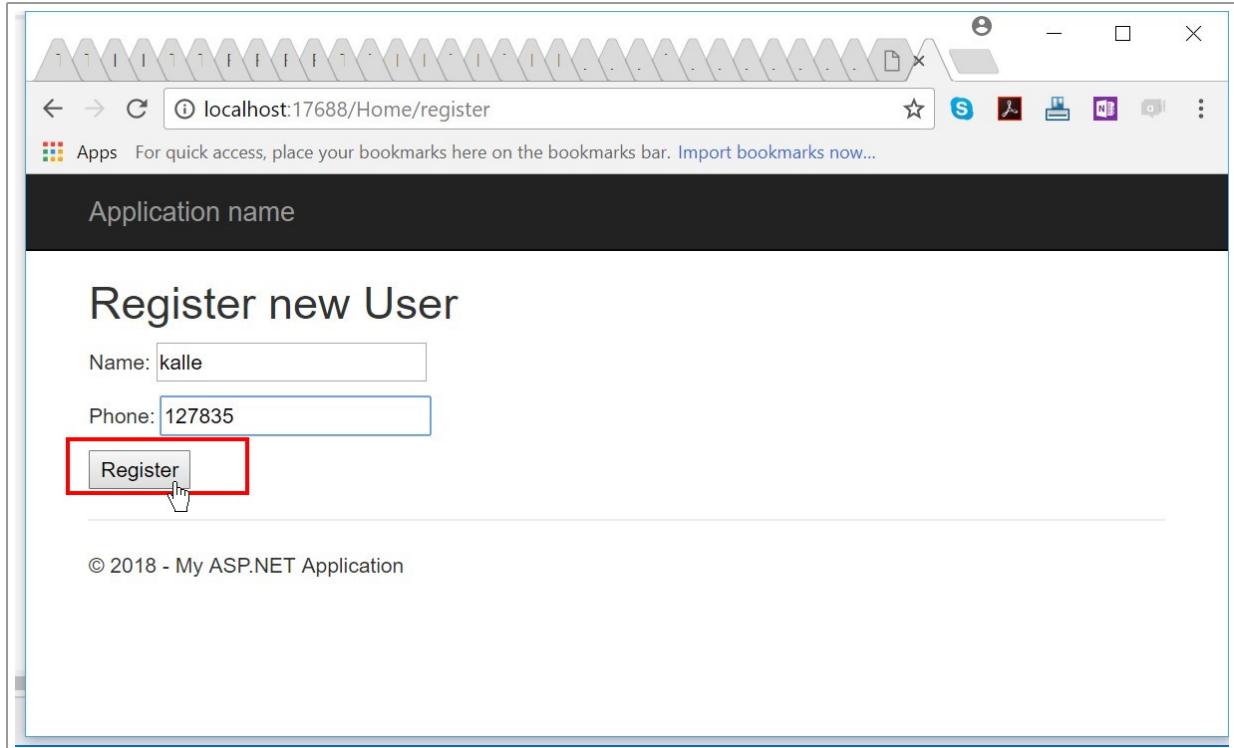
Click on "Text Editor" edit in "Register.cshtml*"



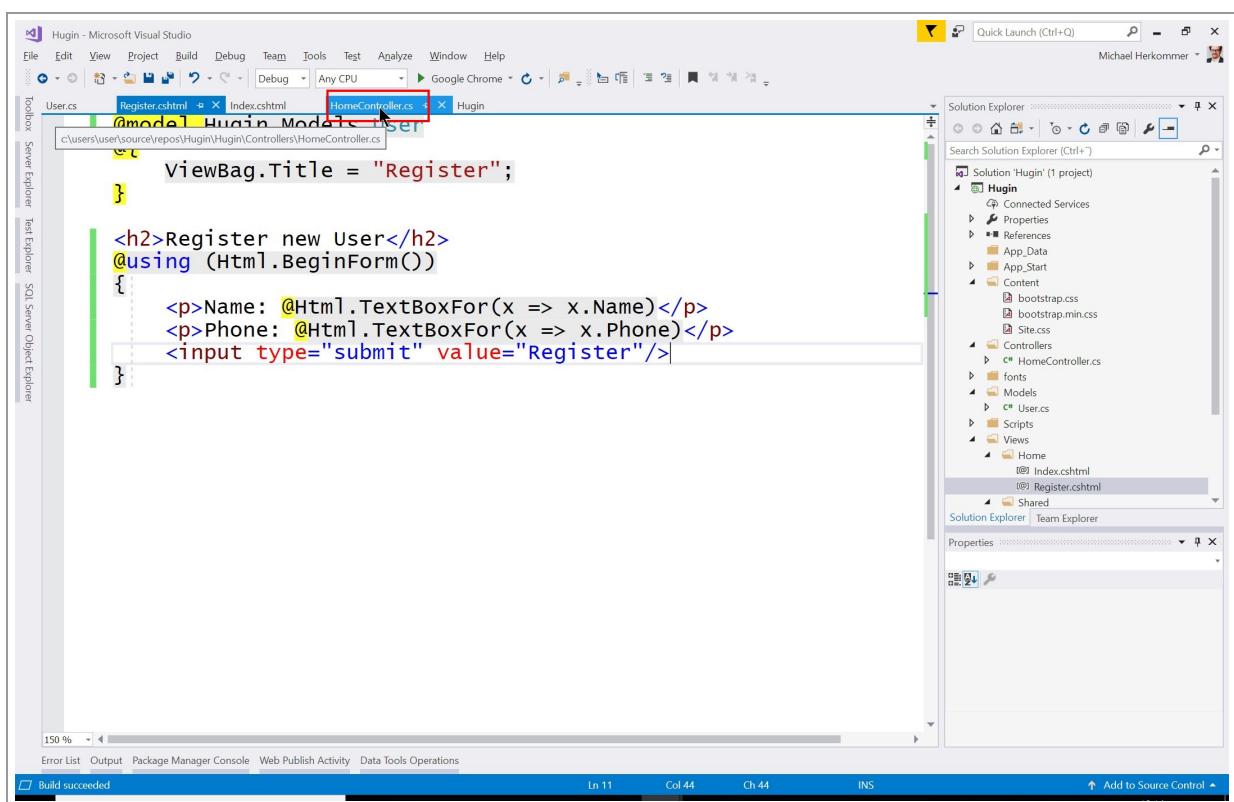
Click on "Chrome Legacy Window" document in "Home - My ASP.NET Application"



Click on "Chrome Legacy Window" document in "Register - My ASP.NET Application"



Click on "Chrome Legacy Window" document in "Register - My ASP.NET Application"



Click on "HomeController.cs" text in "HomeController.cs"

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;

namespace Hugin.Controllers
{
    public class HomeController : Controller
    {
        // GET: Home
        public ActionResult Index()
        {
            return View();
        }

        [HttpPost]
        public ActionResult Register()
        {
            return View();
        }
    }
}
```

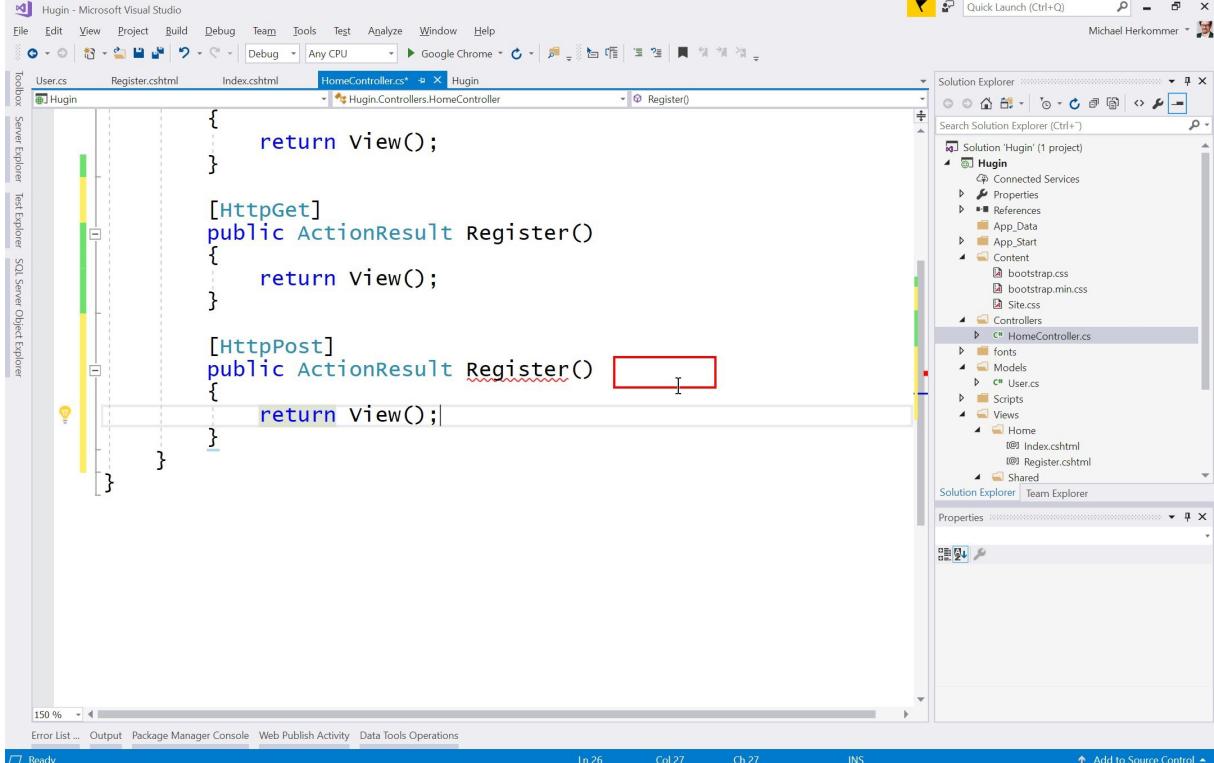
Click on "Text Editor" edit in "HomeController.cs"

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;

namespace Hugin.Controllers
{
    public class HomeController : Controller
    {
        // GET: Home
        public ActionResult Index()
        {
            return View();
        }

        [HttpPost]
        public ActionResult Register()
        {
            return View();
        }
    }
}
```

Click on "Text Editor" edit in "HomeController.cs*"



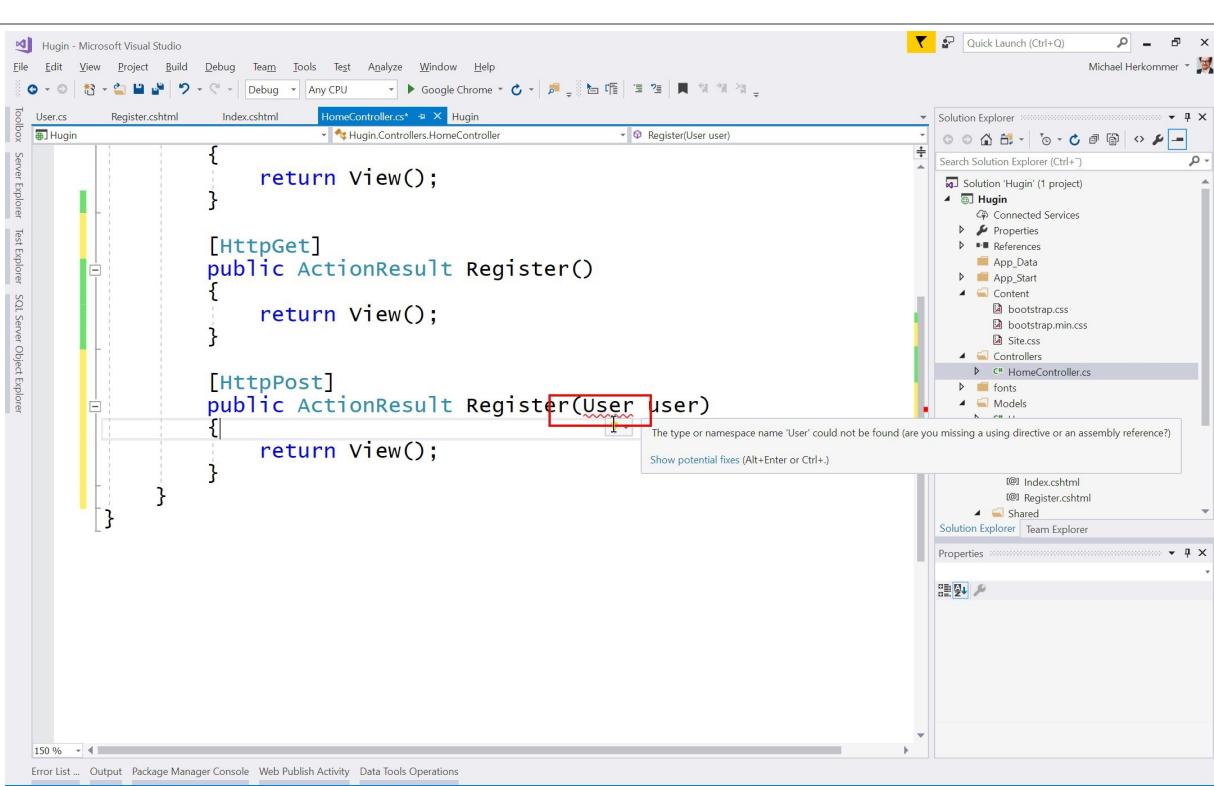
Click on "Text Editor" edit in "HomeController.cs"

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using Hugin.Models;

namespace Hugin.Controllers
{
    public class HomeController : Controller
    {
        public ActionResult Index()
        {
            return View();
        }

        [HttpGet]
        public ActionResult Register()
        {
            return View();
        }

        [HttpPost]
        public ActionResult Register()
        {
            return View();
        }
    }
}
```



Click on "Text Editor" edit in "HomeController.cs"

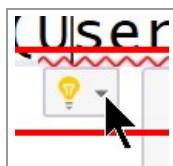
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using Hugin.Models;

namespace Hugin.Controllers
{
    public class HomeController : Controller
    {
        public ActionResult Index()
        {
            return View();
        }

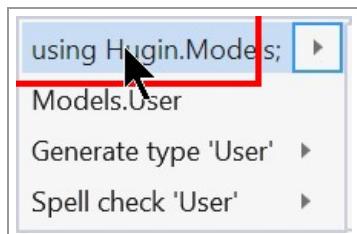
        [HttpGet]
        public ActionResult Register()
        {
            return View();
        }

        [HttpPost]
        public ActionResult Register(User user)
        {
            return View();
        }
    }
}
```

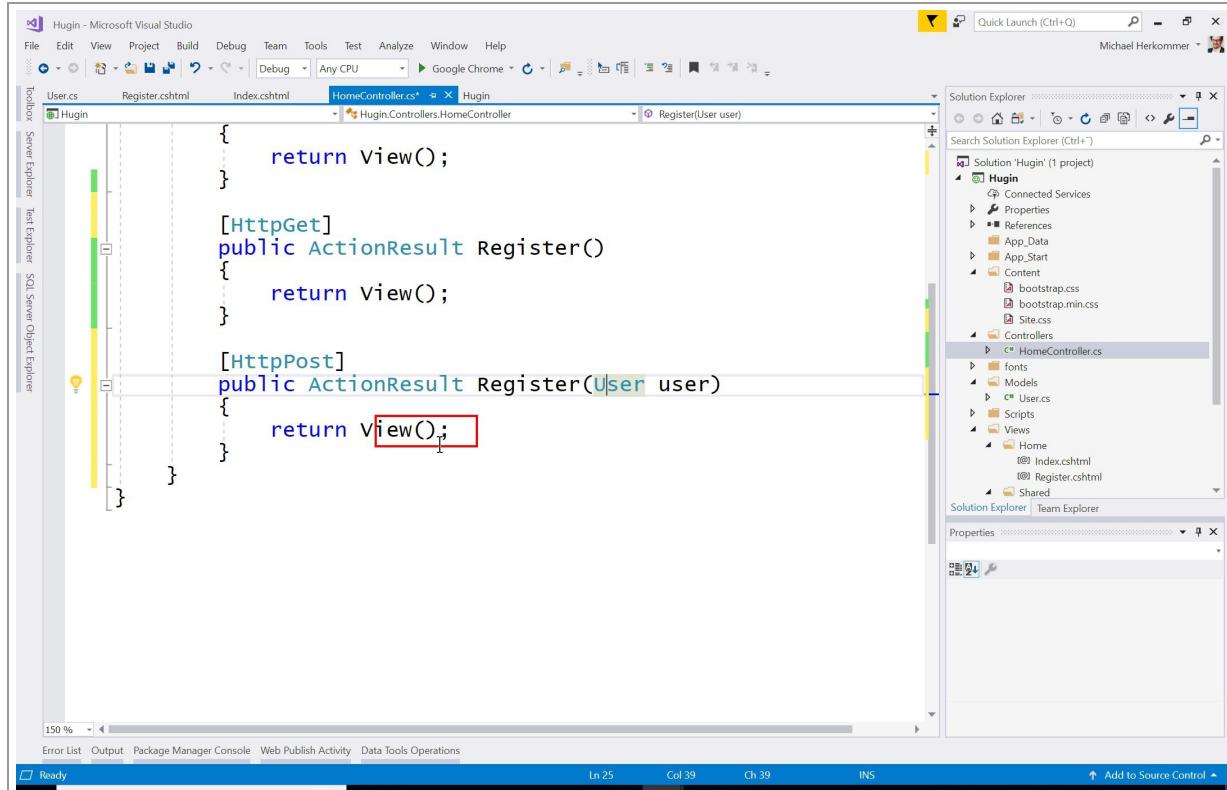
Click on "Text Editor" edit in "HomeController.cs"



