

## ЛАБОРАТОРНА РОБОТА № 5

### Аутентифікація та авторизація. Робота з Web API. JSON Web Token.

Мета роботи: набути навичок роботи з механізмами аутентифікації та авторизації користувачів в ASP.NET, набути навичок роботи з Web API.

#### Завдання 1.

Реалізувати реєстрацію, автентифікацію та авторизацію користувача з використанням пакету Microsoft Identity.

**Завдання на додаткові бали (5 балів):** реалізувати роботу з ролями користувачів.

Передбачити в проекті можливість реєстрації двох типів користувачів, наприклад пацієнт і лікар. Якщо тематика проекту не передбачає роботу з кількома типами користувачів, то реалізувати ролі звичайного користувача та адміністратора.

**Замінений** вміст Program.cs:

```
using Microsoft.EntityFrameworkCore;
using CinemaBooking.Models;
using CinemaBooking.Infrastructure;
using Microsoft.AspNetCore.Identity;

var builder = WebApplication.CreateBuilder(args);

builder.Services.AddControllersWithViews();

builder.Services.AddDbContext<CinemaDbContext>(opts => {
    opts.UseSqlServer(
        builder.Configuration["ConnectionStrings:CinemaBookingConnection"]);
});

builder.Services.AddScoped<ICinemaRepository, EFCinemaRepository>();

builder.Services.AddDistributedMemoryCache();
builder.Services.AddSession(options => {
    options.IdleTimeout = TimeSpan.FromMinutes(30);
    options.Cookie.HttpOnly = true;
    options.Cookie.IsEssential = true;
});
builder.Services.AddSingleton<IHttpContextAccessor, HttpContextAccessor>();
builder.Services.AddScoped<Booking>(sp => SessionBooking.GetBooking(sp));

builder.Services.AddDbContext<AppIdentityDbContext>(options =>
    options.UseSqlServer(
        builder.Configuration["ConnectionStrings:IdentityConnection"]));
```

```
builder.Services.AddIdentity<IdentityUser, IdentityRole>(opts => {
    opts.User.RequireUniqueEmail = true;
```

ЛВ «Житомирська політехніка». 24.121.01.000 – Лр1						
Розроб.	Круковський Д.С.			Літ.	Арк.	Аркушів
Перевір.	Українець М.О.				1	12
Керівник				Звіт з лабораторної роботи ФІКТ Гр. ІІЗ-23-3		
Н. контр.						
Зав. каф.						

```

    opts.Password.RequiredLength = 8;
    opts.Password.RequireNonAlphanumeric = false;
    opts.Password.RequireLowercase = true;
    opts.Password.RequireUppercase = true;

    opts.Password.RequireDigit = true;
})
.AddEntityFrameworkStores<AppIdentityDbContext>();

var app = builder.Build();

app.UseStaticFiles();
app.UseSession();
app.UseAuthentication();
app.UseAuthorization();

app.MapDefaultControllerRoute();

await IdentitySeedData.EnsurePopulatedAsync(app);

SeedData.EnsurePopulated(app);

app.Run();

```

Новий файл Models/ViewModels/LoginModel.cs:

```

using System.ComponentModel.DataAnnotations;

namespace CinemaBooking.Models.ViewModels
{
    public class LoginModel
    {
        [Required(ErrorMessage = "Будь ласка, введіть ім'я користувача")]
        public string Name { get; set; } = string.Empty;

        [Required(ErrorMessage = "Будь ласка, введіть пароль")]
        [UIHint("password")]
        public string Password { get; set; } = string.Empty;

        public string ReturnUrl { get; set; } = "/";
    }
}

```

НОВИЙ файл Models/ViewModels/RegisterModel.cs:

```

using System.ComponentModel.DataAnnotations;

namespace CinemaBooking.Models.ViewModels
{
    public class RegisterModel
    {
        [Required(ErrorMessage = "Будь ласка, введіть ім'я користувача")]
        public string UserName { get; set; } = string.Empty;

        [Required(ErrorMessage = "Будь ласка, введіть Email")]
        [EmailAddress(ErrorMessage = "Будь ласка, введіть коректний Email")]
        public string Email { get; set; } = string.Empty;

        [Required(ErrorMessage = "Будь ласка, введіть пароль")]
    }
}

```

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				2
Змн.	Арк.	№ докум.	Підпис	Дата		

```

[UIHint("password")]
public string Password { get; set; } = string.Empty;

[Required(ErrorMessage = "Будь ласка, підтвердіть пароль")]
[Compare("Password", ErrorMessage = "Паролі не збігаються")]
[UIHint("password")]
[Display(Name = "Підтвердіть пароль")]
public string ConfirmPassword { get; set; } = string.Empty;

public string returnUrl { get; set; } = "/";
}
}

```

НОВИЙ файл Models/ViewModels/ProfileViewModel.cs:

```

using Microsoft.AspNetCore.Identity;

namespace CinemaBooking.Models.ViewModels
{
    public class ProfileViewModel
    {
        public IdentityUser? User { get; set; }
    }
}

```

Створений Controllers/AccountController.cs:

```

using Microsoft.AspNetCore.Authorization;
using Microsoft.AspNetCore.Identity;
using Microsoft.AspNetCore.Mvc;
using CinemaBooking.Models.ViewModels;
using System.Threading.Tasks;

namespace CinemaBooking.Controllers
{
    public class AccountController : Controller
    {
        private UserManager<IdentityUser> userManager;
        private SignInManager<IdentityUser> signInManager;
        private RoleManager<IdentityRole> roleManager;

        public AccountController(UserManager<IdentityUser> userMgr,
                                SignInManager<IdentityUser> signInMgr,
                                RoleManager<IdentityRole> roleMgr)
        {
            userManager = userMgr;
            signInManager = signInMgr;
            roleManager = roleMgr;
        }

        [AllowAnonymous]
        public IActionResult Login(string returnUrl)
        {
            return View(new LoginModel
            {
                ReturnUrl = returnUrl ?? "/"
            });
        }

        [HttpPost]
        [AllowAnonymous]
        [ValidateAntiForgeryToken]
        public async Task<IActionResult> Login(LoginModel loginModel)

```

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

```

{
    if (ModelState.IsValid)
    {
        IdentityUser? user =
            await userManager.FindByNameAsync(loginModel.Name);

        if (user != null)
        {
            var result = await signInManager.PasswordSignInAsync(user,
                loginModel.Password, isPersistent: false, lockoutOnFailure:
false);

            if (result.Succeeded)
            {
                return Redirect(loginModel?.ReturnUrl ?? "/");
            }
        }
        ModelState.AddModelError("", "Невірне ім'я користувача або пароль");
    }
    return View(loginModel);
}

[AllowAnonymous]
public ViewResult Register(string returnUrl)
{
    return View(new RegisterModel
    {
        ReturnUrl = returnUrl ?? "/"
    });
}

[HttpPost]
[AllowAnonymous]
[ValidateAntiForgeryToken]
public async Task<IActionResult> Register(RegisterModel registerModel)
{
    if (ModelState.IsValid)
    {
        var user = new IdentityUser
        {
            UserName = registerModel.UserName,
            Email = registerModel.Email
        };

        IdentityResult result = await userManager.CreateAsync(user,
registerModel.Password);

        if (result.Succeeded)
        {
            if (!await roleManager.RoleExistsAsync("User"))
            {
                await roleManager.CreateAsync(new IdentityRole("User"));
            }
            await userManager.AddToRoleAsync(user, "User");

            await signInManager.SignInAsync(user, isPersistent: false);
            return Redirect(registerModel?.ReturnUrl ?? "/");
        }
        else
        {
            foreach (IdentityError error in result.Errors)
            {
                ModelState.AddModelError("", error.Description);
            }
        }
    }
}

```

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

```

    }
    return View(registerModel);
}

[Authorize]
public async Task<RedirectResult> Logout(string returnUrl = "/")
{
    await signInManager.SignOutAsync();
    return Redirect(returnUrl);
}

[Authorize]
public async Task<IActionResult> Profile()
{
    IdentityUser? user = await userManager.GetUserAsync(User);

    if (user == null)
    {
        return Challenge();
    }

    var model = new ProfileViewModel
    {
        User = user
    };

    return View(model);
}

[AllowAnonymous]
public IActionResult AccessDenied()
{
    return View();
}
}
}

```

**Створений** Views/Account/Login.cshtml:

```

@model LoginModel

@{
    ViewBag.Title = "Вхід";
}

<h2 class="text-center">@ViewBag.Title</h2>

<div asp-validation-summary="All" class="text-danger"></div>

<form asp-action="Login" asp-controller="Account" method="post">
    <input type="hidden" asp-for="ReturnUrl" />

    <div class="mb-3">
        <label asp-for="Name" class="form-label">Ім'я користувача</label>
        <input asp-for="Name" class="form-control" />
        <span asp-validation-for="Name" class="text-danger"></span>
    </div>

    <div class="mb-3">
        <label asp-for="Password" class="form-label">Пароль</label>
        <input asp-for="Password" class="form-control" />
        <span asp-validation-for="Password" class="text-danger"></span>
    </div>

```

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

```

<div class="d-grid gap-2">
  <button class="btn btn-primary" type="submit">Увійти</button>
</div>
</form>

<div class="text-center mt-3">
  <p>Ще не маєте акаунту? <a asp-action="Register" asp-controller="Account" asp-route-
returnUrl="@Model.ReturnUrl">Зареєструватися</a></p>
</div>

```

Створений Views/Account/Register.cshtml:

```

@model RegisterModel

@{
  ViewBag.Title = "Реєстрація";
}

<h2 class="text-center">@ViewBag.Title</h2>

<div asp-validation-summary="All" class="text-danger"></div>

<form asp-action="Register" asp-controller="Account" method="post">
  <!-- Приховане поле для returnUrl -->
  <input type="hidden" asp-for="ReturnUrl" />

  <div class="mb-3">
    <label asp-for="UserName" class="form-label">Ім'я користувача</label>
    <input asp-for="UserName" class="form-control" />
    <span asp-validation-for="UserName" class="text-danger"></span>
  </div>

  <div class="mb-3">
    <label asp-for="Email" class="form-label">Email</label>
    <input asp-for="Email" class="form-control" />
    <span asp-validation-for="Email" class="text-danger"></span>
  </div>

  <div class="mb-3">
    <label asp-for="Password" class="form-label">Пароль</label>
    <input asp-for="Password" class="form-control" />
    <span asp-validation-for="Password" class="text-danger"></span>
    <small class="form-text text-muted">Мінімум 8 символів, великі та малі літери,
цифри.</small>
  </div>

  <div class="mb-3">
    <label asp-for="ConfirmPassword" class="form-label">Підтвердіть пароль</label>
    <input asp-for="ConfirmPassword" class="form-control" />
    <span asp-validation-for="ConfirmPassword" class="text-danger"></span>
  </div>

  <div class="d-grid gap-2">
    <button class="btn btn-primary" type="submit">Зареєструватися</button>
  </div>
</form>

<div class="text-center mt-3">
  <p>Вже маєте акаунт? <a asp-action="Login" asp-controller="Account" asp-route-
returnUrl="@Model.ReturnUrl">Увійти</a></p>
</div>

