**PapersController**

IPG521\_SA.Data;

using IPG521\_SA.Models;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

namespace IPG521\_SA.Controllers

{

[Authorize(Roles = "Client")]

public class PApersController : Controller

{

PApersDbContext PApersDb = new PApersDbContext();

// GET: Product

[AllowAnonymous]

public ActionResult Index()

{

return View(PApersDb.Papers.ToList());

}

// GET: Product

[AllowAnonymous]

public ActionResult List()

{

var paper = PApersDb.Papers.Where(s => s.PaperAuthor == User.Identity.Name);

return View();

}

public ActionResult Create()

{

return View();

}

[HttpPost]

public ActionResult Create(Papers paper, FormCollection formCollection)

{

if (ModelState.IsValid)

{

paper.PaperAuthor = User.Identity.Name;

paper.TopicID = int.Parse(paper.TopicID.ToString());

paper.PaperDateSubmitted = DateTime.Now.ToShortDateString();

PApersDb.Papers.Add(paper);

PApersDb.SaveChanges();

return RedirectToAction("List");

}

return View(paper);

}

[HttpGet]

public ActionResult Edit(int id)

{

var papers = PApersDb.Papers.Find(id);

return View(papers);

}

[HttpPost]

public ActionResult Edit(int id, FormCollection collection)

{

var papers = PApersDb.Papers.Find(id);

if (ModelState.IsValid)

{

if (TryUpdateModel(papers))

{

PApersDb.SaveChanges();

return RedirectToAction("List");

}

else

ModelState.AddModelError("Database Failed to update", "");

}

return View(papers);

}

[HttpGet]

public ActionResult Delete(int id)

{

var papers = PApersDb.Papers.Find(id);

return View(papers);

}

[HttpPost]

public ActionResult Delete(int id, FormCollection collection)

{

try

{

var papers = PApersDb.Papers.Find(id);

PApersDb.Papers.Remove(papers);

PApersDb.SaveChanges();

return RedirectToAction("List");

}

catch (Exception ex)

{

ModelState.AddModelError(ex.Message, "");

}

return View();

}

}

}

**Home Controller**

using IPG521\_SA.Data;

using IPG521\_SA.Models;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

namespace IPG521\_SA.Controllers

{

public class HomeController : Controller

{

PApersDbContext PApersDb = new PApersDbContext();

public ActionResult Index()

{

return View();

}

public ActionResult News()

{

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

return View();

}

public ActionResult Contact()

{

ViewBag.Message = "Your contact page.";

return View();

}

public ActionResult Create2()

{

return View();

}

[HttpPost]

public ActionResult Create2(Papers paper, FormCollection formCollection)

{

if (ModelState.IsValid)

{

paper.PaperAuthor = User.Identity.Name;

paper.TopicID = int.Parse(paper.TopicID.ToString());

paper.PaperDateSubmitted = DateTime.Now.ToShortDateString();

PApersDb.Papers.Add(paper);

PApersDb.SaveChanges();

return RedirectToAction("List");

}

return View(paper);

}

}

}

**Cofig Identity**

namespace IPG521\_SA.Migrations

{

using IPG521\_SA.Models;

using Microsoft.AspNet.Identity;

using Microsoft.AspNet.Identity.EntityFramework;

using System;

using System.Data.Entity;

using System.Data.Entity.Migrations;

using System.Linq;

internal sealed class Configuration\_Identity : DbMigrationsConfiguration<IPG521\_SA.Models.ApplicationDbContext>

{

public Configuration\_Identity()

{

AutomaticMigrationsEnabled = true;

}

protected override void Seed(IPG521\_SA.Models.ApplicationDbContext context)

{

var store = new RoleStore<IdentityRole>(context);

var manager = new RoleManager<IdentityRole>(store);

var roleAdmin = new IdentityRole { Name = "Admin" };

var roleUser = new IdentityRole { Name = "Client" };

if (!manager.RoleExists(roleAdmin.Name))

manager.Create(roleAdmin);

if (!manager.RoleExists(roleUser.Name))

manager.Create(roleUser);

// Default user

UserStore<ApplicationUser> userStore = new UserStore<ApplicationUser>(context);

var userManager = new UserManager<ApplicationUser>(userStore);

ApplicationUser adminUser = new ApplicationUser

{

Email = "admin@ctu.co.za",

UserName = "admin@ctu.co.za"

};

//Admin user password

var results = userManager.Create(adminUser, "Secret123$");

if (results.Succeeded)

userManager.AddToRoles(adminUser.Id, roleAdmin.Name);

base.Seed(context);