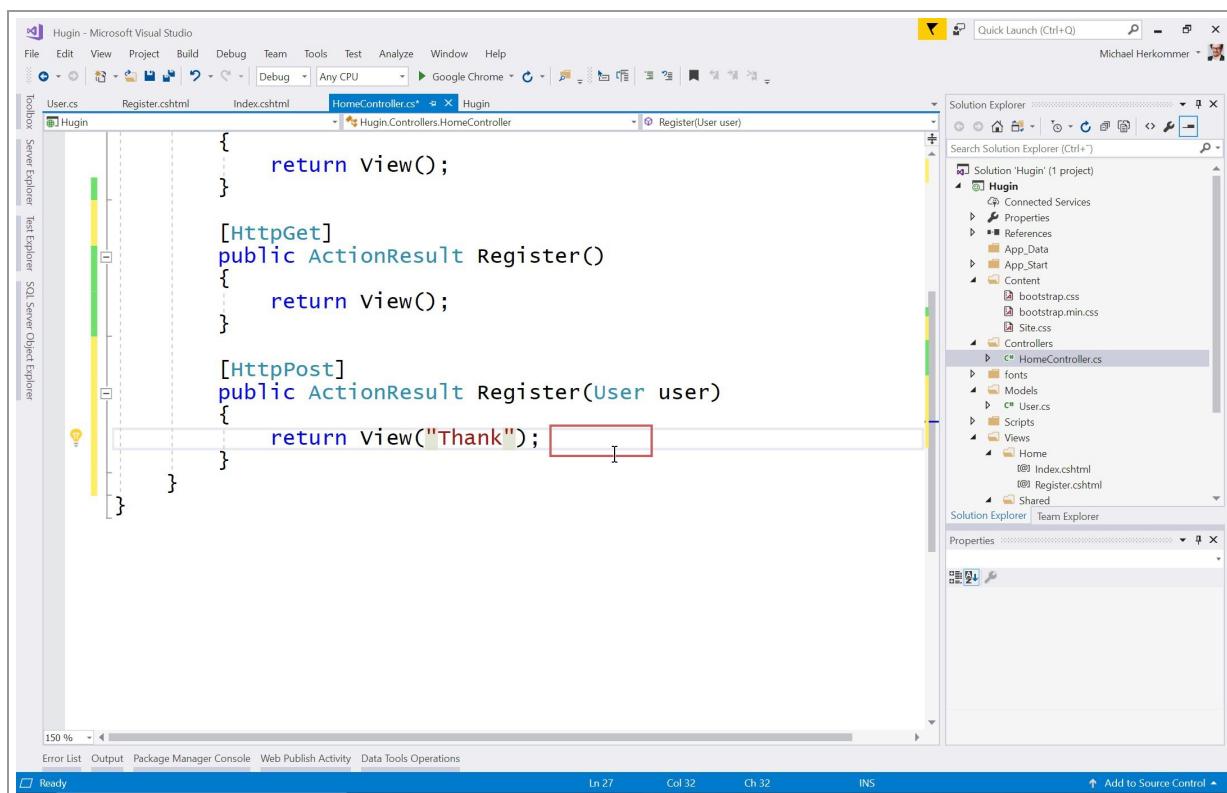
Click on "Light Bulb" button in ""



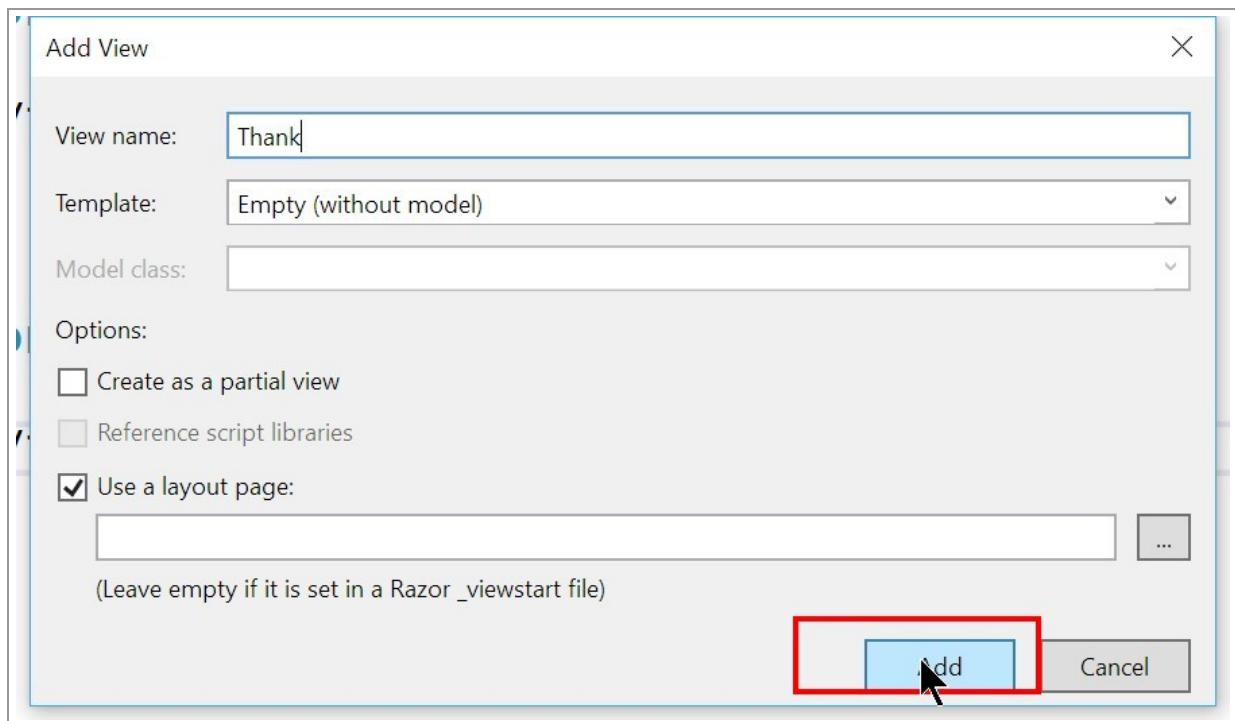
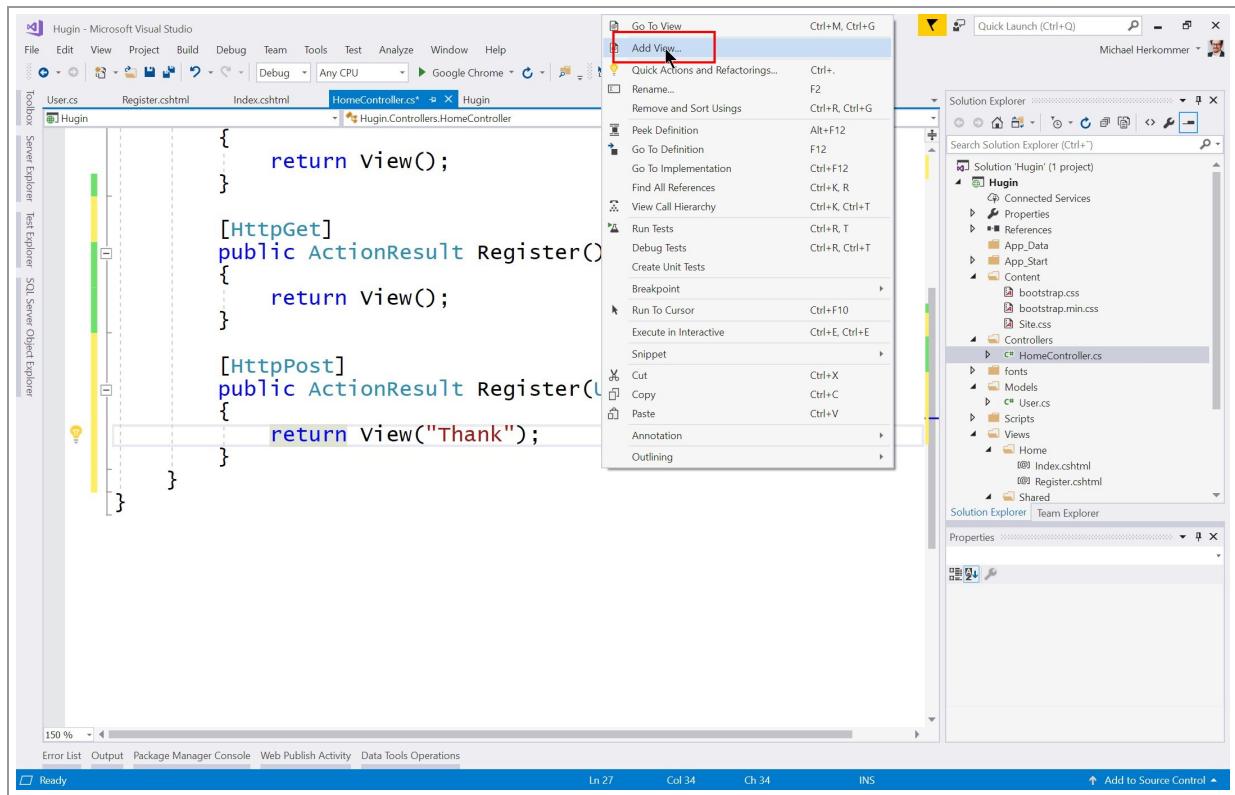
Click on "using Hugin.Models;" text in ""



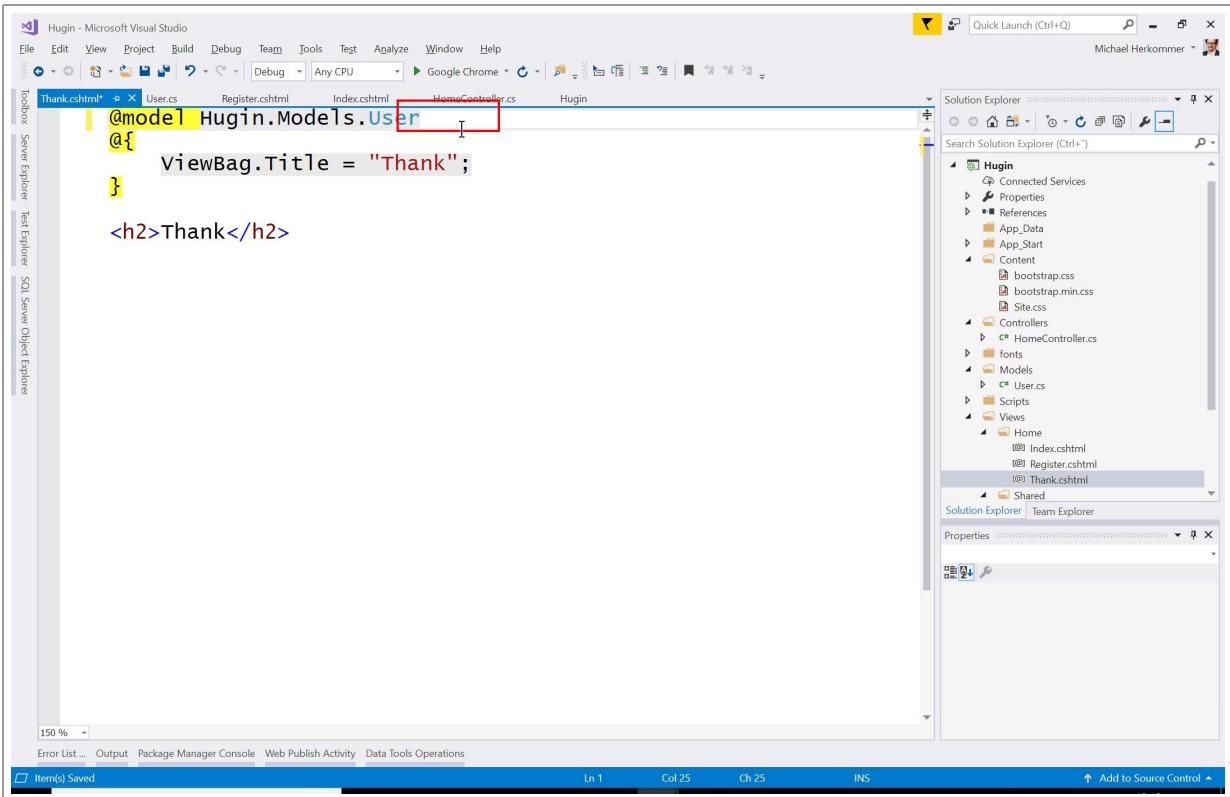
Click on "Text Editor" edit in "HomeController.cs*"



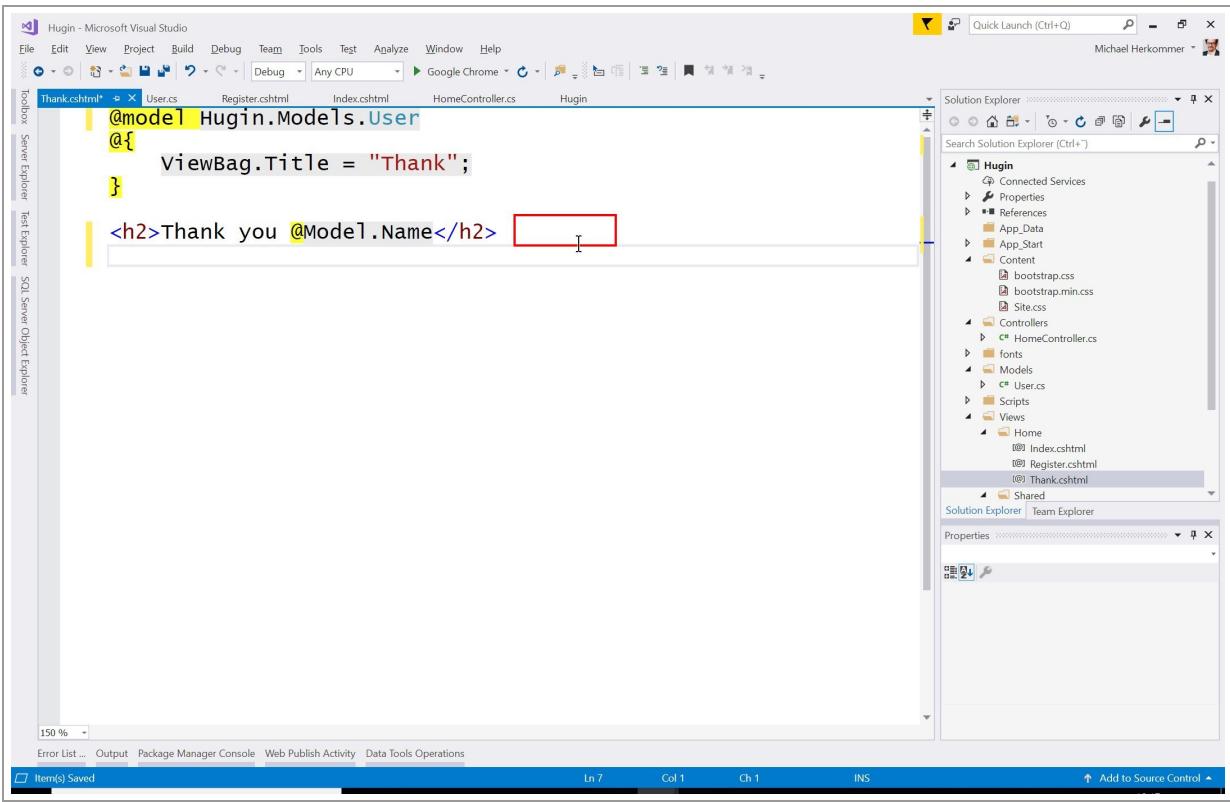
Right click on "Text Editor" edit in "HomeController.cs*"



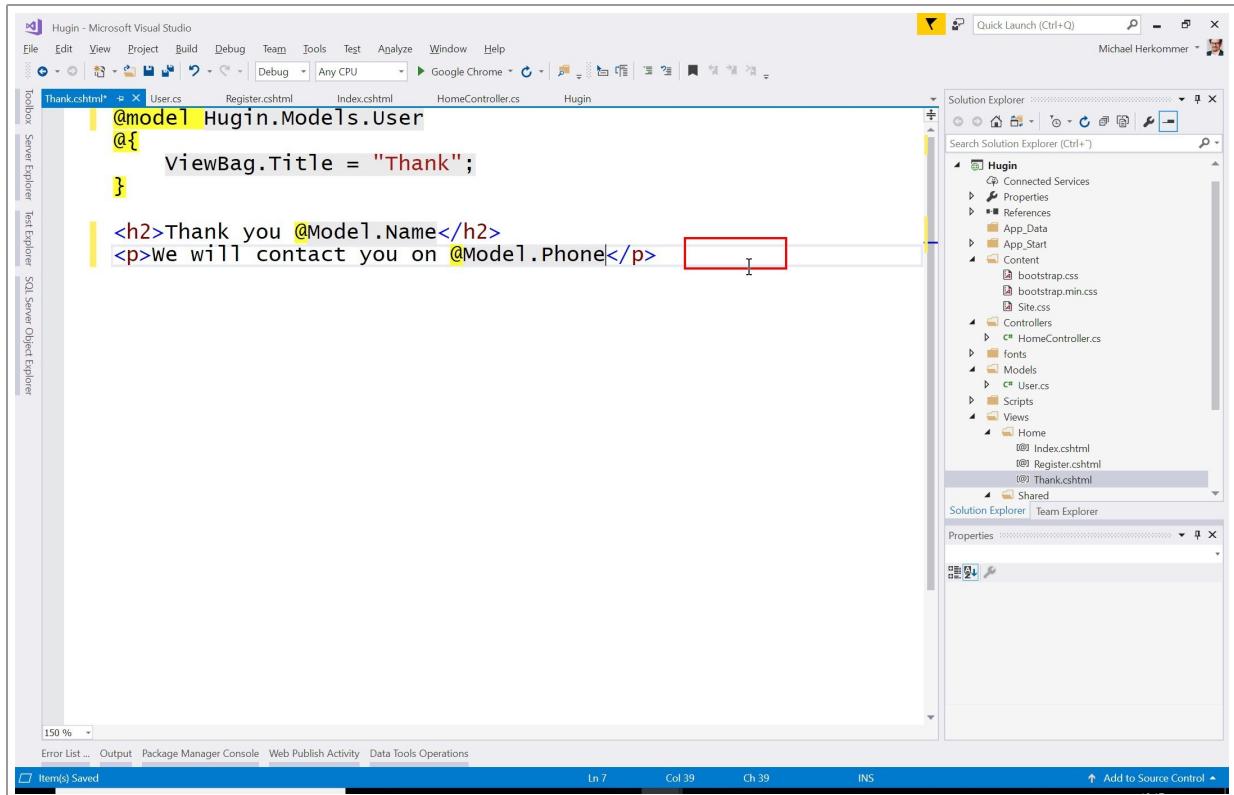
Click on "Add" text in "Add View"



