Omar Ibrahim

Ottawa, Ontario, Canada
343-297-3668 omarmgmi08@gmail.com linkedin.com/in/omar-ibrahim github.com/omaribrahim6

Education

Carleton University **Ottawa, ON**
Bachelor of Engineering in Software Engineering *Expected 2028*

Technical Skills

Languages: Python, Java, C++, C, Typescript, SQL, Bash — English, Arabic
Developer Tools: Git, Docker, PyCharm, VS Code, Kali Linux, WSL
Technologies/Frameworks: Linux, Express, JUnit, PlantUML, Nginx, Cloudflare, Tailscale, Redis, Proxmox, WireGaurd, PostgreSQL, Next.js, Tailwind CSS
Practices: CI/CD, Agile Methodology, Secure Coding, Object Oriented Programming, API Design, Unit Testing, UML Modeling

Experience

Cybersecurity Analyst **Oct 2025 – Dec 2025**
Journale AI *Remote*

- Conducted penetration testing across **3 core attack surfaces** (authentication, payments, APIs), identifying authorization and data exposure vulnerabilities.
- Reproduced exploits and collaborated with engineers to deploy secure remediations under responsible disclosure.
- Assessed access control failures including IDOR, privilege escalation paths, session handling weaknesses, and advised on secure deployment practices.

Freelance Web Developer **Sep 2025 – Present**
Self-Employed *Ottawa, Ontario*

- Designed, deployed, and maintained **containerized** production websites using **Next.js** for **multiple clients**, handling full-stack development and infrastructure ownership.
- Integrated and consumed **third-party APIs** (REST-based) for forms, email delivery, analytics, and external data services.
- Managed **Linux VPS** environments including domain configuration, SSL, firewall hardening, backups, and uptime monitoring.
- Implemented **CI/CD pipelines** to automate deployments and reduce manual release overhead.

Projects

RoadSense – AI Road Damage Intelligence | *Python, YOLOv8, OpenCV, Supabase, Next.js, TypeScript, MapLibre* **February 2026**

- Built an **end-to-end** road damage detection platform that processes dashcam video to identify potholes, cracks, ruts, and debris using a **fine-tuned YOLOv8 model**.
- Developed a backend pipeline for frame extraction, inference, severity scoring, GPS resolution, and deduplication to produce clean, geotagged detection records.
- Designed and integrated **Supabase** storage/database workflows to upload detection images and persist structured metadata (confidence, severity, bbox, status, source).
- Implemented a **Next.js** dashboard with interactive map and table views, enabling engineers to filter, review, and monitor road damage findings by type, severity, and location.

CUMSA Security Auditor | *Web Security, Cloud Security, Linux* **October 2025**

- Conducted an authorized security audit of a student-facing web platform supporting **2,000+ students**, identifying and reporting multiple high-impact vulnerabilities including broken access control, credential exposure, path traversal, unsafe file handling, and missing security headers.
- Documented step-by-step reproduction and remediation guidance for **8+ distinct security issues**
- Hardened authentication and API authorization flows, eliminating unauthenticated event creation/deletion and reducing the site's effective attack surface by **multiple orders of magnitude**.

University of Ottawa MSA Platform | *Next.js, Tailwind CSS, SQL (Supabase), Docker* **November 2025**

- Collaborated within an **8-developer** engineering team to build a platform supporting **1,000+ students**.
- Designed UI components, refactored database schemas, and developed secure admin dashboard features.
- Worked in Agile sprints using GitHub **PR**s, **CI/CD** pipelines, and structured code reviews.
- Built a multi-organization library system with React and Supabase used by **5+ organizations** to manage book sharing.

Secure Home Lab & Cloud Infrastructure | *Linux, Docker, CI/CD, Cloudflare Tunnels, Horizontal Scaling* **January 2026**

- Designed and deployed a hybrid infrastructure spanning a Linux VPS and **3 private Windows/Linux nodes** for remote development and compute workloads.
- Implemented zero-trust remote access using encrypted overlay networking, resulting in **zero public inbound ports**.
- Hardened headless systems with full-disk encryption, firewall-scoped access rules, and least-privilege authentication.