

Лабораторна робота №3

Тема: Часткові представлення. Робота з View Components. Навігація. Фільтрація.
Використання сесії.

Мета: отримати навички роботи з частковими представленнями, навчитися працювати з параметрами рядку запиту, набути навичок роботи з сесіями.

Хід роботи:

Завдання 1. Створення часткового представлення (Partial View)

1. У папці Views/Shared створіть .cshtml файл для одиничного елемента зі списку сутностей (для SportsStore це буде ProductSummary.cshtml)

Лістинг:

```
@model JobPortal.Models.Vacancy

<div class="col-md-4 mb-4">
  <div class="card h-100 shadow-sm">
    <div class="card-body">
      <h5 class="card-title">@Model.Title</h5>
      <h6 class="card-subtitle text-muted">@Model.Company</h6>
      <p class="card-text mt-2">@Model.Description</p>
      <p><strong>Місцезнаходження:</strong> @Model.Location</p>
      <p>
        <strong>Зарплата:</strong>
        @Model.Salary.ToString("C", new System.Globalization.CultureInfo("uk-UA"))
      </p>
    </div>
    <a asp-controller="Apply"
      asp-action="Apply"
      asp-route-vacancyId="@Model.VacancyID"
      class="btn btn-outline-primary mt-3">
      Подати заявку
    </a>
  </div>
</div>
```

2. Перемістіть код для одиничного елемента з загального View Index.cshtml в створений файл.
3. Використайте створений Partial View в Index.cshtml за допомогою Tag Helper'a <partial>

Лістинг:

```
@model JobPortal.Models.ViewModels.VacanciesListViewModel

@{
  ViewData["Title"] = "Вакансії";
}

<div class="row">
  @foreach (var v in Model.Vacancies)
  {
    <partial name="_VacancySummary" model="v" />
  }
</div>
```

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.25.121.13.000 – Лр.3							
Змн.	Арк.	№ докум.	Підпис	Дата								
Розроб.		Кравченко К.М.			Звіт з лабораторної роботи №3				Літ.	Арк.	Аркушів	
Перевір.		Українець М.О.									1	
Реценз.									ФІКТ, гр. ІПЗ-22-2			
Н. Контр.												
Зав.каф.												

```
}  
</div>
```

Завдання 2. Навігація по категоріям

Реалізуйте навігацію за категорією (або за іншою ознакою, яка більше підходить за вашим варіантом), дотримуючись наступних кроків:

1. В папці проекту створіть папку Components, в ній створіть файл з назвою `NavigationMenuViewComponent.cs`
2. В загальному шаблоні (файл `_Layout.cshtml`) додайте розмітку для сайд-бару. В доданій розмітці додайте виклик створеного `NavigationMenuViewComponent`.
3. Змініть реалізацію `NavigationMenuViewComponent` таким чином, щоб компонент отримував перелік категорій з БД і повертав результат типу `IViewComponentResult`.

Лістинг:

```
<div class="container mb-5">  
  <div class="row">  
  
    <!-- Sidebar -->  
    <div class="col-3">  
      @await Component.InvokeAsync("NavigationMenu",  
        new { selectedLocation = ViewBag.SelectedLocation })  
    </div>  
  
    <!-- Main content -->  
    <div class="col-9">  
      @RenderBody()  
    </div>  
  
  </div>  
</div>
```

4. В папці `Views/Shared/Components/NavigationMenu` створіть файл `Default.cshtml`.

Лістинг:

```
5. using JobPortal.Models;  
using Microsoft.AspNetCore.Mvc;  
using Microsoft.EntityFrameworkCore;  
  
namespace JobPortal.Components  
{  
    public class NavigationMenuViewComponent : ViewComponent  
    {  
        private readonly PortalDbContext _context;  
  
        public NavigationMenuViewComponent(PortalDbContext context)  
        {  
            _context = context;  
        }  
  
        public IViewComponentResult Invoke(string selectedLocation)  
        {  
            var locations = _context.Vacancies  
                .Select(v => v.Location)  
                .Distinct()  
                .OrderBy(l => l)  
                .ToList();  
  
            ViewBag.SelectedLocation = selectedLocation;  
        }  
    }  
}
```

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.25.121.13.000 – Лр.3	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        return View(locations);
    }
}

```

4. В створеному файлі представлення реалізуйте виведення списку посилань, отриманих з ViewComponent'a.
5. У файлі Controllers/HomeController.cs в методі Index() реалізуйте підтримку фільтрації по категоріям.

Лістинг:

```

@model IEnumerable<string>

<div class="list-group">

    <a class="list-group-item list-group-item-action"
        @(ViewBag.SelectedLocation == null ? "active" : "")"
asp-controller="Home"
asp-action="Index"
asp-route-location="">
        Усі локації
    </a>

    @foreach (var loc in Model)
    {
        <a class="list-group-item list-group-item-action"
            @(ViewBag.SelectedLocation == loc ? "active" : "")"
            asp-controller="Home"
            asp-action="Index"
            asp-route-location="@loc">
            @loc
        <a>
    }
</div>

```

6. Змініть ViewModel таким чином, щоб мати інформацію про обрану категорію на представленні.

Лістинг:

```

using Microsoft.AspNetCore.Mvc;
using JobPortal.Models;
using JobPortal.Models.ViewModels;
using System.Linq;

namespace JobPortal.Controllers
{
    public class HomeController : Controller
    {
        private readonly IPortalRepository repository;
        private const int PageSize = 4;

        public HomeController(IPortalRepository repo)
        {
            repository = repo;
        }

        public IActionResult Index(string? location, int page = 1)
        {
            // Фільтрація за локацією
            var filteredVacancies = repository.Vacancies;

            if (!string.IsNullOrEmpty(location))
            {
                filteredVacancies = filteredVacancies
                    .Where(v => v.Location == location);
            }
        }
    }
}

```

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.25.121.13.000 – Лр.3	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		

```

    }

    int totalItems = filteredVacancies.Count();

    var vacancies = filteredVacancies
        .OrderBy(v => v.VacancyID)
        .Skip((page - 1) * PageSize)
        .Take(PageSize)
        .ToList();

    var viewModel = new VacanciesListViewModel
    {
        Vacancies = vacancies,
        PagingInfo = new PagingInfo
        {
            CurrentPage = page,
            ItemsPerPage = PageSize,
            TotalItems = totalItems
        },
        CurrentLocation = location
    };

    ViewBag.SelectedLocation = location;

    return View(viewModel);
}
}
}

```

7. Реалізуйте виділення категорії, обраної користувачем. Для цього передавайте інформацію про обрану категорію з View Component в View.

```

namespace JobPortal.Models.ViewModels
{
    public class VacanciesListViewModel
    {
        public IEnumerable<Vacancy> Vacancies { get; set; } =
            Enumerable.Empty<Vacancy>();
        public PagingInfo PagingInfo { get; set; } = new PagingInfo();

        public string? CurrentLocation { get; set; }
    }
}

```

8. Змініть реалізацію пейджингу таким чином, щоб він відображав кількість доступних сторінок відповідно до обраної категорії.

```

<a class="list-group-item list-group-item-action
    @(ViewBag.SelectedLocation == loc ? "active" : "")"

```

Результат:

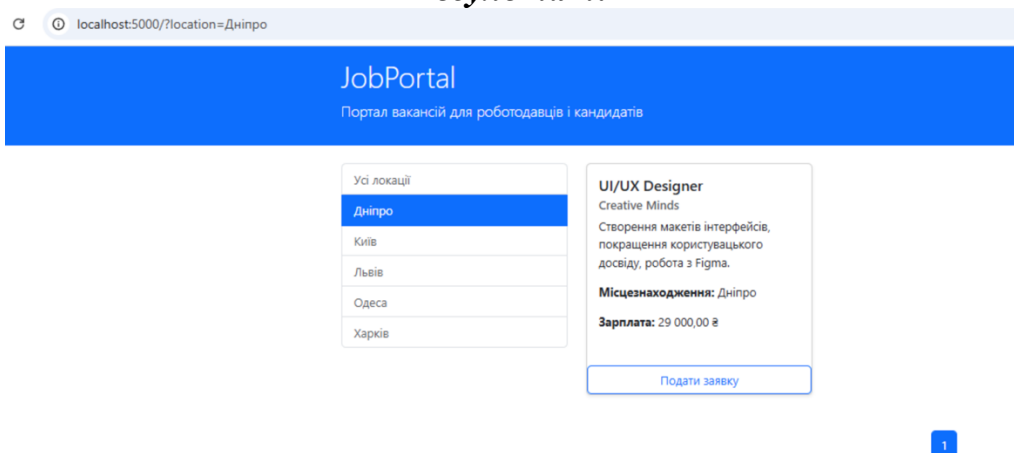


Рис.2.1. – Меню навігації

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.25.121.13.000 – Лр.3	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		

Завдання 3. Сесії

Увімкнемо сесії в Program.cs:

```
using Microsoft.EntityFrameworkCore;
using JobPortal.Models;

var builder = WebApplication.CreateBuilder(args);

// Add services to the container.
builder.Services.AddControllersWithViews();
builder.Services.AddSession();

builder.Services.AddDbContext<PortalDbContext>(opts => {

opts.UseSqlServer(builder.Configuration["ConnectionStrings:JobPortalConnection"]);
});

builder.Services.AddScoped<IPortalRepository, EFPortalRepository>();

var app = builder.Build();
...
app.UseSession();

app.UseAuthorization();

app.MapControllerRoute(
    name: "default",
    pattern: "{controller=Home}/{action=Index}/{id?}");

SeedData.EnsurePopulated(app);

app.Run();
```

Модель заявки (Application):

```
namespace JobPortal.Models
{
    public class Application
    {
        public long VacancyID { get; set; }
        public string ApplicantName { get; set; } = string.Empty;
        public string Email { get; set; } = string.Empty;
        public string Message { get; set; } = string.Empty;
    }
}
```

ViewModel для форми заявки:

```
using JobPortal.Models;

namespace JobPortal.Models.ViewModels
{
    public class ApplicationViewModel
    {
        public Vacancy Vacancy { get; set; }
        public Application Application { get; set; }
    }
}
```

Розширення для збереження об'єкта у сесію (JSON):

```
using Microsoft.AspNetCore.Http;
using Newtonsoft.Json;

namespace JobPortal.Infrastructure
{
    public static class SessionExtensions
    {
        public static void SetJson(this ISession session, string key, object
```

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.25.121.13.000 – Лр.3	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		

```

value)
    {
        session.SetString(key, JsonConvert.SerializeObject(value));
    }

    public static T? GetJson<T>(this ISession session, string key)
    {
        var data = session.GetString(key);
        return data == null ? default :
        JsonConvert.DeserializeObject<T>(data);
    }
}

```

Контролер для заявки ApplyController.cs:

```

using Microsoft.AspNetCore.Mvc;
using JobPortal.Models;
using JobPortal.Models.ViewModels;
using JobPortal.Infrastructure;

namespace JobPortal.Controllers
{
    public class ApplyController : Controller
    {
        private readonly IPortalRepository repository;
        private const string SessionKey = "ApplicationDraft";

        public ApplyController(IPortalRepository repo)
        {
            repository = repo;
        }

        // GET: форма заявки
        public IActionResult Apply(long vacancyId)
        {
            var vacancy = repository.Vacancies
                .FirstOrDefault(v => v.VacancyID == vacancyId);

            if (vacancy == null)
                return NotFound();

            // Якщо є незавершена форма – підставляємо її
            var draft = HttpContext.Session.GetJson<Application>(SessionKey);

            var vm = new ApplicationViewModel
            {
                Vacancy = vacancy,
                Application = draft ?? new Application { VacancyID = vacancyId }
            };

            return View(vm);
        }

        // POST: отримуємо дані
        [HttpPost]
        public IActionResult Apply(ApplicationViewModel vm)
        {
            if (!ModelState.IsValid)
            {
                var errors = ModelState.Values.SelectMany(v => v.Errors);
                foreach (var e in errors)
                    Console.WriteLine(e.ErrorMessage);

                vm.Vacancy = repository.Vacancies
                    .FirstOrDefault(v => v.VacancyID == vm.Application.VacancyID);
            }
        }
    }
}

```

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.25.121.13.000 – Лр.3	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        return View(vm);
    }

    // Якщо заявка завершена – очистити чернетку
    HttpContext.Session.Remove("ApplicationDraft");

    return RedirectToAction("Completed");
}

public IActionResult Completed()
{
    return View();
}
}

```

View для заявки (Apply.cshtml):

```

@model JobPortal.Models.ViewModels.ApplicationViewModel

<h3>Заявка на вакансію: @Model.Vacancy.Title</h3>

<form method="post">

    <input type="hidden" asp-for="Application.VacancyID" />

    <div class="mb-3">
        <label class="form-label">Ваше ім'я</label>
        <input asp-for="Application.ApplicantName" class="form-control" />
    </div>

    <div class="mb-3">
        <label class="form-label">Email</label>
        <input asp-for="Application.Email" class="form-control" />
    </div>

    <div class="mb-3">
        <label class="form-label">Повідомлення</label>
        <textarea asp-for="Application.Message" class="form-control"></textarea>
    </div>

    <button type="submit" class="btn btn-primary">
        Надіслати заявку
    </button>
</form>

```

Кнопка “Подати заявку” на сторінці вакансій у VacancySummary.cshtml:

```

<a asp-controller="Apply"
    asp-action="Apply"
    asp-route-vacancyId="@Model.VacancyID"
    class="btn btn-outline-primary mt-3">
    Подати заявку
</a>

```

Результат:

Junior .NET Developer
 SoftLine Group
 Розробка веб-додатків на
 платформі .NET, робота з базами
 даних, підтримка існуючих систем.

Місцезнаходження: Київ

Зарплата: 25 000,00 ₴

Подати заявку

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.25.121.13.000 – Лр.3	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		

Усі локації

Дніпро

Київ

Львів

Одеса

Харків

Заявка на вакансію: Junior .NET Developer

Ваше ім'я

Катерина

Email

uuu111@mail.com

Повідомлення

Hello world

Надіслати заявку

© 2025 — JobPortal. Усі права захищено.

Always match Chrome's language

Switch DevTools to Ukrainian

Memory

Application

Privacy and security

Lighthouse

Recorder

✕

☐ Only show cookies with an issue

	Value	Domain	Path	Expires / Max...	Size	HttpOnly	Secure
o	CfDJ8CcCHUDeG7dEnDORbhplysVtoXrJkUXkCjy-rHO0BjenN7_oKo1wdlcxzoDniWK...	localhost	/	Session	190	✓	

Рис.3.1. – 3.2. Реалізація сесій

Висновок: У ході виконання лабораторної роботи №3 було опрацьовано основні можливості ASP.NET Core, пов'язані з побудовою гнучкого та зручного інтерфейсу додатка. Було створено та використано часткові представлення, що дозволило оптимізувати структуру сторінок і забезпечити повторне використання елементів інтерфейсу.