```

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

Створений Views/Account/Profile.cshtml:

```
@model ProfileViewModel
```

```
@{
```

```
    ViewBag.Title = "Особистий кабінет";
```

```
}
```

```
<h2 class="text-center">@ViewBag.Title</h2>
```

```
<div class="row justify-content-center">
```

```
    <div class="col-md-6">
```

```
        <div class="card shadow-sm">
```

```
            <div class="card-body">
```

```
                <h5 class="card-title">Інформація про користувача</h5>
```

```
                <hr />
```

```
                <dl class="row">
```

```
                    <dt class="col-sm-4">Ім'я користувача:</dt>
```

```
                    <dd class="col-sm-8">@Model.User?.UserName</dd>
```

```
                    <dt class="col-sm-4">Email:</dt>
```

```
                    <dd class="col-sm-8">@Model.User?.Email</dd>
```

```
                </dl>
```

```
                <div class="text-end mt-3">
```

```
                    <a asp-action="Logout" asp-controller="Account" class="btn btn-danger">Вийти</a>
```

```
                </div>
```

```
            </div>
```

```
        </div>
```

```
    </div>
```

```
</div>
```

замінений вміст Views/Shared/\_Layout.cshtml:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <meta name="viewport" content="width=device-width" />
```

```
    <title>@ViewData["Title"] - CinemaBooking</title>
```

```
    <link rel="stylesheet" href="~/lib/bootstrap/css/bootstrap.min.css" />
```

```
    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.11.3/font/bootstrap-icons.min.css">
```

```
    <link rel="stylesheet" href="~/css/style.css" />
```

```
</head>
```

```
<body class="d-flex flex-column min-vh-100">
```

```
    <header>
```

```
        <nav class="navbar navbar-expand-sm navbar-dark bg-dark border-bottom box-shadow mb-3">
```

```
            <div class="container">
```

```
                <a class="navbar-brand" asp-area="" asp-controller="Home" asp-action="Index">CinemaBooking</a>
```

```
                <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target=".navbar-collapse" aria-controls="navbarSupportedContent"
```

```
                    aria-expanded="false" aria-label="Toggle navigation">
```

```
                    <span class="navbar-toggler-icon"></span>
```

```
                </button>
```

```
                <div class="navbar-collapse collapse d-sm-inline-flex justify-content-between">
```

```
                    <ul class="navbar-nav flex-grow-1">
```

		Круковський Д.С.			ДУ «Житомирська політехніка».24.121.01.000 - Лр1	Арк.
		Українець М.О.				7
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        <li class="nav-item">
            <a class="nav-link text-light" asp-area="" asp-
controller="Home" asp-action="Index">Головна</a>
        </li>
        <li class="nav-item">
            <a class="nav-link text-light" asp-area="" asp-
controller="Admin" asp-action="Index">Адмін</a>
        </li>
    </ul>

    <div class="navbar-nav">
        <a class="btn btn-sm btn-warning me-2" asp-controller="Booking"
asp-action="Index">
            <i class="bi bi-ticket-perforated"></i>
            Обрані
        </a>

        @if (User.Identity?.IsAuthenticated ?? false)
        {
            <span class="navbar-text text-white me-2">Привіт,
@User.Identity.Name!</span>
            <a class="btn btn-sm btn-outline-light me-2" asp-
controller="Account" asp-action="Profile">Кабінет</a>
            <a class="btn btn-sm btn-outline-danger" asp-
controller="Account" asp-action="Logout">Вийти</a>
        }
        else
        {
            <a class="btn btn-sm btn-outline-light me-2" asp-
controller="Account" asp-action="Login">Вхід</a>
            <a class="btn btn-sm btn-primary" asp-controller="Account"
asp-action="Register">Реєстрація</a>
        }
    </div>

</div>
</div>
</nav>
</header>

<div class="container content-wrapper">
    <main role="main" class="pb-3">
        <div class="row">
            <div class="col-md-3">
                <vc:navigation-menu />
            </div>
            <div class="col-md-9">
                @if (TempData["message"] != null)
                {
                    <div class="alert alert-success">@TempData["message"]</div>
                }
                @RenderBody()
            </div>
        </div>
    </main>
</div>

<footer class="footer border-top bg-light text-muted mt-auto">
    <div class="container text-center py-3">
        &copy; 2025 - CinemaBooking - Усі права захищено
    </div>
</footer>

<script src="~/lib/bootstrap/js/bootstrap.bundle.min.js"></script>
</body>

```

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



</html>

Оновлений Controllers/AdminController.cs:

```
using Microsoft.AspNetCore.Mvc;
using CinemaBooking.Models;
using System.Linq;
using Microsoft.AspNetCore.Mvc.Rendering;
using Microsoft.AspNetCore.Authorization;

namespace CinemaBooking.Controllers
{
    [Authorize(Roles = "Admin")]
    public class AdminController : Controller
    {
        private ICinemaRepository repository;

        public AdminController(ICinemaRepository repo)
        {
            repository = repo;
        }

        public IActionResult Index() => View(repository.Movies);

        public IActionResult MovieDetails(long movieId)
        {
            Movie? movie = repository.Movies
                .FirstOrDefault(m => m.MovieID == movieId);
            return View(movie);
        }

        public IActionResult CreateMovie() => View("MovieEditor", new Movie());

        public IActionResult EditMovie(long movieId)
        {
            Movie? movie = repository.Movies
                .FirstOrDefault(m => m.MovieID == movieId);
            return View("MovieEditor", movie);
        }

        [HttpPost]
        public IActionResult SaveMovie(Movie movie)
        {
            if (ModelState.IsValid)
            {
                repository.SaveMovie(movie);
                TempData["message"] = $"{movie.Title} було збережено";
                return RedirectToAction("Index");
            }
            else
            {
                return View("MovieEditor", movie);
            }
        }

        public IActionResult DeleteMovieConfirmation(long movieId)
        {
            Movie? movie = repository.Movies
                .FirstOrDefault(m => m.MovieID == movieId);
            return View(movie);
        }

        [HttpPost]
        public IActionResult DeleteMovie(long movieId)
        {
            repository.DeleteMovie(movieId);
            TempData["message"] = "Фільм було видалено";
            return RedirectToAction("Index");
        }
    }
}
```

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

```

        Movie? movieToDelete = repository.Movies.FirstOrDefault(m => m.MovieID ==
movieId);
        if (movieToDelete != null)
        {
            repository.DeleteMovie(movieToDelete);
            TempData["message"] = $"{movieToDelete.Title} було видалено";
        }
        return RedirectToAction("Index");
    }

    public ViewResult SessionIndex() => View(repository.MovieSessions);

    public ViewResult SessionDetails(long sessionId)
    {
        MovieSession? session = repository.MovieSessions
            .FirstOrDefault(s => s.MovieSessionID == sessionId);
        return View(session);
    }

    public ViewResult CreateSession()
    {
        ViewBag.Movies = new SelectList(repository.Movies, "MovieID", "Title");
        return View("SessionEditor", new MovieSession());
    }

    public ViewResult EditSession(long sessionId)
    {
        MovieSession? session = repository.MovieSessions
            .FirstOrDefault(s => s.MovieSessionID == sessionId);
        ViewBag.Movies = new SelectList(repository.Movies, "MovieID", "Title",
session?.MovieID);
        return View("SessionEditor", session);
    }

    [HttpPost]
    public IActionResult SaveSession(MovieSession session)
    {
        if (ModelState.IsValid)
        {
            repository.SaveSession(session);
            TempData["message"] = $"Сеанс для фільму ID {session.MovieID} на
{session.SessionTime:g} було збережено";
            return RedirectToAction("SessionIndex");
        }
        else
        {
            ViewBag.Movies = new SelectList(repository.Movies, "MovieID", "Title",
session.MovieID);
            return View("SessionEditor", session);
        }
    }

    public ViewResult DeleteSessionConfirmation(long sessionId)
    {
        MovieSession? session = repository.MovieSessions
            .FirstOrDefault(s => s.MovieSessionID == sessionId);
        return View(session);
    }

    [HttpPost]
    public IActionResult DeleteSession(long sessionId)
    {
        MovieSession? sessionToDelete = repository.MovieSessions.FirstOrDefault(s =>
s.MovieSessionID == sessionId);
        if (sessionToDelete != null)

```

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

```

        {
            repository.DeleteSession(sessionToDelete);
            TempData["message"] = $"Сеанс на {sessionToDelete.SessionTime:g} було
видалено";
        }
        return RedirectToAction("SessionIndex");
    }
}
}

```

**Оновлений** Controllers/BookingController.cs:

```

using Microsoft.AspNetCore.Mvc;
using CinemaBooking.Models;
using CinemaBooking.Models.ViewModels;
using Microsoft.AspNetCore.Authorization;

namespace CinemaBooking.Controllers
{
    [Authorize]
    public class BookingController : Controller
    {
        private ICinemaRepository repository;
        private Booking booking;

        public BookingController(ICinemaRepository repo, Booking bookingService)
        {
            repository = repo;
            booking = bookingService;
        }

        public IActionResult Index(string returnUrl)
        {
            return View(new BookingIndexViewModel
            {
                Booking = booking,
                ReturnUrl = returnUrl ?? "/"
            });
        }

        [HttpPost]
        public IActionResult AddToBooking(long movieSessionID, string returnUrl)
        {
            MovieSession? session = repository.MovieSessions
                .FirstOrDefault(p => p.MovieSessionID == movieSessionID);

            if (session != null)
            {
                booking.AddItem(session, 1);
            }
            return Redirect(returnUrl ?? "/");
        }

        [HttpPost]
        public IActionResult RemoveFromBooking(long movieSessionID, string returnUrl)
        {
            MovieSession? session = repository.MovieSessions
                .FirstOrDefault(p => p.MovieSessionID == movieSessionID);

            if (session != null)
            {
                booking.RemoveLine(session);
            }
            return RedirectToAction("Index", new { returnUrl });
        }
    }
}

```

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

}

Реєстрація

Ім'я користувача

Email

Пароль

Мінімум 8 символів, великі та малі літери, цифри.