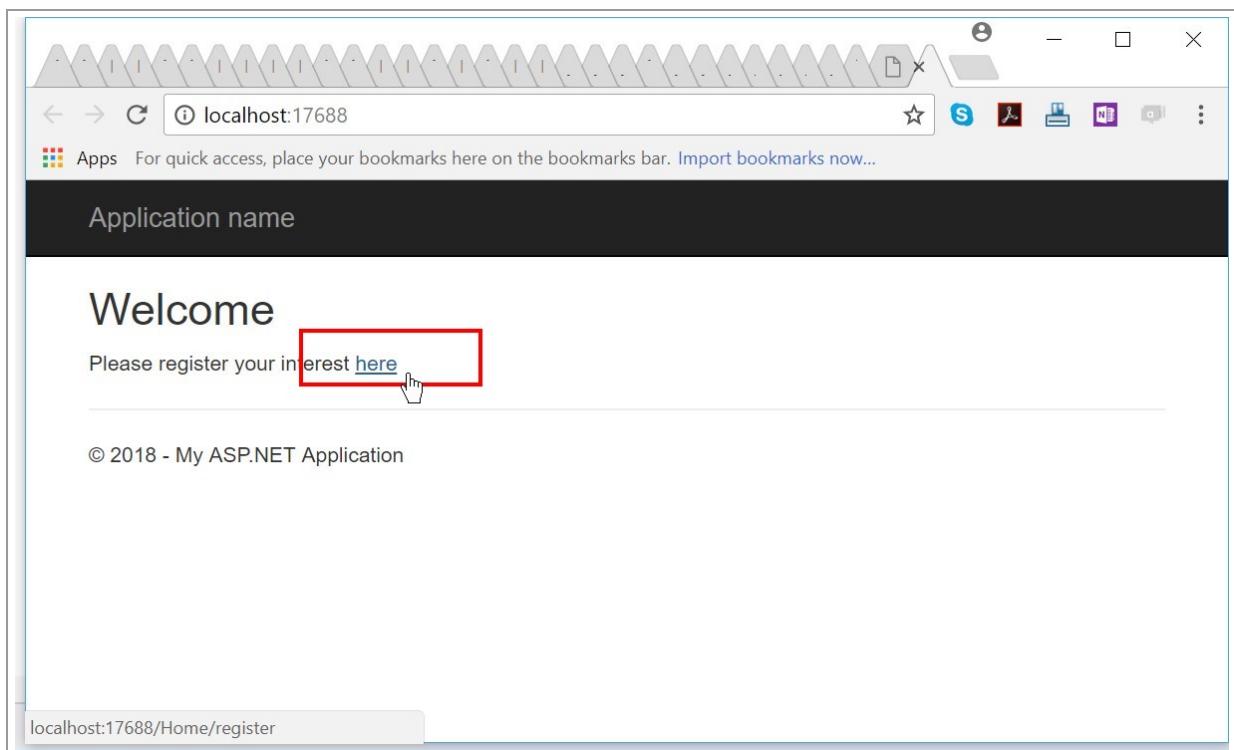
Click on "Text Editor" edit in "Thank.cshtml"



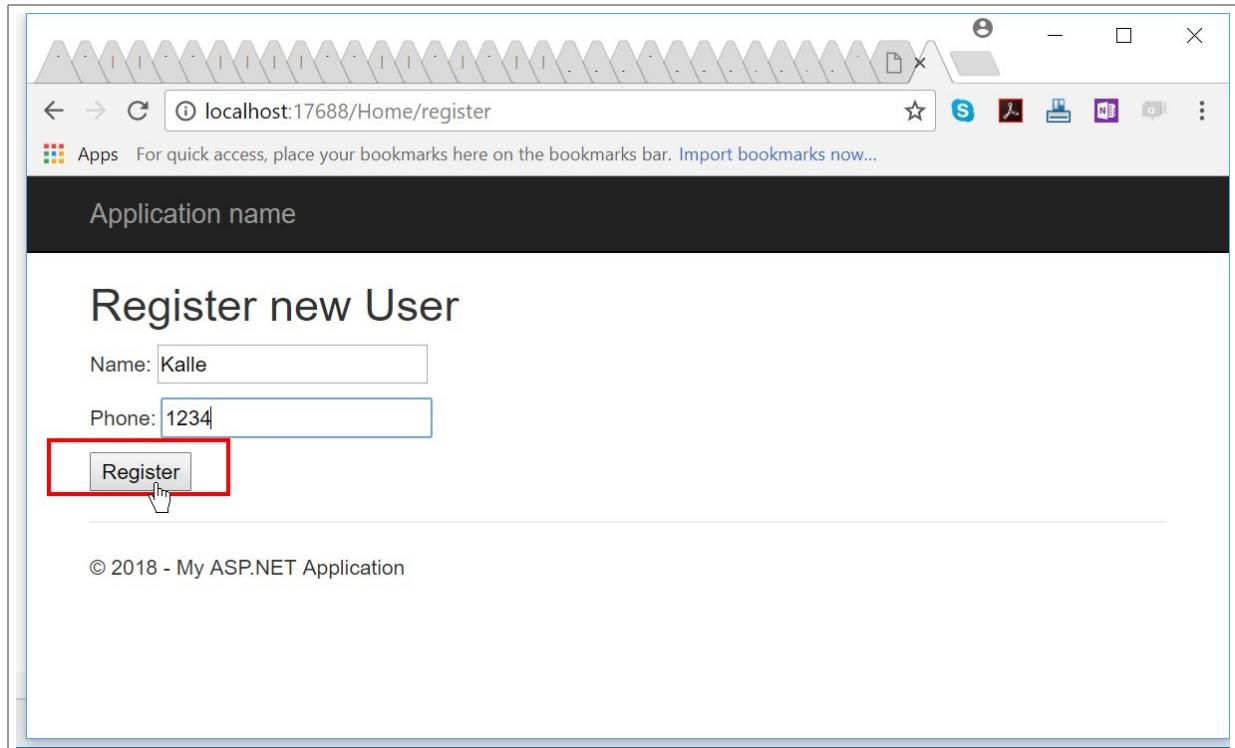
Click on "Text Editor" edit in "Thank.cshtml"



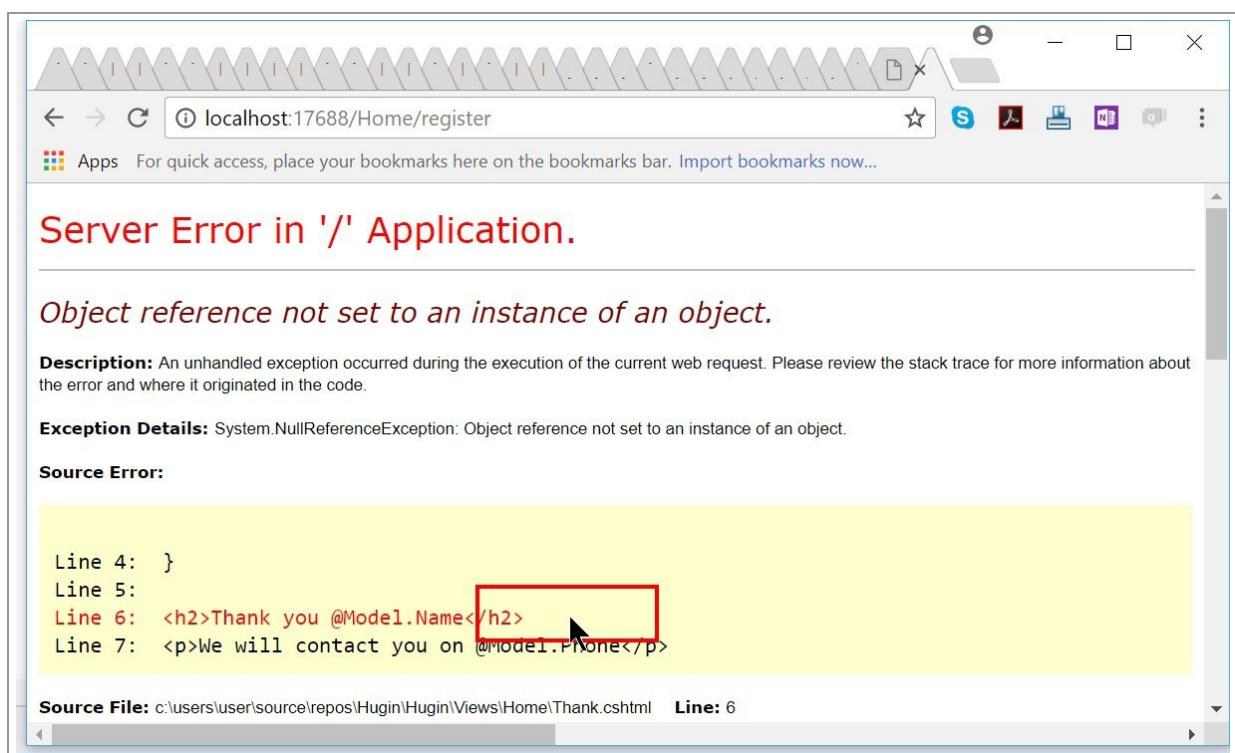
Click on "Text Editor" edit in "Thank.cshtml"



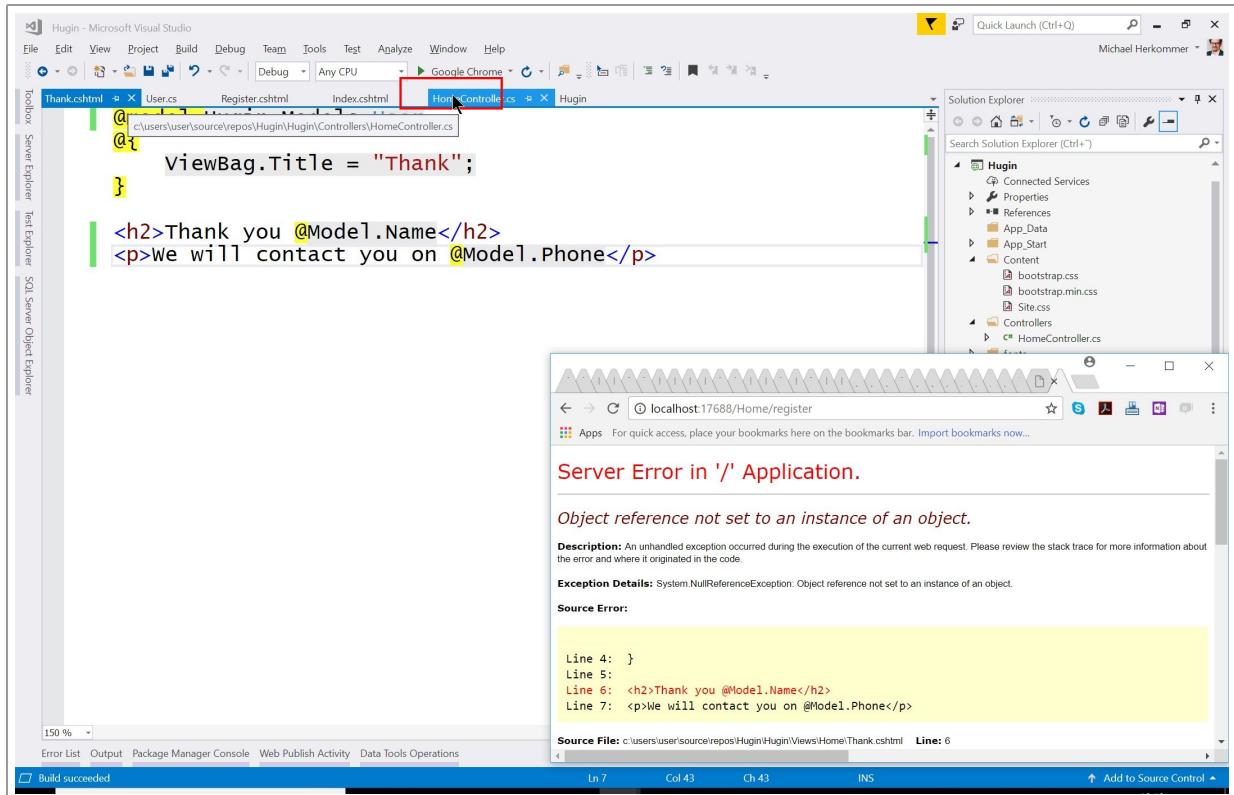
Click on "Chrome Legacy Window" document in "Home - My ASP.NET Application"



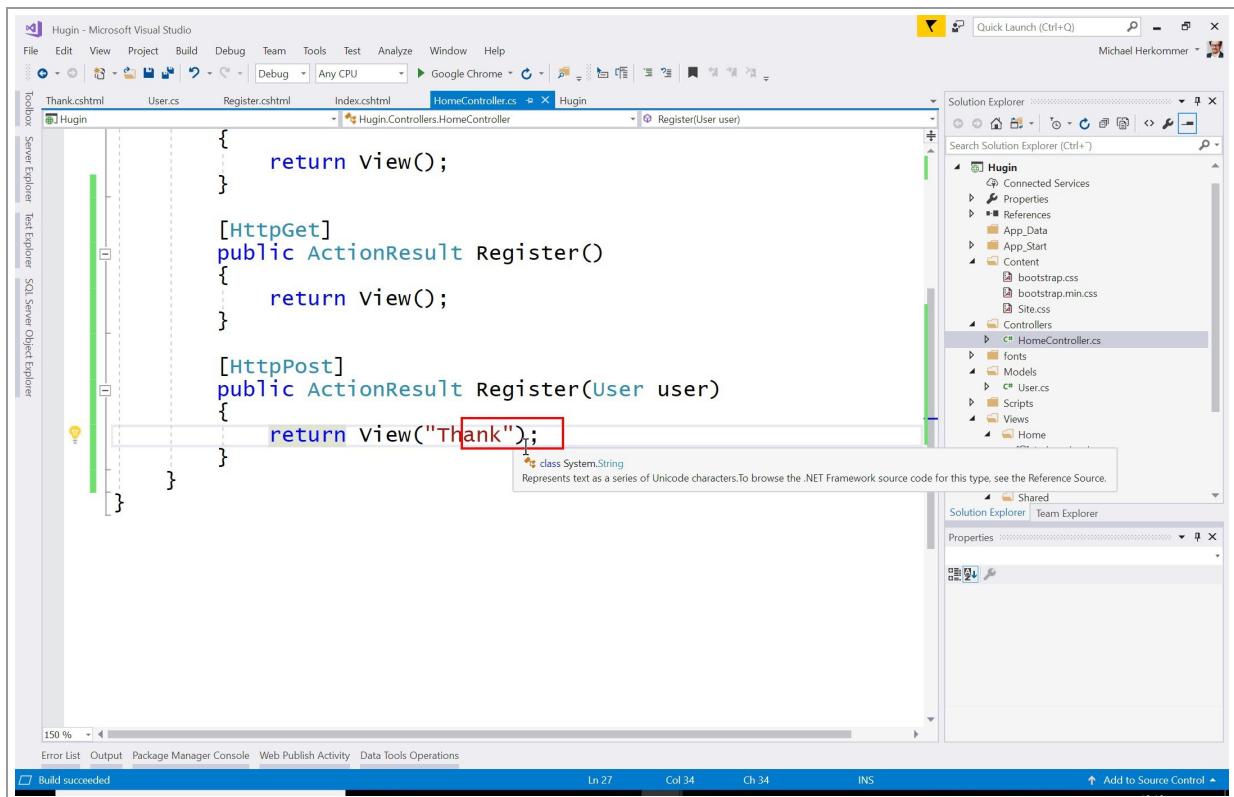
Click on "Chrome Legacy Window" document in "Register - My ASP.NET Application"



Click on "Chrome Legacy Window" document in "Object reference not set to an instance of an object."



Click on "HomeController.cs" text in "HomeController.cs"



Click on "Text Editor" edit in "HomeController.cs"

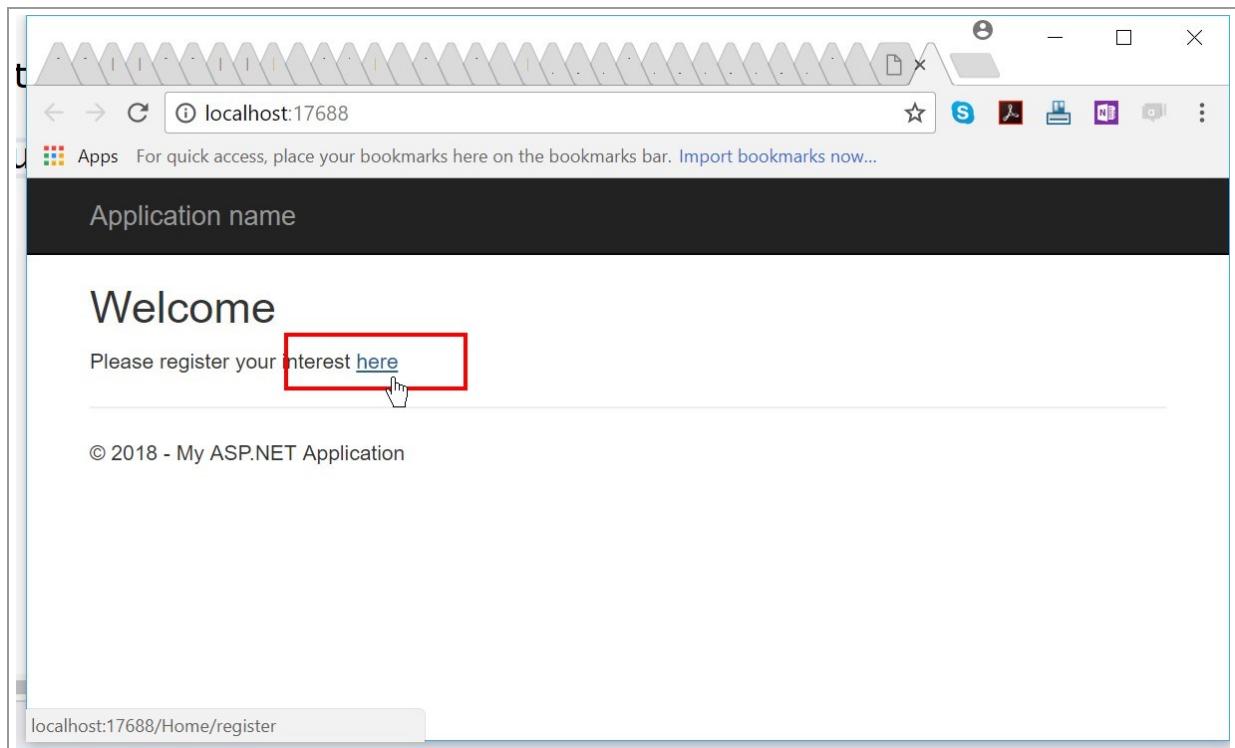
```
    return View();}

[HttpGet]
public ActionResult Register()
{
    return View();
}

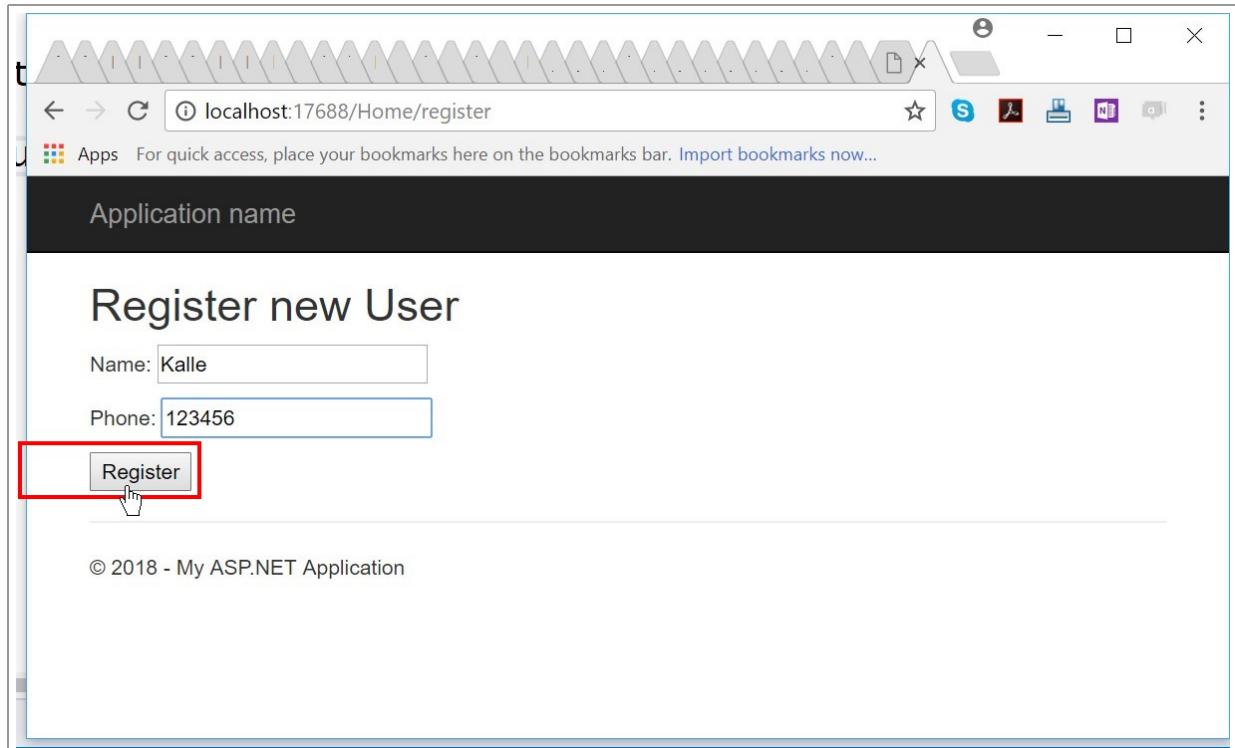
[HttpPost]
public ActionResult Register(User user)
{
    return View("Thank", user);
}
```