}

}

}

**Configurations**

namespace IPG521\_SA.Migrations

{

using IPG521\_SA.Models;

using System;

using System.Data.Entity;

using System.Data.Entity.Migrations;

using System.Linq;

internal sealed class Configuration : DbMigrationsConfiguration<IPG521\_SA.Data.PApersDbContext>

{

public Configuration()

{

AutomaticMigrationsEnabled = false;

}

protected override void Seed(IPG521\_SA.Data.PApersDbContext context)

{

context.Topics.AddOrUpdate(

s => s.TopicID,

new Topics { TopicName = "Data Science" },

new Topics { TopicName = "Education" },

new Topics { TopicName = "Human-Computer Interaction" },

new Topics { TopicName = "Internet of Things" });

}

}

}

**PapersDbContext**

using IPG521\_SA.Models;

using System;

using System.Collections.Generic;

using System.Data.Entity;

using System.Linq;

using System.Web;

namespace IPG521\_SA.Data

{

public class PApersDbContext : DbContext

{

public PApersDbContext()

: base("PaperConnection")

{ }

public DbSet<Papers> Papers { get; set; }

public DbSet<Topics> Topics { get; set; }

}

}

**Topics**

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations;

using System.Linq;

using System.Web;

using System.Web.ModelBinding;

namespace IPG521\_SA.Models

{

public class Topics

{

[Key]

[BindNever]

public int TopicID { get; set; }

[Display(Name = "Topic")]

[Required]

public string TopicName { get; set; }

}

}

**Papers**

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations;

using System.Linq;

using System.Web;

using System.Web.ModelBinding;

namespace IPG521\_SA.Models

{

public class Papers

{

[Key]

[BindNever]

[Display(Name = "ID")]

public int PaperId { get; set; }

[Required]

[Display(Name = "Title")]

public string PaperTitle { get; set; }

[Required]

[Display(Name = "Abstract")]

public string PaperAbstract { get; set; }

[Required]

[Display(Name = "Author")]

public string PaperAuthor { get; set; }

[Display(Name = "Submission Date")]

public string PaperDateSubmitted { get; set; }

[Required]

[BindNever]

[Display(Name = "Topic")]

public int TopicID { get; set; }

}

}

**Contact**

<body style="background-image: url('../../Pics/Contact1 (1).jpg'); ">

<h3 style="color:white;">Physical Address:</h3>

<h4 style="color:white;">Southern African Computer Lecturer’s Association (SACLA)</h4>

<h4 style="color:white;">School of Computer Science and Information Systems</h4>

<h4 style="color:white;">North-West University</h4>

<h4 style="color:white;">11 Hoffman Street</h4>

<h4 style="color:white;">Potchefstroom</h4>

<h4 style="color:white;">2531</h4>

<h3 style="color:white;">Postal Address:</h3>

<h4 style="color:white;">Southern African Computer Lecturer’s Association (SACLA)</h4>

<h4 style="color:white;">School of Computer Science and Information Systems</h4>

<h4 style="color:white;">Private Bag X6001</h4>

<h4 style="color:white;">Potchefstroom</h4>

<h4 style="color:white;">2520</h4>

</body>

**Index**

@{

ViewBag.Title = "Home Page";

}

<div class="jumbotron">

<h2 style="font-size:46.5px">SACLA Conference Paper Management System</h2>

</div>

<div class="row">

<div class="col-md-4">

<h2>Welcome to SACLA</h2>

<p style="font-size:21px">

This is a application that was created for assessment purpose as part of the CSIS3734/CSIQ3734 module. All rights reserved.

</p>

</div>

<div class="col-md-4">

<h2>News</h2>

<p style="font-size:21px">Visit our News page to find out what is happening.</p>

<p><a class="btn btn-primary" href="@Url.Action("News","Home")">News &raquo;</a></p>

</div>

<div class="col-md-4">

<h2>Contact us</h2>

<p style="font-size:20px">Give us a call or drop us an email to find out more about upcoming confereces and workshops.</p>

<p><a class="btn btn-primary" href="@Url.Action("Contact","Home")">See contact details &raquo;</a></p>

</div>

</div>

News

@{

ViewBag.Title = "News";

}

<h3 style="color:white">News</h3>

<body style="background-image: url('../../Pics/news (1).jpg');">

<p style="color:white">

SACLA (the Southern African Computer Lecturers’ Association) is a relatively informal

association of people involved in the teaching of Computer Science and Information Systems

at universities throughout Southern Africa. The main activity of SACLA is the annual conference,

at which issues concerning teaching and research in ICT in Southern Africa are discussed.