Підтвердіть пароль

Зареєструватися

Вже маєте акаунт? [Увійти](#)

© 2025 - CinemaBooking - Усі права захищено

Рис.1 Результат виконання

Вхід

Ім'я користувача

Пароль

Увійти

Ще не маєте акаунту? [Зареєструватися](#)

© 2025 - CinemaBooking - Усі права захищено

Рис.2 Результат виконання

Обрані

Привіт, Admin!

Кабінет

Вийти

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

Рис.3 Результат виконання

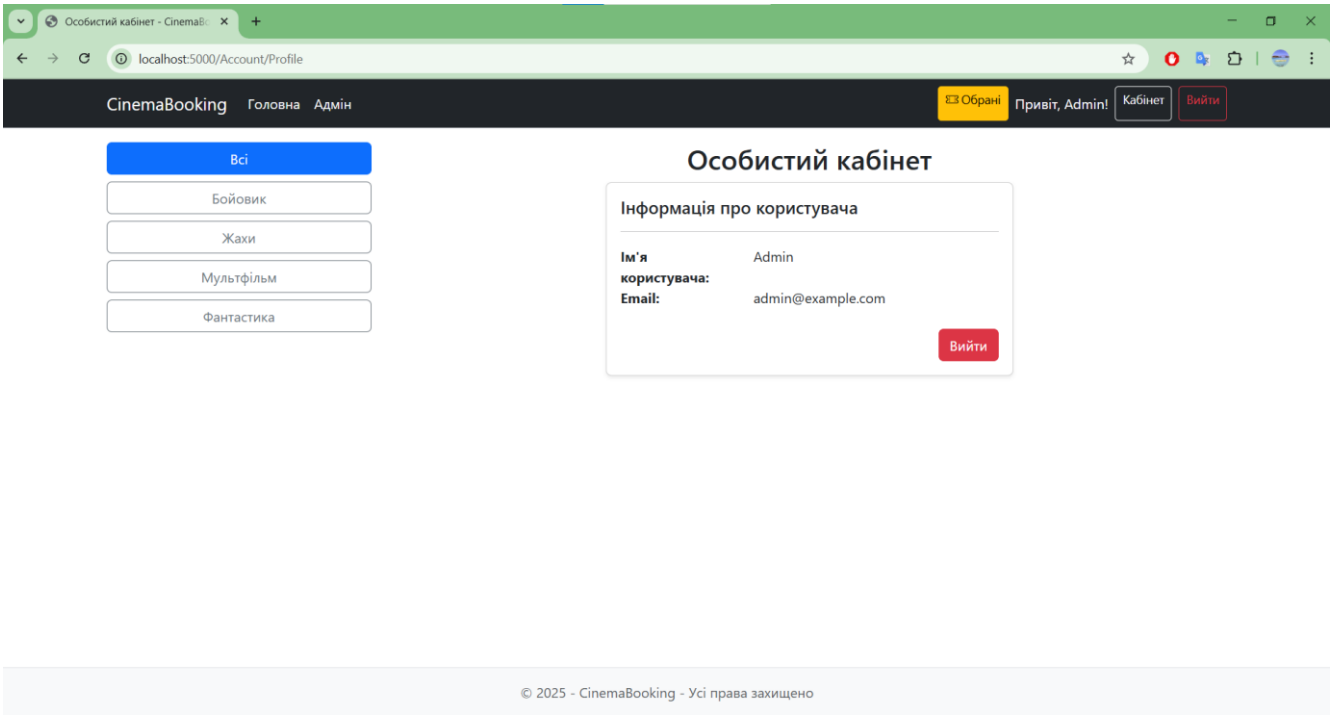


Рис.4 Результат виконання

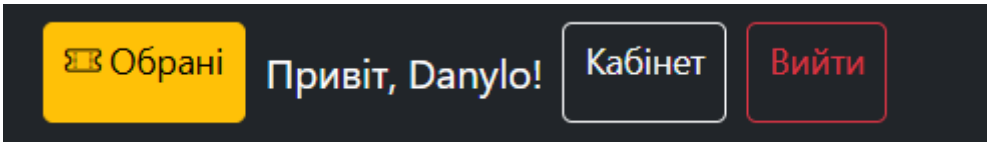


Рис.5 Результат виконання

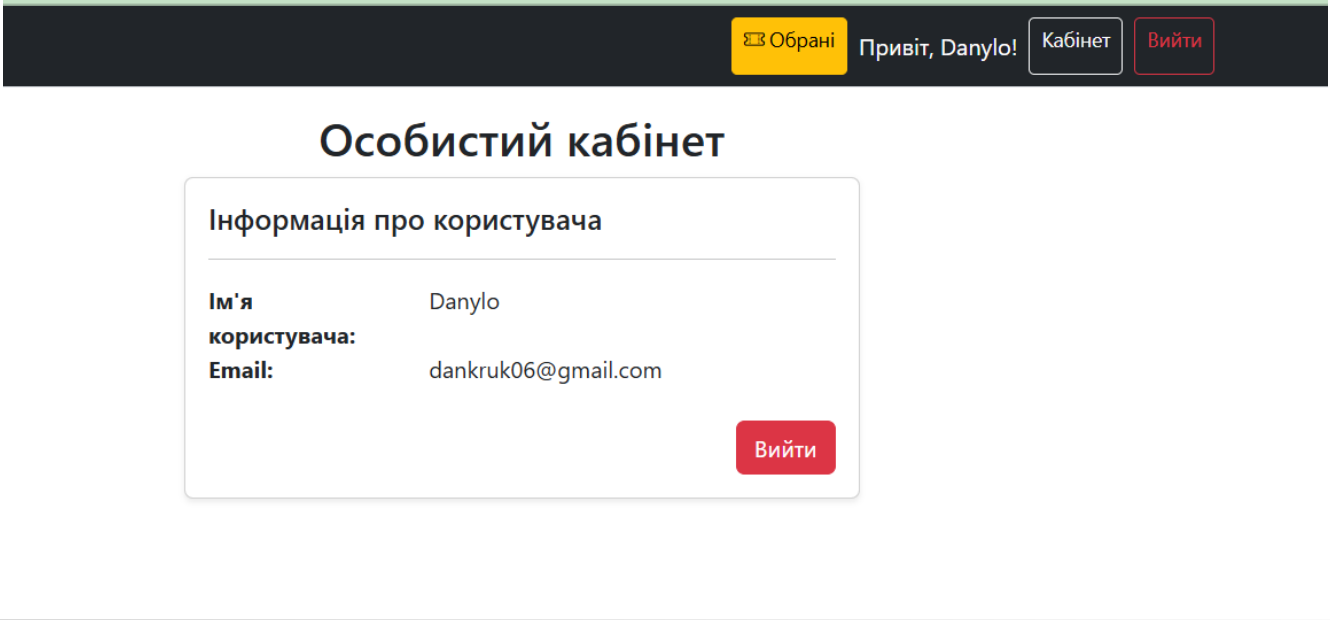


Рис.6 Результат виконання

Ім'я користувача

Будь ласка, введіть ім'я користувача

Email

Будь ласка, введіть Email

Пароль

Будь ласка, введіть пароль Мінімум 8 символів, великі та малі літери, цифри.

Підтвердіть пароль

Будь ласка, підтвердіть пароль

Зареєструватися

Вже маєте акаунт? [Увійти](#)

Рис.7 Результат виконання

## Завдання 2.

В рішенні створити новий WebAPI проект. Цей проект повинен реалізовувати всі функціональності, які наявні в MVC проекті у вигляді REST-ендпоїнтів. Для кращої організації і перевикористання коду слід перенести файли для роботи з БД (репозиторії, контексти, моделі) в окрему бібліотеку і додати посилання на неї в MVC та WebAPI проектах. Встановіть необхідні залежності для проекту бібліотеки.

Після цього налаштуйте `ConnectionString`'и для створених БД для WebAPI проекту в файлі `appsettings.json`. Таким чином, після перенесення файлів для роботи з БД в окрему бібліотеку ми можемо використовувати всі необхідні сервіси в обох проектах.

**Реалізуйте всі необхідні CRUD-операції в вашому WebAPI проекті, включаючи реєстрацію користувача. Перевірте працездатність ваших ендпоїнтів за допомогою Postman, Swagger або будь-яким іншим зручним інструментом**

Створення Бібліотеки `CinemaBooking.Data` та Перенесення Файлів

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

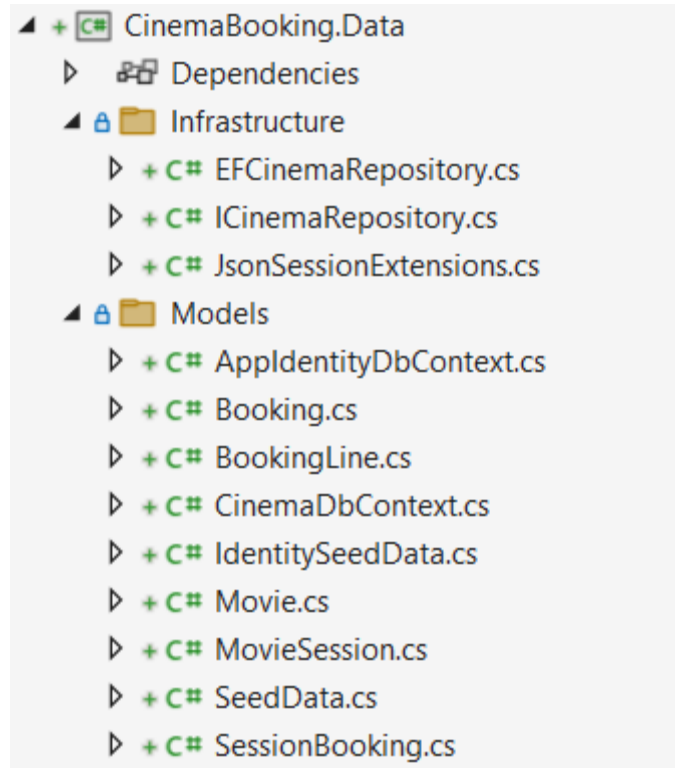


Рис.8 Результат виконання

Рефакторинг SeedData та IdentitySeedData:

SeedData.cs:

```
using Microsoft.EntityFrameworkCore;
using System.Globalization;
using System.Linq;
using System;
using Microsoft.Extensions.DependencyInjection;

namespace CinemaBooking.Data.Models
{
    public static class SeedData
    {
        public static void EnsurePopulated(IServiceProvider services)
        {
            using (var scope = services.CreateScope())
            {
                var context = scope.ServiceProvider
                    .GetRequiredService<CinemaDbContext>();

                if (context.Database.GetPendingMigrations().Any())
                {
                    context.Database.Migrate();
                }

                if (!context.Movies.Any())
                {
                    var dune = new Movie
                    {
                        Title = "Дюна: Частина друга",
                        Description = "Пол Атрід об'єднується з Чані та Фременами, щоб помститися змовникам, які знищили його родину.",
                        Genre = "Фантастика"
                    };
                }
            }
        }
    }
}
```

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

```

var joker = new Movie
{
    Title = "Джокер: Божевільня на двох",
    Description = "Артур Флек у психіатричній лікарні Аркхем зустрі-
чає Гарлі Квінн.",
    Genre = "Трилер / Мюзикл"
};

var panda = new Movie
{
    Title = "Панда Кунг-Фу 4",
    Description = "Воїн Дракона По має знайти собі наступника, перш
ніж стати Духовним лідером Долини Миру.",
    Genre = "Мультфільм"
};

var civilWar = new Movie
{
    Title = "Повстання Штатів",
    Description = "У найближчому майбутньому команда журналістів по-
дорожує Сполученими Штатами під час громадянської війни.",
    Genre = "Бойовик"
};

var mavka = new Movie
{
    Title = "Мавка. Лісова пісня",
    Description = "Мавка, душа Лісу, стикається з неможливим вибором
між коханням і обов'язком берегині.",
    Genre = "Мультфільм"
};

context.Movies.AddRange(dune, joker, panda, civilWar, mavka);

context.MovieSessions.AddRange(
    new MovieSession
    {
        Movie = dune,
        Hall = "Зал 1 (IMAX)",
        Price = 220,
        SessionTime = DateTime.ParseExact("28.10.2025 18:00",
"dd.MM.yyyy HH:mm", CultureInfo.InvariantCulture)
    },
    new MovieSession
    {
        Movie = joker,
        Hall = "Зал 2",
        Price = 180,
        SessionTime = DateTime.ParseExact("28.10.2025 19:30",
"dd.MM.yyyy HH:mm", CultureInfo.InvariantCulture)
    },
    new MovieSession
    {
        Movie = panda,
        Hall = "Зал 3 (Дитячий)",
        Price = 150,
        SessionTime = DateTime.ParseExact("28.10.2025 16:00",
"dd.MM.yyyy HH:mm", CultureInfo.InvariantCulture)
    },
    new MovieSession
    {
        Movie = civilWar,
        Hall = "Зал 1 (IMAX)",
        Price = 210,

```

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



IdentitySeedData.cs:

		Круковський Д.С.			ДУ «Житомирська політехніка».24.121.01.000 - Лр1	Арк.
		Українець М.О.				17
Змн.	Арк.	№ докум.	Підпис	Дата		

```

    }
  }
}

```

Створено ASP.NET Core Web API Project:

Program.cs для API:

```

using CinemaBooking.Data.Infrastructure;
using CinemaBooking.Data.Models;
using Microsoft.AspNetCore.Identity;
using Microsoft.EntityFrameworkCore;
using Microsoft.OpenApi.Models;
using System.Text.Json.Serialization;

var builder = WebApplication.CreateBuilder(args);

builder.Services.AddCors(options =>
{
    options.AddDefaultPolicy(policy =>
    {
        policy.AllowAnyOrigin()
            .AllowAnyMethod()
            .AllowAnyHeader();
    });
});

builder.Services.AddControllers().AddJsonOptions(opts =>
{
    opts.JsonSerializerOptions.ReferenceHandler = ReferenceHandler.IgnoreCycles;
});

builder.Services.AddEndpointsApiExplorer();
builder.Services.AddSwaggerGen(c => {
    c.SwaggerDoc("v1", new OpenApiInfo { Title = "CinemaBooking API", Version = "v1" });

    c.AddSecurityDefinition("Bearer", new OpenApiSecurityScheme
    {
        Description = "JWT Authorization header using the Bearer scheme. Example:
        \"Bearer {token}\"",
        Name = "Authorization",
        In = ParameterLocation.Header,
        Type = SecuritySchemeType.ApiKey,
        Scheme = "Bearer"
    });
    c.AddSecurityRequirement(new OpenApiSecurityRequirement() {
        {
            new OpenApiSecurityScheme {
                Reference = new OpenApiReference {
                    Type = ReferenceType.SecurityScheme,
                    Id = "Bearer"
                },
                Scheme = "oauth2",
                Name = "Bearer",
                In = ParameterLocation.Header,
            },
            new List<string>()
        }
    });
});

```

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

```

});
});

builder.Services.AddDbContext<CinemaDbContext>(opts => {
    opts.UseSqlServer(
        builder.Configuration["ConnectionStrings:CinemaBookingConnection"]);
});

builder.Services.AddDbContext<AppIdentityDbContext>(options =>
    options.UseSqlServer(
        builder.Configuration["ConnectionStrings:IdentityConnection"])
);

builder.Services.AddScoped<ICinemaRepository, EFCinemaRepository>();

builder.Services.AddIdentity<IdentityUser, IdentityRole>(opts => {
    opts.User.RequireUniqueEmail = true;
    opts.Password.RequiredLength = 8;
    opts.Password.RequireNonAlphanumeric = false;
    opts.Password.RequireLowercase = true;
    opts.Password.RequireUppercase = true;
    opts.Password.RequireDigit = true;
})
    .AddEntityFrameworkStores<AppIdentityDbContext>();

var app = builder.Build();

if (app.Environment.IsDevelopment())
{
    app.UseSwagger();
    app.UseSwaggerUI();
}

app.UseHttpsRedirection();

app.UseCors();

app.UseAuthentication();
app.UseAuthorization();

app.MapControllers();

_ = Task.Run(() => IdentitySeedData.EnsurePopulatedAsync(app.Services));
SeedData.EnsurePopulated(app.Services);

app.Run();

```

Тепер створимо ендпоїнти. Ми будемо використовувати DTO (Data Transfer Objects) для передачі даних.

**Створено** DTOs/RegisterDto.cs:

```

using System.ComponentModel.DataAnnotations;

namespace CinemaBooking.API.DTOs
{
    public class RegisterDto
    {
        [Required(ErrorMessage = "Ім'я користувача є обов'язковим")]

```

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

```

        public string Username { get; set; } = string.Empty;

        [Required(ErrorMessage = "Email є обов'язковим")]
        [EmailAddress(ErrorMessage = "Некоректний формат Email")]
        public string Email { get; set; } = string.Empty;

        [Required(ErrorMessage = "Пароль є обов'язковим")]
        public string Password { get; set; } = string.Empty;
    }
}

```

**Створено** DTOs/LoginDto.cs:

```

using System.ComponentModel.DataAnnotations;

namespace CinemaBooking.API.DTOs
{
    public class LoginDto
    {
        [Required(ErrorMessage = "Ім'я користувача є обов'язковим")]
        public string Username { get; set; } = string.Empty;

        [Required(ErrorMessage = "Пароль є обов'язковим")]
        public string Password { get; set; } = string.Empty;
    }
}

```

**Створено** DTOs/MovieDto.cs:

```

namespace CinemaBooking.API.DTOs
{
    public class MovieDto
    {
        public long MovieID { get; set; }
        public string Title { get; set; } = string.Empty;
        public string Description { get; set; } = string.Empty;
        public string Genre { get; set; } = string.Empty;
    }
}

```

**Створено** DTOs/MovieSessionDto.cs:

```

namespace CinemaBooking.API.DTOs
{
    public class MovieSessionDto
    {
        public long MovieSessionID { get; set; }
        public string Hall { get; set; } = string.Empty;
        public decimal Price { get; set; }
        public DateTime SessionTime { get; set; }

        public long? MovieID { get; set; }

        public string? MovieTitle { get; set; }
        public string? MovieGenre { get; set; }
    }
}

```

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

Створено DTOs/CreateUpdateMovieDto.cs:

```
using System.ComponentModel.DataAnnotations;

namespace CinemaBooking.API.DTOs
{
    public class CreateUpdateMovieDto
    {
        [Required(ErrorMessage = "Будь ласка, введіть назву фільму")]
        public string Title { get; set; } = string.Empty;

        [Required(ErrorMessage = "Будь ласка, введіть опис")]
        public string Description { get; set; } = string.Empty;

        [Required(ErrorMessage = "Будь ласка, введіть жанр")]
        public string Genre { get; set; } = string.Empty;
    }
}
```

Створено DTOs/CreateUpdateSessionDto.cs:

```
using System.ComponentModel.DataAnnotations;

namespace CinemaBooking.API.DTOs
{
    public class CreateUpdateSessionDto
    {
        [Required(ErrorMessage = "Будь ласка, введіть назву залу")]
        public string Hall { get; set; } = string.Empty;

        [Required(ErrorMessage = "Будь ласка, введіть ціну")]
        [Range(0.01, double.MaxValue, ErrorMessage = "Будь ласка, введіть позитивну ці-
ну")]
        public decimal Price { get; set; }

        [Required(ErrorMessage = "Будь ласка, введіть час сеансу")]
        public DateTime SessionTime { get; set; }

        [Required(ErrorMessage = "Будь ласка, вкажіть ID фільму")]
        public long MovieID { get; set; }
    }
}
```

Створено Controllers/MoviesController.cs:

```
using CinemaBooking.API.DTOs;
using CinemaBooking.Data.Infrastructure;
using CinemaBooking.Data.Models;
using Microsoft.AspNetCore.Http;
using Microsoft.AspNetCore.Mvc;
using Microsoft.EntityFrameworkCore;

namespace CinemaBooking.API.Controllers
{
    [Route("api/[controller]")]
    [ApiController]
    public class MoviesController : ControllerBase
    {
        private readonly ICinemaRepository _repository;

        public MoviesController(ICinemaRepository repository)
        {
            _repository = repository;
        }
    }
}
```