Creates a ViewResult object that renders the specified IView object.
model: The model that is rendered by the view.

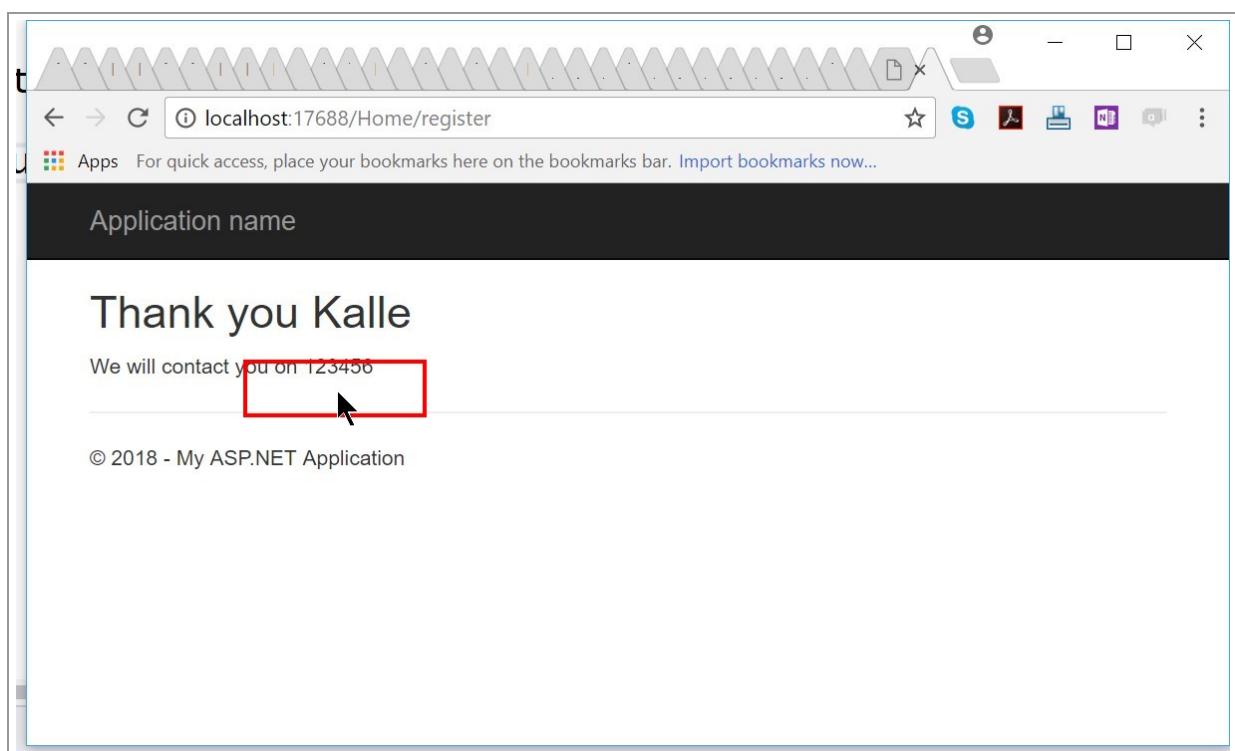
Click on "Text Editor" edit in "HomeController.cs"



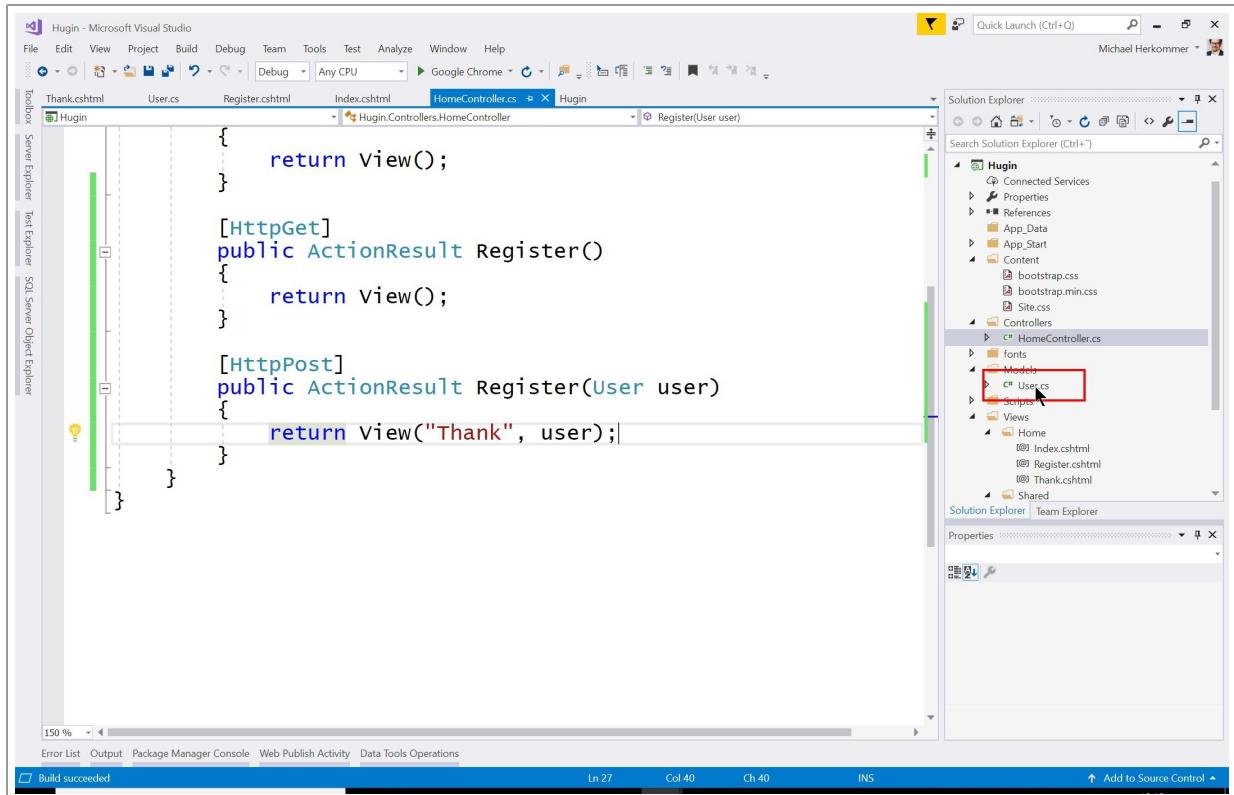
Click on "Chrome Legacy Window" document in "Home - My ASP.NET Application"



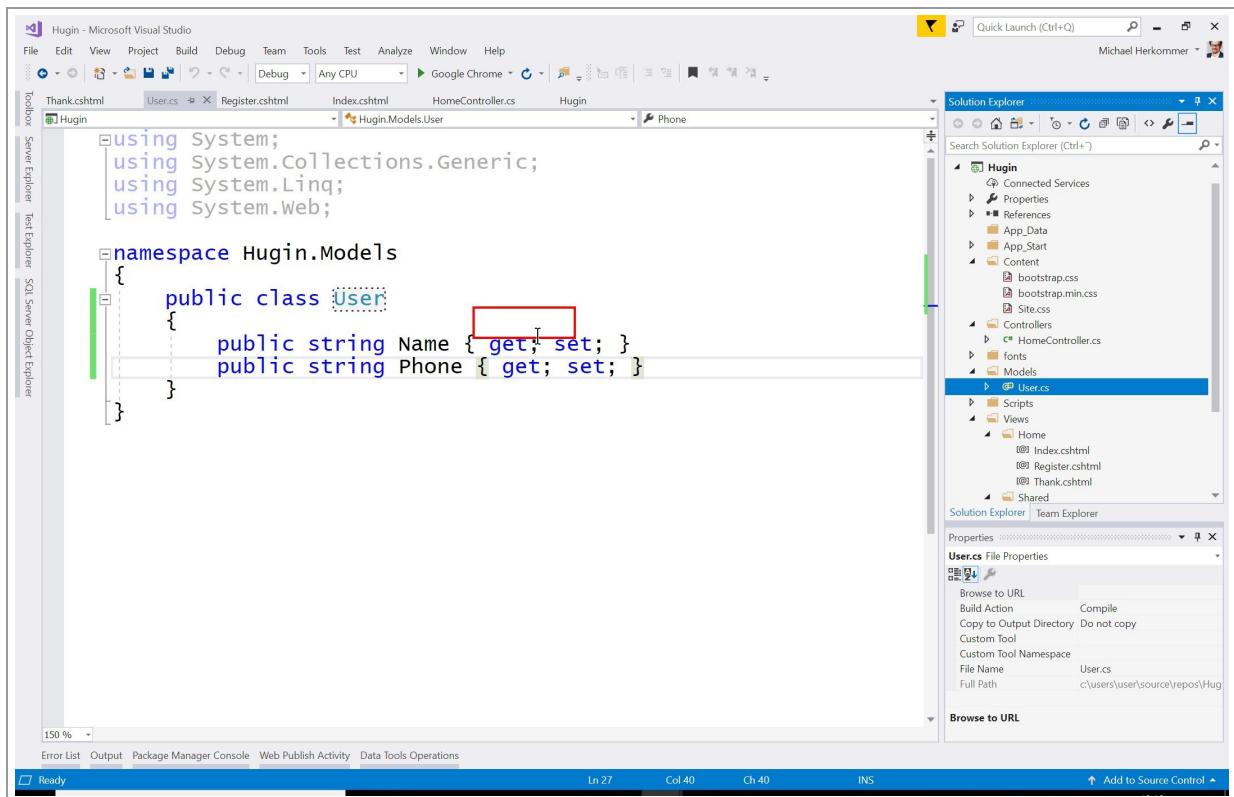
Click on "Chrome Legacy Window" document in "Register - My ASP.NET Application"



Click on "Chrome Legacy Window" document in "Thank - My ASP.NET Application"



Click on "User.cs" tree view item in "Solution Explorer"



Click on "Text Editor" edit in "User.cs"

A screenshot of Microsoft Visual Studio showing the User.cs file in the Text Editor. The code defines a User class with two properties: Name and Phone, both annotated with [Required]. A red box highlights the 'using System.ComponentModel.DataAnnotations;' line. The Solution Explorer shows the project structure, including files like Thank.cshtml, Register.cshtml, Index.cshtml, HomeController.cs, and the User.cs file itself.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.ComponentModel.DataAnnotations; [Red Box]

namespace Hugin.Models
{
    public class User
    {
        [Required()]
        public string Name { get; set; }
        public string Phone { get; set; }
    }
}
```

Click on "Text Editor" edit in "User.cs"

A screenshot of Microsoft Visual Studio showing the User.cs file in the Text Editor. The cursor is positioned over the [Required] attribute of the Name property. A red box highlights the entire line of code: 'public string Name { get; set; }'. The Solution Explorer shows the project structure, including files like Thank.cshtml, Register.cshtml, Index.cshtml, HomeController.cs, and the User.cs file itself.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.ComponentModel.DataAnnotations;

namespace Hugin.Models
{
    public class User
    {
        [Required]
        public string Name { get; set; }
        public string Phone { get; set; }
    }
}
```

Click on "Text Editor" edit in "User.cs"

A screenshot of Microsoft Visual Studio showing the User.cs code editor. The code defines a User class with two properties: Name and Phone, both annotated with [Required]. The Solution Explorer shows the project structure, including files like Thank.cshtml, Register.cshtml, Index.cshtml, HomeController.cs, and Hugin.csproj. The Properties and Shared nodes are also visible.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.ComponentModel.DataAnnotations;

namespace Hugin.Models
{
    public class User
    {
        [Required()]
        public string Name { get; set; }

        [Required()]
        public string Phone { get; set; }
    }
}
```

Click on "Text Editor" edit in "User.cs"

A screenshot of Microsoft Visual Studio showing the User.cs code editor. The file tab is highlighted with a red box. The code is identical to the previous screenshot. The Solution Explorer shows the project structure, including files like Thank.cshtml, Register.cshtml, Index.cshtml, HomeController.cs, and Hugin.csproj. The Properties and Shared nodes are also visible.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.ComponentModel.DataAnnotations;

namespace Hugin.Models
{
    public class User
    {
        [Required()]
        public string Name { get; set; }

        [Required()]
        public string Phone { get; set; }
    }
}
```

Click on "HomeController.cs" text in "HomeController.cs"

```
    {
        return View();
    }

    [HttpPost]
    public ActionResult Register()
    {
        return View();
    }

    [HttpPost]
    public ActionResult Register(User user)
    {
        return View("Thank", user);
    }
}
```

Click on "Text Editor" edit in "HomeController.cs"

```
    {
        return View();
    }

    [HttpPost]
    public ActionResult Register()
    {
        return View();
    }

    [HttpPost]
    public ActionResult Register(User user)
    {
        if (ModelState.IsValid)
        {
            return View("Thank", user);
        }
    }
}
```

