

# Youssef Bouryal

---

Software Developer | Finance Background

## Contact

---

- **Email:** [josephbouryal@gmail.com](mailto:josephbouryal@gmail.com)
- **Phone:** +212 643 336 201
- **Location:** Rabat, Morocco
- [GitHub](#) | [LinkedIn](#) | [Twitter](#)

## Professional Summary

---

Software developer with 2+ years of programming experience and a bachelor's degree in Applied Finance. Currently advancing technical skills at 1337 coding school (42 Network), focusing on low-level systems programming and web development. Combining analytical finance background with technical problem-solving abilities to create efficient, scalable software solutions.

## Technical Skills

---

- **Languages:** C, Python, JavaScript, Bash
- **Web Development:** HTML, CSS, Django, REST APIs
- **Database:** PostgreSQL, SQL fundamentals
- **DevOps:** Docker, Git/GitHub
- **Computer Science:** Data Structures, Algorithms, UNIX Systems

## Education

---

**1337 Coding School (42 Network)** | Rabat, Morocco *Computer Science Intensive Program* |  
September 2024 - Present

- Peer-to-peer learning model focusing on project-based mastery of programming concepts
- Selected through rigorous "Piscine" admission process with <15% acceptance rate
- Developing strong problem-solving skills through algorithm challenges and complex system design

**Cadi Ayyad University** | Morocco *Bachelor's Degree in Applied Finance* | 2021 - 2024

- Developed strong analytical and quantitative skills applicable to data-driven software development
- Relevant coursework: Statistics, Econometrics, Data Analysis

## Projects

---

### Sportify | [GitHub](#)

- Developed a full-stack sports management application during school hackathon to coordinate university athletics
- Implemented backend REST API with Django, enabling competition management and real-time leaderboards
- Designed database schema and integration with Sqlite3 using Django ORM
- Containerized application with Docker for consistent deployment across environments

### Minishell | [GitHub](#)

- Built a POSIX-compliant shell in C mimicking core bash functionality
- Implemented parser using recursive descent techniques to create abstract syntax trees
- Developed execution logic for commands, pipes, logical operators, and I/O redirections
- Applied low-level programming concepts including process management, signal handling, and memory management

### LibFT | [GitHub](#)

- Created comprehensive C library with over 40 utility functions replicating standard C library functionality
- Implemented memory management, string manipulation, linked list handling, and I/O functions
- Optimized for performance and memory usage while maintaining readability

## Certifications

---

- [Programming in Python](#) - Coursera
- [Introduction to Backend Development](#) - Coursera

## Languages

---

- **English:** Professional working proficiency
- **Arabic:** Native