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

```

[HttpGet]
public async Task<ActionResult<IEnumerable<MovieDto>>> GetMovies()
{
    var movies = await _repository.Movies
        .Select(m => new MovieDto
        {
            MovieID = m.MovieID,
            Title = m.Title,
            Description = m.Description,
            Genre = m.Genre
        })
        .ToListAsync();

    return Ok(movies);
}

[HttpGet("{id}")]
public async Task<ActionResult<MovieDto>> GetMovie(long id)
{
    var movie = await _repository.Movies
        .Select(m => new MovieDto
        {
            MovieID = m.MovieID,
            Title = m.Title,
            Description = m.Description,
            Genre = m.Genre
        })
        .FirstOrDefaultAsync(m => m.MovieID == id);

    if (movie == null)
    {
        return NotFound();
    }

    return Ok(movie);
}

[HttpPost]
public ActionResult<MovieDto> CreateMovie(CreateUpdateMovieDto createMovieDto)
{
    if (!ModelState.IsValid)
    {
        return BadRequest(ModelState);
    }

    var movie = new Movie
    {
        Title = createMovieDto.Title,
        Description = createMovieDto.Description,
        Genre = createMovieDto.Genre
    };

    _repository.SaveMovie(movie);

    var movieDto = new MovieDto
    {
        MovieID = movie.MovieID,
        Title = movie.Title,
        Description = movie.Description,
        Genre = movie.Genre
    };

    return CreatedAtAction(nameof(GetMovie), new { id = movie.MovieID },
movieDto);
}

```

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

```

[HttpPut("{id}")]
public async Task<IActionResult> UpdateMovie(long id, CreateUpdateMovieDto
updateMovieDto)
{
    if (!ModelState.IsValid)
    {
        return BadRequest(ModelState);
    }

    var movieToUpdate = await _repository.Movies
        .FirstOrDefaultAsync(m => m.MovieID == id);

    if (movieToUpdate == null)
    {
        return NotFound();
    }

    movieToUpdate.Title = updateMovieDto.Title;
    movieToUpdate.Description = updateMovieDto.Description;
    movieToUpdate.Genre = updateMovieDto.Genre;

    _repository.SaveMovie(movieToUpdate);

    return NoContent();
}

[HttpDelete("{id}")]
public async Task<IActionResult> DeleteMovie(long id)
{
    var movieToDelete = await _repository.Movies
        .FirstOrDefaultAsync(m => m.MovieID == id);

    if (movieToDelete == null)
    {
        return NotFound();
    }

    _repository.DeleteMovie(movieToDelete);

    return NoContent();
}
}
}

```

**Створено** Controllers/SessionsController.cs:

```

using CinemaBooking.API.DTOs;
using CinemaBooking.Data.Infrastructure;
using CinemaBooking.Data.Models;
using Microsoft.AspNetCore.Http;
using Microsoft.AspNetCore.Mvc;
using Microsoft.EntityFrameworkCore;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;

namespace CinemaBooking.API.Controllers
{
    [Route("api/[controller]")]
    [ApiController]
    public class SessionsController : ControllerBase
    {
        private readonly ICinemaRepository _repository;
        private readonly CinemaDbContext _context;
    }
}

```

		Круковський Д.С.			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				23
Змн.	Арк.	№ докум.	Підпис	Дата		

```

public SessionsController(ICinemaRepository repository, CinemaDbContext context)
{
    _repository = repository;
    _context = context;
}

[HttpGet]
public async Task<ActionResult<IEnumerable<MovieSessionDto>>> GetSessions()
{
    var sessions = await _repository.MovieSessions
        .OrderBy(s => s.SessionTime)
        .Select(s => new MovieSessionDto
        {
            MovieSessionID = s.MovieSessionID,
            Hall = s.Hall,
            Price = s.Price,
            SessionTime = s.SessionTime,
            MovieID = s.MovieID,
            MovieTitle = s.Movie != null ? s.Movie.Title : "N/A",
            MovieGenre = s.Movie != null ? s.Movie.Genre : "N/A"
        })
        .ToListAsync();

    return Ok(sessions);
}

[HttpGet("{id}")]
public async Task<ActionResult<MovieSessionDto>> GetSession(long id)
{
    var session = await _repository.MovieSessions
        .Where(s => s.MovieSessionID == id)
        .Select(s => new MovieSessionDto
        {
            MovieSessionID = s.MovieSessionID,
            Hall = s.Hall,
            Price = s.Price,
            SessionTime = s.SessionTime,
            MovieID = s.MovieID,
            MovieTitle = s.Movie != null ? s.Movie.Title : "N/A",
            MovieGenre = s.Movie != null ? s.Movie.Genre : "N/A"
        })
        .FirstOrDefaultAsync();

    if (session == null)
    {
        return NotFound($"Сеанс з ID {id} не знайдено.");
    }

    return Ok(session);
}

[HttpPost]
public async Task<ActionResult<MovieSessionDto>>
CreateSession(CreateUpdateSessionDto createSessionDto)
{
    if (!ModelState.IsValid)
    {
        return BadRequest(ModelState);
    }

    var movieExists = await _repository.Movies.AnyAsync(m => m.MovieID ==
createSessionDto.MovieID);
    if (!movieExists)
    {

```

		Круковський Д.С.			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				24
Змн.	Арк.	№ докум.	Підпис	Дата		



```

        ModelState.AddModelError(nameof(createSessionDto.MovieID), $"Фільм з ID
{createSessionDto.MovieID} не знайдено.");
        return BadRequest(ModelState);
    }

    var session = new MovieSession
    {
        Hall = createSessionDto.Hall,
        Price = createSessionDto.Price,
        SessionTime = createSessionDto.SessionTime,
        MovieID = createSessionDto.MovieID
    };

    _context.MovieSessions.Add(session);
    _repository.SaveSession(session);

    await _context.Entry(session).Reference(s => s.Movie).LoadAsync();

    var sessionDto = new MovieSessionDto
    {
        MovieSessionID = session.MovieSessionID,
        Hall = session.Hall,
        Price = session.Price,
        SessionTime = session.SessionTime,
        MovieID = session.MovieID,
        MovieTitle = session.Movie != null ? session.Movie.Title : "N/A",
        MovieGenre = session.Movie != null ? session.Movie.Genre : "N/A"
    };

    return CreatedAtAction(nameof(GetSession), new { id = session.MovieSessionID
}, sessionDto);
}

[HttpPut("{id}")]
public async Task<IActionResult> UpdateSession(long id, CreateUpdateSessionDto
updateSessionDto)
{
    if (!ModelState.IsValid)
    {
        return BadRequest(ModelState);
    }

    var movieExists = await _repository.Movies.AnyAsync(m => m.MovieID ==
updateSessionDto.MovieID);
    if (!movieExists)
    {
        ModelState.AddModelError(nameof(updateSessionDto.MovieID), $"Фільм з ID
{updateSessionDto.MovieID} не знайдено.");
        return BadRequest(ModelState);
    }

    var sessionExists = await _context.MovieSessions
        .AsNoTracking()
        .AnyAsync(s => s.MovieSessionID == id);

    if (!sessionExists)
    {
        return NotFound($"Сеанс з ID {id} не знайдено.");
    }

    var updatedSession = new MovieSession
    {
        MovieSessionID = id,
        Hall = updateSessionDto.Hall,
        Price = updateSessionDto.Price,

```

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				25
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        SessionTime = updateSessionDto.SessionTime,
        MovieID = updateSessionDto.MovieID
    };

    _context.MovieSessions.Update(updatedSession);
    _repository.SaveSession(updatedSession);

    return NoContent();
}

[HttpDelete("{id}")]
public async Task<IActionResult> DeleteSession(long id)
{
    var sessionToDelete = await _context.MovieSessions
        .FirstOrDefaultAsync(s => s.MovieSessionID ==
id);

    if (sessionToDelete == null)
    {
        return NotFound($"Сеанс з ID {id} не знайдено.");
    }

    _repository.DeleteSession(sessionToDelete);

    return NoContent();
}
}
}

```

**Створено** Controllers/AccountController.cs:

```

using CinemaBooking.API.DTOs;
using CinemaBooking.Data.Models;
using Microsoft.AspNetCore.Http;
using Microsoft.AspNetCore.Identity;
using Microsoft.AspNetCore.Mvc;
using System.Threading.Tasks;

namespace CinemaBooking.API.Controllers
{
    [Route("api/[controller]")]
    [ApiController]
    public class AccountController : ControllerBase
    {
        private readonly UserManager<IdentityUser> _userManager;
        private readonly SignInManager<IdentityUser> _signInManager;
        private readonly RoleManager<IdentityRole> _roleManager;

        public AccountController(UserManager<IdentityUser> userManager,
SignInManager<IdentityUser> signInManager, RoleManager<IdentityRole> roleManager)
        {
            _userManager = userManager;
            _signInManager = signInManager;
            _roleManager = roleManager;
        }

        [HttpPost("register")]
        public async Task<IActionResult> Register(RegisterDto registerDto)
        {
            if (!ModelState.IsValid)
            {
                return BadRequest(ModelState);
            }
        }
    }
}

```

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				26
Змн.	Арк.	№ докум.	Підпис	Дата		

```

var user = new IdentityUser
{
    UserName = registerDto.Username,
    Email = registerDto.Email,
    EmailConfirmed = true
};

var result = await _userManager.CreateAsync(user, registerDto.Password);

if (result.Succeeded)
{
    if (!await _roleManager.RoleExistsAsync("User"))
        await _roleManager.CreateAsync(new IdentityRole("User"));
    if (!await _roleManager.RoleExistsAsync("Admin"))
        await _roleManager.CreateAsync(new IdentityRole("Admin"));

    await _userManager.AddToRoleAsync(user, "User");

    return StatusCode(StatusCodes.Status201Created, new { message = "User
registered successfully." });
}
else
{
    foreach (var error in result.Errors)
    {
        ModelState.AddModelError(string.Empty, error.Description);
    }
    return BadRequest(ModelState);
}
}

