Bahria University,

Karachi Campus



LAB EXPERIMENT NO.

3

LIST OF TASKS

|  |  |
| --- | --- |
| TASK NO | OBJECTIVE |
| 1 | Create login and registration form using simple MVC net core |
| 2 | Create feedback form using simple MVC net core. |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Submitted On:

**\_\_\_\_\_\_\_\_\_\_\_**

Task 01: Create login and registration form using simple MVC net core

Solution:

Model (User):

namespace CC\_Task3.Models

{

public class UserViewModel

{

public string? Name { get; set; }

public string Email { get; set; }

public string Password { get; set; }

}

}

View (Login):

@model CC\_Task3.Models.UserViewModel;

@\*

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

\*@

@{

}

<style>

.gradient-custom {

/\* fallback for old browsers \*/

background: #6a11cb;

/\* Chrome 10-25, Safari 5.1-6 \*/

background: -webkit-linear-gradient(to right, rgba(106, 17, 203, 1), rgba(37, 117, 252, 1));

/\* W3C, IE 10+/ Edge, Firefox 16+, Chrome 26+, Opera 12+, Safari 7+ \*/

background: linear-gradient(to right, rgba(106, 17, 203, 1), rgba(37, 117, 252, 1))

}

</style>

<section class="vh-100 gradient-custom">

<div class="container py-5 h-100">

<div class="row d-flex justify-content-center align-items-center h-100">

<div class="col-12 col-md-8 col-lg-6 col-xl-5">

<div class="card bg-dark text-white" style="border-radius: 1rem;">

<div class="card-body p-5 text-center">

<div class="mb-md-5 mt-md-4 pb-5">

<h2 class="fw-bold mb-2 text-uppercase">Login</h2>

<p class="text-white-50 mb-5">Please enter your login and password!</p>

<**form** method="post" **asp-action**="processLogin">

<div class="form-outline form-white mb-4">

<**input** **type**="email" id="typeEmailX" **asp-for**="Email" class="form-control form-control-lg" required />

<label class="form-label" for="typeEmailX">Email</label>

</div>

<div class="form-outline form-white mb-4">

<**input** **type**="password" id="typePasswordX" **asp-for**="Password" class="form-control form-control-lg" required />

<label class="form-label" for="typePasswordX">Password</label>

</div>

<p class="small mb-5 pb-lg-2">@ViewBag.Error</p>

<button class="btn btn-outline-light btn-lg px-5" type="submit">Login</button>

</**form**>

<div class="d-flex justify-content-center text-center mt-4 pt-1">

<a href="#!" class="text-white"><i class="fab fa-facebook-f fa-lg"></i></a>

<a href="#!" class="text-white"><i class="fab fa-twitter fa-lg mx-4 px-2"></i></a>

<a href="#!" class="text-white"><i class="fab fa-google fa-lg"></i></a>

</div>

</div>

<div>

<p class="mb-0">

Don't have an account? <button onclick="redirectToRegister()" class="btn btn-dark">Sign Up</button>

</p>

</div>

</div>

</div>

</div>

</div>

</div>

</section>

<script>

function redirectToRegister() {

window.location.href = "/Account/Register";

}

</script>

View(Register):

@model CC\_Task3.Models.UserViewModel;

@\*

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

\*@

@{

}

<style>

.gradient-custom {

/\* fallback for old browsers \*/

background: #6a11cb;

/\* Chrome 10-25, Safari 5.1-6 \*/

background: -webkit-linear-gradient(to right, rgba(106, 17, 203, 1), rgba(37, 117, 252, 1));

/\* W3C, IE 10+/ Edge, Firefox 16+, Chrome 26+, Opera 12+, Safari 7+ \*/

background: linear-gradient(to right, rgba(106, 17, 203, 1), rgba(37, 117, 252, 1))

