

ЛАБОРАТОРНА РОБОТА № 6

Розробка клієнтських додатків з використанням Blazor WebAssembly

Мета: отримати навички створення та інтеграції клієнтського додатку з RESTful Web API за допомогою Blazor WebAssembly.

Завдання 1.

Створіть у своєму рішенні проект Blazor WebAssembly, який взаємодіятиме з WebAPI. Розробіть сторінки та необхідні компоненти для виконання CRUD-операцій з існуючими сутностями. Додайте валідацію для створених форм.

Оновлений CinemaBooking.Web/Program.cs:

```
using Microsoft.AspNetCore.Components.Web;
using Microsoft.AspNetCore.Components.WebAssembly.Hosting;

namespace CinemaBooking.Web
{
    public class Program
    {
        public static async Task Main(string[] args)
        {
            var builder = WebAssemblyHostBuilder.CreateDefault(args);
            builder.RootComponents.Add<App>("#app");
            builder.RootComponents.Add<HeadOutlet>("head::after");
            builder.Services.AddHttpClient("MyApi", client =>
            {
                client.BaseAddress = new Uri("https://localhost:7102/");
            });

            await builder.Build().RunAsync();
        }
    }
}
```

Додано Movies.razor:

```
@page "/movies"
@using CinemaBooking.Shared.DTOs
@inject IHttpClientFactory HttpClientFactory
@inject NavigationManager Navigation

<div class="d-flex justify-content-between align-items-center mb-3">
    <h3>Фільми</h3>
    <a href="/movies/add" class="btn btn-primary">Додати Фільм</a>
</div>
```

					ДУ «Житомирська політехніка».24.121.01.000 – Лр1			
Змн.	Арк.	№ докум.	Підпис	Дата	Звіт з лабораторної роботи			
Розро б.		Круковський Д.С.						
Перевір.		Українець М.О.						
Керівник								
Н. контр.								
Зав. каф.					ФІКТ Гр. ІІЗ-23-3			
					Літ.		Арк.	Аркушів
							1	12

```

@if (!string.IsNullOrEmpty(errorMessage))
{
    <div class="alert alert-danger">@errorMessage</div>
}

@if (movies == null)
{
    <p><em>Завантаження...</em></p>
}
else
{
    <table class="table">
        <thead>
            <tr>
                <th>Назва</th>
                <th>Жанр</th>
                <th>Опис</th>
                <th>Дії</th>
            </tr>
        </thead>
        <tbody>
            @foreach (var movie in movies)
            {
                <tr>
                    <td>@movie.Title</td>
                    <td>@movie.Genre</td>
                    <td>@movie.Description</td>
                    <td>
                        <a href="@($"movies/edit/{movie.MovieID}")" class="btn btn-sm
btn-secondary">
                            Редагувати
                        </a>
                        <button class="btn btn-sm btn-danger" @onclick="() =>
DeleteMovie(movie.MovieID)">
                            Видалити
                        </button>
                    </td>
                </tr>
            }
        </tbody>
    </table>
}

@code {
    private MovieDto[]? movies;
    private HttpClient? Http;
    private string? errorMessage;

    protected override async Task OnInitializedAsync()
    {
        Http = HttpClientFactory.CreateClient("MyApi");
        await LoadMovies();
    }

    private async Task LoadMovies()
    {
        if (Http == null) return;

        try
        {
            movies = await Http.GetFromJsonAsync<MovieDto[]>("api/movies");
            StateHasChanged();
        }
    }
}

```

```

        catch (Exception ex)
        {
            errorMessage = $"Помилка завантаження фільмів: {ex.Message}";
        }
    }

    private async Task DeleteMovie(long movieId)
    {
        if (Http == null) return;
        errorMessage = null;

        try
        {
            var response = await Http.DeleteAsync($"api/movies/{movieId}");

            if (response.IsSuccessStatusCode)
            {
                await LoadMovies();
            }
            else if (response.StatusCode == System.Net.HttpStatusCode.Unauthorized)
            {
                errorMessage = "Помилка 401: Ви не авторизовані для цієї дії.";
            }
            else
            {
                errorMessage = $"Помилка видалення: {response.StatusCode}";
            }
        }
        catch (Exception ex)
        {
            errorMessage = $"Виникла помилка: {ex.Message}";
        }
    }
}

```

Створено новий файл AddMovie.razor:

```

@page "/movies/add"
@using CinemaBooking.Shared.DTOs
@inject IHttpClientFactory HttpClientFactory
@inject NavigationManager Navigation

<h3>Додати новий фільм</h3>

<EditForm Model="@movieModel" OnValidSubmit="@HandleValidSubmit">
    <DataAnnotationsValidator />

    <div class="mb-3">
        <label class="form-label">Назва:</label>
        <InputText class="form-control" @bind-Value="movieModel.Title" />
        <ValidationMessage For="@(() => movieModel.Title)" />
    </div>

    <div class="mb-3">
        <label class="form-label">Жанр:</label>
        <InputText class="form-control" @bind-Value="movieModel.Genre" />
        <ValidationMessage For="@(() => movieModel.Genre)" />
    </div>

    <div class="mb-3">
        <label class="form-label">Опис:</label>
        <InputTextArea class="form-control" @bind-Value="movieModel.Description" />
        <ValidationMessage For="@(() => movieModel.Description)" />
    </div>

