

Mustapha Jlem software engineer

jlemmustapha@gmail.com ❖ +212673764374 ❖ Rabat, Morocco ❖ [GitHub link](#)

EDUCATION

1337 Coding School (42 Network)

Morocco

- Immersive Learning Approach Emphasis on peer-to-peer learning, practical projects, and self-directed, hands-on training.
- Specialized in problem-solving, critical thinking, and teamwork, with extensive experience in real-world software development scenarios.

PROJECTS & EXPERIENCE

Full-Stack Web App

- Developed a real-time multiplayer Pong game platform using NestJS for backend operations and Next.js for frontend development
- Integrating PostgreSQL for robust data management and implementing comprehensive security measures, including OAuth for user authentication.
- Implemented real-time communication features using WebSocket to enable live chat functionalities, significantly enhancing user interaction and engagement on the platform.

Custom Web Server

- Developed a custom HTTP server in C++, ensuring compatibility with web standards and high-performance non-blocking I/O operations.
- Configured dynamic server behavior similar to NGINX, supporting multiple HTTP methods, file uploads, and static content serving.
- Ensured robustness through comprehensive stress testing and implemented advanced features including cookies, session management, and multiple CGI support.

Docker Infrastructure

- Successfully completed a Docker-based system administration project, establishing a robust virtualized infrastructure.
- Developed and managed Docker-based infrastructure using custom Dockerfiles and Docker Compose, orchestrating essential services such as NGINX, WordPress, and MariaDB.

Minishell

- Developed a custom command-line interface in C, simulating fundamental bash shell functionalities such as command history, environment variable handling, and built-in commands like “echo”, “cd”, “exit”, and more.
- Implemented process management and signal handling features, implementing I/O redirections, pipes, and system calls (e.g., fork, execve) to manage execution flows and command parsing effectively.

3D Game

- Implemented a ray-casting engine in C to simulate a 3D maze navigation system, enhancing user experience through interactive controls using miniLibX.
- Programmed wall detection, texture mapping, and user input handling to support real-time movement within the game environment.

SKILLS

- **Languages:** C++, C, JavaScript, TypeScript Bash.
- **Technologies:** Docker, Linux, Node.js, Express, Nest.js, Next.js, React, HTML, CSS, GIT, GitHub, GNU Make, PostgreSQL, Prisma.