}

</style>

<section class="vh-200 gradient-custom">

<div class="container py-5 h-100">

<div class="row d-flex justify-content-center align-items-center h-100">

<div class="col-12 col-md-8 col-lg-6 col-xl-5">

<div class="card bg-dark text-white" style="border-radius: 1rem;">

<div class="card-body p-5 text-center">

<**form** method="post" **asp-action**="processRegister">

<div class="mb-md-5 mt-md-4 pb-5">

<h2 class="fw-bold mb-2 text-uppercase">Register</h2>

<p class="text-white-50 mb-5">Please enter your details to register!</p>

<div class="form-outline form-white mb-4">

<**input** **asp-for**="Name" **type**="text" id="Name" class="form-control form-control-lg" required />

<label class="form-label" for="Name">Name</label>

</div>

<div class="form-outline form-white mb-4">

<**input** **asp-for**="Email" **type**="email" id="Email" class="form-control form-control-lg" required />

<label class="form-label" for="Email">Email</label>

</div>

<div class="form-outline form-white mb-4">

<**input** **asp-for**="Password" **type**="password" id="Password" class="form-control form-control-lg" required />

<label class="form-label" for="Password">Password</label>

</div>

<div class="form-outline form-white mb-4">

<input type="password" id="Confirm\_Password" class="form-control form-control-lg" />

<label class="form-label" for="Password">Confirm Password</label>

</div>

<p class="small mb-5 pb-lg-2">@ViewBag.Error</p>

<button class="btn btn-outline-light btn-lg px-5" type="submit">Register</button>

<div class="d-flex justify-content-center text-center mt-4 pt-1">

<a href="#!" class="text-white"><i class="fab fa-facebook-f fa-lg"></i></a>

<a href="#!" class="text-white"><i class="fab fa-twitter fa-lg mx-4 px-2"></i></a>

<a href="#!" class="text-white"><i class="fab fa-google fa-lg"></i></a>

</div>

</div>

</**form**>

<div>

<p class="mb-0">

Already have an account? <button onclick="redirectToLogin()" class="btn btn-dark">Login</button>

</p>

</div>

</div>

</div>

</div>

</div>

</div>

</section>

<script>

function redirectToLogin() {

window.location.href="Account/Login"

}

</script>

Controller(Account):

using CC\_Task3.Models;

using Microsoft.AspNetCore.Mvc;

namespace CC\_Task3.Controllers

{

public class AccountController : Controller

{

private static List<UserViewModel> \_users = new List<UserViewModel>();

public IActionResult Login()

{

return View();

}

public IActionResult Register()

{

return View();

}

[HttpPost]

public IActionResult processRegister(UserViewModel user)

{

if (user == null)

{

ViewBag.Error = "Fill all fileds!";

return View("Register");

}

//add confirm password condition

if(!ModelState.IsValid)

{

ViewBag.Error = "Incorrect formate";

return View("Register");

}

if (emailExist(user.Email))

{

ViewBag.Error = "Email already exist!";

return View("Register");

}

\_users.Add(user);

ViewBag.Error = "Account Created Successfully";

return View("Login");

}

[HttpPost]

public IActionResult processLogin(UserViewModel user)

{

UserViewModel userToLogin = UserExist(user.Email, user.Password);

if (userToLogin != null)

{

TempData["Name"] = userToLogin.Name;

return RedirectToAction("Index","Home");

}

ViewBag.Error = "Incorrect Email or Password";

return View("Login");

}

public UserViewModel UserExist(string email, string password)

{

return \_users.Find(user => user.Email == email && user.Password == password);

}

public bool emailExist(string email)

