

Лабораторна робота № 7

Створення веб-застосунків з підтримкою взаємодії в реальному часі з використанням SignalR.

Мета: отримати практичний досвід інтеграції SignalR у веб-застосунок для забезпечення обміну даними в реальному часі за допомогою SignalR.

Хід роботи

Завдання 1.

Додайте SignalR до вашого MVC проекту розробленого під час попередніх лабораторних робіт. Реалізуйте функціональність, яка працюватиме в реальному часі, відповідно до тематики вашого проекту (табл. 1).

Таблиця 1. Функціональності для реалізації

Номер за списком	Тема проекту	Функціональність
3	Портал вакансій	Миттєві сповіщення кандидатів про публікацію нової вакансії в бажаній категорії.

Лістинг зміненого Program.cs:

```
using JobPortal.Data;
using JobPortal.Models;
using Microsoft.AspNetCore.Identity;
using Microsoft.EntityFrameworkCore;
using Microsoft.AspNetCore.Authentication.JwtBearer;
using Microsoft.IdentityModel.Tokens;
using System.Text;
using System.Reflection.Metadata.Ecma335;
using JobPortal.Hubs;

namespace JobPortal
{
    public class Program
    {
        public static async Task Main(string[] args)
        {
            var builder = WebApplication.CreateBuilder(args);

            ...

            builder.Services.AddSignalR();
        }
    }
}
```

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.25.121.23.000 – Лр.7							
Змн.	Арк.	№ докум.	Підпис	Дата	Звіт з лабораторної роботи №7			Літ.	Арк.	Аркушів		
Розроб.		Трофімчук М.О.								1		
Перевір.		Українець М.О.						ФІКТ, гр. ІПЗ-22-2				
Реценз.												
Н. Контр.												
Зав.каф.												

```

// Session
builder.Services.AddDistributedMemoryCache();
builder.Services.AddSession(options =>
{
    options.IdleTimeout = TimeSpan.FromMinutes(30);
    options.Cookie.HttpOnly = true;
    options.Cookie.IsEssential = true;
});

var app = builder.Build();

...

// Маршрутизація для API-контролерів
app.MapControllers();

// SeedData.EnsurePopulated(app); // Припускаємо, що SeedData існує
app.MapHub<NotificationHub>("/notificationHub");

app.Run();
}

private static async Task CreateRolesAsync(WebApplication app)
{
    using var scope = app.Services.CreateScope();
    var roleManager = scope.ServiceProvider.GetRequiredService<RoleManager<IdentityRole>>();

    string[] roles = { "Candidate", "Employer" };

    foreach (var role in roles)
    {
        if (!await roleManager.RoleExistsAsync(role))
        {
            await roleManager.CreateAsync(new IdentityRole(role));
        }
    }
}
}

```

Лістинг зміненого JobsController.cs (Create method):

```

using JobPortal.Models;
using Microsoft.AspNetCore.Authorization;
using Microsoft.AspNetCore.Mvc;
using Microsoft.AspNetCore.SignalR;
using JobPortal.Hubs;

namespace JobPortal.Controllers
{
    public class JobsController : Controller
    {
        private readonly IPortalRepository _repository;
        private readonly IHubContext<NotificationHub> _hubContext;

        public JobsController(IPortalRepository repo, IHubContext<NotificationHub>
hubContext)
        {
            _repository = repo;
            _hubContext = hubContext;
        }

        [Authorize(Roles = "Employer")]
        [HttpPost]
        [ValidateAntiForgeryToken]
        public IActionResult Create(Job job)
        {

```

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.25.121.23.000 – Лр.7	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		2

```

        if (ModelState.IsValid)
        {
            _repository.CreateJob(job);

            // ВІДПРАВКА СПОВІЩЕННЯ
            _hubContext.Clients.All.SendAsync("ReceiveJobNotification", job.Title);

            return RedirectToAction("Index");
        }
        return View("Edit", job);
    }
}
}

```

Лістинг зміненого Layout для відображення сповіщень:

```

@await RenderSectionAsync("Scripts", required: false)
<script src="~/lib/jquery/dist/jquery.min.js"></script>
<script src="~/lib/bootstrap/dist/js/bootstrap.bundle.min.js"></script>
<script src="~/js/site.js" asp-append-version="true"></script>

<div id="notification-area" style="position: fixed; bottom: 20px; right: 20px; z-
index: 9999;"></div>

<script src="https://cdnjs.cloudflare.com/ajax/libs/microsoft-
signalr/7.0.12/signalr.min.js"></script>

<script>
    const connection = new signalR.HubConnectionBuilder()
        .withUrl("/notificationHub")
        .build();

    // команда від сервера
    connection.on("ReceiveJobNotification", function (jobTitle) {
        const alertHtml = `
            <div class="alert alert-success alert-dismissible fade show shadow"
role="alert">
                <strong>Нова вакансія!</strong> <br/>
                ${jobTitle}
                <button type="button" class="btn-close" data-bs-dismiss="alert"
aria-label="Close"></button>
            </div>
        `;

        const notificationArea = document.getElementById("notification-area");
        notificationArea.innerHTML += alertHtml;

        setTimeout(() => {
            const alerts = notificationArea.getElementsByClassName('alert');
            if (alerts.length > 0) {
                alerts[0].remove();
            }
        }, 5000);
    });

    connection.start().catch(function (err) {
        return console.error(err.toString());
    });
</script>
</body>
</html>

