Minh Tran

450-331-1489 | minh@minhtrannhat.com | linkedin.com/in/minh-tran-nhat | minhtrannhat.com | git.minhtrannhat.com/explore/repos

EDUCATION

Concordia University   Montreal, QC, Canada
Bachelor of Engineering in Computer Engineering.   *September 2019 – December 2025*

EXPERIENCE

Software Engineer   March 2026 – Present
Amazon   *Toronto, Ontario, Canada*

Software Engineer Intern   May 2025 – August 2025
Amazon   *Toronto, Ontario, Canada*
- Built an **LLM+RAG+MCP** test automation platform for Amazon Flex; auto-retrieved setup docs and executed mobile test orders, cutting setup from 5 min to 30 sec (10x) for 250+ daily cross-functional users.
- Implemented a **Java AWS Lambda** exposing an **MCP** server to 140+ org APIs, using reflection for dynamic method resolution and Gson to serialize test orders.
- Delivered a multi-AZ Knowledge Base via **AWS CDK** with **OpenSearch Serverless**, **S3**, and **Bedrock** ingestion Lambda, enabling scalable RAG workloads.
- Shipped a secure **TypeScript/React** upload + chat app using **AWS S3 SDK** and **AWS Bedrock SDK**, cutting time-to-first-token 86% (30s → 4s).

Software Engineer Intern   January 2023 – May 2023
Cisco   *Remote*
- Built a multithreaded **Python** microservice for internal SaaS to ingest Cat9kv telemetry from thousands of GNS3 projects into **ElasticSearch**.
- Integrated **ElasticSearch** and **Kibana** into the microservices stack by containerizing and deploying both services.
- Produced **Kibana** dashboards on daily Cat9kv usage and user counts; selected to present to Cisco Enterprise Networking Group executives.
- Authored **Ansible** modules to sync folders across managed servers with 100% integrity via **Python** SHA1 hashing.
- Automated **Ansible** module tests with **Bash**; built **Vagrant** + **Docker** environments to emulate production **Linux** servers.

PROJECTS

IncidentOps | *Python, FastAPI, PostgreSQL, Celery, Redis, OpenTelemetry, Kubernetes, Helm*   Dec 2025
- Built a multi-tenant incident API with **FastAPI/asyncpg**, adding JWT auth with refresh rotation, reuse detection, and RBAC (admin/member/viewer) via dependency injection.
- Implemented an incident state machine with **PostgreSQL** optimistic locking, cursor pagination, and an append-only audit timeline to prevent races.
- Delivered a **Celery** background system with **Redis/SQS** brokers, notification fan-out, webhook retries with exponential backoff, and delayed escalation checks.
- Added observability via **OpenTelemetry + Grafana** stack (**Loki/Tempo/Prometheus**) and structured JSON logs; containerized with **Docker** and deployed using **Helm/Skaffold** on **Kubernetes**.

TECHNICAL SKILLS

Languages: Python, Java, Javascript, Typescript, SQL, Rust, HTML, CSS
Frameworks: Flask, Node.js, FastAPI, Django, Express, Next.Js
Tools: Git, Docker, Ansible, Kibana, Grafana, Vagrant, Bash, IntelliJ, Emacs, Vim, Postman, Insomnia, LaTex, VS Code
Cloud Technologies: AWS S3, AWS Lambda, AWS OpenSearch Serverless, AWS DynamoDB, AWS Bedrock, AWS CDK
Databases: PostgreSQL, MySQL, MongoDB, ElasticSearch, etcd, Firestore