

Michael Wong

347-538-1802 | michaelwong3049@gmail.com | [linkedin.com/in/michaelwong44](https://www.linkedin.com/in/michaelwong44) | github.com/michaelwong3049

EDUCATION

Hunter College (CUNY)

Expected Graduation - May 2028

Bachelor's in Computer Science **GPA: 3.61/4.0**

New York, NY

Relevant Coursework | Intro to Computer Science (Python), Object-Oriented Programming in C++, Data Structures & Algorithms, Computer Architecture, Operating Systems, Discrete Mathematics

TECHNICAL SKILLS

Languages: TypeScript, JavaScript, Java, Rust, Python, C++, Go, YAML, Bash, WGS, SQL, HTML, CSS

Frameworks: Spring Boot, Next.js/React.js, MongoDB, Node.js, WebGPU, Express.js, PostgreSQL, GraphQL

Developer Tools: AWS, Azure, Git/GitHub, GitHub Actions, Kubernetes, Docker, Postman, Prometheus, Grafana

EXPERIENCE

Capital One

June 2026 - August 2026

Software Engineer Intern

Richmond, VA

- Integrated end-to-end **Azure Key Vault** key management by deploying **Lowkey Vault** for local emulation using **KIND (Kubernetes in Docker)** to establish environment parity with production, validating Azure services
- Developed integration tests in **Rust** using **Cucumber** to validate data flow and verify **Databricks** and **Snowflake** integration, improving test coverage and reliability from **60%** to **80%**
- Authored a design document outlining test architecture, proposed solutions, and tradeoffs to present to engineers
- Configured **Jenkins** pipeline jobs and provisioned **AWS** infrastructure (**ECR, EKS, IAM**) to support fully automated CI/CD, and authored runbooks to cut onboarding time for new engineers

Meta & MLH Fellowship

June 2025 - September 2025

Site Reliability Engineer Fellow

New York, NY

- Built and containerized a personal portfolio page using **Docker** and DigitalOcean using HTML, Flask, and Jinja
- Instrumented **Prometheus** and **Grafana** dashboards with to monitor CPU, memory, and network I/O metrics across Docker containers to notify on threshold breaches for a proactive incident response
- Configured **NGINX** load balancing with automated testing via **GitHub Actions**, reducing downtime under load

AuriStor

March 2025 - May 2025

Software Engineer Intern

New York, NY

- Led a team of 4 interns using Agile methodologies to develop metrics telemetry pipelines to analyze AuriStorFS server system health and downtime during production, accelerating the project by several weeks
- Engineered a **Go** backend exporting **150+** AFS metrics to **Prometheus** and **Grafana** dashboards with **Alertmanager** tracking Fileserver, client, and caching performance, reducing incident detection time by **40%**
- Designed a centralized, **singleton-based TypeScript** API system with built-in caching and error handling, reducing redundant backend calls by **70%** while safely handling **5K+** container volume queries

PROJECTS

Backlight Simulator | *React.js, TypeScript, Jest.js, Canvas API, WGS, WebGPU*

January 2026

- Simulated the dynamic backlighting effect of RGB lights behind TVs/monitors using image convolutions with real-time videos using **React.js**, **Canvas API**, and memory/time optimizations using byte manipulation
- Optimized up to **20x** faster per-frame pixel processing by implementing data structures and algorithmic logic in **WebGPU/WGS GPU** shaders, cutting latency from **~97ms** to **~6ms** and sustaining smooth **60+ FPS** video
- Explored **unit testing** using **Jest** to validate pixel color manipulation algorithmic functions to ensure proper array return values and function logic, resulting in **100%** test case coverage
- Deployed a CI/CD pipeline using **Playwright**, **Chai**, and **Sinon** with GitHub Actions to ensure unit tests, integration tests, **screenshot tests**, and build systems pass prior to deployment

mockr | *Next.js, Java, Spring Boot, PostgreSQL, LiveKit, OpenAI Realtime API*

August 2025

- Developed an AI-powered mock coding interview application where users can practice both communication and technical skills by leveraging agentic speech-to-speech to simulate the software engineering interview experience
- Built a **Java (Spring Boot)** backend with PostgreSQL to persist user profiles, code, and interview results
- Integrated **OpenAI's Realtime API** with **LiveKit's WebRTC** modules using **MCP (Model Context Protocol)** architecture for code analysis, tailored AI responses, and hints resulting in a more realistic interview experience