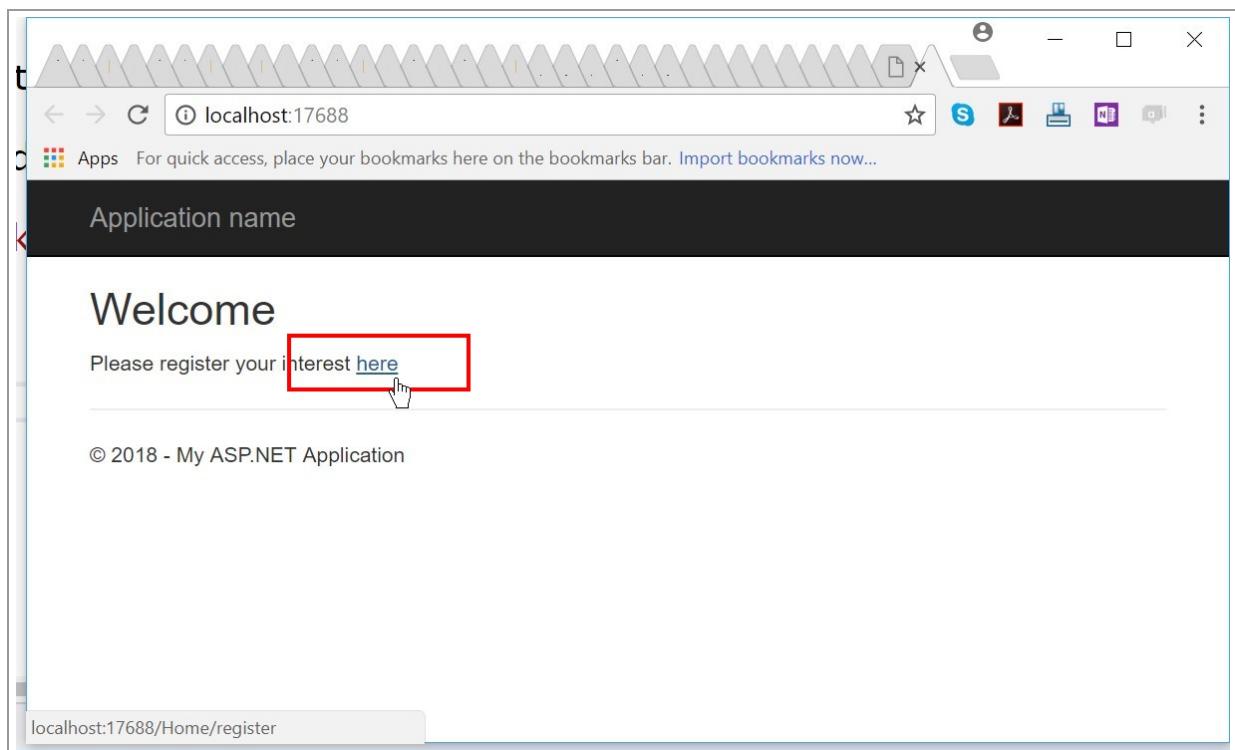
Click on "Text Editor" edit in "HomeController.cs*"

```
        {
            return View();
        }

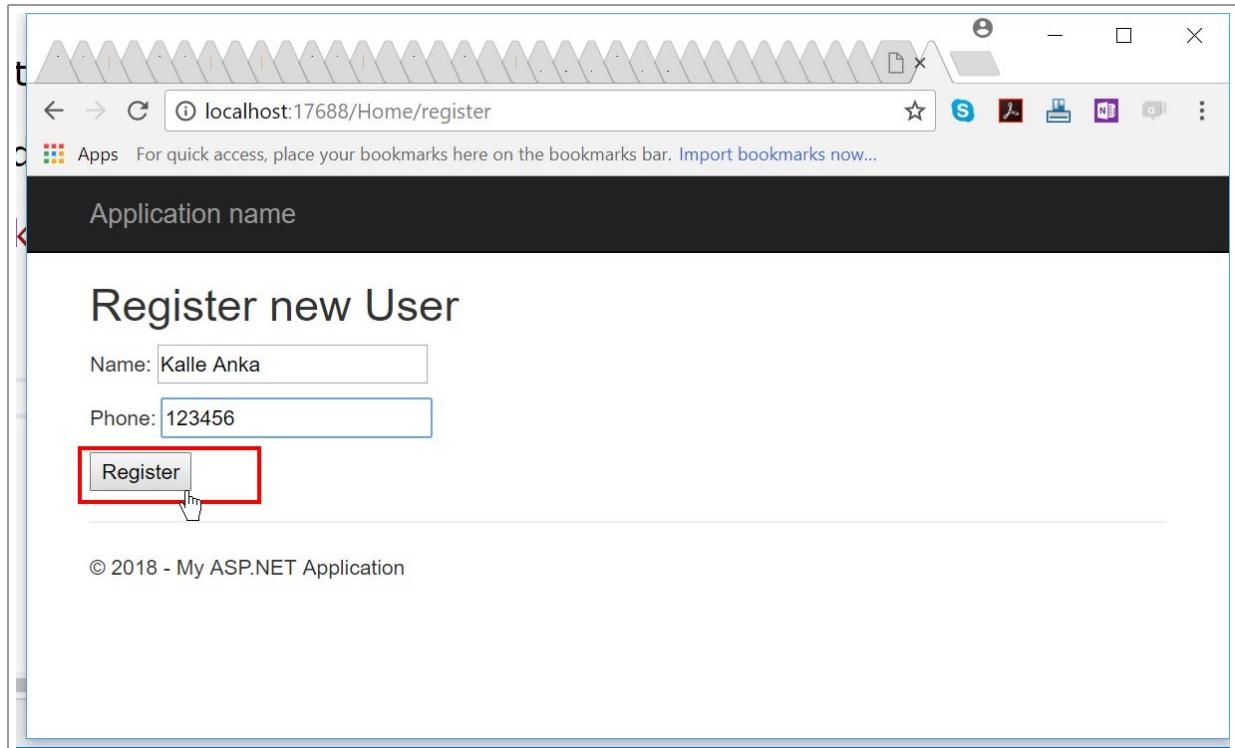
        [HttpGet]
        public ActionResult Register()
        {
            return View();
        }

        [HttpPost]
        public ActionResult Register(User user)
        {
            if (ModelState.IsValid)
            {
                return View("Thank", user);
            }
            else
            {
                return View(); // Red box highlights this line
            }
        }
    }
```

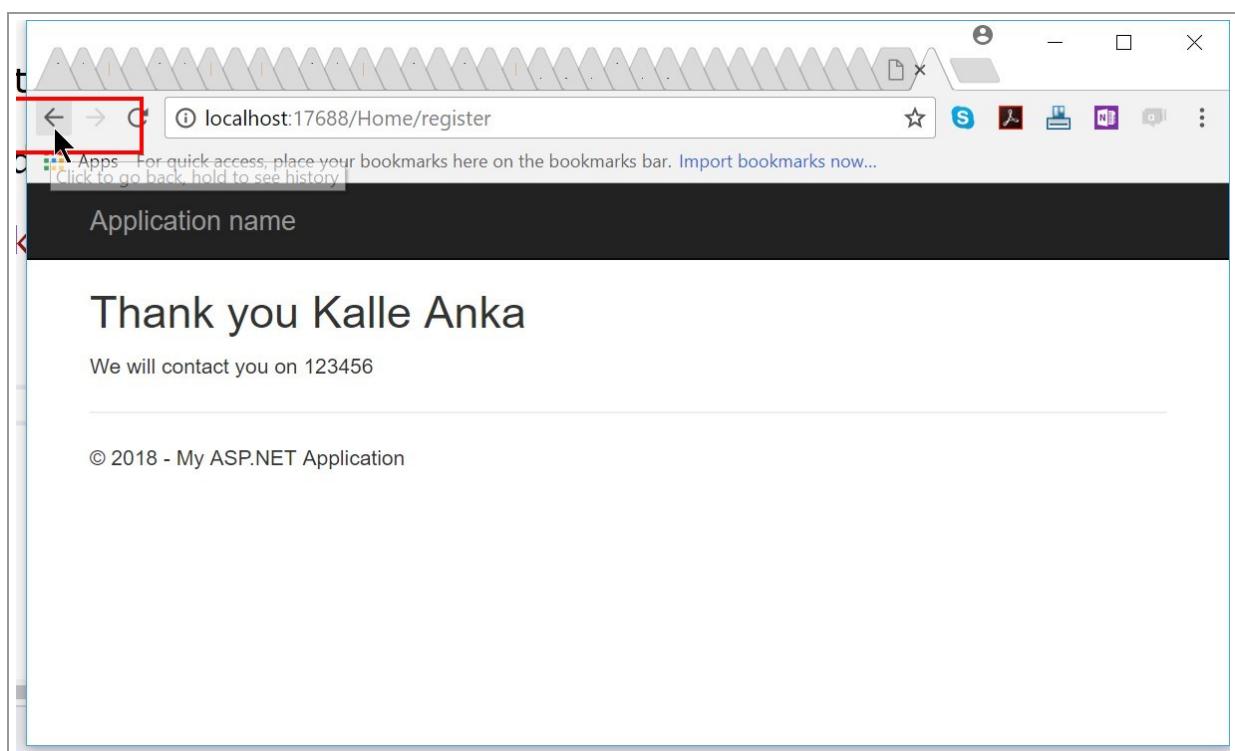
Click on "Text Editor" edit in "HomeController.cs"



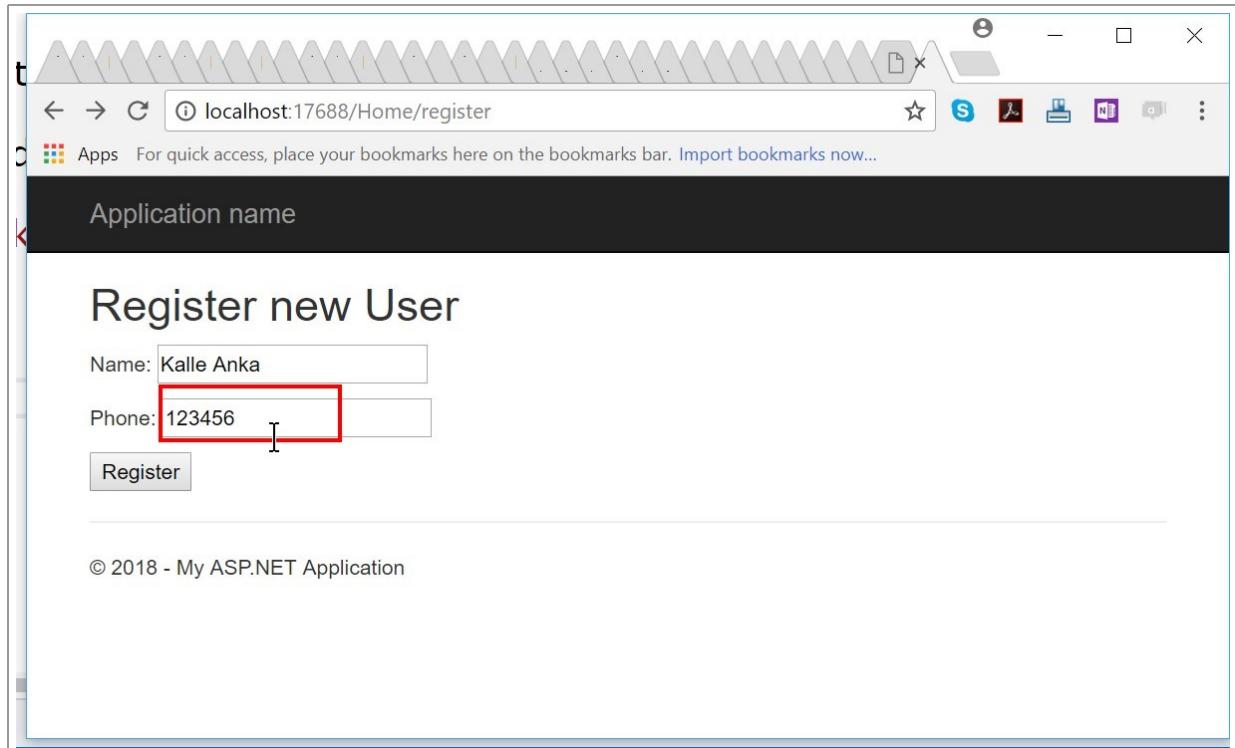
Click on "Chrome Legacy Window" document in "Home - My ASP.NET Application"



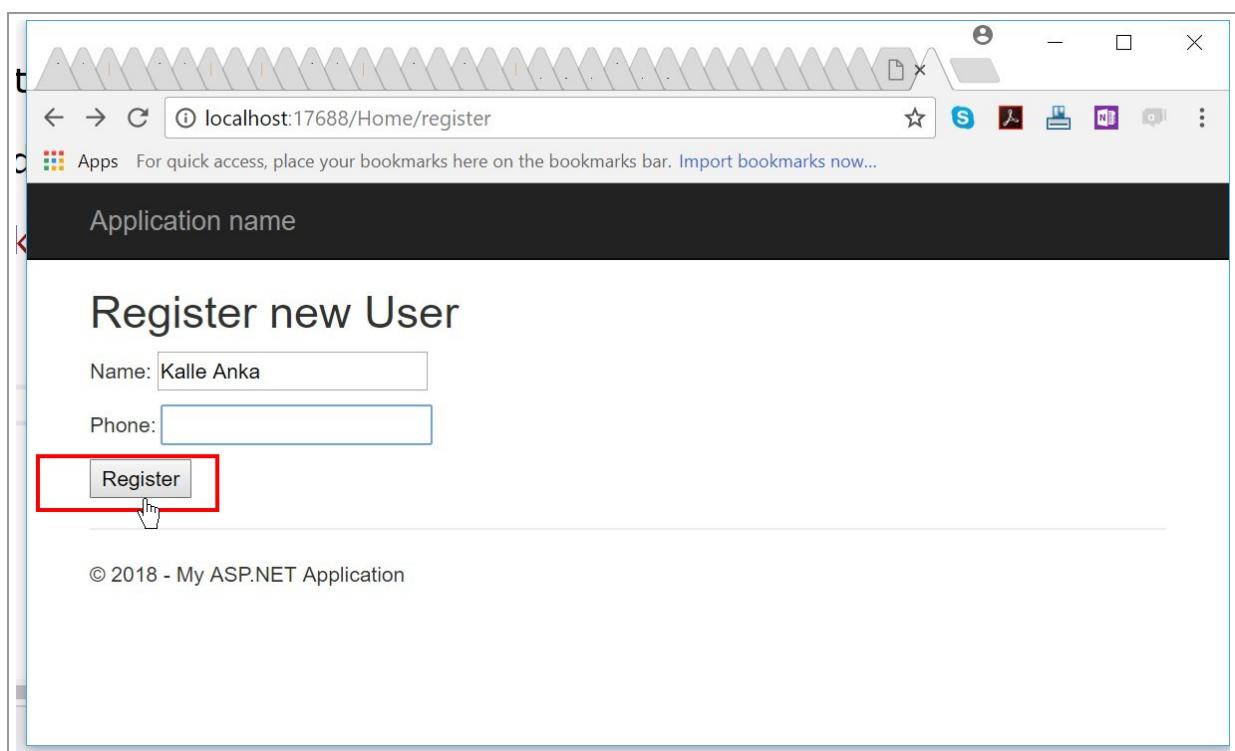
Click on "Chrome Legacy Window" document in "Register - My ASP.NET Application"



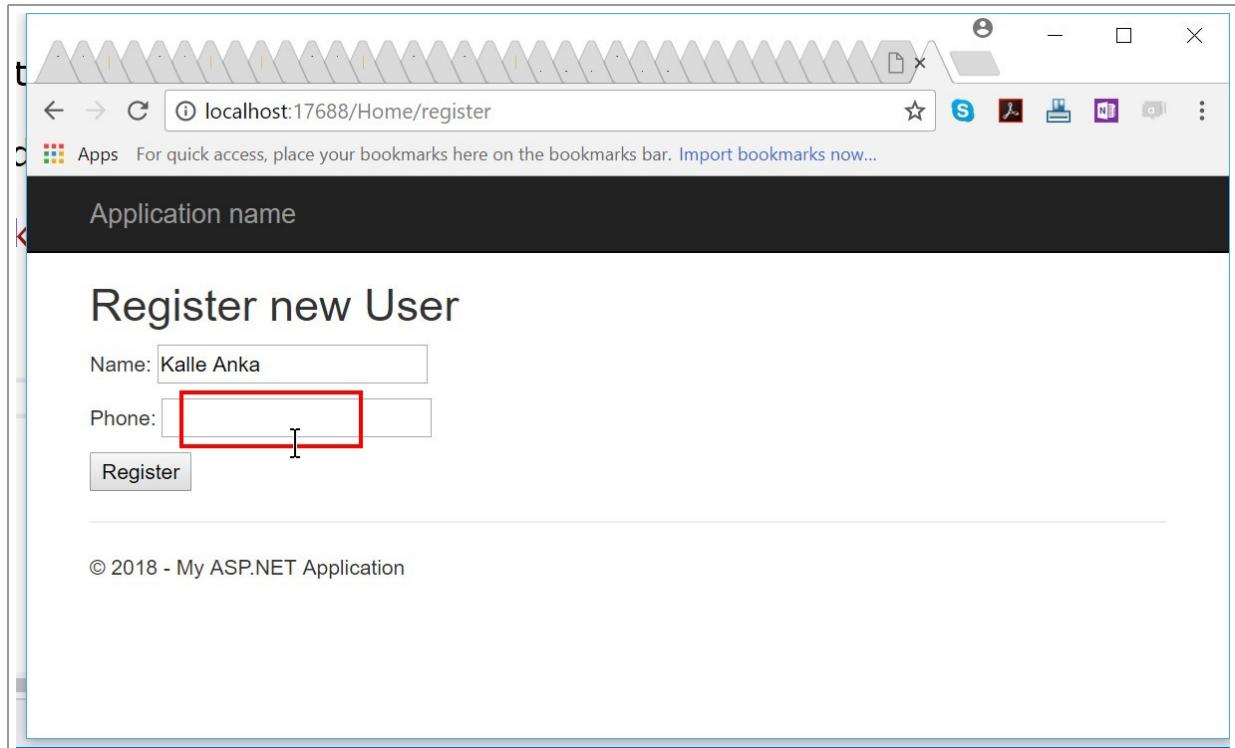
Click on "Thank - My ASP.NET Application" window in "Thank - My ASP.NET Application"



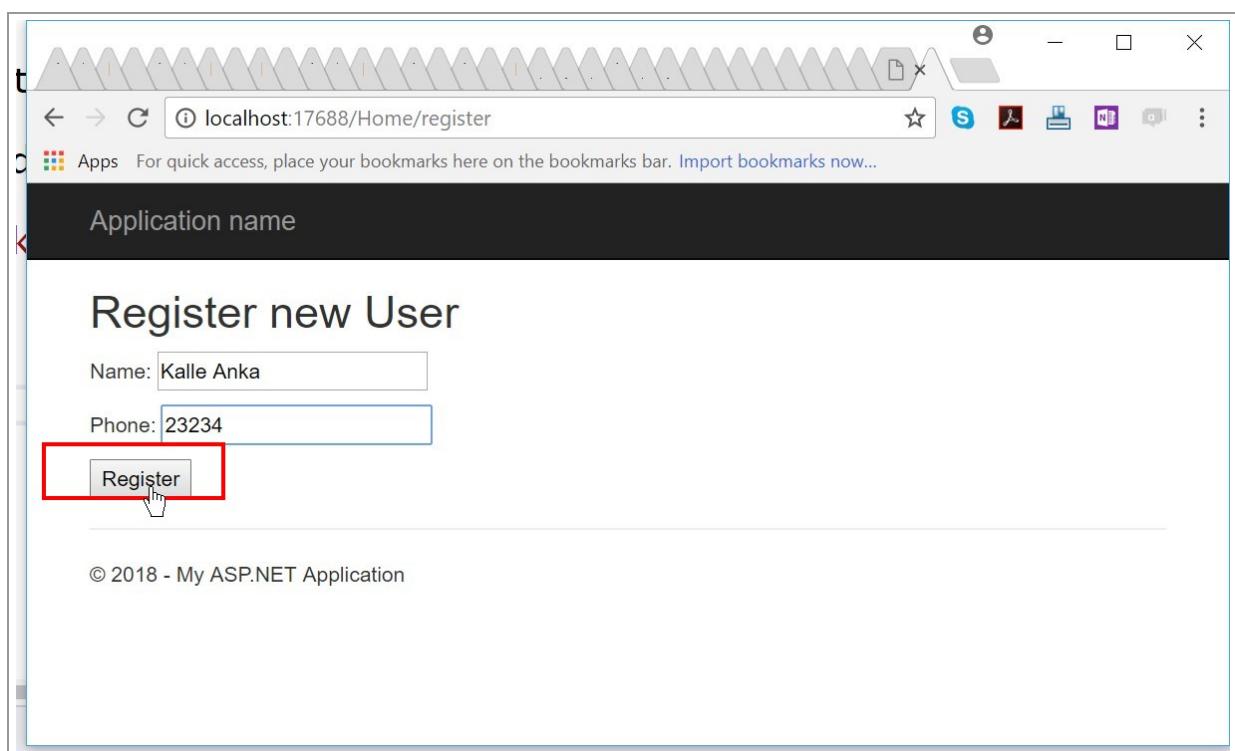
Click on "Chrome Legacy Window" document in "Register - My ASP.NET Application"



Click on "Chrome Legacy Window" document in "Register - My ASP.NET Application"



Click on "Chrome Legacy Window" document in "Register - My ASP.NET Application"



Click on "Chrome Legacy Window" document in "Register - My ASP.NET Application"

The screenshot shows the Microsoft Visual Studio interface with the "Register.cshtml" file open in the main editor window. The code is as follows:

```
        return View();  
  
    }  
  
    [HttpGet]  
    public ActionResult Register()  
    {  
        return View();  
    }  
  
    [HttpPost]  
    public ActionResult Register(User user)  
    {  
        if (ModelState.IsValid)  
        {  
            return View("Thank", user);  
        }  
        else  
        {  
            return View();  
        }  
    }  
}
```

The "Register.cshtml" tab is highlighted with a red box. The Solution Explorer on the right shows the project structure with files like "Index.cshtml", "Register.cshtml", and "Thank.cshtml" under the "Views\Home" folder.