```

		Круковський Д.С.			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				3
Змн.	Арк.	№ докум.	Підпис	Дата		

```

</div>

@if (!string.IsNullOrEmpty(errorMessage))
{
    <div class="alert alert-danger">@errorMessage</div>
}

<button type="submit" class="btn btn-primary">Зберегти</button>
<a href="/movies" class="btn btn-link">Повернутися до списку</a>
</EditForm>

@code {
    private CreateUpdateMovieDto movieModel = new();

    private HttpClient? Http;
    private string? errorMessage;

    protected override void OnInitialized()
    {
        Http = HttpClientFactory.CreateClient("MyApi");
    }

    private async Task HandleValidSubmit()
    {
        if (Http == null) return;
        errorMessage = null;

        try
        {
            var response = await Http.PostAsJsonAsync("api/movies", movieModel);

            if (response.IsSuccessStatusCode)
            {
                Navigation.NavigateTo("/movies");
            }
            else if (response.StatusCode == System.Net.HttpStatusCode.Unauthorized)
            {
                errorMessage = "Помилка 401: Ви не авторизовані для цієї дії.";
            }
            else
            {
                errorMessage = $"Помилка збереження: {response.StatusCode}";
            }
        }
        catch (Exception ex)
        {
            errorMessage = $"Виникла помилка: {ex.Message}";
        }
    }
}

```

Створено новий файл EditMovie.razor:

```

@page "/movies/edit/{MovieId:long}"
@using CinemaBooking.Shared.DTOS
@inject IHttpClientFactory HttpClientFactory
@inject NavigationManager Navigation

<h3>Редагувати фільм</h3>

@if (movieModel == null)
{
    <p>Завантаження даних фільму...</p>
}

```

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				4
Змн.	Арк.	№ докум.	Підпис	Дата		

```

else
{
    <EditForm Model="@movieModel" OnValidSubmit="@HandleValidSubmit">
        <DataAnnotationsValidator />

        <div class="mb-3">
            <label class="form-label">Назва:</label>
            <InputText class="form-control" @bind-Value="movieModel.Title" />
            <ValidationMessage For="@(() => movieModel.Title)" />
        </div>

        <div class="mb-3">
            <label class="form-label">Жанр:</label>
            <InputText class="form-control" @bind-Value="movieModel.Genre" />
            <ValidationMessage For="@(() => movieModel.Genre)" />
        </div>

        <div class="mb-3">
            <label class="form-label">Опис:</label>
            <InputTextArea class="form-control" @bind-Value="movieModel.Description" />
            <ValidationMessage For="@(() => movieModel.Description)" />
        </div>

        @if (!string.IsNullOrEmpty(errorMessage))
        {
            <div class="alert alert-danger">@errorMessage</div>
        }

        <button type="submit" class="btn btn-primary">Оновити</button>
        <a href="/movies" class="btn btn-link">Повернутися до списку</a>
    </EditForm>
}

@code {
    [Parameter]
    public long MovieId { get; set; }

    private CreateUpdateMovieDto? movieModel;

    private HttpClient? Http;
    private string? errorMessage;

    protected override async Task OnInitializedAsync()
    {
        Http = HttpClientFactory.CreateClient("MyApi");

        var movieDto = await Http.GetFromJsonAsync<MovieDto>($"api/movies/{MovieId}");

        if (movieDto != null)
        {
            movieModel = new CreateUpdateMovieDto
            {
                Title = movieDto.Title,
                Description = movieDto.Description,
                Genre = movieDto.Genre
            };
        }
    }

    private async Task HandleValidSubmit()
    {
        if (Http == null || movieModel == null) return;
        errorMessage = null;
    }
}

```

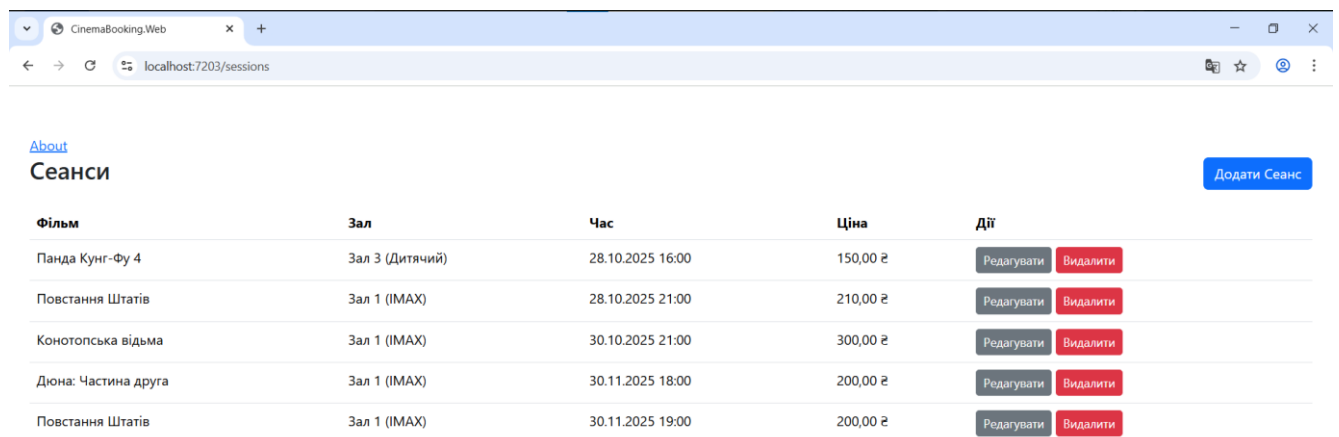
		Круковський Д.С.			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				5
Змн.	Арк.	№ докум.	Підпис	Дата		

```

try
{
    var response = await Http.PutAsJsonAsync($"api/movies/{MovieId}",
movieModel);

    if (response.IsSuccessStatusCode)
    {
        Navigation.NavigateTo("/movies");
    }
    else if (response.StatusCode == System.Net.HttpStatusCode.Unauthorized)
    {
        errorMessage = "Помилка 401: Ви не авторизовані для цієї дії.";
    }
    else
    {
        errorMessage = $"Помилка оновлення: {response.StatusCode}";
    }
}
catch (Exception ex)
{
    errorMessage = $"Виникла помилка: {ex.Message}";
}
}
}