```

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.25.121.23.000 – Лр.7	Арк.
						3
Змн.	Арк.	№ докум.	Підпис	Дата		

Результат:

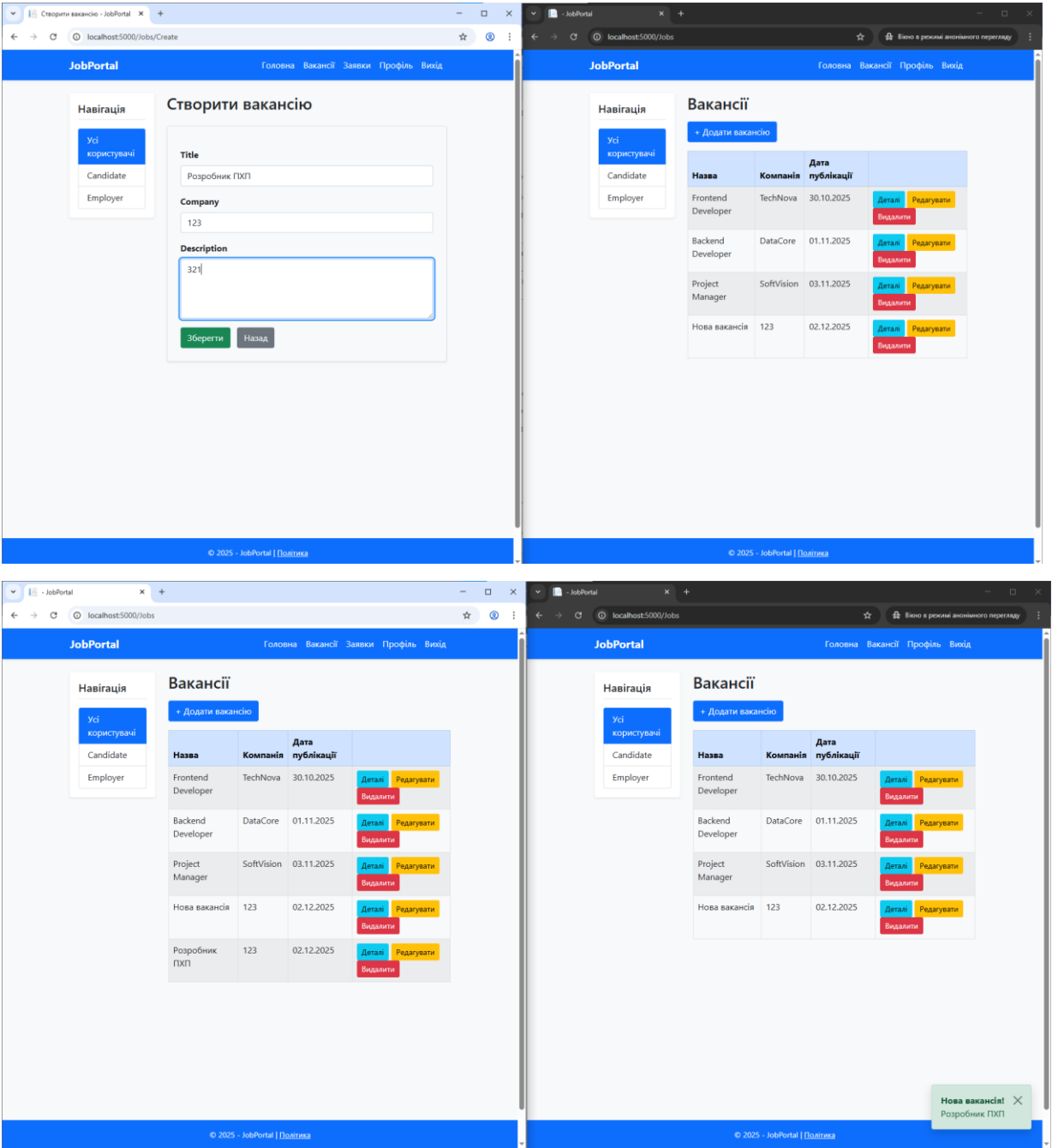


Рис. 1.1-2. Створення нової вакансії та відображення сповіщення у користувача

Висновок: в рамках цієї лабораторної роботи було отримано практичний досвід інтеграції SignalR у веб-застосунок для забезпечення обміну даними в реальному часі за допомогою SignalR.