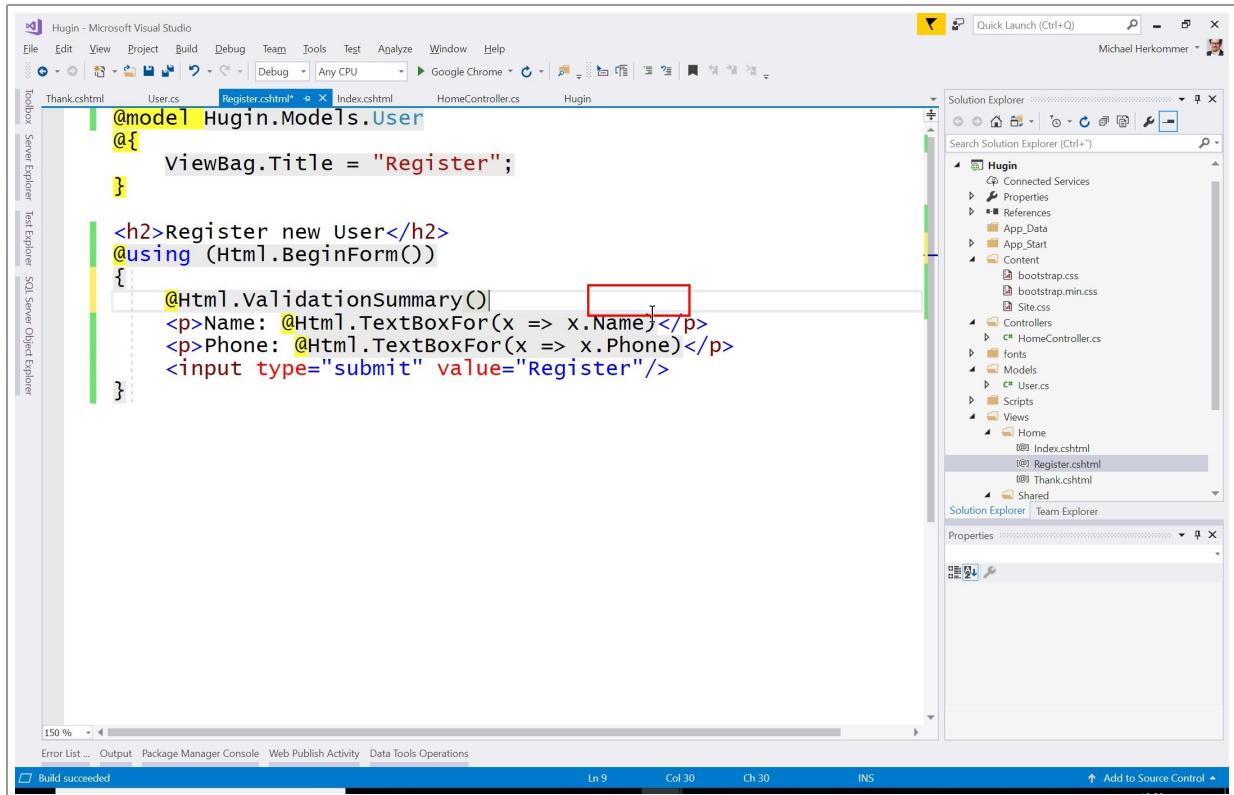
Click on "Text Editor" edit in "HomeController.cs"

The screenshot shows the Microsoft Visual Studio interface with the "Register.cshtml" file open in the main editor window. The code is as follows:

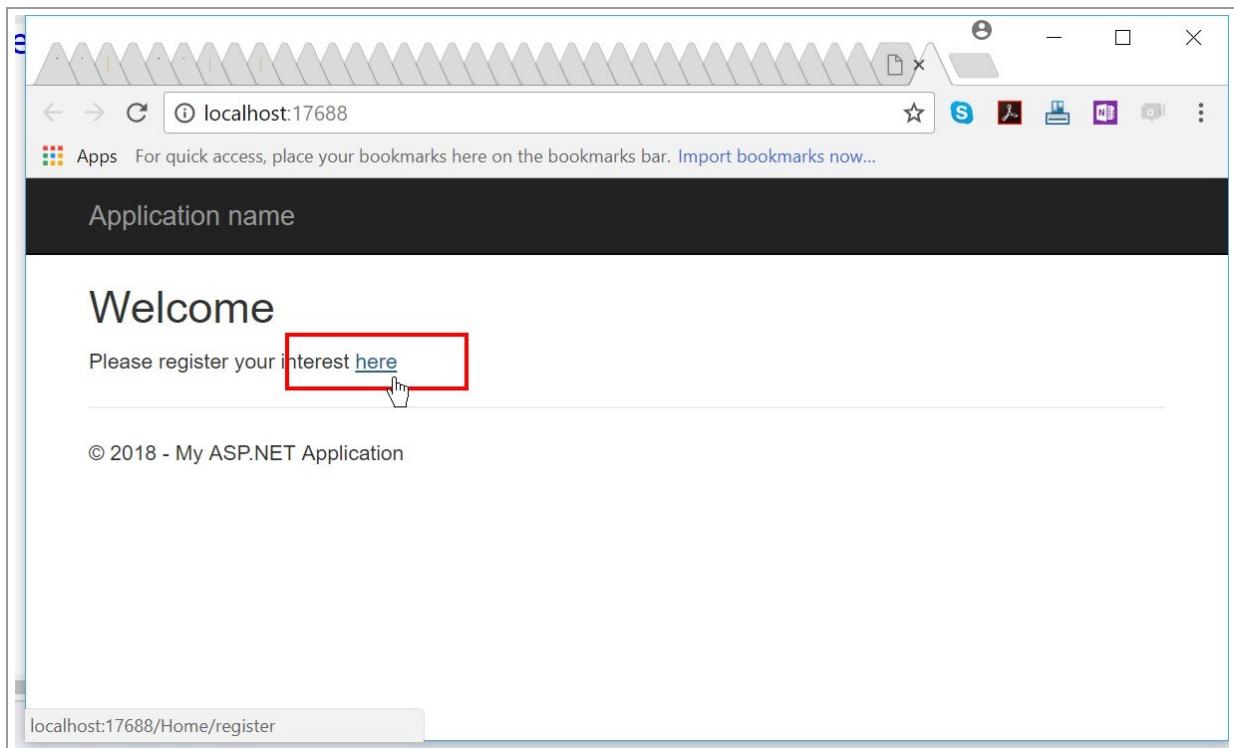
```
@model Hugin.Models.User  
@{  
    ViewBag.Title = "Register";  
}  
  
<h2>Register new User</h2>  
@using (Html.BeginForm())  
{  
    <p>Name: @Html.TextBoxFor(x => x.Name)</p>  
    <p>Phone: @Html.TextBoxFor(x => x.Phone)</p>  
    <input type="submit" value="Register"/>  
}
```

The "Register.cshtml" tab is highlighted with a red box. The Solution Explorer on the right shows the project structure with files like "Index.cshtml", "Register.cshtml", and "Thank.cshtml" under the "Views\Home" folder.

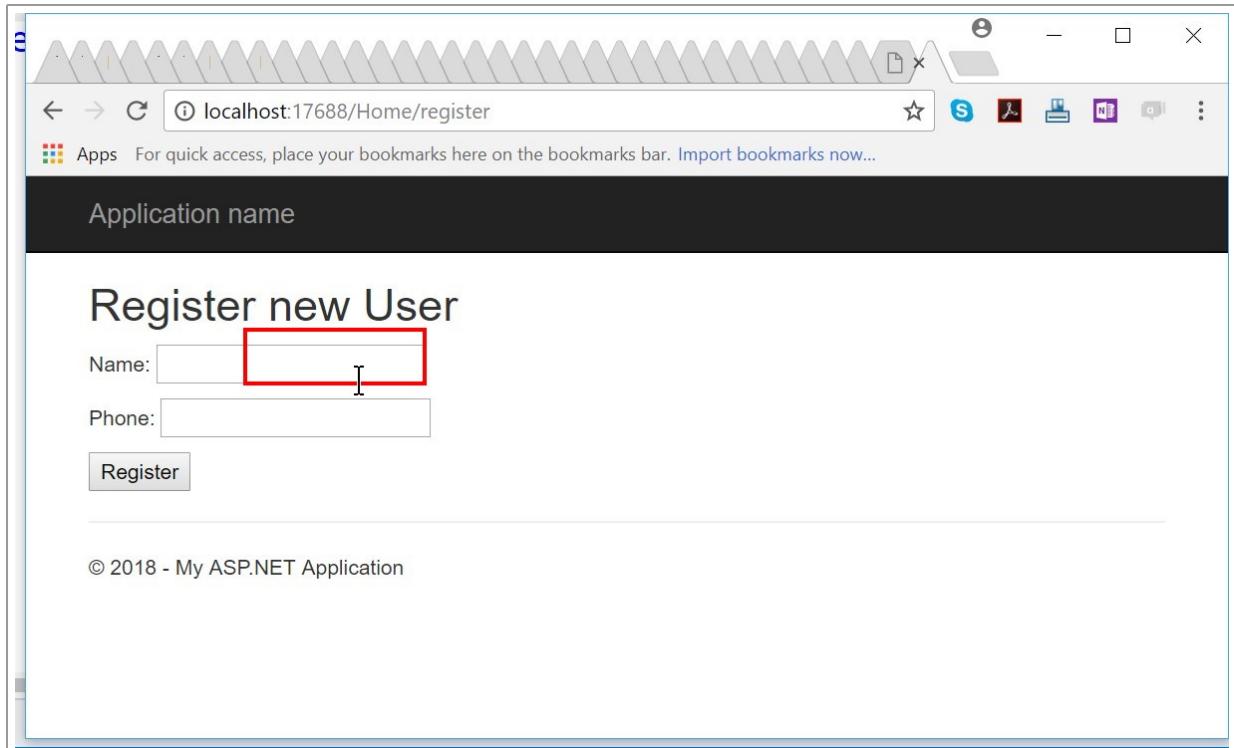
Click on "Text Editor" edit in "Register.cshtml"



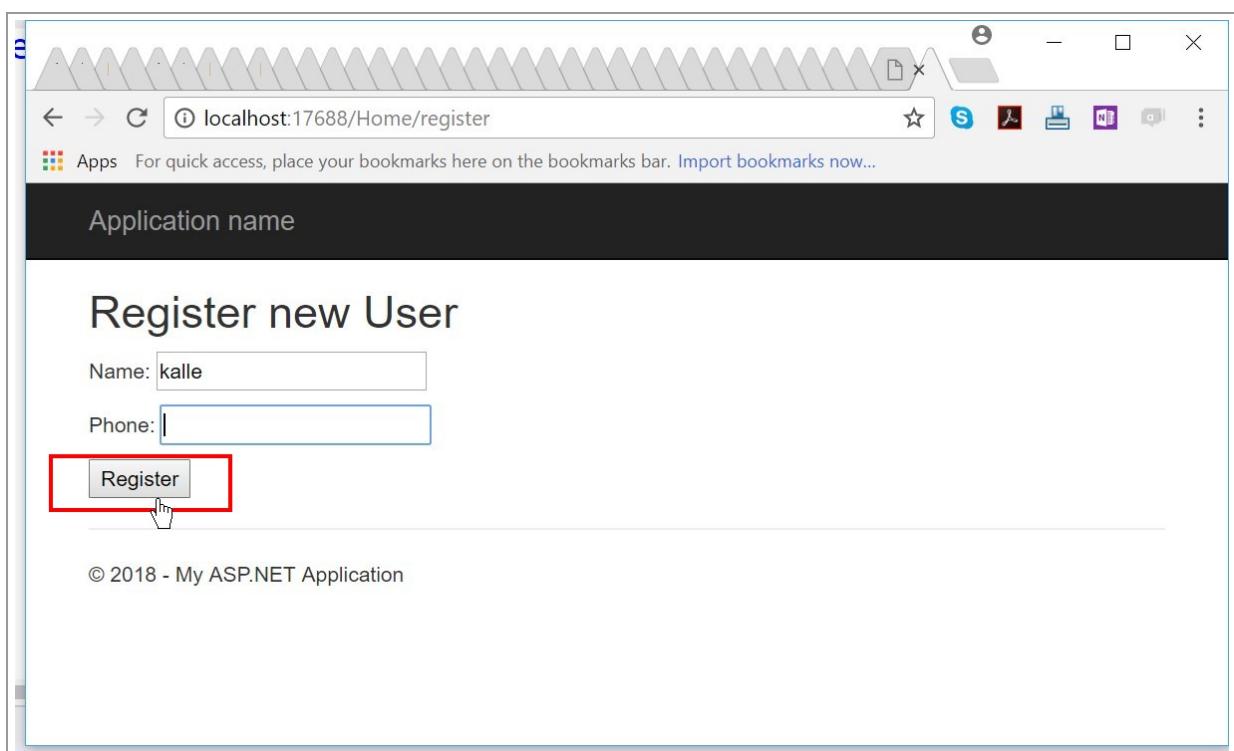
Click on "Text Editor" edit in "Register.cshtml*"



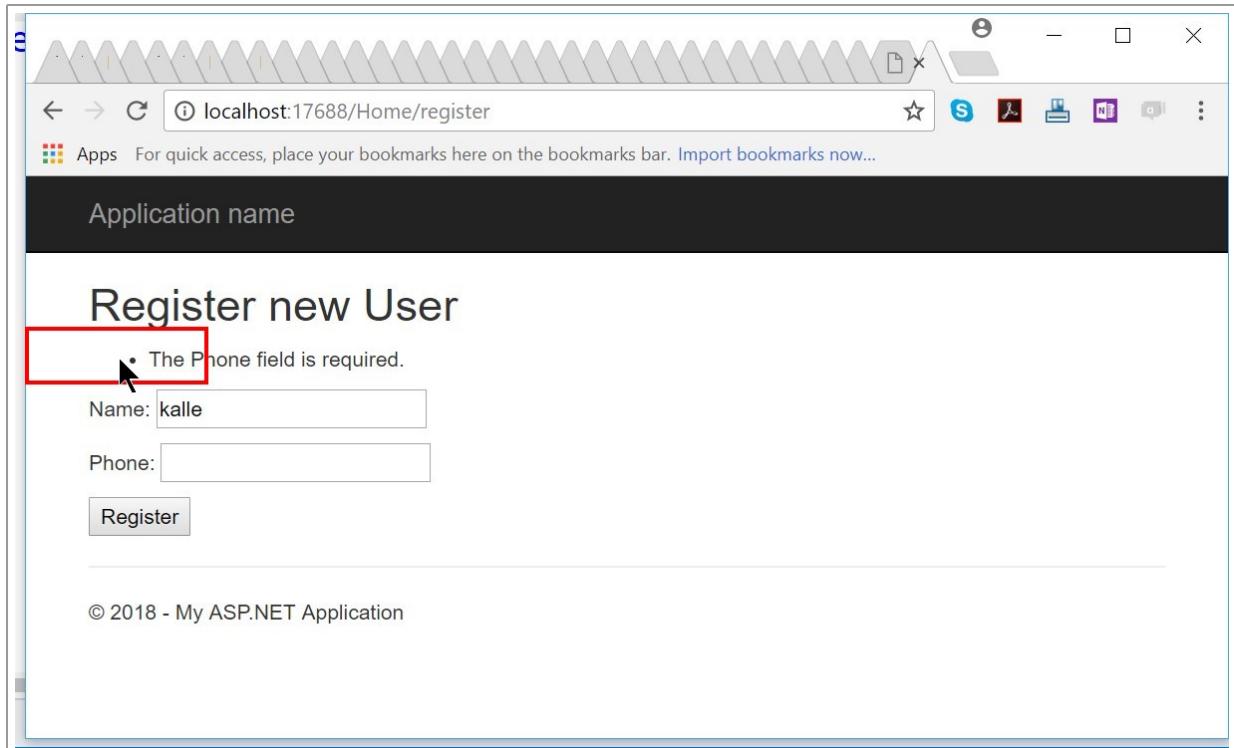
Click on "Chrome Legacy Window" document in "Home - My ASP.NET Application"