```



about

Сеанси Додати Сеанс

Фільм	Зал	Час	Ціна	Дії
Панда Кунг-Фу 4	Зал 3 (Дитячий)	28.10.2025 16:00	150,00 ₪	Редагувати Видалити
Повстання Штатів	Зал 1 (IMAX)	28.10.2025 21:00	210,00 ₪	Редагувати Видалити
Конотопська відьма	Зал 1 (IMAX)	30.10.2025 21:00	300,00 ₪	Редагувати Видалити
Дюна: Частина друга	Зал 1 (IMAX)	30.11.2025 18:00	200,00 ₪	Редагувати Видалити
Повстання Штатів	Зал 1 (IMAX)	30.11.2025 19:00	200,00 ₪	Редагувати Видалити

Рис.1 Результат виконання

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				6
Змн.	Арк.	№ докум.	Підпис	Дата		



Рис.2 Результат виконання

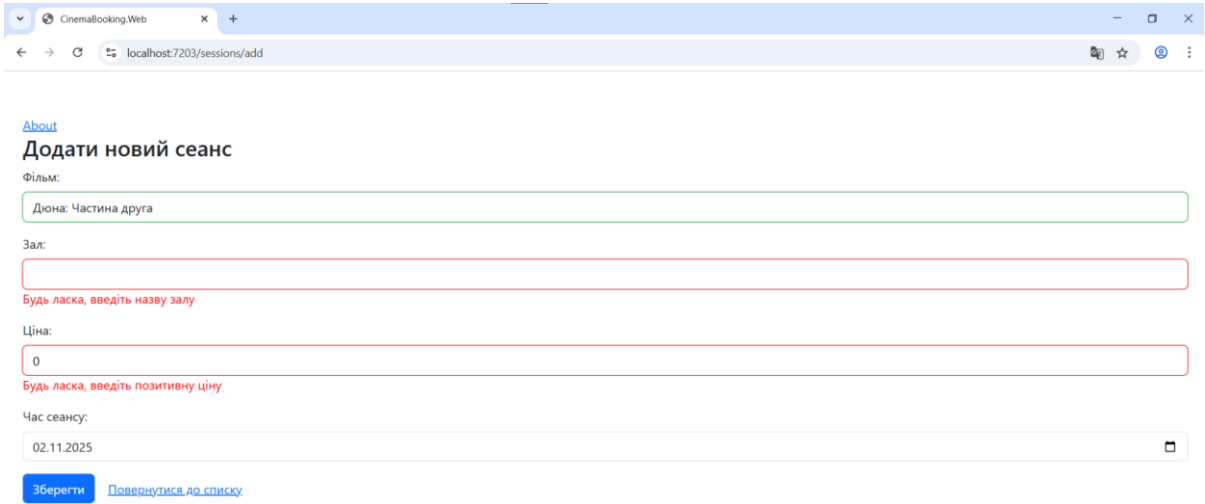


Рис.3 Результат виконання

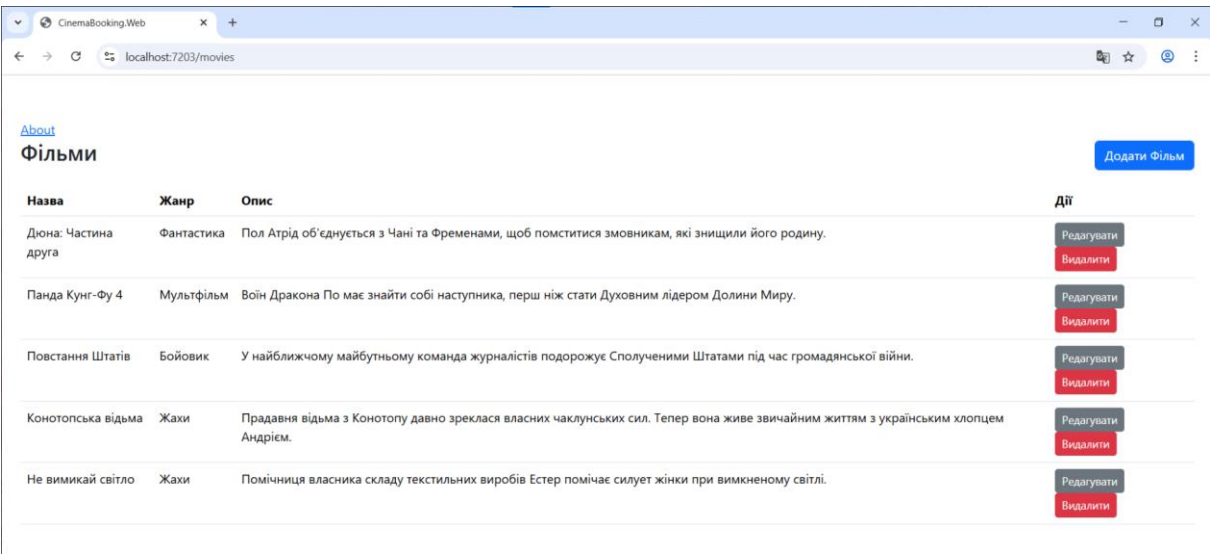


Рис.4 Результат виконання

Рис.5 Результат виконання

Рис.6 Результат виконання

Завдання 2.

Додайте автентифікацію та авторизацію до клієнта. Додайте сторінки та відповідні форми для реєстрації та входу. Реалізуйте автентифікацію на основі JWT.