{

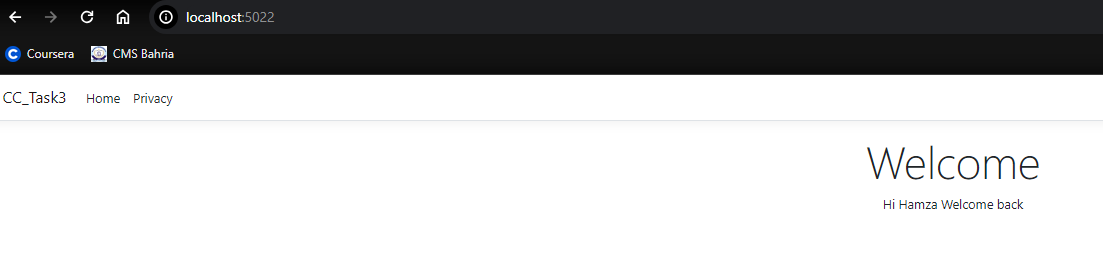
return \_users.Exists(user => user.Email == email);

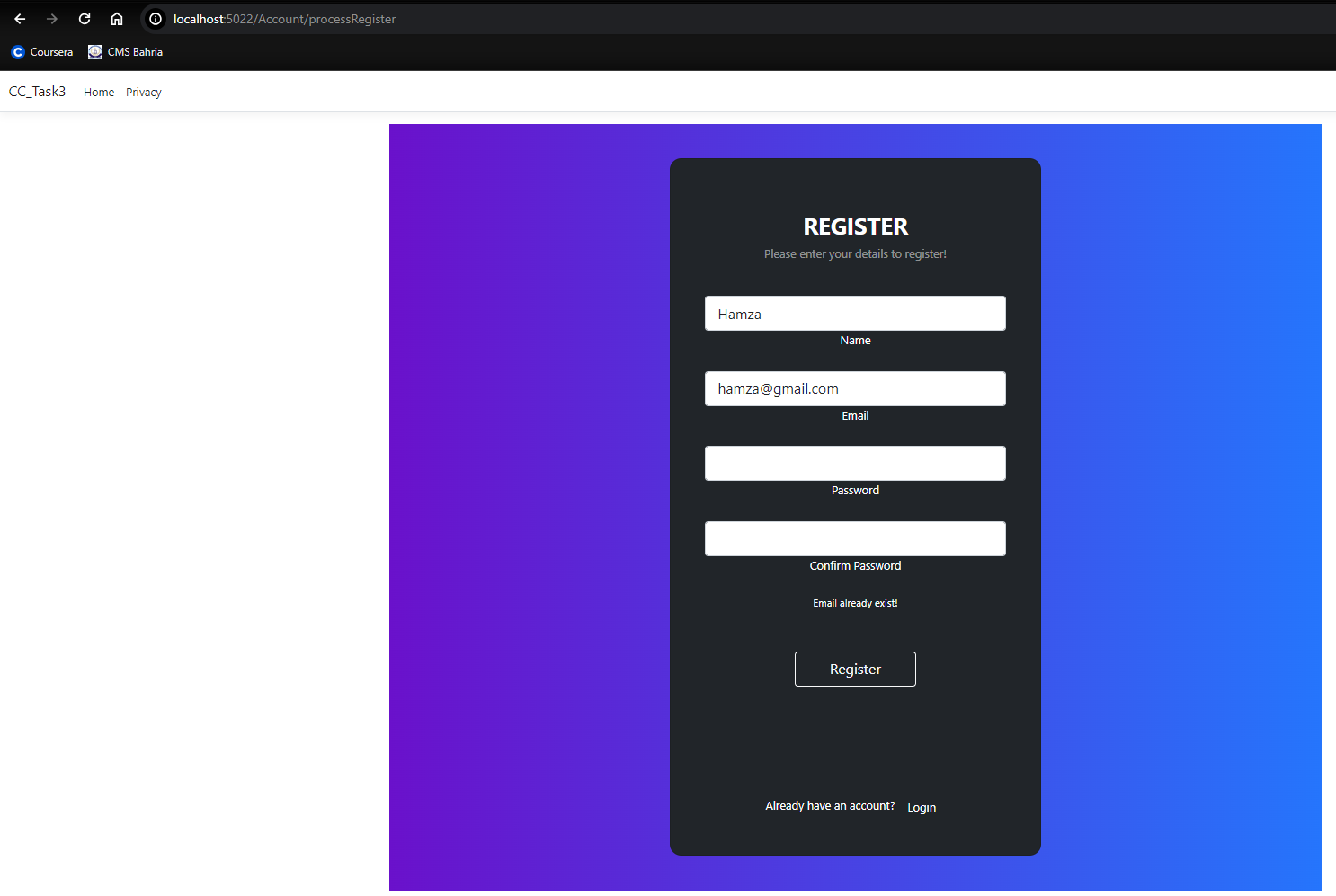
