**Bahria University,**

**Karachi Campus**



**LAB EXPERIMENT NO.**

**\_13**

**LIST OF TASKS**

|  |  |
| --- | --- |
| TASK NO | **OBJECTIVE** |
| 01 | Develop a Two-Factor Authentication (2FA) Micro service using Google Authenticator in ASP.NET MVC for User Account Security Enhancement. |
|  |  |

**Submitted On:**

**Date: 16/May/2024**

**TASK:** Develop a Two-Factor Authentication (2FA) Micro service using Google Authenticator in ASP.NET MVC for User Account Security Enhancement.

**LOGIN.CS:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

namespace CC\_LAB\_10.Models

{

public class LoginModel

{

public string UserName { get; set; }

public string Password { get; set; }

}

}

@model CC\_LAB\_10.Models.LoginModel

@{

ViewBag.Title = "Login";

}

<center>

<h2>Login Page</h2>

@if (ViewBag.Status == null || !ViewBag.Status)

{

<div>@ViewBag.Message</div>

<div>

@using (Html.BeginForm())

{

<div class="form-group">

<label for="UserName">UserName : </label>

@Html.TextBoxFor(a => a.UserName, new { @class = "form-control" })

</div>

<div class="form-group">

<label for="Password">Password : </label>

@Html.TextBoxFor(a => a.Password, new

{

@class = "form-control",

type = "Password"

})

</div>

<input type="submit" value="Login" class="btn btn-default" />

}

</div>

}

else

{

<div>@ViewBag.Message</div>

<div>

<img src="@ViewBag.BarcodeImageUrl" width="300" height="300" />

</div>

<div>

Manual Setup Code : @ViewBag.SetupCode

</div>

<div>

@using (Html.BeginForm("TwoFactorAuthenticate", "Home", FormMethod.Post))

{

<input type="text" name="CodeDigit" />

<input type="submit" class="btn btn-success" />

}

</div>

}

</center>

**HOME CONTROLLER:**

using CC\_LAB\_10.Models;

using Google.Authenticator;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Web;

using System.Web.Configuration;

using System.Web.Mvc;

using System.Web.Security;

using System.Web.UI.WebControls;

namespace CC\_LAB\_10.Controllers

{

public class HomeController : Controller

{

public ActionResult Index()

{

if (Session["Username"] == null || Session["IsValidTwoFactorAuthentication"] ==

null || !(bool)Session["IsValidTwoFactorAuthentication"])

{

return RedirectToAction("Login");

}

return View();

}

public ActionResult About()

{

if (Session["Username"] == null || Session["IsValidTwoFactorAuthentication"] ==

null || !(bool)Session["IsValidTwoFactorAuthentication"])

{

return RedirectToAction("Login");

}

ViewBag.Message = "Your application description page.";

return View();

}

public ActionResult Contact()

{

if (Session["Username"] == null || Session["IsValidTwoFactorAuthentication"] ==

null || !(bool)Session["IsValidTwoFactorAuthentication"])

{

return RedirectToAction("Login");

}

ViewBag.Message = "Your contact page.";

return View();

}

public ActionResult Login()

{

Session["UserName"] = null;

Session["IsValidTwoFactorAuthentication"] = null;

return View();

}

[HttpPost]

public ActionResult Login(LoginModel login)

{

bool status = false;

if (Session["Username"] == null || Session["IsValidTwoFactorAuthentication"] == null ||

!(bool)Session["IsValidTwoFactorAuthentication"])

{

string googleAuthKey = WebConfigurationManager.AppSettings["GoogleAuthKey"];

string UserUniqueKey = (login.UserName + googleAuthKey);

if (login.UserName == "Aiman" && login.Password == "54321")

{

Session["UserName"] = login.UserName;

TwoFactorAuthenticator TwoFacAuth = new TwoFactorAuthenticator();

var setupInfo = TwoFacAuth.GenerateSetupCode("UdayDodiyaAuthDemo.com",

login.UserName, ConvertSecretToBytes(UserUniqueKey, false), 300);

Session["UserUniqueKey"] = UserUniqueKey;

ViewBag.BarcodeImageUrl = setupInfo.QrCodeSetupImageUrl;

ViewBag.SetupCode = setupInfo.ManualEntryKey;

status = true;

}

}

else

{

return RedirectToAction("Index");

}

ViewBag.Status = status;

return View();

}

private static byte[] ConvertSecretToBytes(string secret, bool secretIsBase32) =>

secretIsBase32 ? Base32Encoding.ToBytes(secret) : Encoding.UTF8.GetBytes(secret);

public ActionResult TwoFactorAuthenticate()

{

var token = Request["CodeDigit"];

TwoFactorAuthenticator TwoFacAuth = new TwoFactorAuthenticator();

string UserUniqueKey = Session["UserUniqueKey"].ToString();

bool isValid = TwoFacAuth.ValidateTwoFactorPIN(UserUniqueKey, token, false);

if (isValid)

{

HttpCookie TwoFCookie = new HttpCookie("TwoFCookie");

string UserCode

= Convert.ToBase64String(MachineKey.Protect(Encoding.UTF8.GetBytes(UserUniqueKey)));

Session["IsValidTwoFactorAuthentication"] = true;

return RedirectToAction("Index");

}

ViewBag.Message = "Google Two Factor PIN is expired or wrong";

return RedirectToAction("Login");

}

public ActionResult Logoff()

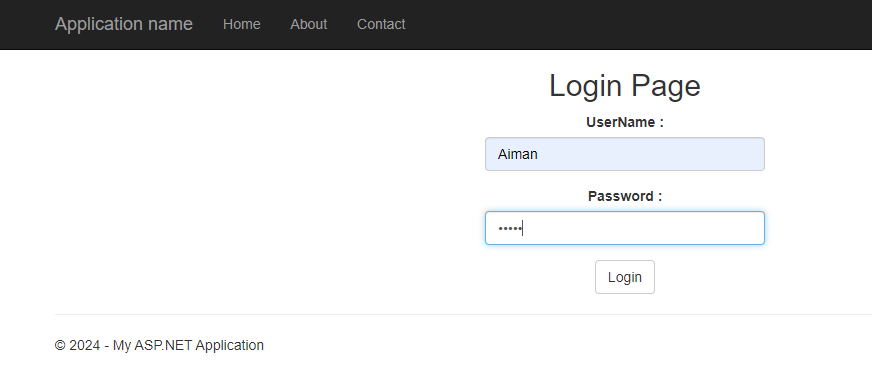
{

Session["UserName"] = null;

Session["IsValidTwoFactorAuthentication"] = null;

return RedirectToAction("Login");

} } }



A qr code on a screen

Description automatically generated