</p>

<style>

table {

font-family: arial, sans-serif;

border-collapse: collapse;

width: 100%;

background-color:black;

}

td, th {

border: 1px solid #dddddd;

text-align: left;

padding: 8px;

}

</style>

<h3 style="color:yellow">The following is dates & Place for the upcoming conferences </h3>

<table style="color:yellow">

<tr>

<th>Place</th>

<th>Speaker</th>

<th>Date</th>

</tr>

<tr>

<td>CTU</td>

<td>Mac van der Merwe</td>

<td>25 July 2021</td>

</tr>

<tr>

<td>NWU</td>

<td>Prof Jean Greyling</td>

<td>30 September 2021</td>

</tr>

<tr>

<td>UNISA</td>

<td>Prof Bruce Watson</td>

<td>4 October 2021</td>

</tr>

</table>

</body>

**Create**

@model IPG521\_SA.Models.Papers

@{

ViewBag.Title = "Create";

}

<h2>Submitted New Papers</h2>

@using (Html.BeginForm())

{

@Html.AntiForgeryToken()

<div class="form-horizontal">

<h4>Paper</h4>

@Html.ValidationMessageFor(model => model.PaperTitle, "", new { @class = "text-danger" })

@Html.ValidationMessageFor(model => model.PaperAbstract, "", new { @class = "text-danger" })

@Html.ValidationMessageFor(model => model.TopicID, "", new { @class = "text-danger" })

<hr />

@Html.ValidationSummary(true, "", new { @class = "text-danger" })

<h4 style="left:-200px">Title</h4>

<div class="form-group">

<div class="col-md-10">

@Html.EditorFor(model => model.PaperTitle, new { htmlAttributes = new { @class = "form-control" } })

</div>

</div>

<h4 style="left:-20px">Abstract</h4>

<div class="form-group">

<div class="col-md-10">

@Html.EditorFor(model => model.PaperAbstract, new { htmlAttributes = new { @class = "form-control" } })

</div>

</div>

<h4 style="left:-20px">Topic</h4>

<div class="form-group">

<div class="col-md-10" style="height:50px ">

@Html.DropDownListFor(model => model.TopicID,

new SelectList

(

new List<SelectListItem> {

new SelectListItem { Selected = true, Text = "-- Select Topic --", Value = "-1" },

new SelectListItem { Selected = false, Text = "Data Science", Value = "1" },

new SelectListItem { Selected = false, Text = "Education", Value = "2"},

new SelectListItem { Selected = false, Text = "Human-Computer Interaction", Value = "3" },

new SelectListItem { Selected = false, Text = "Internet of Things", Value = "4"}

}, "Value", "Text")

)

</div>

</div>

<div class="form-group">

<div class="col-md-10">

<input type="submit" value="Submit" class="btn btn-primary" style="background-color:blue" />

<a class="btn btn-default" href="@Url.Action("Index","Home")" style="border-color:blue; color:blue;">Back to List</a>

</div>

</div>

</div>

}

@section Scripts {

@Scripts.Render("~/bundles/jqueryval")

}

**Delete**

@model IPG521\_SA.Models.Papers

@{

ViewBag.Title = "Delete";

}

<h2>Delete</h2>

<h3>Are you sure you want to delete this?</h3>

<div>

<h4>Papers</h4>

<hr />

<dl class="dl-horizontal">

<dt>

@Html.DisplayNameFor(model => model.PaperTitle)

</dt>

<dd>

@Html.DisplayFor(model => model.PaperTitle)

</dd>

<dt>

@Html.DisplayNameFor(model => model.PaperAbstract)

</dt>

<dd>

@Html.DisplayFor(model => model.PaperAbstract)

</dd>

<dt>

@Html.DisplayNameFor(model => model.PaperAuthor)

</dt>

<dd>

@Html.DisplayFor(model => model.PaperAuthor)

</dd>

<dt>

@Html.DisplayNameFor(model => model.PaperDateSubmitted)

</dt>

<dd>

@Html.DisplayFor(model => model.PaperDateSubmitted)

</dd>

<dt>

@Html.DisplayNameFor(model => model.TopicID)

</dt>

<dd>

@Html.DisplayFor(model => model.TopicID)

</dd>

</dl>

@using (Html.BeginForm()) {

@Html.AntiForgeryToken()

<div class="form-actions no-color">

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

@Html.ActionLink("Back to List", "Index")

</div>

}

</div>

**Edit**

@model IPG521\_SA.Models.Papers

@{

ViewBag.Title = "Edit";

