IT2030 Assignment 15 Matt Brown

**Index.cshtml**

@model IEnumerable<Class>

@{

ViewData["Title"] = "Class Schedule";

}

<div class="text-center">

<h1 class="display-4">Class Schedule</h1>

<vc:day-filter></vc:day-filter>

</div>

<a asp-action="Add" asp-controller="Class" class="ml-3">Add Class</a>

<table class="table">

<thead>

<tr><th>Title</th><th>Number</th><th>Teacher</th><th>Day</th><th>Time</th><th></th></tr>

</thead>

<tbody>

@foreach (Class c in Model) {

<tr>

<td>@c.Title</td>

<td>@c.Number</td>

<td>@c.Teacher.FullName</td>

<td>@c.Day.Name</td>

<td>@c.MilitaryTime</td>

<td>

<my-link-button action="Edit" controller="Class" id="@c.ClassId">Edit</my-link-button>

<my-link-button action="Delete" controller="Class" id="@c.ClassId">Delete</my-link-button>

</td>

</tr>

}

</tbody>

</table>

**\_ViewImports.cshtml**

@using ClassSchedule

@using ClassSchedule.Models

@addTagHelper \*, Microsoft.AspNetCore.Mvc.TagHelpers

@addTagHelper \*, ClassSchedule

**SubmitButtonTagHelper.cs**

using Microsoft.AspNetCore.Razor.TagHelpers;

namespace ClassSchedule.TagHelpers

{

[HtmlTargetElement(Attributes = "[type=submit]")]

public class SubmitButtonTagHelper : TagHelper

{

public override void Process(TagHelperContext context,

TagHelperOutput output)

{

output.Attributes.AppendCssClass("btn btn-dark");

}

}

}

**MyLinkButtonTagHelper.cs**

using Microsoft.AspNetCore.Razor.TagHelpers;

using Microsoft.AspNetCore.Mvc.ViewFeatures;

using Microsoft.AspNetCore.Mvc.Rendering;

using Microsoft.AspNetCore.Routing;

namespace ClassSchedule.TagHelpers

{

public class MyLinkButtonTagHelper : TagHelper

{

private LinkGenerator linkBuilder;

public MyLinkButtonTagHelper(LinkGenerator lg) => linkBuilder = lg;

[ViewContext]

[HtmlAttributeNotBound]

public ViewContext ViewCtx { get; set; }

public string Action { get; set; }

public string Controller { get; set; }

public string Id { get; set; }

public bool IsActive { get; set; }

public override void Process(TagHelperContext context, TagHelperOutput output)

{

string action = Action ?? ViewCtx.RouteData.Values["action"].ToString();

string controller = Controller ?? ViewCtx.RouteData.Values["controller"].ToString();

var segment = (string.IsNullOrEmpty(Id)) ? null : new { Id };

string url = linkBuilder.GetPathByAction(action, controller, segment);

string css = (IsActive) ? "btn btn-dark" : "btn btn-outline-dark";

output.BuildLink(url, css);

}

}

}

**DayFilter.cs**

using Microsoft.AspNetCore.Mvc;

using ClassSchedule.Models;

namespace ClassSchedule.Components

{

public class DayFilter : ViewComponent

{

private IRepository<Day> data { get; set; }

public DayFilter(IRepository<Day> rep) => data = rep;

public IViewComponentResult Invoke()

{

var days = data.List(new QueryOptions<Day>

{

OrderBy = d => d.DayId

});

return View(days);

}

}

}

**Default.cshtml**

@model IEnumerable<Day>

@{

int id = Convert.ToInt32(ViewContext.RouteData.Values["id"]);

}

@foreach (Day d in Model)

{

<my-link-button action="Index" id="@d.DayId" class="mr-1 mb-2"

is-active="@(id == d.DayId)">@d.Name</my-link-button>

}