[HttpPost("login")]
public async Task<IActionResult> Login(LoginDto loginDto)
{
    if (!ModelState.IsValid)
    {
        return BadRequest(ModelState);
    }

    var user = await _userManager.FindByNameAsync(loginDto.Username);

    if (user != null && await _userManager.CheckPasswordAsync(user,
loginDto.Password))
    {
        var result = await _signInManager.PasswordSignInAsync(user,
loginDto.Password, isPersistent: false, lockoutOnFailure: false);

        if (result.Succeeded)
        {
            return Ok(new { message = "Login successful" });
        }
        if (result.IsLockedOut)
        {
            return BadRequest(new { message = "User account locked out." });
        }
        if (result.IsNotAllowed)
        {
            return BadRequest(new { message = "User is not allowed to sign in
(check email confirmation?)." });
        }
    }
}

```

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				27
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        return Unauthorized(new { message = "Invalid username or password" });
    }

    [HttpPost("logout")]
    public async Task<IActionResult> Logout()
    {
        await _signInManager.SignOutAsync();
        return Ok(new { message = "Logout successful" });
    }
}

```

**Account**

**POST** /api/Account/register

Parameters: No parameters

Request body: application/json

```
{
  "username": "Dmytro",
  "email": "Dmytro@gmail.com",
  "password": "1234zxcl!"
}
```

Execute Clear

Рис.9 Результат виконання

**Responses**

Curl

```
curl -X 'POST' \
  'https://localhost:7102/api/Account/register' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "username": "Dmytro",
    "email": "Dmytro@gmail.com",
    "password": "1234zxcl!"
  }'
```

Request URL: https://localhost:7102/api/Account/register

Server response

Code	Details
201	<p>Response body</p> <pre>{   "message": "User registered successfully." }</pre> <p>Response headers</p> <pre>access-control-allow-origin: * content-type: application/json; charset=utf-8 date: Wed, 29 Oct 2025 20:54:43 GMT server: Kestrel</pre>

Responses

Code	Description	Links
200	OK	No links

Рис.10 Результат виконання

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				28
Змн.	Арк.	№ докум.	Підпис	Дата		

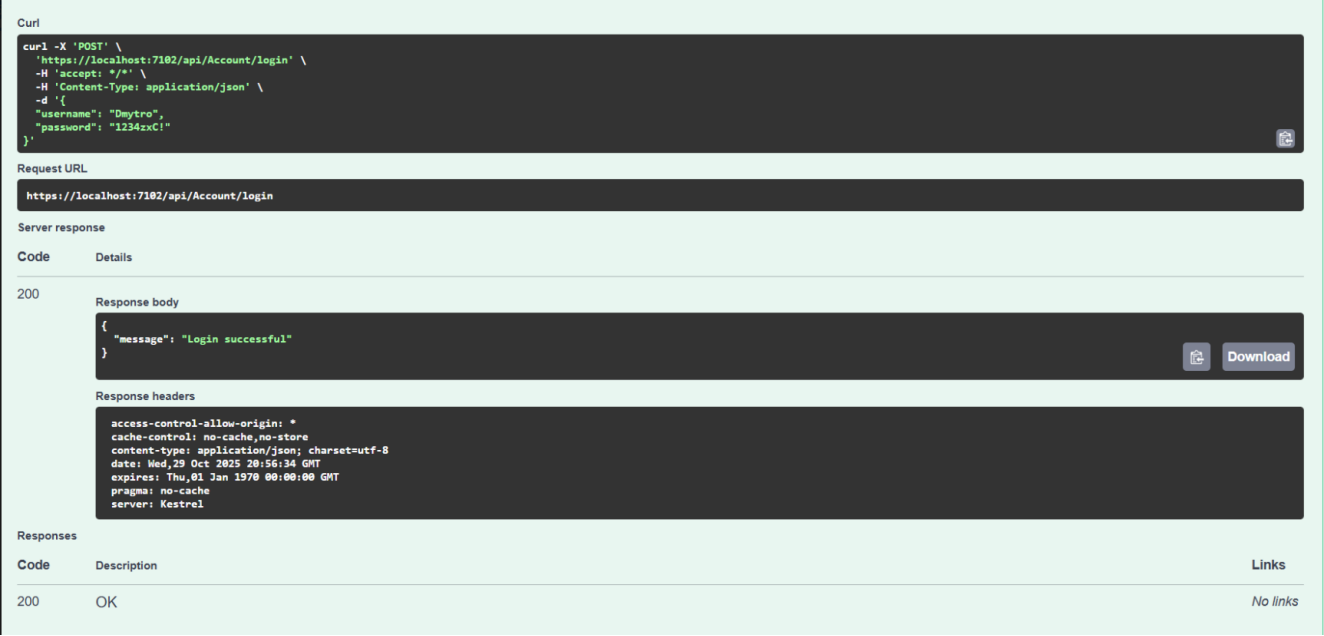


Рис.11 Результат виконання

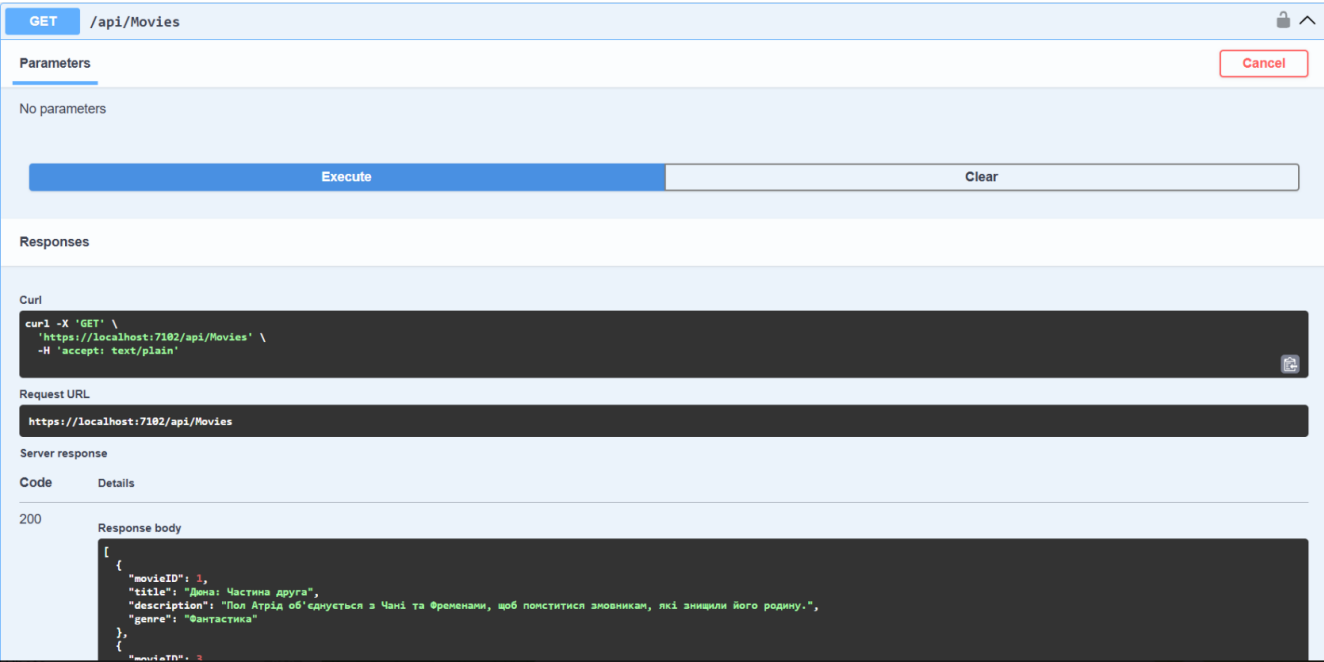


Рис.12 Результат виконання

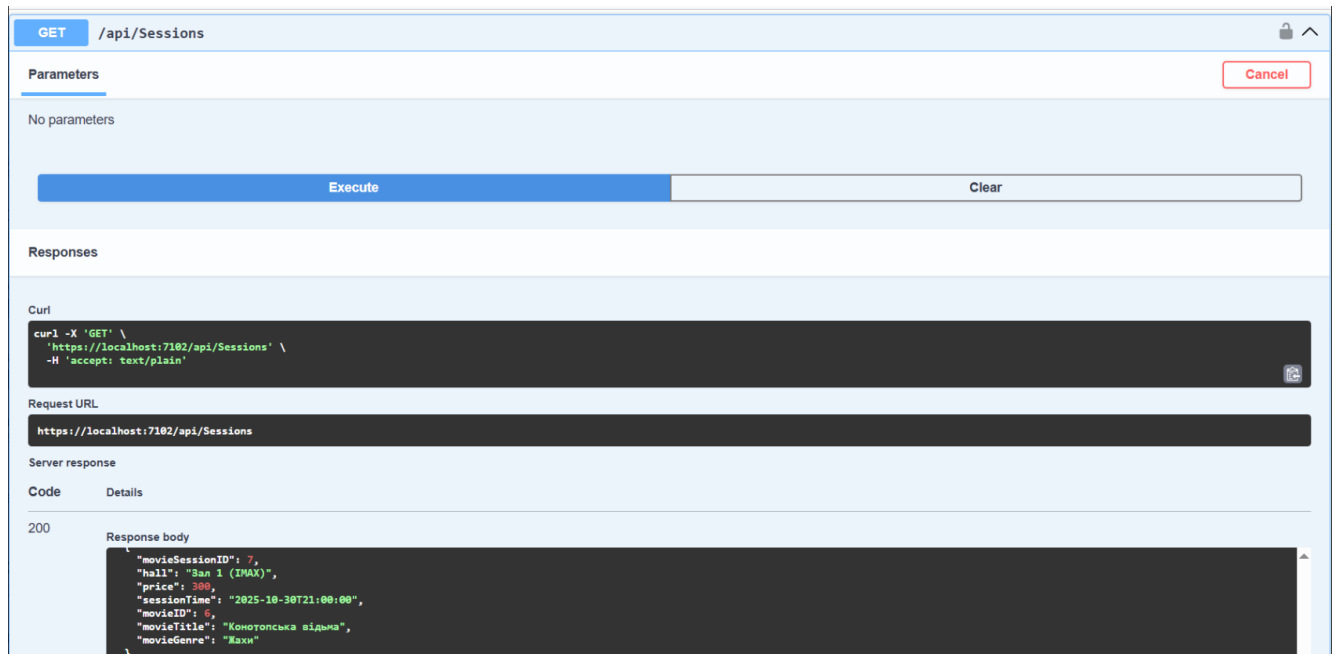


Рис.13 Результат виконання

### Завдання 3.

Реалізуйте автентифікацію та авторизацію користувача з використанням механізму JSON Web Token (JWT). Вона повинна працювати наступним чином:

1. Клієнт відправляє запит на login-ендпоїнт зі своїм логіном та паролем.
2. Якщо користувач з таким логіном та паролем існує, тоді сервер повинен повернути згенерований JSON Web Token.
3. Для отримання доступу до дій, які вимагають авторизації, клієнт відправляє в запиті заголовок Authorization зі значенням Bearer TokenValue, де TokenValue – токен, який було згенеровано під час автентифікації.
4. Якщо токен валідний – сервер виконає запит, інакше поверне помилку 401 Unauthorized.

Нам потрібен пакет для роботи з JWT:

`dotnet add package Microsoft.AspNetCore.Authentication.JwtBearer --version 8.0.0`

Налаштування JWT в `appsettings.json`:

```
{
  "JwtSettings": {
    "Secret": "fX9dcAmaY7edV0npTihS8ylCsmK2vsxL3myB8ys",
    "Issuer": "CinemaBookingAPI",
    "Audience": "CinemaBookingClient"
  }
}
```

Налаштуємо сервіси автентифікації для використання JWT.

**Замінений** вміст `CinemaBooking.API/Program.cs`:

```
using CinemaBooking.Data.Infrastructure;
using CinemaBooking.Data.Models;
using Microsoft.AspNetCore.Identity;
```

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				30
Змн.	Арк.	№ докум.	Підпис	Дата		

```

using Microsoft.EntityFrameworkCore;
using Microsoft.OpenApi.Models;
using System.Text.Json.Serialization;
using Microsoft.AspNetCore.Authentication.JwtBearer;
using Microsoft.IdentityModel.Tokens;
using System.Text;

var builder = WebApplication.CreateBuilder(args);

builder.Services.AddCors(options =>
{
    options.AddDefaultPolicy(policy =>
    {
        policy.AllowAnyOrigin()
            .AllowAnyMethod()
            .AllowAnyHeader();
    });
});

builder.Services.AddControllers().AddJsonOptions(opts =>
{
    opts.JsonSerializerOptions.ReferenceHandler = ReferenceHandler.IgnoreCycles;
});

builder.Services.AddEndpointsApiExplorer();
builder.Services.AddSwaggerGen(c => {
    c.SwaggerDoc("v1", new OpenApiInfo { Title = "CinemaBooking API", Version = "v1" });
    c.AddSecurityDefinition("Bearer", new OpenApiSecurityScheme
    {
        Description = "JWT Authorization header using the Bearer scheme. Example:
        \"Bearer {token}\"",
        Name = "Authorization",
        In = ParameterLocation.Header,
        Type = SecuritySchemeType.ApiKey,
        Scheme = "Bearer"
    });
    c.AddSecurityRequirement(new OpenApiSecurityRequirement() {
        {
            new OpenApiSecurityScheme {
                Reference = new OpenApiReference {
                    Type = ReferenceType.SecurityScheme,
                    Id = "Bearer"
                },
                Scheme = "oauth2",
                Name = "Bearer",
                In = ParameterLocation.Header,
            },
            new List<string>()
        }
    });
});

builder.Services.AddDbContext<CinemaDbContext>(opts => {
    opts.UseSqlServer(
        builder.Configuration["ConnectionStrings:CinemaBookingConnection"]);
});
builder.Services.AddDbContext<AppIdentityDbContext>(options =>
    options.UseSqlServer(
        builder.Configuration["ConnectionStrings:IdentityConnection"])
);

builder.Services.AddScoped<ICinemaRepository, EFCinemaRepository>();

builder.Services.AddIdentity<IdentityUser, IdentityRole>(opts => {
    opts.User.RequireUniqueEmail = true;
});

```

		Круковський Д.С.			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				31
Змн.	Арк.	№ докум.	Підпис	Дата		

```

    opts.Password.RequiredLength = 8;
    opts.Password.RequireNonAlphanumeric = false;
    opts.Password.RequireLowercase = true;
    opts.Password.RequireUppercase = true;
    opts.Password.RequireDigit = true;
})
.AddEntityFrameworkStores<AppIdentityDbContext>();

var jwtSecretKey = builder.Configuration["JwtSettings:Secret"];
if (string.IsNullOrEmpty(jwtSecretKey))
{
    throw new ArgumentNullException("JwtSettings:Secret",
        "JWT Secret key is not configured in appsettings.json");
}
var key = new SymmetricSecurityKey(Encoding.UTF8.GetBytes(jwtSecretKey));

builder.Services.AddAuthentication(options =>
{
    options.DefaultAuthenticateScheme = JwtBearerDefaults.AuthenticationScheme;
    options.DefaultChallengeScheme = JwtBearerDefaults.AuthenticationScheme;
})
.AddJwtBearer(options =>
{
    options.TokenValidationParameters = new TokenValidationParameters
    {
        ValidateIssuer = true,
        ValidateAudience = true,
        ValidateLifetime = true,
        ValidateIssuerSigningKey = true,

        ValidIssuer = builder.Configuration["JwtSettings:Issuer"],
        ValidAudience = builder.Configuration["JwtSettings:Audience"],
        IssuerSigningKey = key
    };
});

var app = builder.Build();

if (app.Environment.IsDevelopment())
{
    app.UseSwagger();
    app.UseSwaggerUI();
}

app.UseHttpsRedirection();
app.UseCors();

app.UseAuthentication();
app.UseAuthorization();

app.MapControllers();

_ = Task.Run(() => IdentitySeedData.EnsurePopulatedAsync(app.Services));
SeedData.EnsurePopulated(app.Services);

app.Run();

```

**Замінений** Вміст CinemaBooking.API/Controllers/AccountController.cs:

```

using CinemaBooking.API.DTOS;
using CinemaBooking.Data.Models;
using Microsoft.AspNetCore.Http;
using Microsoft.AspNetCore.Identity;
using Microsoft.AspNetCore.Mvc;

```

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				32
Змн.	Арк.	№ докум.	Підпис	Дата		



```

using System.Threading.Tasks;
using System.IdentityModel.Tokens.Jwt;
using System.Security.Claims;
using Microsoft.IdentityModel.Tokens;
using System.Text;
using Microsoft.Extensions.Configuration;
using System.Collections.Generic;
using System;

namespace CinemaBooking.API.Controllers
{
    [Route("api/[controller]")]
    [ApiController]
    public class AccountController : ControllerBase
    {
        private readonly UserManager<IdentityUser> _userManager;
        private readonly RoleManager<IdentityRole> _roleManager;
        private readonly IConfiguration _configuration;

        public AccountController(UserManager<IdentityUser> userManager,
            RoleManager<IdentityRole> roleManager, IConfiguration configuration)
        {
            _userManager = userManager;
            _roleManager = roleManager;
            _configuration = configuration;
        }

        [HttpPost("register")]
        public async Task<IActionResult> Register(RegisterDto registerDto)
        {
            if (!ModelState.IsValid)
            {
                return BadRequest(ModelState);
            }
            var user = new IdentityUser
            {
                Username = registerDto.Username,
                Email = registerDto.Email,
                EmailConfirmed = true
            };
            var result = await _userManager.CreateAsync(user, registerDto.Password);

            if (result.Succeeded)
            {
                if (!await _roleManager.RoleExistsAsync("User"))
                    await _roleManager.CreateAsync(new IdentityRole("User"));
                if (!await _roleManager.RoleExistsAsync("Admin"))
                    await _roleManager.CreateAsync(new IdentityRole("Admin"));

                await _userManager.AddToRoleAsync(user, "User");

                return StatusCode(StatusCodes.Status201Created, new { message = "User
registered successfully." });
            }
            else
            {
                foreach (var error in result.Errors)
                {
                    ModelState.AddModelError(string.Empty, error.Description);
                }
                return BadRequest(ModelState);
            }
        }

        [HttpPost("login")]
    }
}

```

		Круковський Д.С.			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				33
Змн.	Арк.	№ докум.	Підпис	Дата		

```

public async Task<IActionResult> Login(LoginDto loginDto)
{
    if (!ModelState.IsValid)
    {
        return BadRequest(ModelState);
    }

    var user = await _userManager.FindByNameAsync(loginDto.Username);

    if (user != null && await _userManager.CheckPasswordAsync(user,
loginDto.Password))
    {
        var authClaims = new List<Claim>
        {
            new Claim(ClaimTypes.Name, user.UserName!),
            new Claim(JwtRegisteredClaimNames.Jti, Guid.NewGuid().ToString()),
            new Claim(ClaimTypes.NameIdentifier, user.Id)
        };

        var userRoles = await _userManager.GetRolesAsync(user);
        foreach (var userRole in userRoles)
        {
            authClaims.Add(new Claim(ClaimTypes.Role, userRole));
        }

        var secret = _configuration["JwtSettings:Secret"];
        if (string.IsNullOrEmpty(secret))
        {
            throw new InvalidOperationException("JWT Secret key is not configured
in appsettings.json");
        }
        var authSigningKey = new
SymmetricSecurityKey(Encoding.UTF8.GetBytes(secret));

        var token = new JwtSecurityToken(
            issuer: _configuration["JwtSettings:Issuer"],
            audience: _configuration["JwtSettings:Audience"],
            expires: DateTime.Now.AddHours(3),
            claims: authClaims,
            signingCredentials: new SigningCredentials(authSigningKey,
SecurityAlgorithms.HmacSha256)
        );

        return Ok(new
        {
            token = new JwtSecurityTokenHandler().WriteToken(token),
            expiration = token.ValidTo
        });
    }

    return Unauthorized(new { message = "Invalid username or password" });
}
}

```

## Захист API Контролерів:

**Оновлений** CinemaBooking.API/Controllers/MoviesController.cs:

```

using CinemaBooking.API.DTOS;
using CinemaBooking.Data.Infrastructure;
using CinemaBooking.Data.Models;
using Microsoft.AspNetCore.Http;

