МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

«Харківський політехнічний інститут»

Кафедра стратегічного управління

ЗВІТ

з лабораторної роботи №4

з дисципліни «Стек технологій .NET»

Перевірила

Виконав:

Харків – 2019

**Тема:** Робота з даними в ASP.NET

**Ціль:** Познайомитися з рядом засобів ASP.NETдля представлення та відображення даних.

**Варіант:** №2 – Склад. Розробити простий web-застосунок та реалізувати базовий функціонал.

Опис механізму розробленого за стосунку:

Адміністратор має можливість здійснювати стандартні операції з додаванням товару на склад, добавлянням нових клієнтів, нових адміністраторів та нових товари.

Структура за стосунку наведена на рисунку 1.

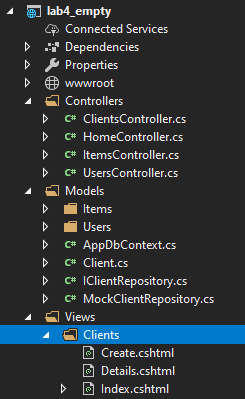


Рисунок 1 – Структура за стосунку

Код програми (для компоненту Клієнт) наведено нижче.

Контролер

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

using lab4\_empty.Models;

using Microsoft.AspNetCore.Mvc;

using System.Data;

using System.Data.SqlClient;

using System.Data.SqlTypes;

namespace lab4\_empty.Controllers

{

public class ClientsController : Controller

{

private readonly IClientRepository \_clientRepository;

public ClientsController(IClientRepository clientRepository)

{

\_clientRepository = clientRepository;

}

[Route("Clients")]

public ViewResult Index()

{

return View(\_clientRepository.GetClients());

}

[Route("Clients/Details/{id?}")]

public ViewResult Details(int id = 1)

{

Client model = \_clientRepository.GetClient(id);

return View(model);

}

[Route("Clients/Delete/{id}")]

public RedirectToActionResult Delete(int id)

{

\_clientRepository.Delete(id);

return RedirectToAction("Index");

}

[HttpGet]

public ViewResult Create()

{

return View();

}

[HttpPost]

public RedirectToActionResult Create(Client client)

{

Client newClient =\_clientRepository.Add(client);

return RedirectToAction("Details", new { id = newClient.Id });

}

}

}

Модель

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

namespace lab4\_empty.Models

{

public class Client

{

public int Id { get; set; }

public string Name { get; set; }

public string Address { get; set; }

public string PhoneNumber { get; set; }

}

}

Уявлення

@model IEnumerable<lab4\_empty.Models.Client>

<h3>@ViewBag.PageTitle</h3>

<a class="btn btn-info" **asp-controller**="Clients" **asp-action**="Create">Добавить</a>

<div class="card-deck">

@foreach (var client in Model)

{

<div class="card m-3">

<div class="card-header">

<h3>@client.Name</h3>

</div>

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

<a **asp-controller**="clients" **asp-action**="details" **asp-route-id**="@client.Id" class="btn btn-primary">View</a>

<a **asp-controller**="clients" **asp-action**="details" **asp-route-id**="@client.Id" class="btn btn-primary">Edit</a>

<a **asp-controller**="clients" **asp-action**="delete" **asp-route-id**="@client.Id" class="btn btn-danger">Delete</a>

</div>

</div>

}

</div>

@section Scripts {

<script src="~/js/script.js"></script>

}

Для лабораторної роботи застосовувалися «несправжні» дані, що були запрограмовані в код.

Фрагмент класу MockClientRepository

\_clientList = new List<Client>()

{

new Client() { Id = 1, Name = "Mary", Address = "gagarina", PhoneNumber = "123" },

new Client() { Id = 2, Name = "John", Address = "pushka", PhoneNumber = "456" },

new Client() { Id = 3, Name = "Sam", Address = "sumska", PhoneNumber = "789" },

};

}

Вид сторінки з клієнтами зображено на рисунку 2.

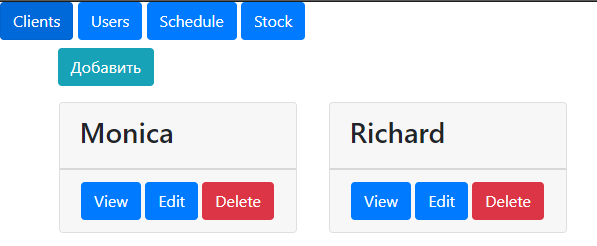


Рисунок 2 – Сторінка з клієнтами

Висновки: при виконанні даної лабораторної роботи було здійснено знайомство з фреймворком ASP.NET Core, технологією Razor Pages, тегами Razor Pages.