}

<h2>Edit</h2>

@using (Html.BeginForm())

{

@Html.AntiForgeryToken()

<div class="form-horizontal">

<h4>Papers</h4>

<hr />

@Html.ValidationSummary(true, "", new { @class = "text-danger" })

@Html.HiddenFor(model => model.PaperId)

<div class="form-group">

@Html.LabelFor(model => model.PaperTitle, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-10">

@Html.EditorFor(model => model.PaperTitle, new { htmlAttributes = new { @class = "form-control" } })

@Html.ValidationMessageFor(model => model.PaperTitle, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.PaperAbstract, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-10">

@Html.EditorFor(model => model.PaperAbstract, new { htmlAttributes = new { @class = "form-control" } })

@Html.ValidationMessageFor(model => model.PaperAbstract, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.TopicID, htmlAttributes: new { @class = "control-label col-md-2" })

<div class="col-md-10">

@Html.DropDownListFor(model => model.TopicID,

new SelectList

(

new List<SelectListItem> {

new SelectListItem { Selected = true, Text = "-- Select Topic --", Value = "-1" },

new SelectListItem { Selected = false, Text = "Data Science", Value = "1" },

new SelectListItem { Selected = false, Text = "Education", Value = "2"},

new SelectListItem { Selected = false, Text = "Human-Computer Interaction", Value = "3" },

new SelectListItem { Selected = false, Text = "Internet of Things", Value = "4"}

}, "Value", "Text")

)

@Html.ValidationMessageFor(model => model.TopicID, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

<div class="col-md-offset-2 col-md-10">

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

@Html.ActionLink("Back to List", "Index")

</div>

</div>

</div>

}

@section Scripts {

@Scripts.Render("~/bundles/jqueryval")

}

Index

@model IEnumerable<IPG521\_SA.Models.Papers>

@{

ViewBag.Title = "Index";

}

<h2>My Papers</h2>

<p>

@Html.ActionLink("Submit New Paper", "Create")

</p>

<table class="table">

<tr>

<th>

ID

</th>

<th>

Title

</th>

<th>

Submission date

</th>

<th>

Topic

</th>

<th></th>

</tr>

@foreach (var item in Model) {

<tr>

<td>

@Html.DisplayFor(model => item.PaperId)

</td>

<td>

@Html.DisplayFor(modelItem => item.PaperTitle)

</td>

<td>

@Html.DisplayFor(modelItem => item.PaperDateSubmitted)

</td>

<td>

@Html.DisplayFor(modelItem => item.TopicID)

</td>

<td>

@Html.ActionLink("Edit", "Edit", new { id = item.PaperId }, new { @class = "btn btn-warning" })

@Html.ActionLink("Delete", "Delete", new { id = item.PaperId }, new { @class = "btn btn-danger" })

</td>

</tr>

}

</table>

**List**

@model IEnumerable<IPG521\_SA.Models.Papers>

@{

ViewBag.Title = "List2";

}

<h2>List2</h2>

<table class="table">

<tr>

<th>

@Html.DisplayNameFor(model => model.TopicID)

</th>

<th>

@Html.DisplayNameFor(model => model.PaperTitle)

</th>

<th>

@Html.DisplayNameFor(model => model.PaperAuthor)

</th>

</tr>

@foreach (var item in Model) {

<tr>

<td>

@Html.DisplayFor(modelItem => item.PaperTitle)

</td>

<td>

@Html.DisplayFor(modelItem => item.PaperAuthor)

</td>

<td>

@Html.DisplayFor(modelItem => item.TopicID)

</td>

</tr>

}

</table>

**Layout**

<!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>

@Styles.Render("~/Content/css")

@Scripts.Render("~/bundles/modernizr")

</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>

<h3 style="color:white">SACLA</h3>

</div>

<div class="navbar-collapse collapse">

<ul class="nav navbar-nav">

<li>@Html.ActionLink("Home", "Index", "Home")</li>

<li>@Html.ActionLink("Papers", "List", "Papers")</li>

<li>@Html.ActionLink("News", "News", "Home")</li>

<li>@Html.ActionLink("Contact Us", "Contact", "Home")</li>

</ul>

@Html.Partial("\_LoginPartial")

</div>

</div>

</div>

<div class="container body-content">

@RenderBody()

<hr />

<footer>

<p>&copy; @DateTime.Now.Year - Devon Economon (7770)</p>

</footer>

</div>

@Scripts.Render("~/bundles/jquery")

@Scripts.Render("~/bundles/bootstrap")

@RenderSection("scripts", required: false)

</body>

</html>