

Hamza Boutaleb

boutaleb.hm@gmail.com | +212708158971

LinkedIn | GitHub

Summary

Software developer with a strong foundation in frontend, backend, and low-level development. Combines problem-solving and critical thinking with design patterns, SOLID principles, and clean code practices to build maintainable software.

Education

1337

Oct 2024 – Oct 2026

Software Engineering

Core subjects: C, C++, algorithms, Unix, networking, Java, Spring Boot, microservices, Kafka, and Angular.

Zone01 Oujda

Apr 2024 – Apr 2026

Software Development Program

Technologies: Go, JavaScript, Rust, Java, and Spring Boot.

Additional training: Soft Skills Training.

Cadi Ayyad University, Marrakech

2019 – 2024

Bachelor's Degree in Physics — Electronics and Automation

Core subjects: Electronic circuits, signals, automated systems, mathematics, and algorithms.

Programming: C, C++, and Arduino.

ALX

Feb 2023 – Mar 2024

Software Engineering Program — Backend Specialization

Core subjects: C, Python, JavaScript, data structures and algorithms, Linux, and SQL.

Backend: Node.js, TypeScript, MySQL, NoSQL, Redis, authentication, caching, and unit and integration testing.

Experience

Software Developer Intern

Aug 2026 – Sep 2026

HPS

- Contributed to feature updates and bug fixes in PowerCARD Switch using Java and Spring Boot microservices.
- Developed a C HTTP service using libevhttp and json-c to receive requests from POS terminals and return JSON responses.
- Configured Oracle database connectivity for the C service.

Web Developer Intern

Jul 2022 – Oct 2022

ICF Communication — Digital Marketing Agency

- Developed backend features using PHP and Laravel.
- Built frontend pages using HTML and CSS.

Technical Skills

Languages: Java, JavaScript, Go, Rust, C, C++, Python, PHP, TypeScript, SQL.

Frontend: Angular, React, HTML, CSS, TanStack Query.

Backend: Spring Boot, Spring Security, Django, Flask, Express.js, Laravel, Node.js.

Databases: PostgreSQL, MySQL, SQLite, Redis.

Systems and Infrastructure: Linux, Docker, Docker Compose, Kafka, Nginx.

Engineering: Data structures and algorithms, design patterns, SOLID principles, REST APIs, microservices, WebSockets, Unix systems programming.

Projects

7irafi — Local Services Marketplace

Java, Spring Boot, Angular, Kafka, PostgreSQL, Docker Compose

- Co-developed a marketplace connecting clients with local professionals, contributing across the Angular frontend and Spring Boot microservices.
- Implemented service discovery with Eureka, request routing through Spring Cloud Gateway, and JWT authentication.
- Used a separate PostgreSQL database for each service and Kafka for asynchronous events, including notifications.
- Built real-time chat using WebSockets and containerized the application with Docker Compose.

01Blog — Social Blogging Platform

Java, Spring Boot, Spring Security, Angular, PostgreSQL

- Developed a blogging platform with multimedia posts, user subscriptions, likes, and comments.
- Implemented secure authentication and role-based access for users and administrators.
- Built personalized feeds, subscription notifications, and moderation tools for managing users, posts, and reports.

Social Network — Fullstack Social Platform

Go, SQLite, WebSockets, Docker

- Developed social features including user profiles, follow requests, multimedia posts, groups, and events.
- Implemented session and cookie authentication, with access controls for private profiles, posts, and group content.
- Built private and group messaging using WebSockets, alongside activity notifications.
- Managed SQLite schema migrations and containerized the frontend and backend separately.

Real-Time Forum — Discussion and Messaging Platform

Go, JavaScript, HTML, CSS, SQLite, WebSockets, Docker

- Built a single-page forum with categorized posts, comments, likes/dislikes, and post filtering.
- Implemented registration, login, and cookie-based session management.
- Added real-time private messaging and online/offline presence through WebSockets.
- Implemented paginated chat history with debounced scroll handling to limit repeated requests.

AirBnB Clone — Property Listing Platform

Python, Flask, SQLAlchemy, MySQL, JavaScript, jQuery, Nginx, Gunicorn

- Built a fullstack application for browsing and filtering property listings.
- Developed a management console and data models supporting JSON file storage and MySQL persistence through SQLAlchemy.
- Implemented REST API endpoints and integrated them with a dynamic JavaScript/jQuery frontend.
- Deployed the application using Nginx and Gunicorn.