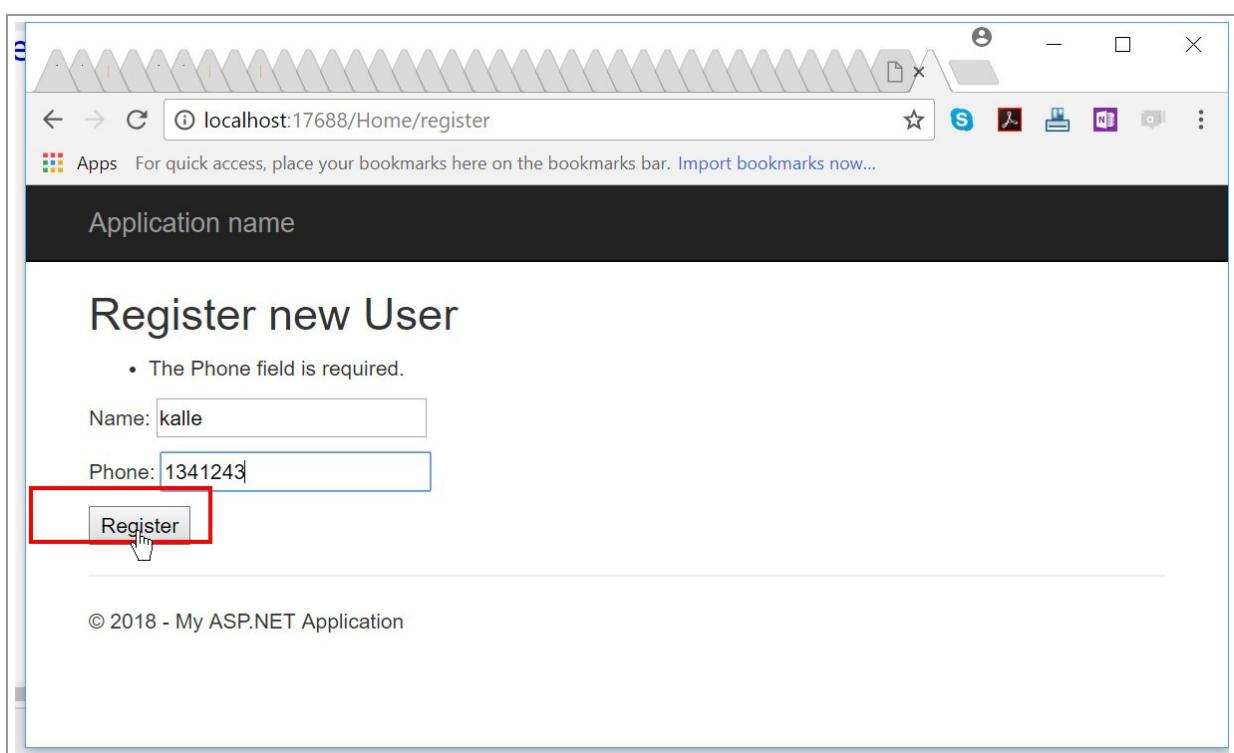
Click on "Chrome Legacy Window" document in "Register - My ASP.NET Application"



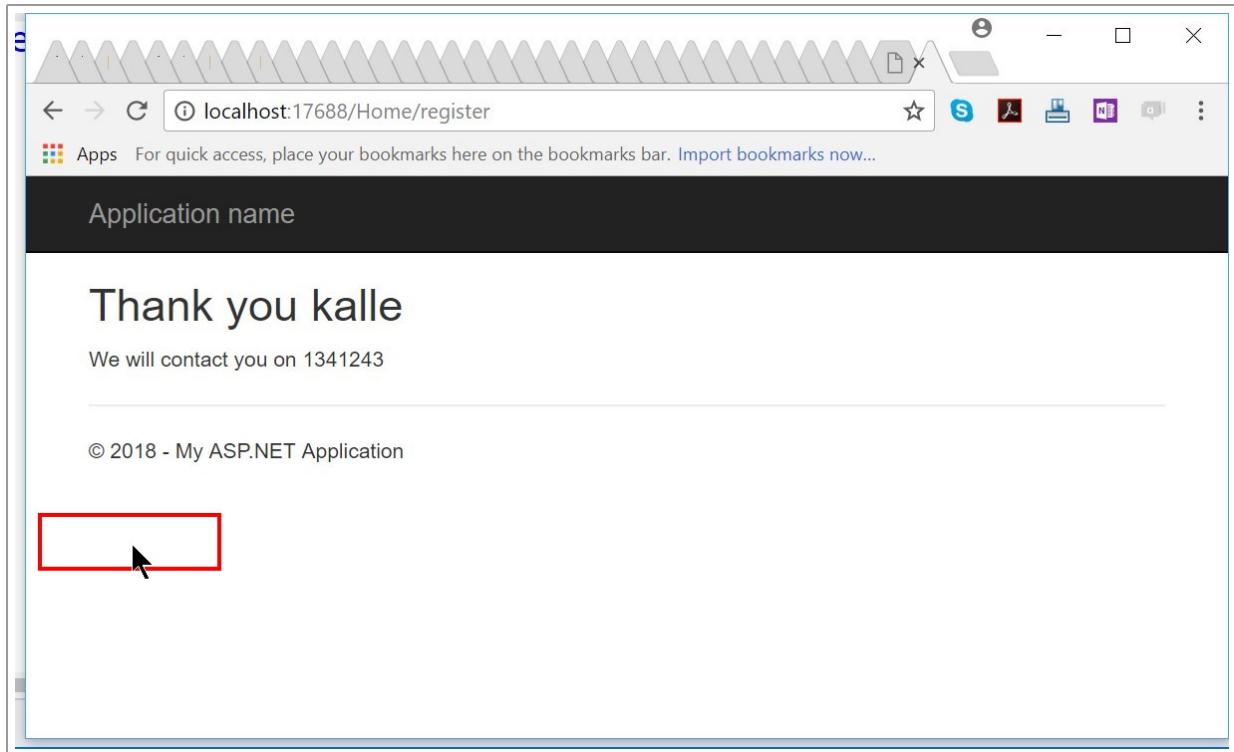
Click on "Chrome Legacy Window" document in "Register - My ASP.NET Application"



Click on "Chrome Legacy Window" document in "Register - My ASP.NET Application"



Click on "Chrome Legacy Window" document in "Register - My ASP.NET Application"



Click on "Chrome Legacy Window" document in "Thank - My ASP.NET Application"

The screenshot shows a Chrome browser window with the URL `localhost:17688/Home/register`. The page content is as follows:

```
<h1>Application name</h1>


## Register new User



- The Phone field is required.



Name:



Phone:



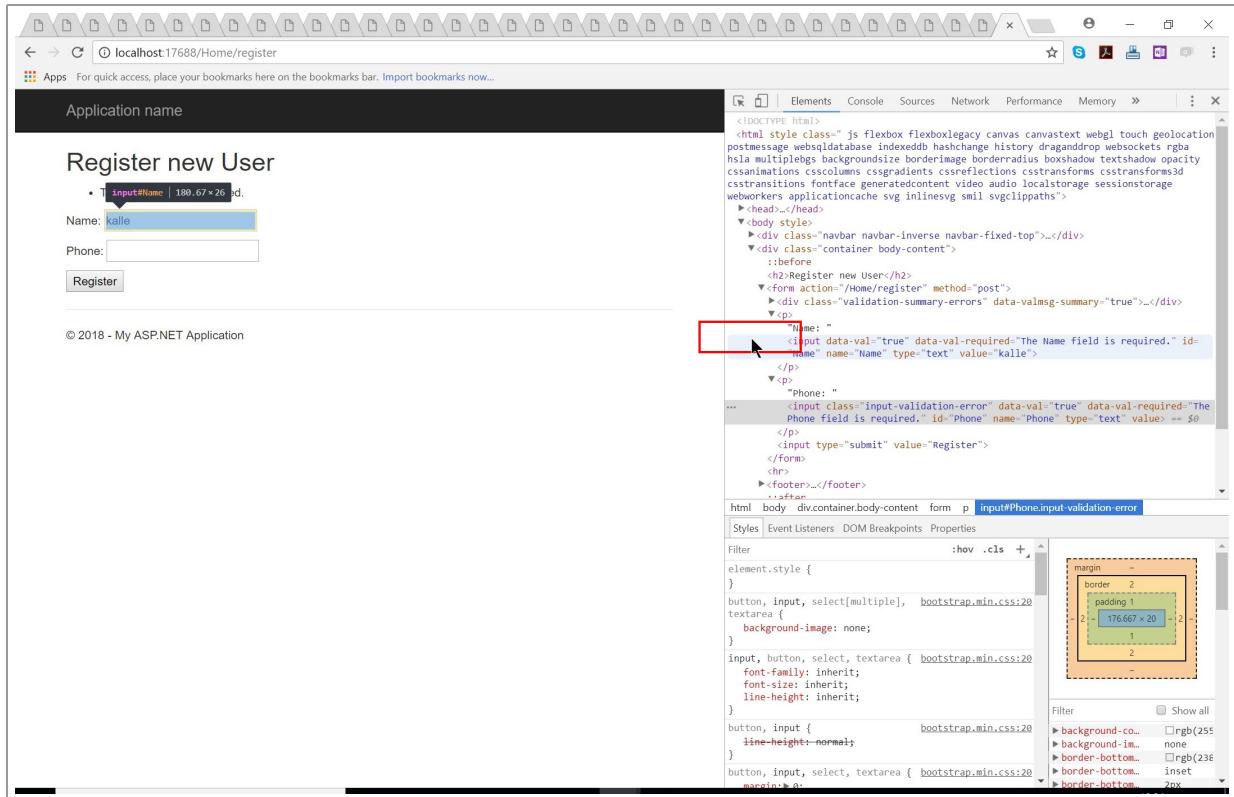
---



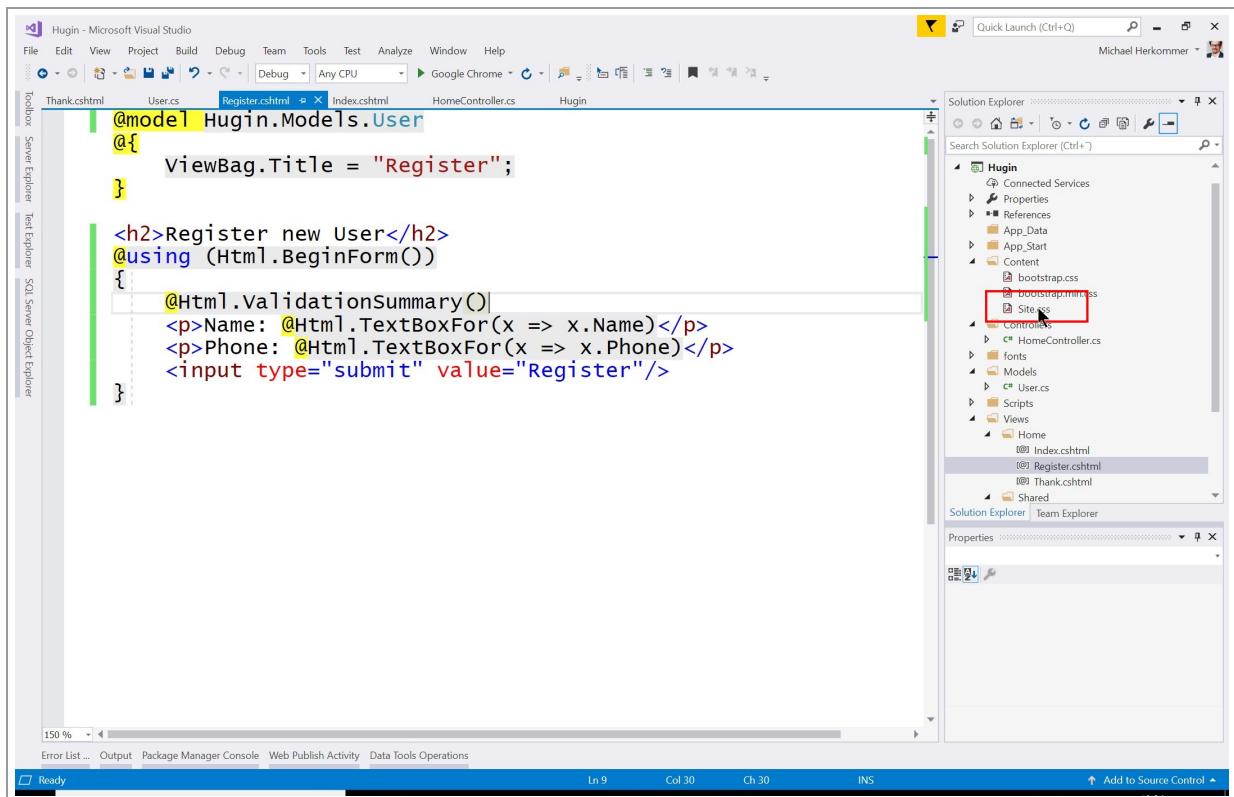
© 2018 - My ASP.NET Application


```

Click on "Chrome Legacy Window" pane in "Register - My ASP.NET Application"



Click on "Chrome Legacy Window" document in "Register - My ASP.NET Application"



Click on "Site.css" tree view item in "Solution Explorer"

```
body {
    padding-top: 50px;
    padding-bottom: 20px;
}

/* Set padding to keep content from hitting the edges */
.body-content {
    padding-left: 15px;
    padding-right: 15px;
}

/* Set width on the form input elements since they're 100% wide */
input,
select,
textarea {
    max-width: 280px;
}
```

Click on "Text Editor" edit in "Site.css"

```
body {
    padding-top: 50px;
    padding-bottom: 20px;
}

/* Set padding to keep content from hitting the edges */
.body-content {
    padding-left: 15px;
    padding-right: 15px;
}

/* Set width on the form input elements since they're 100% wide */
input,
select,
textarea {
    max-width: 280px;
}

.input-validation-error {
    background-color: red;
}
```

Click on "Text Editor" edit in "Site.css*"

The screenshot shows the Microsoft Visual Studio interface. In the center, the Text Editor displays the `Site.css` file with CSS code. On the right, the Solution Explorer shows the project structure, including files like `bootstrap.css`, `bootstrap.min.css`, and `Site.cshtml`. A red box highlights the `_Layout.cshtml` file under the `Shared` folder.

```
body { padding-top: 50px; padding-bottom: 20px; } /* Set padding to keep content from hitting the edges */ .body-content { padding-left: 15px; padding-right: 15px; } /* Set width on the form input elements since they're 100% wide */ input, select, textarea { max-width: 280px; } .input-validation-error { background-color: red; }
```

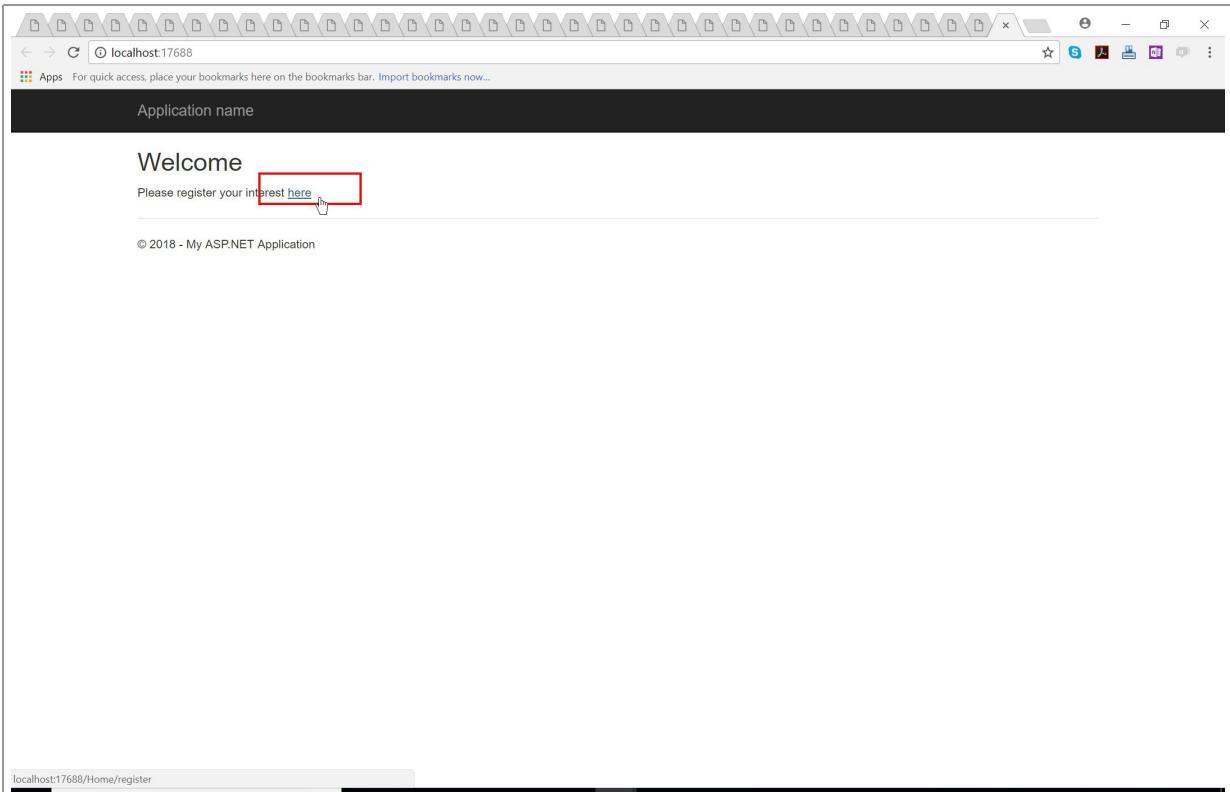
Click on "`_Layout.cshtml`" tree view item in "Solution Explorer"

The screenshot shows the Microsoft Visual Studio interface. In the center, the Text Editor displays the `_Layout.cshtml` file with HTML and C# code. On the right, the Solution Explorer shows the project structure, including files like `bootstrap.css`, `bootstrap.min.css`, and `Site.cshtml`. A red box highlights the `_Layout.cshtml` file under the `Shared` folder.

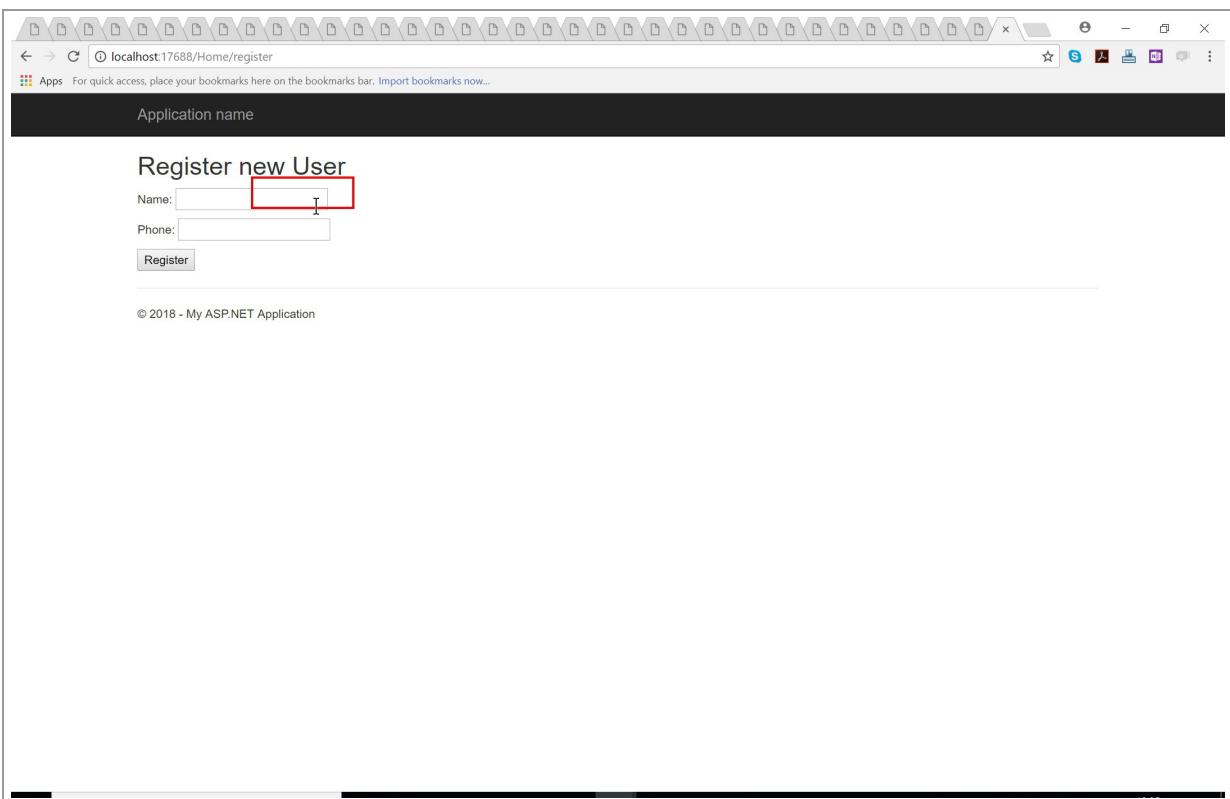
```
<!DOCTYPE html>
<html>
    <head>
        <meta charset="utf-8" />
        <meta name="viewport" content="width=device-width, initial-scale=1.0" />
        <title>@ViewBag.Title - My ASP.NET Application</title>
        <link href="~/Content/Site.css" rel="stylesheet" type="text/css" />
        <link href="~/Content/bootstrap.min.css" rel="stylesheet" type="text/css" />
        <script src="~/Scripts/modernizr-2.6.2.js"></script>
    </head>
    <body>
        <div class="navbar navbar-inverse navbar-fixed-top">
            <div class="container">
                <div class="navbar-header">
                    <button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#navbar-collapse">
                        <span class="icon-bar"></span>
                        <span class="icon-bar"></span>
                        <span class="icon-bar"></span>
                    </button>
                    @Html.ActionLink("Application name", "Index", "Home")
                </div>
                <div class="navbar-collapse collapse">
                    <ul class="nav navbar-nav">
                    </ul>
                </div>
            </div>
        </div>
    </body>

```

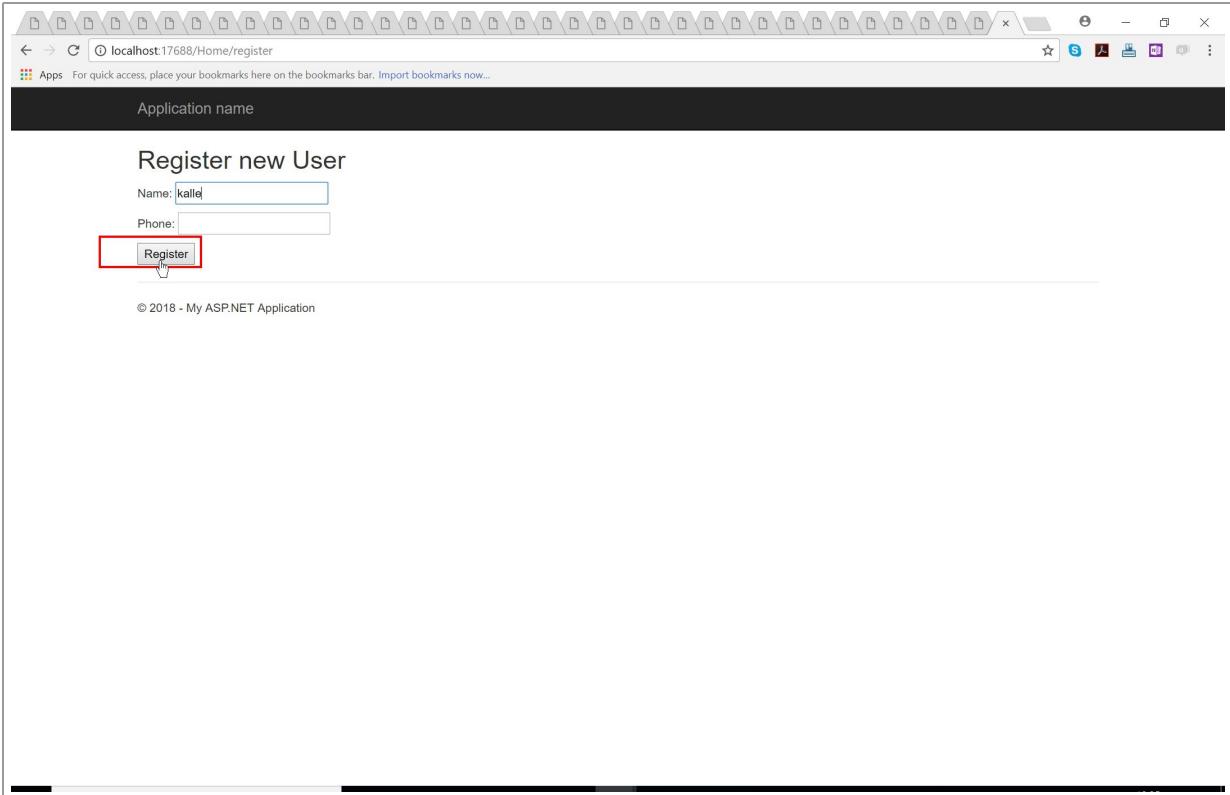
Click on "Text Editor" edit in "`_Layout.cshtml`"



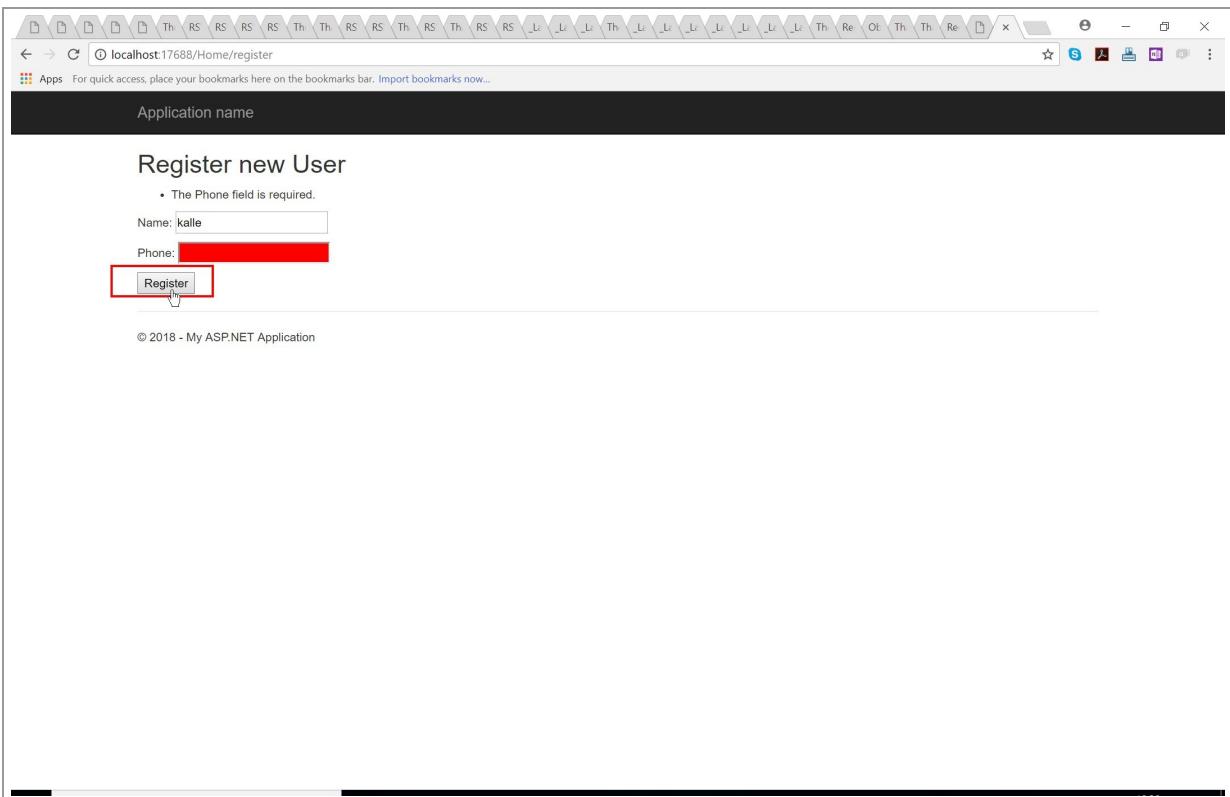
Click on "Chrome Legacy Window" pane in "Home - My ASP.NET Application"



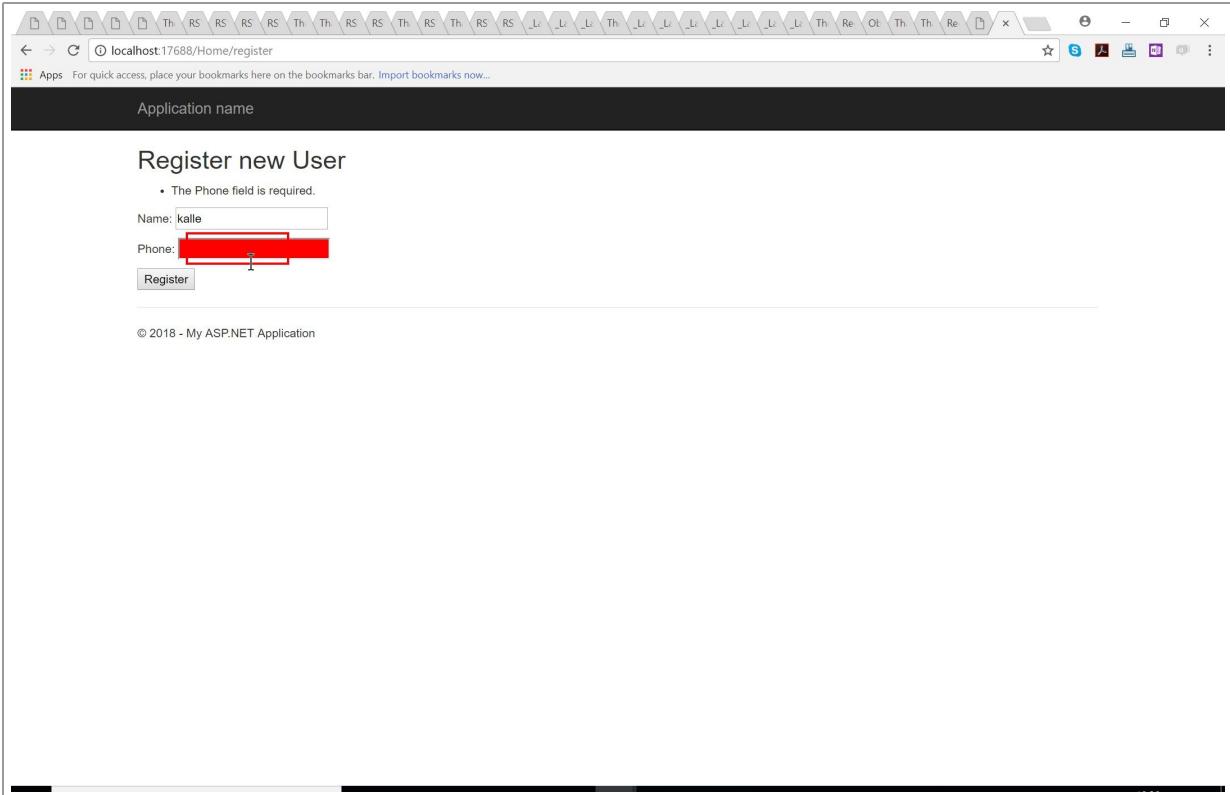
Click on "Chrome Legacy Window" pane in "Register - My ASP.NET Application"



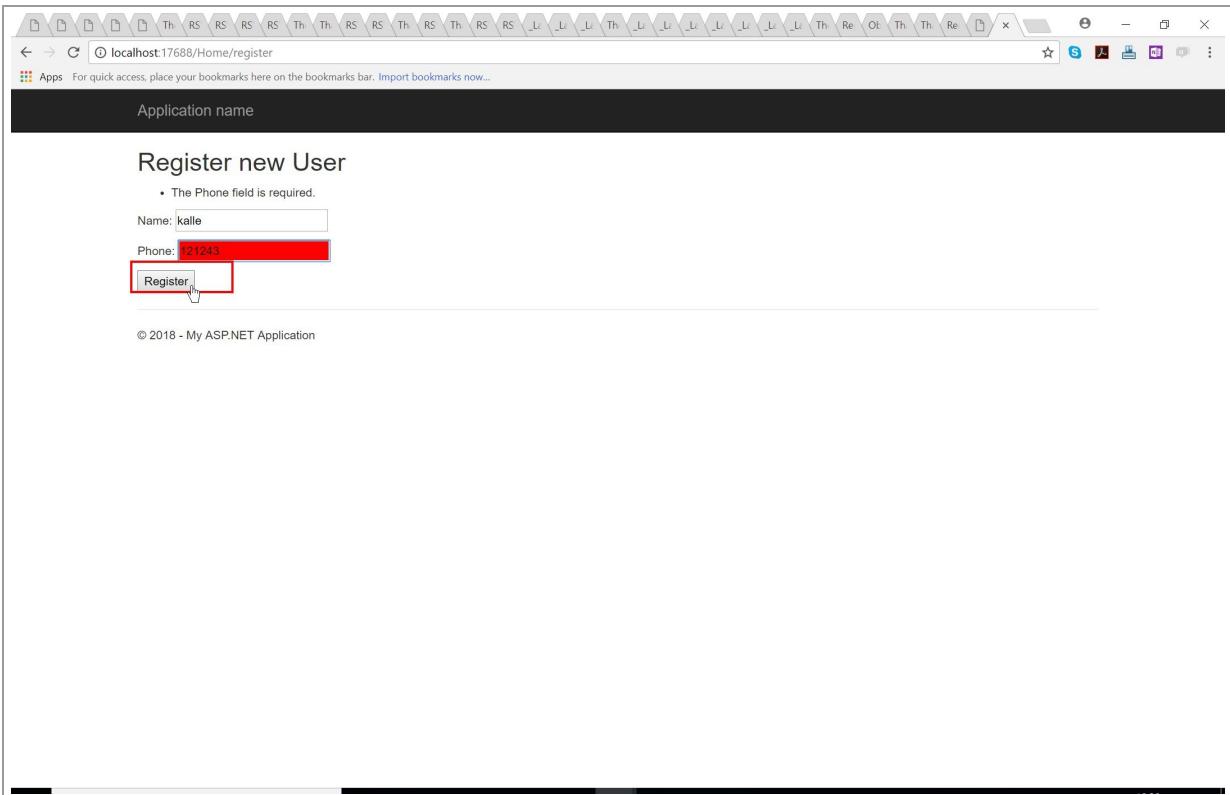
Click on "Chrome Legacy Window" document in "Register - My ASP.NET Application"



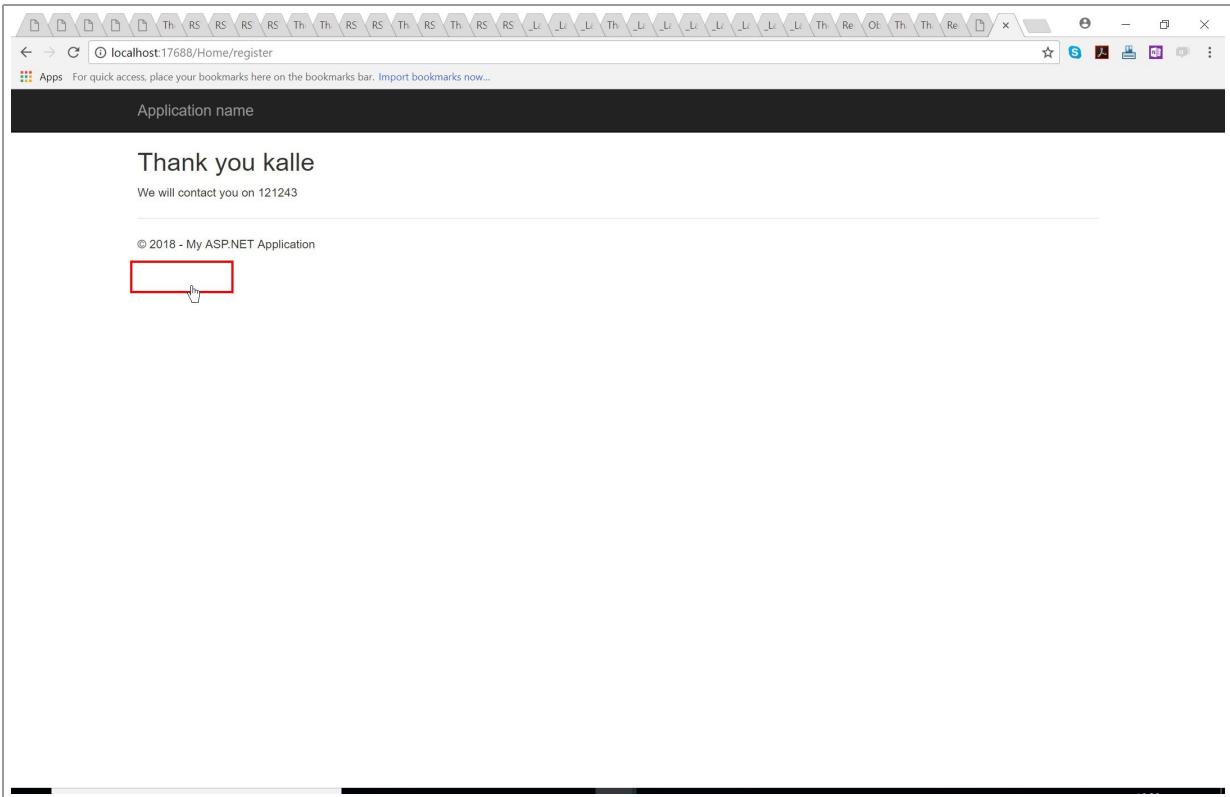
Click on "Chrome Legacy Window" document in "Register - My ASP.NET Application"



Click on "Chrome Legacy Window" document in "Register - My ASP.NET Application"



Click on "Chrome Legacy Window" document in "Register - My ASP.NET Application"



Click on "Chrome Legacy Window" document in "Thank - My ASP.NET Application"