Emrullah Yildirim

Email | GitHub | LinkedIn

Gebze, Kocaeli, Turkey

EDUCATION

Marmara University | *GPA: 85.2 out of 100* *Jun 2025*
Associate Degree | *Computer Programming*

42 Istanbul *Dec 2023 – Present*
Software Development Training | *Core Curriculum (Project-Based)*

SKILLS

Focus Areas: Backend Development | Frontend Development | Full-Stack Web Applications | Systems Programming (C/C++) | DevOps Fundamentals
Programming Languages: JavaScript | TypeScript | C | C++ | Node.js
Libraries & Frameworks: React.js | Next.js | Express.js | Electron.js
Databases: PostgreSQL | MySQL | MongoDB
Tools: Git | Docker | Postman

EXPERIENCE

Part-time IT Staff | *Marmara University* *Nov 2024 - Jun 2025*
Providing essential hardware and software support to university departments.

- Provided technical user support by managing computer hardware and software across university units.
- Managed the installation and configuration of printers, scanners, and other peripherals within network and system components.

Software Development Intern | *Marmara University* *Jul 2025 - Aug 2025*
Contributed to an application generating educational content via AI integration.

- Designed and integrated API services to manage communication between the mobile interface (Frontend) and the question generation service (Backend).
- Contributed to the development of backend logic for analyzing and interpreting PDF/Text content, which served as the core functionality of the application.
- Followed and contributed to Git version control standards for project versioning and team collaboration.

PROJECTS

Command Line Interface (Shell Application) | *GitHub* *Nov 2024*
An advanced command line application developed in C.

- Successfully completed the project in a two-person team, utilizing Git version control effectively to manage parallel development.
- Developed a parser to transform user inputs into an Abstract Syntax Tree (AST).
- Designed a modular and easily extensible architecture for managing Built-in Commands.
- Maximized application performance and efficiency by using core system calls for system resource and memory management, rather than relying on high-level standard libraries.