```

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				34
Змн.	Арк.	№ докум.	Підпис	Дата		

```

using Microsoft.AspNetCore.Mvc;
using Microsoft.EntityFrameworkCore;
using Microsoft.AspNetCore.Authorization;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;

namespace CinemaBooking.API.Controllers
{
    [Route("api/[controller]")]
    [ApiController]
    public class MoviesController : ControllerBase
    {
        private readonly ICinemaRepository _repository;

        public MoviesController(ICinemaRepository repository)
        {
            _repository = repository;
        }

        [HttpGet]
        public async Task<ActionResult<IEnumerable<MovieDto>>> GetMovies()
        {
            var movies = await _repository.Movies
                .Select(m => new MovieDto
                {
                    MovieID = m.MovieID,
                    Title = m.Title,
                    Description = m.Description,
                    Genre = m.Genre
                })
                .ToListAsync();

            return Ok(movies);
        }

        [HttpGet("{id}")]
        public async Task<ActionResult<MovieDto>> GetMovie(long id)
        {
            var movie = await _repository.Movies
                .Select(m => new MovieDto
                {
                    MovieID = m.MovieID,
                    Title = m.Title,
                    Description = m.Description,
                    Genre = m.Genre
                })
                .FirstOrDefaultAsync(m => m.MovieID == id);

            if (movie == null)
            {
                return NotFound();
            }

            return Ok(movie);
        }

        [HttpPost]
        [Authorize(Roles = "Admin")]
        public ActionResult<MovieDto> CreateMovie(CreateUpdateMovieDto createMovieDto)
        {
            if (!ModelState.IsValid)
            {
                return BadRequest(ModelState);
            }
        }
    }
}

```

		Круковський Д.С.			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				35
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        var movie = new Movie
        {
            Title = createMovieDto.Title,
            Description = createMovieDto.Description,
            Genre = createMovieDto.Genre
        };

        _repository.SaveMovie(movie);

        var movieDto = new MovieDto
        {
            MovieID = movie.MovieID,
            Title = movie.Title,
            Description = movie.Description,
            Genre = movie.Genre
        };

        return CreatedAtAction(nameof(GetMovie), new { id = movie.MovieID },
movieDto);
    }

    [HttpPut("{id}")]
    [Authorize(Roles = "Admin")]
    public async Task<IActionResult> UpdateMovie(long id, CreateUpdateMovieDto
updateMovieDto)
    {
        if (!ModelState.IsValid)
        {
            return BadRequest(ModelState);
        }

        var movieToUpdate = await _repository.Movies
                                                .FirstOrDefaultAsync(m => m.MovieID ==
id);

        if (movieToUpdate == null)
        {
            return NotFound();
        }

        movieToUpdate.Title = updateMovieDto.Title;
        movieToUpdate.Description = updateMovieDto.Description;
        movieToUpdate.Genre = updateMovieDto.Genre;

        _repository.SaveMovie(movieToUpdate);

        return NoContent();
    }

    [HttpDelete("{id}")]
    [Authorize(Roles = "Admin")]
    public async Task<IActionResult> DeleteMovie(long id)
    {
        var movieToDelete = await _repository.Movies
                                                .FirstOrDefaultAsync(m => m.MovieID ==
id);

        if (movieToDelete == null)
        {
            return NotFound();
        }

        _repository.DeleteMovie(movieToDelete);

        return NoContent();
    }

```

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				36
Змн.	Арк.	№ докум.	Підпис	Дата		

```

    }
}

```

## Оновлений CinemaBooking.API/Controllers/SessionsController.cs:

```

using CinemaBooking.API.DTOs;
using CinemaBooking.Data.Infrastructure;
using CinemaBooking.Data.Models;
using Microsoft.AspNetCore.Http;
using Microsoft.AspNetCore.Mvc;
using Microsoft.EntityFrameworkCore;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using Microsoft.AspNetCore.Authorization;

namespace CinemaBooking.API.Controllers
{
    [Route("api/[controller]")]
    [ApiController]
    public class SessionsController : ControllerBase
    {
        private readonly ICinemaRepository _repository;
        private readonly CinemaDbContext _context;

        public SessionsController(ICinemaRepository repository, CinemaDbContext context)
        {
            _repository = repository;
            _context = context;
        }

        [HttpGet]
        public async Task<ActionResult<IEnumerable<MovieSessionDto>>> GetSessions()
        {
            var sessions = await _repository.MovieSessions
                .OrderBy(s => s.SessionTime)
                .Select(s => new MovieSessionDto
                {
                    MovieSessionID = s.MovieSessionID,
                    Hall = s.Hall,
                    Price = s.Price,
                    SessionTime = s.SessionTime,
                    MovieID = s.MovieID,
                    MovieTitle = s.Movie != null ? s.Movie.Title : "N/A",
                    MovieGenre = s.Movie != null ? s.Movie.Genre : "N/A"
                })
                .ToListAsync();

            return Ok(sessions);
        }

        [HttpGet("{id}")]
        public async Task<ActionResult<MovieSessionDto>> GetSession(long id)
        {
            var session = await _repository.MovieSessions
                .Where(s => s.MovieSessionID == id)
                .Select(s => new MovieSessionDto
                {
                    MovieSessionID = s.MovieSessionID,
                    Hall = s.Hall,
                    Price = s.Price,
                    SessionTime = s.SessionTime,
                    MovieID = s.MovieID,

```

		Круковський Д.С.			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				37
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        MovieTitle = s.Movie != null ? s.Movie.Title : "N/A",
        MovieGenre = s.Movie != null ? s.Movie.Genre : "N/A"
    })
    .FirstOrDefaultAsync();

    if (session == null)
    {
        return NotFound($"Сеанс з ID {id} не знайдено.");
    }

    return Ok(session);
}

[HttpPost]
[Authorize(Roles = "Admin")]
public async Task<ActionResult<MovieSessionDto>>
CreateSession(CreateUpdateSessionDto createSessionDto)
{
    if (!ModelState.IsValid)
    {
        return BadRequest(ModelState);
    }

    var movieExists = await _repository.Movies.AnyAsync(m => m.MovieID ==
createSessionDto.MovieID);
    if (!movieExists)
    {
        ModelState.AddModelError(nameof(createSessionDto.MovieID), $"Фільм з ID
{createSessionDto.MovieID} не знайдено.");
        return BadRequest(ModelState);
    }

    var session = new MovieSession
    {
        Hall = createSessionDto.Hall,
        Price = createSessionDto.Price,
        SessionTime = createSessionDto.SessionTime,
        MovieID = createSessionDto.MovieID
    };

    _context.MovieSessions.Add(session);
    _repository.SaveSession(session);

    await _context.Entry(session).Reference(s => s.Movie).LoadAsync();

    var sessionDto = new MovieSessionDto
    {
        MovieSessionID = session.MovieSessionID,
        Hall = session.Hall,
        Price = session.Price,
        SessionTime = session.SessionTime,
        MovieID = session.MovieID,
        MovieTitle = session.Movie != null ? session.Movie.Title : "N/A",
        MovieGenre = session.Movie != null ? session.Movie.Genre : "N/A"
    };

    return CreatedAtAction(nameof(GetSession), new { id = session.MovieSessionID
}, sessionDto);
}

[HttpPut("{id}")]
[Authorize(Roles = "Admin")]
public async Task<IActionResult> UpdateSession(long id, CreateUpdateSessionDto
updateSessionDto)

```

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				38
Змн.	Арк.	№ докум.	Підпис	Дата		

```

{
    if (!ModelState.IsValid)
    {
        return BadRequest(ModelState);
    }

    var movieExists = await _repository.Movies.AnyAsync(m => m.MovieID ==
updateSessionDto.MovieID);
    if (!movieExists)
    {
        ModelState.AddModelError(nameof(updateSessionDto.MovieID), $"Фільм з ID
{updateSessionDto.MovieID} не знайдено.");
        return BadRequest(ModelState);
    }

    var sessionExists = await _context.MovieSessions
        .AsNoTracking()
        .AnyAsync(s => s.MovieSessionID == id);

    if (!sessionExists)
    {
        return NotFound($"Сеанс з ID {id} не знайдено.");
    }

    var updatedSession = new MovieSession
    {
        MovieSessionID = id,
        Hall = updateSessionDto.Hall,
        Price = updateSessionDto.Price,
        SessionTime = updateSessionDto.SessionTime,
        MovieID = updateSessionDto.MovieID
    };

    _context.MovieSessions.Update(updatedSession);
    _repository.SaveSession(updatedSession);

    return NoContent();
}

[HttpDelete("{id}")]
[Authorize(Roles = "Admin")]
public async Task<IActionResult> DeleteSession(long id)
{
    var sessionToDelete = await _context.MovieSessions
        .FirstOrDefaultAsync(s => s.MovieSessionID ==
id);

    if (sessionToDelete == null)
    {
        return NotFound($"Сеанс з ID {id} не знайдено.");
    }

    _repository.DeleteSession(sessionToDelete);

    return NoContent();
}
}
}

```

		Круковський Д.С.			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				39
Змн.	Арк.	№ докум.	Підпис	Дата		

Movies

GET /api/Movies

POST /api/Movies

Parameters

No parameters

Request body

application/json

```
{  "title": "Не виникай світло",  "description": "Помічниця власника складу текстильних виробів Естер помічає силует жінки при виникненні світла.",  "genre": "Жахи"}  
```

Execute

Clear

Рис.14 Результат виконання

Request URL

https://localhost:7102/api/Movies

Server response

Code

Details

201

Undocumented

Response body

```
{  "movieID": 7,  "title": "Не виникай світло",  "description": "Помічниця власника складу текстильних виробів Естер помічає силует жінки при виникненні світла.",  "genre": "Жахи"}  
```

Response headers

```
access-control-allow-origin: *  content-type: application/json; charset=utf-8  date: Thu,30 Oct 2025 15:37:20 GMT  location: https://localhost:7102/api/Movies/7  server: Kestrel  
```

Рис.15 результат виконання

Responses

Curl

```
curl -X 'POST' \  'https://localhost:7102/api/Movies' \  -H 'accept: text/plain' \  -H 'Content-Type: application/json' \  -d '{  "title": "Закляття",  "description": "У 1971 році Роджер і Керолайн переїжджають у напівазруйнований будинок у Харрісвіллі, Род-Айленд, зі своїми п'ятьма доньками.",  "genre": "Жахи"}  '  
```

Request URL

https://localhost:7102/api/Movies

Server response

Code

Details

401

Undocumented

Error: response status is 401

Response headers

```
access-control-allow-origin: *  content-length: 0  date: Thu,30 Oct 2025 15:42:13 GMT  server: Kestrel  www-authenticate: Bearer  
```

Рис.16 результат виконання



Висновок: у ході лабораторної роботи ми набули навичок роботи з механізмами аутентифікації та авторизації користувачів в ASP.NET, набули навичок роботи з Web API.

		Круковський Д.С			ДУ «Житомирська політехніка».24.121.01.000 – Лр1	Арк.
		Українець М.О.				41
Змн.	Арк.	№ докум.	Підпис	Дата		