

# Mohamed Ali

## Back-End .NET developer

•Add.: Cairo, Egypt • Mob.: (+20) 01098638861 • Email: [moohamedali571@gmail.com](mailto:moohamedali571@gmail.com)  
•LI: <https://www.linkedin.com/in/moohamedali571/>•GH: <https://github.com/Mohamed-Ali-dev>  
Marital status: Single • Military status: Completed

## Summary

Motivated Backend Developer with foundational knowledge in designing, building, and maintaining backend systems and APIs. Eager to apply my technical skills and passion for backend development to create efficient and scalable solutions that add value to my team and projects.

---

## Technical Skills

-Programming Languages: C#                      -Database Management: SQL  
-Version Control: Git, GitHub                      -Frontend Knowledge: HTML, CSS (Basics)  
-Frameworks & Tools: ASP.NET Core, Entity Framework Core, LINQ, RESTful API, MVC

---

## Soft Skills

-Team Collaboration                      -multi-tasking                      -Critical Thinking & Problem-Solving  
-Communication Skills                      -Self-motivation

---

## Education

### BaniSuef University | 2022

Bachelor's degree in computer science

---

## Projects

### Fast Food E-Commerce

[GitHub](#) | [Live Demo](#)

- Developed a responsive e-commerce platform using ASP.NET MVC, incorporating Stripe payment gateway and third-party authentication (Facebook, Microsoft).
- Implemented role-based access control (User, Manager, Admin) and leveraged the Unit of Work repository pattern for efficient data handling.
- Features: Item catalog, shopping cart, and comprehensive order management with secure authentication.

### Book Shop E-Commerce

[GitHub](#) | [Live Demo](#)

- Designed an e-commerce solution using ASP.NET MVC with similar role-based access and payment integration as the Fast Food project.
- Focused on scalability, secure data handling, and delivering an intuitive shopping experience.

### Pokemon Review

[GitHub](#) | [Live Demo](#)

- Built a review API with CRUD operations, authentication (JWT), and authorization features.
- Employed Unit of Work repository pattern for optimized data interactions and refresh token support.

### Movies Api

[GitHub](#) | [Live Demo](#)

- Developed a .NET API for managing a movie database, allowing users to perform CRUD operations on movies and reviews.
- Key Features: JWT authentication, role-based access, logging, refresh token support, sorting, pagination, filtering, and the Unit of Work repository pattern.

### eCommerce App

[GitHub](#) | [Live Demo](#)

- Developed a scalable API following Clean Architecture principles, incorporating a generic repository pattern for efficient and reusable data management.
  - Features include robust exception handling, JWT authentication, Stripe payment integration, and role-based access control.
-