DopShop — Moroccan Products Marketplace

React, Django, TanStack Query, Axios, React Router

- Built an e-commerce platform for Moroccan products with product search, a shopping cart, and checkout.
- Implemented user accounts and an admin dashboard for managing products.
- Integrated the React frontend with the Django backend using Axios and TanStack Query for API requests, caching, and server-state management.

Bomberman — Real-Time Multiplayer Game

JavaScript, Express.js, Custom Frontend Framework, WebSockets

- Built a browser-based Bomberman game for 2–4 players using a custom JavaScript framework and DOM rendering.
- Implemented multiplayer synchronization and in-game chat through WebSockets and an Express.js backend.
- Developed bomb explosions, destructible blocks, randomized power-ups, and player elimination.
- Maintained a stable 60 FPS using `requestAnimationFrame` for the game loop and animations.

Lem-in — Graph Pathfinding and Ant Simulation

Go, Breadth-First Search, Graph Algorithms

- Built a command-line simulation that routes ants through a graph of rooms and tunnels.
- Adapted breadth-first search to identify routes and schedule ant movements to reduce the total number of turns.
- Enforced room occupancy and tunnel capacity constraints, with validation for malformed input and unreachable destinations.

Unix Shells — C & Rust

C, Rust, Unix System Programming

- Built Unix shells in C and Rust, both supporting command parsing, pipelines, I/O redirection, environment variable expansion, and built-in commands.
- Implemented process execution, signal handling, command history, and exit-status management.
- Developed file and directory utilities directly in Rust, including `ls`, `cat`, `cp`, `mv`, and `rm`.

LocalServer — HTTP/1.1 Server in Rust

Rust, Socket Programming, Non-blocking I/O, CGI

- Built an HTTP/1.1 server from scratch with an event-driven, single-threaded architecture and non-blocking I/O.
- Implemented request parsing, chunked transfer handling, GET/POST/DELETE methods, file uploads, cookies, and sessions.
- Added configurable routes, multiple listening ports, redirects, custom error pages, and CGI execution.

Born2beroot — Linux Server Administration

Debian, Bash, LVM, SSH, UFW, AppArmor, Cron

- Configured a Debian virtual machine with encrypted LVM partitions, SSH access controls, UFW firewall rules, and AppArmor.
- Enforced password policies, user permissions, and sudo logging.
- Automated system monitoring using Bash and cron.
- Deployed WordPress using Lighttpd, MariaDB, and PHP, with an expanded LVM partition layout.

miniRT — 3D Ray Tracer

C, MiniLibX, Ray Tracing, Vector Mathematics

- Built a ray tracer that renders 3D scenes from configurable scene files.
- Implemented ray-object intersections for spheres, planes, and cylinders, with ambient and diffuse lighting and hard shadows.
- Added camera configuration, object transformations, scene validation, and memory management.

ft_irc — IRC Server

C++98, TCP/IP, epoll, Non-blocking I/O

- Built an IRC server handling multiple simultaneous clients through non-blocking sockets and an `epoll` event loop.
- Implemented authentication, nickname registration, private messaging, and channel communication.
- Added channel operator commands and controls for invitations, passwords, user limits, and permissions.

Custom Language Compiler

Rust, x86-64 Assembly, Compiler Design

- Built a compiler in Rust that translates a custom programming language into x86-64 assembly.
- Implemented lexical analysis, parsing, type checking, and assembly code generation.
- Supported variables, typed expressions, arithmetic and logical operations, control flow, and functions with parameters and return values.

Hack Computer — Computer System from First Principles

HDL, Computer Architecture, Assembly, Compiler Design

- Built the Hack computer from logic gates, implementing an ALU, registers, memory, and CPU.
- Developed an assembler, virtual machine translator, and compiler for the Jack programming language.
- Implemented the Jack OS library for memory management, input/output, graphics, and basic system services.

Certifications

- **Nand to Tetris — Parts I & II** — Hebrew University of Jerusalem.
- **Learn to Code with Rust Specialization** — Packt Aug 2026
- **IBM Java Developer Specialization** — IBM, SkillUp Apr 2026
- **Advanced TypeScript with OOP** — Edureka Nov 2025
- **Data Structures and Algorithms Specialization** — University of California San Diego.
- **Back End Development and APIs** — freeCodeCamp Nov 2023

Leadership & Achievements

- **Founder, Competitive Programming Club — Zone01 Oujda:** Founded the club and organized multiple programming contests.
- **2nd Place, AUSIM Cyberdrill — SecDojo, 2026:** Earned second place as part of a team in a cybersecurity competition.

Languages

Arabic: Native | **English:** Professional working proficiency | **French:** Intermediate