}

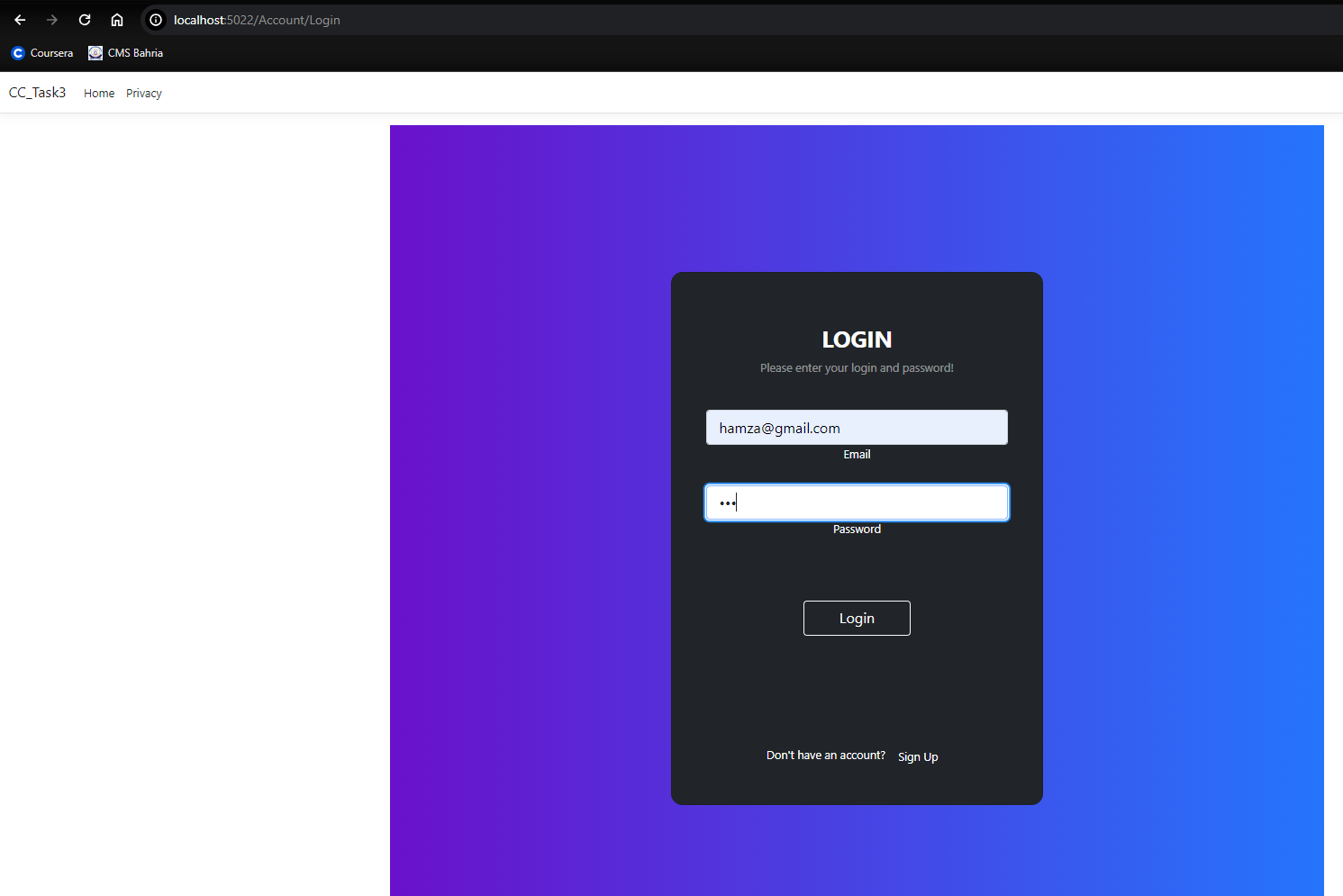
}

}

Output:







Task 02: Create feedback form using simple MVC net core.

