Abdeljalil EL HIDAOUI

Morocco | abdeljalileihidaoui21@gmail.com | +212 606441097

github.com/abel-hid | abel-hid.vercel.app

Professional Summary

Software engineer trained at 1337 (42 Network) at Mohammed VI Polytechnic University (UM6P), a project-based software engineering school. I build and deploy software across the stack, including REST APIs and backend systems in Python, Node.js, and C++, web applications with Flask and Next.js, and containerized deployments with Docker, Kubernetes, and Ansible. Projects include a multi-client IRC server built from raw sockets, a real-time web application, and a multiplayer game platform

Work Experience

Warriors Team *Aug 2025 – Feb 2026*

Full Stack Developer

- Built full-stack web applications at an agency, in a small cross-functional team, using React, Next.js, and Tailwind CSS on the front end and Node.js with Supabase on the back end
- Implemented login with role-based access, an admin panel, and a data dashboard
- Developed APIs in Node.js backed by Supabase and optimized backend performance
- Deployed applications to Vercel and shipped features on a recurring release schedule in agile sprints, with regular code reviews

Technical Skills

Programming Languages

- **Proficient:** C, C++, Python, JavaScript, TypeScript, Java, Bash/Shell Scripting

Technologies & Tools

- **Infrastructure & DevOps:** Docker, Docker Compose, Kubernetes (K3s/K3d), Helm, Argo CD (GitOps), Ansible, Vagrant, Nginx, Vercel, CI/CD, Linux/Unix system administration
- **Databases:** SQL, PostgreSQL, Supabase, Redis, MongoDB
- **Web Technologies:** HTML, CSS, Tailwind CSS, React.js, Next.js, Node.js, Express.js, Django, Flask
- **Network & Systems:** Socket Programming, TCP/IP, HTTP/HTTPS, WebSocket, REST APIs, IPC, Process Management

Education

1337 School (42 Network) – Mohammed VI Polytechnic University *Oct 2022 – Present*

Software Engineering Program

Level 17.21 – Common core completed at 100%, currently in the 42 Advanced track

- Member of the 42 Network, a peer-based coding school with a project-only curriculum – no lectures, no professors; progression is earned through project completion and peer evaluation
- Completed the common core at 100% and validated 14 projects with bonus
- Coursework: algorithms and data structures, C++ object-oriented programming (Modules 00 to 09), Unix system programming (processes, signals, sockets), networking and Linux administration, web development, and web security

Projects

Inception of Things (IoT) - Kubernetes Infrastructure from Scratch

- Provisioned a two-node K3s cluster from scratch with Vagrant, configuring one host in controller mode and one in agent mode with static networking and passwordless SSH
- Deployed multiple applications on K3s using Ingress for host-based routing between services
- Built a lightweight K3d cluster and integrated Argo CD for GitOps-based continuous deployment, then extended it with Helm and a self-hosted GitLab instance for a full CI/CD-to-cluster workflow

cloud-1 - Automated Cloud Deployment

- Automated deployment of a containerized web stack (WordPress, MariaDB, Nginx) to a remote cloud server using Ansible playbooks
- Configured persistent data and automatic service recovery on server reboot, with a configuration repeatable across multiple hosts

Inception - Containerized Infrastructure with Docker Compose

- Built a multi-container infrastructure with Docker Compose: NGINX with SSL/TLS, WordPress with php-fpm, and MariaDB

ft_irc - Internet Relay Chat Server

- Built a multi-client IRC server in C++ from raw sockets, supporting real-time messaging, channel management, and user authentication
- Implemented non-blocking I/O and concurrent client handling, following RFC 1459/2812 for protocol compliance

minishell - Custom Unix Shell Implementation

- Built a fully functional shell from scratch in C, with pipes, redirections, and environment variable handling
- Implemented process management and signal handling for reliable command execution

ft_transcendence - Multiplayer Pong Platform (Team of 4)

- Built a real-time multiplayer Pong platform in a team of four, contributing to both frontend and backend
- Implemented remote play over the network for live matches, plus a bracket-style tournament system with matchmaking
- Built user management (profiles, avatars, friends), OAuth 2.0 login, and live chat tied into the game

Matcha - Full-Stack Dating Application

- Built a Flask backend using raw SQL/psycopg2 (no ORM) paired with a Next.js/TypeScript frontend, covering auth, matching, messaging, and profile management
- Implemented real-time notifications and live online-presence tracking with Flask-SocketIO and WebSockets
- Built the full auth flow – email verification, password reset with token versioning, and JWT auth

darkly - Web Vulnerability Audit

- Audited a deliberately vulnerable web platform, exploiting real-world vulnerabilities to escalate from anonymous visitor to administrator
- Documented how each breach could have been prevented, from both a programming and a design perspective

red-tetris - Multiplayer Tetris Game

- Developed a networked multiplayer Tetris game using an exclusively full-stack JavaScript stack