Створення сторінки Реєстрації:

```
@page "/"register"
@using CinemaBooking.Shared.DTOs
@inject NavigationManager Navigation

<h3>Реєстрація</h3>

<EditForm Model="@registerModel" OnValidSubmit="@HandleRegistration">
    <DataAnnotationsValidator />
    <ValidationSummary />

    <div class="mb-3">
        <label class="form-label">Ім'я користувача:</label>
        <InputText class="form-control" @bind-Value="registerModel.Username" />
        <ValidationMessage For="@(( ) => registerModel.Username)" />
    </div>
</EditForm>
```

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				8
Змн.	Арк.	№ докум.	Підпис	Дата		


```

</div>

<div class="mb-3">
    <label class="form-label">Email:</label>
    <InputText type="email" class="form-control" @bind-Value="registerModel.Email" />
    <ValidationMessage For="@(() => registerModel.Email)" />
</div>

<div class="mb-3">
    <label class="form-label">Пароль:</label>
    <InputText type="password" class="form-control" @bind-
Value="registerModel.Password" />
    <ValidationMessage For="@(() => registerModel.Password)" />
</div>

@if (!string.IsNullOrEmpty(errorMessage))
{
    <div class="alert alert-danger">@errorMessage</div>
}
@if (isSuccess)
{
    <div class="alert alert-success">Реєстрація успішна! Тепер ви можете увій-
ти.</div>
}

<button type="submit" class="btn btn-primary"
disabled="@isSuccess">Зареєструватися</button>
</EditForm>

@code {
    private RegisterDto registerModel = new();
    private HttpClient? Http;
    private string? errorMessage;
    private bool isSuccess = false;

    [Inject]
    private IHttpClientFactory HttpClientFactory { get; set; } = default!;

    protected override void OnInitialized()
    {
        Http = HttpClientFactory.CreateClient("MyApi");
    }

    private async Task HandleRegistration()
    {
        if (Http == null) return;
        errorMessage = null;
        isSuccess = false;

        try
        {
            var response = await Http.PostAsJsonAsync("api/account/register",
registerModel);

            if (response.IsSuccessStatusCode)
            {
                isSuccess = true;
            }
            else
            {
                var errorResponse = await response.Content.ReadAsStringAsync();
                errorMessage = $"Помилка реєстрації: {errorResponse}";
            }
        }
        catch (Exception ex)

```

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				9
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        {
            errorMessage = $"Виникла помилка: {ex.Message}";
        }
    }
}

```

Створення сервісу AuthService:

```

using CinemaBooking.Shared.DTOs;
using System.Net.Http.Headers;
using System.Net.Http.Json;
using System.Text.Json.Serialization;

namespace CinemaBooking.Web.Services
{
    public class AuthService
    {
        private readonly IHttpClientFactory _httpClientFactory;

        public string? AuthToken { get; private set; }
        public bool IsLoggedIn => !string.IsNullOrEmpty(AuthToken);

        public AuthService(IHttpClientFactory httpClientFactory)
        {
            _httpClientFactory = httpClientFactory;
        }

        public async Task<bool> LoginAsync(LoginDto loginModel)
        {
            var http = _httpClientFactory.CreateClient("MyApi");
            var response = await http.PostAsJsonAsync("api/account/login", loginModel);

            if (!response.IsSuccessStatusCode)
            {
                AuthToken = null;
                return false;
            }

            var loginResponse = await
response.Content.ReadFromJsonAsync<LoginResponse>();
            if (loginResponse?.Token == null)
            {
                AuthToken = null;
                return false;
            }

            AuthToken = loginResponse.Token;
            return true;
        }

        public void Logout()
        {
            AuthToken = null;
        }

        public void SetAuthHeader(HttpClient client)
        {
            if (IsLoggedIn)
            {
                client.DefaultRequestHeaders.Authorization =
                    new AuthenticationHeaderValue("Bearer", AuthToken);
            }
        }
    }
}

```

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				10
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        private class LoginResponse
        {
            [JsonPropertyName("token")]
            public string? Token { get; set; }
        }
    }
}

```

Створення сторінки Входу:

```

@page "/login"
@using CinemaBooking.Shared.DTOS
@using CinemaBooking.Web.Services
@inject AuthService AuthService
@inject LoginDto loginModel
@inject NavigationManager Navigation

<h3>Вхід</h3>

<EditForm Model="@loginModel" OnValidSubmit="@HandleLogin">
    <DataAnnotationsValidator />
    <ValidationSummary />

    <div class="mb-3">
        <label class="form-label">Ім'я користувача:</label>
        <InputText class="form-control" @bind-Value="loginModel.Username" />
        <ValidationMessage For="@((C) => loginModel.Username)" />
    </div>

    <div class="mb-3">
        <label class="form-label">Пароль:</label>
        <InputText type="password" class="form-control" @bind-Value="loginModel.Password"
    />
        <ValidationMessage For="@((C) => loginModel.Password)" />
    </div>

    @if (!string.IsNullOrEmpty(errorMessage))
    {
        <div class="alert alert-danger">@errorMessage</div>
    }

    <button type="submit" class="btn btn-primary">Увійти</button>
</EditForm>

@code {
    private string? errorMessage;

    private async Task HandleLogin()
    {
        errorMessage = null;

        bool success = await AuthService.LoginAsync(loginModel);

        if (success)
        {
            Navigation.NavigateTo("/movies");
        }
        else
        {
            errorMessage = "Помилка: Неправильний логін або пароль.";
        }
    }
}

```

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				11
Змн.	Арк.	№ докум.	Підпис	Дата		

Оновлений AddMovie.razor:

```
@page "/movies/add"
@using CinemaBooking.Shared.DTOs
@inject IHttpClientFactory HttpClientFactory
@inject NavigationManager Navigation
@using CinemaBooking.Web.Services
@inject AuthService AuthService

<h3>Додати новий фільм</h3>

<EditForm Model="@movieModel" OnValidSubmit="@HandleValidSubmit">
    <DataAnnotationsValidator />

    <div class="mb-3">
        <label class="form-label">Назва:</label>
        <InputText class="form-control" @bind-Value="movieModel.Title" />
        <ValidationMessage For="@(() => movieModel.Title)" />
    </div>

    <div class="mb-3">
        <label class="form-label">Жанр:</label>
        <InputText class="form-control" @bind-Value="movieModel.Genre" />
        <ValidationMessage For="@(() => movieModel.Genre)" />
    </div>

    <div class="mb-3">
        <label class="form-label">Опис:</label>
        <InputTextArea class="form-control" @bind-Value="movieModel.Description" />
        <ValidationMessage For="@(() => movieModel.Description)" />
    </div>

    @if (!string.IsNullOrEmpty(errorMessage))
    {
        <div class="alert alert-danger">@errorMessage</div>
    }

    <button type="submit" class="btn btn-primary">Зберегти</button>
    <a href="/movies" class="btn btn-link">Повернутися до списку</a>
</EditForm>

@code {
    private CreateUpdateMovieDto movieModel = new();
    private HttpClient? Http;
    private string? errorMessage;

    protected override void OnInitialized()
    {
        Http = HttpClientFactory.CreateClient("MyApi");
    }

    private async Task HandleValidSubmit()
    {
        if (Http == null) return;
        errorMessage = null;

        try
        {
            AuthService.SetAuthHeader(Http);

            var response = await Http.PostAsJsonAsync("api/movies", movieModel);

            if (response.IsSuccessStatusCode)
            {

```

		Круковський Д.С.			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				12
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        Navigation.NavigateTo("/movies");
    }
    else if (response.StatusCode == System.Net.HttpStatusCode.Unauthorized)
    {
        errorMessage = "Помилка 401: Ви не авторизовані. Можливо, потрібно увій-
ти.";
    }
    else
    {
        errorMessage = $"Помилка збереження: {response.StatusCode}";
    }
}
catch (Exception ex)
{
    errorMessage = $"Виникла помилка: {ex.Message}";
}
}
}

```

Оновлений EditMovie.razor:

```

@page "/movies/edit/{MovieId:long}"
@using CinemaBooking.Shared.DTOs
@inject IHttpClientFactory HttpClientFactory
@inject NavigationManager Navigation
@using CinemaBooking.Web.Services
@inject AuthService AuthService
<h3>Редагувати фільм</h3>

@if (movieModel == null)
{
    <p>Завантаження даних фільму...</p>
}
else
{
    <EditForm Model="@movieModel" OnValidSubmit="@HandleValidSubmit">
        <DataAnnotationsValidator />

        <div class="mb-3">
            <label class="form-label">Назва:</label>
            <InputText class="form-control" @bind-Value="movieModel.Title" />
            <ValidationMessage For="@(() => movieModel.Title)" />
        </div>

        <div class="mb-3">
            <label class="form-label">Жанр:</label>
            <InputText class="form-control" @bind-Value="movieModel.Genre" />
            <ValidationMessage For="@(() => movieModel.Genre)" />
        </div>

        <div class="mb-3">
            <label class="form-label">Опис:</label>
            <InputTextArea class="form-control" @bind-Value="movieModel.Description" />
            <ValidationMessage For="@(() => movieModel.Description)" />
        </div>

        @if (!string.IsNullOrEmpty(errorMessage))
        {
            <div class="alert alert-danger">@errorMessage</div>
        }

        <button type="submit" class="btn btn-primary">Оновити</button>
        <a href="/movies" class="btn btn-link">Повернутися до списку</a>
    </EditForm>
}

```

		Круковський Д.С.			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				13
Змн.	Арк.	№ докум.	Підпис	Дата		

```

}

@code {
    [Parameter]
    public long MovieId { get; set; }

    private CreateUpdateMovieDto? movieModel;
    private HttpClient? Http;
    private string? errorMessage;

    protected override async Task OnInitializedAsync()
    {
        Http = HttpClientFactory.CreateClient("MyApi");

        var movieDto = await Http.GetFromJsonAsync<MovieDto>($"api/movies/{MovieId}");

        if (movieDto != null)
        {
            movieModel = new CreateUpdateMovieDto
            {
                Title = movieDto.Title,
                Description = movieDto.Description,
                Genre = movieDto.Genre
            };
        }
    }

    private async Task HandleValidSubmit()
    {
        if (Http == null || movieModel == null) return;
        errorMessage = null;

        try
        {
            AuthService.SetAuthHeader(Http);

            var response = await Http.PutAsJsonAsync($"api/movies/{MovieId}",
movieModel);

            if (response.IsSuccessStatusCode)
            {
                Navigation.NavigateTo("/movies");
            }
            else if (response.StatusCode == System.Net.HttpStatusCode.Unauthorized)
            {
                errorMessage = "Помилка 401: Ви не авторизовані. Можливо, потрібно увій-
ти.";
            }
            else
            {
                errorMessage = $"Помилка оновлення: {response.StatusCode}";
            }
        }
        catch (Exception ex)
        {
            errorMessage = $"Виникла помилка: {ex.Message}";
        }
    }
}

```

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				14
Змн.	Арк.	№ докум.	Підпис	Дата		

Оновлений Movies.razor

```
@page "/movies"
@using CinemaBooking.Shared.DTOs
@inject IHttpClientFactory HttpClientFactory
@inject NavigationManager Navigation
@using CinemaBooking.Web.Services
@inject AuthService AuthService

<div class="d-flex justify-content-between align-items-center mb-3">
    <h3>Фільми</h3>
    <a href="/movies/add" class="btn btn-primary">Додати Фільм</a>
</div>

@if (!string.IsNullOrEmpty(errorMessage))
{
    <div class="alert alert-danger">@errorMessage</div>
}