Solution:

Model:

namespace CC\_Task3.Models

{

public class Feedback

{

public string Email { get; set; }

public string Rating { get; set; }

public string Message { get; set; }

}

}

View:

@model CC\_Task3.Models.Feedback;

@\*

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

\*@

@{

}

<div class="mx-0 mx-sm-auto">

<div class="card">

<div class="card-header bg-primary">

<h5 class="card-title text-white mt-2" id="exampleModalLabel">Feedback request</h5>

</div>

<**form** class="px-4" **asp-action**="ProcessFeedback" method="post">

<div class="modal-body">

<div class="text-center">

<i class="far fa-file-alt fa-4x mb-3 text-primary"></i>

<p>

@{

if (TempData.ContainsKey("Name"))

{

<strong>Hi @TempData["Name"] Your opinion matters</strong>

}

else{

<strong>Hi Your opinion matters</strong>

}

}

</p>

<p>

Have some ideas how to improve our product?

<strong>Give us your feedback.</strong>

</p>

</div>

<hr />

<p class="text-center"><strong>Your rating:</strong></p>

<**input** **value**="@TempData["Email"]" **asp-for**="Email" hidden/>

<div class="form-check mb-2">

<**input** class="form-check-input" **type**="radio" **name**="Rating" id="Rating" **asp-for**="Rating" **value**="Very good" />

<label class="form-check-label" for="radio3Example1">

Very good

</label>

</div>

<div class="form-check mb-2">

<**input** class="form-check-input" **type**="radio" **name**="Rating" id="Rating" **asp-for**="Rating" **value**="Good" />

<label class="form-check-label" for="radio3Example2">

Good

</label>

</div>

<div class="form-check mb-2">

<**input** class="form-check-input" **type**="radio" **name**="Rating" id="Rating" **asp-for**="Rating" **value**="Medicore" />

<label class="form-check-label" for="radio3Example3">

Medicore

</label>

</div>

<div class="form-check mb-2">

<**input** class="form-check-input" **type**="radio" **name**="Rating" id="Rating" **asp-for**="Rating" **value**="Bad" />

<label class="form-check-label" for="radio3Example4">

Bad

</label>

</div>

<div class="form-check mb-2">

<**input** class="form-check-input" **type**="radio" **name**="Rating" id="Rating" **asp-for**="Rating" **value**="Very bad" />

<label class="form-check-label" for="radio3Example5">

Very bad

</label>

</div>

<p class="text-center"><strong>What could we improve?</strong></p>

<!-- Message input -->

<div class="form-outline mb-4">

<**textarea** class="form-control" id="form4Example3" rows="4" **asp-for**="Message" required></**textarea**>

<label class="form-label" for="form4Example3">Your feedback</label>

</div>

</div>

<div class="card-footer text-end">

@{

if (TempData.ContainsKey("Error"))

{

<strong>@TempData["Error"]</strong>

}

}

<button type="submit" class="btn btn-primary">Submit</button>

</div>

</**form**>

</div>

</div>

Controller(HomeController):

public IActionResult Feedback()

{

TempData.Keep("Email");

TempData.Keep("Name");

return View();

}

public IActionResult ProcessFeedback(Feedback feedback)

{

if(feedback == null)

{

TempData["Error"] = "Null feedback";

return View("Feedback");

}

if (feedback.Email == null)

{

TempData["Error"] = "First Login to submit feedback! ";

return View("Feedback");

}

TempData["Error"] = "Your Feedback submitted";

return View("Feedback");

}

Output:

