**Dot Net Phase 4 – Custom Support Logger**

**Phase-End Project Problem Statement**

Github: <https://github.com/11812142/EndProject>

**Source Code**

**Program.cs**

using Microsoft.AspNetCore.Hosting;

using Microsoft.Extensions.Configuration;

using Microsoft.Extensions.Hosting;

using Microsoft.Extensions.Logging;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

namespace Loginpage

{

public class Program

{

public static void Main(string[] args)

{

CreateHostBuilder(args).Build().Run();

}

public static IHostBuilder CreateHostBuilder(string[] args) =>

Host.CreateDefaultBuilder(args)

.ConfigureWebHostDefaults(webBuilder =>

{

webBuilder.UseStartup<Startup>();

});

}

}

**Startup.cs**

using Microsoft.AspNetCore.Builder;

using Microsoft.AspNetCore.Hosting;

using Microsoft.AspNetCore.HttpsPolicy;

using Microsoft.Extensions.Configuration;

using Microsoft.Extensions.DependencyInjection;

using Microsoft.Extensions.Hosting;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

namespace Loginpage

{

public class Startup

{

public Startup(IConfiguration configuration)

{

Configuration = configuration;

}

public IConfiguration Configuration { get; }

// This method gets called by the runtime. Use this method to add services to the container.

public void ConfigureServices(IServiceCollection services)

{

services.AddControllersWithViews();

}

// This method gets called by the runtime. Use this method to configure the HTTP request pipeline.

public void Configure(IApplicationBuilder app, IWebHostEnvironment env)

{

if (env.IsDevelopment())

{

app.UseDeveloperExceptionPage();

}

else

{

app.UseExceptionHandler("/Home/Error");

// The default HSTS value is 30 days. You may want to change this for production scenarios, see https://aka.ms/aspnetcore-hsts.

app.UseHsts();

}

app.UseHttpsRedirection();

app.UseStaticFiles();

app.UseRouting();

app.UseAuthorization();

app.UseEndpoints(endpoints =>

{

endpoints.MapControllerRoute(

name: "default",

pattern: "{controller=Login}/{action=Login}/{id?}");

});

}

}

}

**LoginCotroller.cs**

using Microsoft.AspNetCore.Mvc;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

using Loginpage.Models;

namespace Loginpage.Controllers

{

public class LoginController : Controller

{

public IActionResult Login()

{

return View();

}

public List<UserModel> PutValue()

{

var users = new List<UserModel>

{

new UserModel{id=1,username="12345",password="pratibha"},

new UserModel{id=2,username="13579",password="admin"},

new UserModel{id=3,username="24680",password="pratibha2000"},

new UserModel{id=4,username="98765",password="abc"}

};

return users;

}

[HttpPost]

public IActionResult Verify(UserModel usr)

{

var u = PutValue();

var ue = u.Where(u => u.username.Equals(usr.username));

var up = ue.Where(p => p.password.Equals(usr.password));

if(up.Count()==1)

{

ViewBag.message = "Login Success";

return View("LoginSuccess");

}

else

{

ViewBag.message = "Login Failed";

return View("Login");

}

}

}

}

**ErrorViewModel.cs**

using System;

namespace Loginpage.Models

{

public class ErrorViewModel

{

public string RequestId { get; set; }

public bool ShowRequestId => !string.IsNullOrEmpty(RequestId);

}

}

**UserModel.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

namespace Loginpage.Models

{

public class UserModel

{

public int id { get; set; }

public string username { get; set; }

public string password { get; set; }

}

}

**Login.cshtml**

@model Loginpage.Models.UserModel;

@{

Layout = null;

}

<!DOCTYPE html>

<html>

<head>

<meta name="viewport" content="width=device-width, initial-scale=1">

<link href="//maxcdn.bootstrapcdn.com/bootstrap/4.1.1/css/bootstrap.min.css" rel="stylesheet" id="bootstrap-css">

<script src="//maxcdn.bootstrapcdn.com/bootstrap/4.1.1/js/bootstrap.min.js"></script>

<script src="//cdnjs.cloudflare.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>

<style>

body {

margin: 0;

padding: 0;

background-color: #17a2b8;

height: 100vh;

}

#login .container #login-row #login-column #login-box {

margin-top: 120px;

max-width: 400px;

height: 350px;

border: 1px solid #9C9C9C;

background-color: #EAEAEA;

}

#login .container #login-row #login-column #login-box #login-form {

padding: 20px;

}

#login .container #login-row #login-column #login-box #login-form #register-link {

margin-top: -85px;

}

</style>

</head>

<body>

<div id="login">

<div class="container">

<div id="login-row" class="row justify-content-center align-items-center">

<div id="login-column" class="col-md-6">

<div id="login-box" class="col-md-12">

@using (Html.BeginForm("Verify", "Login", FormMethod.Post))

{<h3 class="text-center text-info">Bank Account Login</h3>

<div class="form-group">

<label for="username" class="text-info">Account number</label><br>

@Html.TextBoxFor(m => m.username, new { @class = "form-control", @placeholder = "Enter Account Number", @required = true })

</div>

<div class="form-group">

<label for="password" class="text-info">Password:</label><br>

@Html.PasswordFor(m => m.password, new { @class = "form-control", @placeholder = "Enter Password", @required = true })

</div>

<div class="form-group">

<label for="remember-me" class="text-info"><span>Remember me</span> <span><input id="remember-me" name="remember-me" type="checkbox"></span></label><br>

<input type="submit" name="submit" class="btn btn-info btn-md" value="submit">

</div>

}

<div style="color:red;">@ViewBag.message</div>

</div>

</div>

</div>

</div>

</div>

</body>

</html>

**LoginSuccess.cshtml**

@\*

For more information on enabling MVC for empty projects, visit https://go.microsoft.com/fwlink/?LinkID=397860

\*@

@{

}

@{

Layout = null;

}

<html>

<head>

<title>Login Success</title>

<meta name="viewport" content="width=device-width, initial-scale=1">

<link href="//maxcdn.bootstrapcdn.com/bootstrap/4.1.1/css/bootstrap.min.css" rel="stylesheet" id="bootstrap-css">

<script src="//maxcdn.bootstrapcdn.com/bootstrap/4.1.1/js/bootstrap.min.js"></script>

<script src="//cdnjs.cloudflare.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>

<style>

body {

margin: 0;

padding: 0;

background-color: lightgreen;

height: 100vh;

}

#login .container #login-row #login-column #login-box {

margin-top: 120px;

max-width: 400px;

height: 350px;

border: 1px solid #9C9C9C;

background-color: #EAEAEA;

}

#login .container #login-row #login-column #login-box #login-form {

padding: 20px;

}

#login .container #login-row #login-column #login-box #login-form #register-link {

margin-top: -85px;

}

.centered {

position: fixed;

top: 50%;

left: 50%;

/\* bring your own prefixes \*/

transform: translate(-50%, -50%);

}

.centered {

position: fixed;

top: 50%;

left: 50%;

/\* bring your own prefixes \*/

transform: translate(-50%, -50%);

}

</style>

</head>

<body>

<div class="centered">

<b> <div class="container">@ViewBag.message</div></b>

<br/>

<div class="container">

@using (Html.BeginForm("Verify", "Login", FormMethod.Post))

{

<input type="submit" name="submit" class="btn btn-info btn-md" value="Log out">

}

</div>

</div>

</body>

</html>