@if (movies == null)
{
    <p><em>Завантаження...</em></p>
}
else
{
    <table class="table">
        <thead>
            <tr>
                <th>Назва</th>
                <th>Жанр</th>
                <th>Опис</th>
                <th>Дії</th>
            </tr>
        </thead>
        <tbody>
            @foreach (var movie in movies)
            {
                <tr>
                    <td>@movie.Title</td>
                    <td>@movie.Genre</td>
                    <td>@movie.Description</td>
                    <td>
                        <a href="@($"movies/edit/{movie.MovieID}")" class="btn btn-sm
btn-secondary">
                            Редагувати
                        </a>
                        <button class="btn btn-sm btn-danger" @onclick="() =>
DeleteMovie(movie.MovieID)">
                            Видалити
                        </button>
                    </td>
                </tr>
            }
        </tbody>
    </table>
}

@code {
    private MovieDto[]? movies;
    private HttpClient? Http;
    private string? errorMessage;

    protected override async Task OnInitializedAsync()
    {
```

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				15
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        Http = HttpClientFactory.CreateClient("MyApi");
        await LoadMovies();
    }

    private async Task LoadMovies()
    {
        if (Http == null) return;
        try
        {
            movies = await Http.GetFromJsonAsync<MovieDto[]>("api/movies");
        }
        catch (Exception ex)
        {
            errorMessage = $"Помилка завантаження фільмів: {ex.Message}";
        }
    }

    private async Task DeleteMovie(long movieId)
    {
        if (Http == null) return;
        errorMessage = null;

        try
        {
            AuthService.SetAuthHeader(Http);

            var response = await Http.DeleteAsync($"api/movies/{movieId}");

            if (response.IsSuccessStatusCode)
            {
                await LoadMovies();
            }
            else if (response.StatusCode == System.Net.HttpStatusCode.Unauthorized)
            {
                errorMessage = "Помилка 401: Ви не авторизовані для цієї дії.";
            }
            else
            {
                errorMessage = $"Помилка видалення: {response.StatusCode}";
            }
        }
        catch (Exception ex)
        {
            errorMessage = $"Виникла помилка: {ex.Message}";
        }
    }
}

```

Онволений Sessions.razor:

```

@page "/sessions"
@using CinemaBooking.Shared.DTOS
@using CinemaBooking.Web.Services
@inject IHttpClientFactory HttpClientFactory
@inject NavigationManager Navigation
@inject AuthService AuthService

<div class="d-flex justify-content-between align-items-center mb-3">
    <h3>Сеанси</h3>
    <a href="/sessions/add" class="btn btn-primary">Додати Сеанс</a>
</div>

@if (!string.IsNullOrEmpty(errorMessage))
{

```

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				16
Змн.	Арк.	№ докум.	Підпис	Дата		


```

    <div class="alert alert-danger">@errorMessage</div>
}

@if (sessions == null)
{
    <p><em>Завантаження сеансів...</em></p>
}
else
{
    <table class="table">
        <thead>
            <tr>
                <th>Фільм</th>
                <th>Зал</th>
                <th>Час</th>
                <th>Ціна</th>
                <th>Дії</th>
            </tr>
        </thead>
        <tbody>
            @foreach (var session in sessions)
            {
                <tr>
                    <td>@session.MovieTitle</td>
                    <td>@session.Hall</td>
                    <td>@session.SessionTime.ToString("g")</td>
                    <td>@session.Price.ToString("C")</td>
                    <td>
                        <a href="@($"sessions/edit/{session.MovieSessionID}")" class="btn
btn-sm btn-secondary">
                            Редагувати
                        </a>
                        <button class="btn btn-sm btn-danger" @onclick="() =>
DeleteSession(session.MovieSessionID)">
                            Видалити
                        </button>
                    </td>
                </tr>
            }
        </tbody>
    </table>
}

@code {
    private MovieSessionDto[]? sessions;
    private HttpClient? Http;
    private string? errorMessage;

    protected override async Task OnInitializedAsync()
    {
        Http = HttpClientFactory.CreateClient("MyApi");
        await LoadSessions();
    }

    private async Task LoadSessions()
    {
        if (Http == null) return;
        sessions = await Http.GetFromJsonAsync<MovieSessionDto[]>("api/sessions");
    }

    private async Task DeleteSession(long sessionId)
    {
        if (Http == null) return;
        errorMessage = null;
    }
}

```

```

try
{
    AuthService.SetAuthHeader(Http);

    var response = await Http.DeleteAsync($"api/sessions/{sessionId}");

    if (response.IsSuccessStatusCode)
    {
        await LoadSessions();
    }
    else
    {
        errorMessage = "Помилка 401: Ви не авторизовані.";
    }
}
catch (Exception ex)
{
    errorMessage = $"Виникла помилка: {ex.Message}";
}
}
}

```

Оновлений AddSession.razor:

```

@page "/sessions/add"
@using CinemaBooking.Shared.DTOs
@using CinemaBooking.Web.Services
@inject IHttpClientFactory HttpClientFactory
@inject NavigationManager Navigation
@inject AuthService AuthService

<h3>Додати новий сеанс</h3>

@if (movies == null)
{
    <p><em>Завантаження фільмів...</em></p>
}
else
{
    <EditForm Model="@sessionModel" OnValidSubmit="@HandleValidSubmit">
        <DataAnnotationsValidator />

        <div class="mb-3">
            <label class="form-label">Фільм:</label>
            <InputSelect class="form-control" @bind-Value="sessionModel.MovieID">
                <option value="0">-- Виберіть фільм --</option>
                @foreach (var movie in movies)
                {
                    <option value="@movie.MovieID">@movie.Title</option>
                }
            </InputSelect>
            <ValidationMessage For="@(( ) => sessionModel.MovieID)" />
        </div>

        <div class="mb-3">
            <label class="form-label">Зал:</label>
            <InputText class="form-control" @bind-Value="sessionModel.Hall" />
            <ValidationMessage For="@(( ) => sessionModel.Hall)" />
        </div>

        <div class="mb-3">
            <label class="form-label">Ціна:</label>
            <InputNumber class="form-control" @bind-Value="sessionModel.Price" />
            <ValidationMessage For="@(( ) => sessionModel.Price)" />
        </div>
    </EditForm>
}

```

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				18
Змн.	Арк.	№ докум.	Підпис	Дата		

```

</div>

<div class="mb-3">
    <label class="form-label">Час сеансу:</label>
    <InputDate class="form-control" @bind-Value="sessionModel.SessionTime" />
    <ValidationMessage For="@()" => sessionModel.SessionTime" />
</div>

@if (!string.IsNullOrEmpty(errorMessage))
{
    <div class="alert alert-danger">@errorMessage</div>
}

<button type="submit" class="btn btn-primary">Зберегти</button>
<a href="/sessions" class="btn btn-link">Повернутися до списку</a>
</EditForm>
}

@code {
    private CreateUpdateSessionDto sessionModel = new() { SessionTime = DateTime.Now };
    private MovieDto[]? movies;
    private HttpClient? Http;
    private string? errorMessage;

    protected override async Task OnInitializedAsync()
    {
        Http = HttpClientFactory.CreateClient("MyApi");
        movies = await Http.GetFromJsonAsync<MovieDto[]>("api/movies");
    }

    private async Task HandleValidSubmit()
    {
        if (Http == null) return;
        errorMessage = null;

        try
        {
            AuthService.SetAuthHeader(Http);

            var response = await Http.PostAsJsonAsync("api/sessions", sessionModel);

            if (response.IsSuccessStatusCode)
            {
                Navigation.NavigateTo("/sessions");
            }
            else
            {
                errorMessage = "Помилка 401: Ви не авторизовані.";
            }
        }
        catch (Exception ex)
        {
            errorMessage = $"Виникла помилка: {ex.Message}";
        }
    }
}

```

Оновлений EditSession.razor:

```

@page "/sessions/edit/{SessionId:long}"
@using CinemaBooking.Shared.DTOS
@using CinemaBooking.Web.Services
@Inject IHttpClientFactory HttpClientFactory
@Inject NavigationManager Navigation

```

		Круковський Д.С.			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				19
Змн.	Арк.	№ докум.	Підпис	Дата		

```

@Inject AuthService AuthService

<h3>Редагувати сеанс</h3>

@if (sessionModel == null || movies == null)
{
    <p><em>Завантаження даних...</em></p>
}
else
{
    <EditForm Model="@sessionModel" OnValidSubmit="@HandleValidSubmit">
        <DataAnnotationsValidator />

        <div class="mb-3">
            <label class="form-label">Фільм:</label>
            <InputSelect class="form-control" @bind-Value="sessionModel.MovieID">
                <option value="0">-- Виберіть фільм --</option>
                @foreach (var movie in movies)
                {
                    <option value="@movie.MovieID">@movie.Title</option>
                }
            </InputSelect>
            <ValidationMessage For="@(() => sessionModel.MovieID)" />
        </div>

        <div class="mb-3">
            <label class="form-label">Зал:</label>
            <InputText class="form-control" @bind-Value="sessionModel.Hall" />
            <ValidationMessage For="@(() => sessionModel.Hall)" />
        </div>

        <div class="mb-3">
            <label class="form-label">Ціна:</label>
            <InputNumber class="form-control" @bind-Value="sessionModel.Price" />
            <ValidationMessage For="@(() => sessionModel.Price)" />
        </div>

        <div class="mb-3">
            <label class="form-label">Час сеансу:</label>
            <InputDate class="form-control" @bind-Value="sessionModel.SessionTime" />
            <ValidationMessage For="@(() => sessionModel.SessionTime)" />
        </div>

        @if (!string.IsNullOrEmpty(errorMessage))
        {
            <div class="alert alert-danger">@errorMessage</div>
        }

        <button type="submit" class="btn btn-primary">Оновити</button>
        <a href="/sessions" class="btn btn-link">Повернутися до списку</a>
    </EditForm>
}

@code {
    [Parameter]
    public long SessionId { get; set; }
    private CreateUpdateSessionDto? sessionModel;
    private MovieDto[]? movies;
    private HttpClient? Http;
    private string? errorMessage;

    protected override async Task OnInitializedAsync()
    {
        Http = HttpClientFactory.CreateClient("MyApi");
        movies = await Http.GetFromJsonAsync<MovieDto[]>("api/movies");
    }
}

```

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				20
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        var sessionDto = await
Http.GetFromJsonAsync<MovieSessionDto>($"api/sessions/{SessionId}");

        if (sessionDto != null)
        {
            sessionModel = new CreateUpdateSessionDto
            {
                Hall = sessionDto.Hall,
                Price = sessionDto.Price,
                SessionTime = sessionDto.SessionTime,
                MovieID = sessionDto.MovieID ?? 0
            };
        }
    }

private async Task HandleValidSubmit()
{
    if (Http == null || sessionModel == null) return;
    errorMessage = null;

    try
    {
        AuthService.SetAuthHeader(Http);

        var response = await Http.PutAsJsonAsync($"api/sessions/{SessionId}",
sessionModel);

        if (response.IsSuccessStatusCode)
        {
            Navigation.NavigateTo("/sessions");
        }
        else
        {
            errorMessage = "Помилка 401: Ви не авторизовані.";
        }
    }
    catch (Exception ex)
    {
        errorMessage = $"Виникла помилка: {ex.Message}";
    }
}
}

```

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				21
Змн.	Арк.	№ докум.	Підпис	Дата		

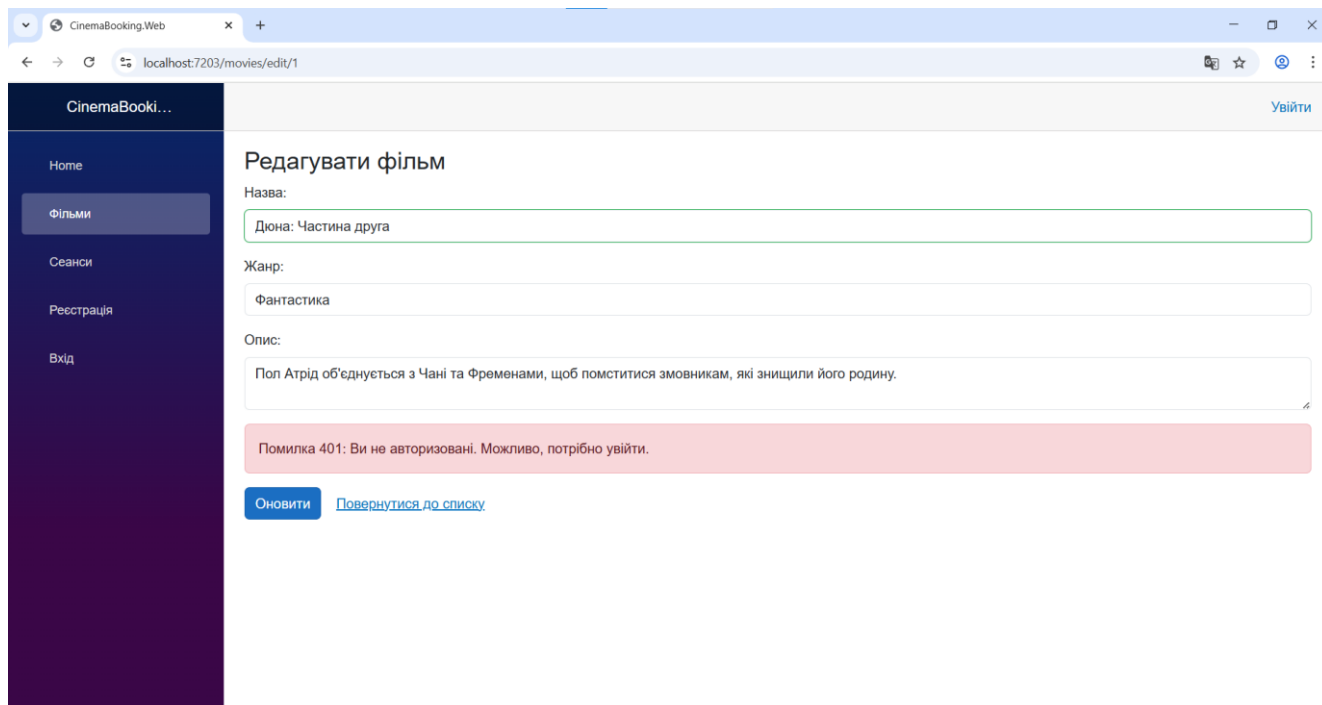


Рис.7 Результат виконання

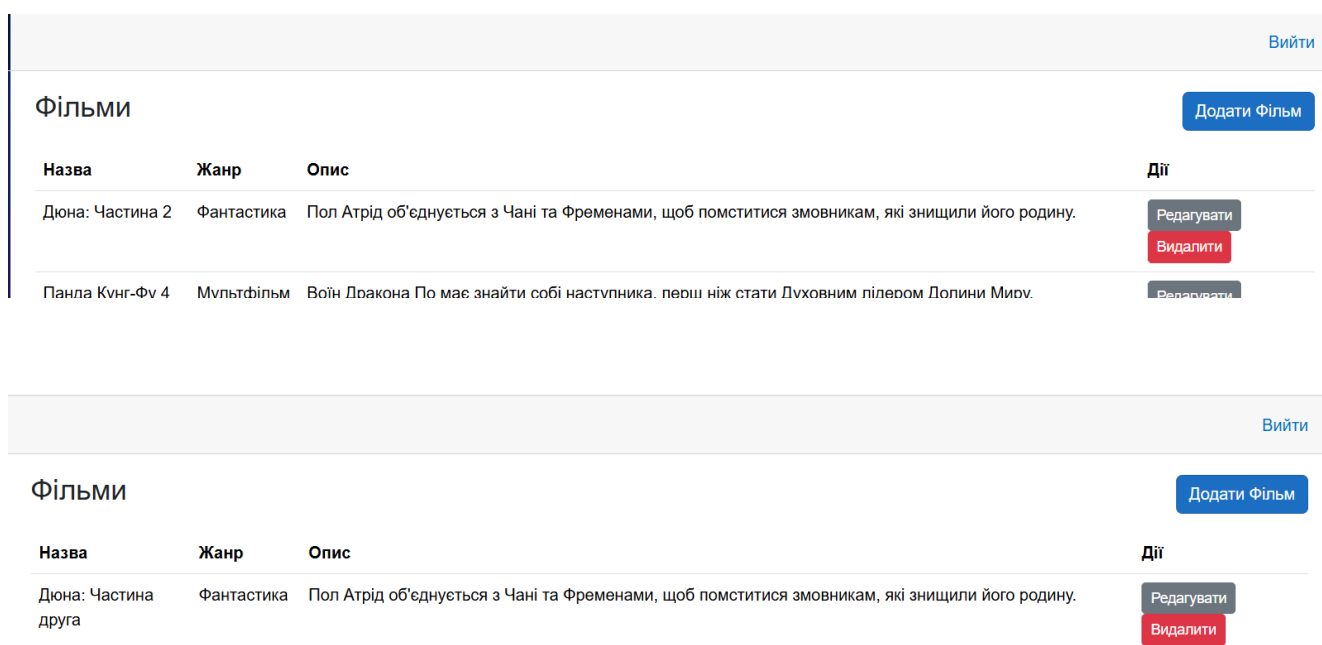


Рис.8-9 успішне оновлення назви фільму після входу адміна

Висновок: у ході лабораторної роботи ми отримали навички створення та інтеграції клієнтського додатку з RESTful Web API за допомогою Blazor WebAssembly.

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				22
Змн.	Арк.	№ докум.	Підпис	Дата		