Bahria University,

Karachi Campus



LAB EXPERIMENT NO.

**12**

LIST OF TASKS

|  |  |
| --- | --- |
| TASK NO | OBJECTIVE |
| **1** | **Two Factor Authentication for Login** |
| **2** | **Two Factor Authentication for signup using google authentication** |
|  |  |

**Task No. 1: Two Factor Authentication for Login**

**Solution:**

**Home Controller:**

namespace GoogleAuthentication.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);

//Take UserName And Password As Static - Admin As User And 12345 As Password

if (login.UserName == "Admin" && login.Password == "12345")

{

Session["UserName"] = login.UserName;

//Two Factor Authentication Setup

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");

}

}

}

**Login html**

@model GoogleAuthentication.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" />

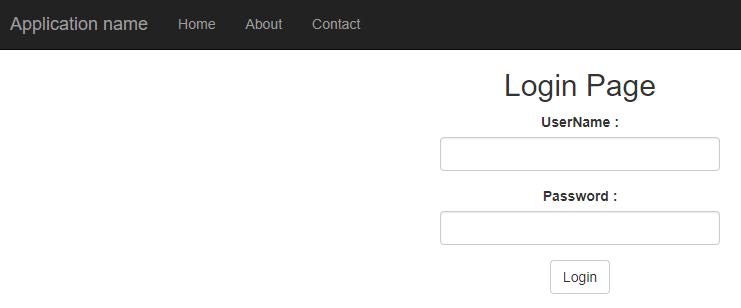
}

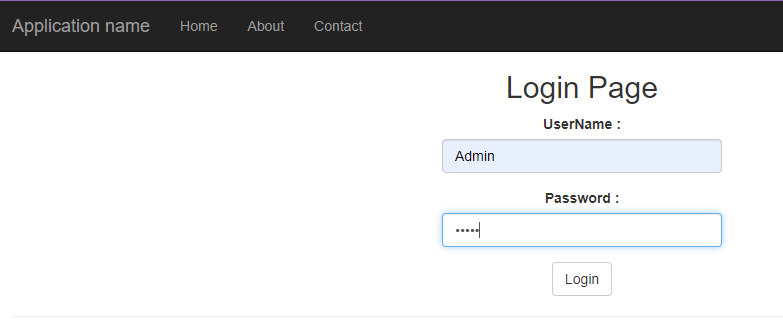
</div>

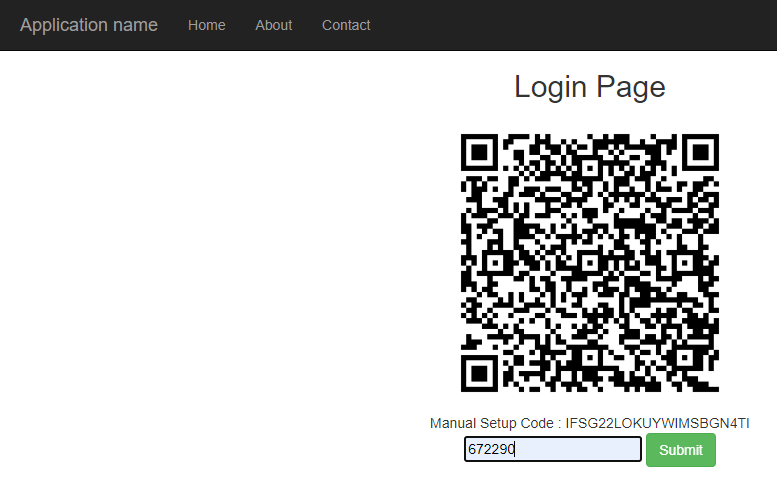
}

</center>

**Output:**







A screenshot of a computer

Description automatically generated with medium confidence

**Task No. 2: Two Factor Authentication for signup using google authentication.**

**Class Code:**

namespace Authentication

{

public partial class SignUp : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

}

protected void btnSignUp\_Click(object sender, EventArgs e)

{

string email = txtEmail.Text;

string password = txtPassword.Text;

string secretKey = GenerateSecretKey();

string qrCodeUrl = GenerateQrCodeUrl(email, secretKey);

Session["SecretKey"] = secretKey;

Response.Redirect(qrCodeUrl);

}

private string GenerateSecretKey()

{

var secretKeyBytes = KeyGeneration.GenerateRandomKey();

var base32Encoder = new Base32Encoder();

var secretKey = base32Encoder.EncodeData(secretKeyBytes);

return secretKey;

}

private string GenerateQrCodeUrl(string email, string secretKey)

{

var authenticatorUri = $"otpauth://totp/{email}?secret={secretKey}&issuer=Authentication";

return $"https://chart.googleapis.com/chart?chs=200x200&chld=M|0&cht=qr&chl={Uri.EscapeDataString(authenticatorUri)}";

}

}

}

**Output:**

A picture containing text, screenshot, font, line

Description automatically generated

muazshahzad@gmail.com