

# ANDRIAN KOBRYN

.NET Developer

+380932347803  
Lviv, UA

andrikkobryn@gmail.com  
<https://github.com/TheEnemy25>

## CONTACT ME

### Gmail

andrikkobryn@gmail.com

### LinkedIn

[https://www.linkedin.com/in/andrian-kobryn/?locale=xx\\_XX](https://www.linkedin.com/in/andrian-kobryn/?locale=xx_XX)

### Telegram

[https://t.me/ENEMY\\_25](https://t.me/ENEMY_25)

### GitHub

<https://github.com/TheEnemy25>

## SUMMARY

Entry-level back-end developer with practical knowledge of C#, ASP.NET Core, EF Core, MS SQL, and foundational front-end skills in HTML, CSS, React, Next.js, and Tailwind. Passionate about building efficient and scalable solutions that meet modern development standards. Committed to continuous professional growth and eager to apply my knowledge to solve real-world challenges as part of a team of professionals.

## ELEKS UNIVERSITY .NET INTERNSHIP

### Seat Reservation System:

Developed a service for reserving seats on transportation with the ability to change routes during the journey. This system allowed passengers to reserve seats on buses and other types of transportation with the option to modify their route during the trip. My role included designing and developing the seat reservation functionality and interacting with the API, using AutoMapper for object mapping, FluentValidation for input validation, and Entity Framework for database management. All requests and checks were conducted through Swagger, allowing effective testing and interaction with the API without frontend development. The project was built using a microservices architecture, ensuring scalability and flexibility.

<https://github.com/TheEnemy25/BookingService>

### Error Handling in the ReservationSystem:

During testing of the reservation functionality, I discovered that the system did not handle cases where a user attempted to book an already occupied seat. I implemented an additional check at the business logic level, which included verifying seat availability before confirming a reservation, and introduced detailed error handling with logging for easier debugging.

### RoutingSystem:

Developed a service for optimizing routes and planning trips with transfers. This service assisted passengers in finding the most optimal routes with possible transfers and planning their journeys. My role involved developing routing algorithms that took into account various factors such as travel time, transfers, and traffic conditions, as well as integrating the algorithms with other system components using Swagger for testing. To create flexible and efficient data handling, I used AutoMapper and FluentValidation, and for data management, I used Entity Framework.

<https://github.com/TheEnemy25/RouteService>

### Issues with Docker:

This was my first experience using Docker for project containerization. While setting up the containers, I encountered difficulties with configuring the environment and correctly integrating dependencies between microservices. I faced challenges in configuring the Dockerfile and docker-compose.yml.



## SKILLS

C# · ASP.NET Core · EF Core · MS SQL ·

HTML · CSS · React · NextJs · Tailwind

## LANGUAGES

Ukrainian

Native

English

Intermediate

## HOBBIES

Computer games, football, basketball, chess, listening to music

## EDUCATION

Computer Science Bachelor

Lviv Polytechnic National University

09/2019 - 06/2023

Lviv, Ukraine

Computer Science Master

Lviv Polytechnic National University

09/2023 - 12/2024

Lviv, Ukraine

## EXPERIENCE

### Cinema project (Myself project)

01/2024 - 12/2024

- Currently developing a platform for managing cinematic events, focusing on a robust back-end architecture built with ASP.NET Core Web API.
- **Back-end:** Implemented authentication and authorization using IdentityServer4, datamanagement with Entity Framework Core, and MS SQL as the database. Utilized AutoMapper for object mapping, FluentValidation for input validation, and MediatR to implement the CQRS pattern.
- **Front-end:** Initially developed the user interface using React and SCSS. Recently transitioned the project to Next.js to improve performance and enable server-side rendering.
- **Encountered issues:** with configuring IdentityServer4, particularly with the multiple startup setup during back-end initialization. Initially, everything was running on a single URL in Swagger (<https://localhost:5445> for the API), while IdentityServer4 was hosted on a different URL (<https://localhost:5443>). The problem was that Swagger only responds to one domain. So, if the domain was <https://localhost:5445> and I tried to perform actions on <https://localhost:5443>, an error occurred, even though everything worked fine in Postman. The solution was straightforward: creating a configuration that would open two tabs (<https://localhost:5445> and <https://localhost:5443>) during the multiple startup process. However, at first, due to unfamiliarity, it was not clear what was happening.

🔗 Server: <https://github.com/TheEnemy25/Cinema>  
Client: <https://github.com/TheEnemy25/k-spot>

### Device shop project (Project for bachelor thesis) 05/2023 - 06/2023

- Computer Equipment E-commerce Project  
Developed an e-commerce platform for purchasing computer hardware, focusing on a user-friendly and efficient shopping experience.
- Technologies Used: Implemented with ASP.NET Core MVC for the back-end, Entity Framework for database management, and MS SQL as the database. Created a responsive and interactive front-end using HTML, SCSS, and Gulp for task automation and efficient build processes.
- Challenges: As this project was my first experience with MVC projects and my first time creating an admin panel, I encountered some difficulties. One specific issue arose when a user who had zero items in their wishlist would see items that were added by other users, leading to duplication of wishlist data across all users. To resolve this problem, I adapted the logic for interacting with the database and set up proper data segregation between users, ensuring the uniqueness and accuracy of each user's wishlist.

🔗 <https://github.com/TheEnemy25/ProductsShop>