Teddy Malhan

malhan.ca | github.com/teddymalhan | ama367@sfu.ca | linkedin.com/in/teddymalhan | 672-339-2503

Education

Simon Fraser University Burnaby, BC
Bachelor of Computer Science *May 2027 (expected)*

- GPA: 3.81, Dean's List (2024 – Present)
- Courses: Data Structures & Algorithms, Artificial Intelligence, Linear Algebra, Object-Oriented Programming, Discrete Mathematics, Computer Systems, Software Engineering, Web Development

Skills

Languages: Java, Python, SQL, JavaScript, TypeScript, C++, C, HTML, CSS
Libraries/Frameworks: Spring Boot, FastAPI, Flask, Django, LangGraph, LangChain, React.js, Node.js
Databases/Cloud: PostgreSQL, Redis, MongoDB, AWS (EC2, RDS, S3, DynamoDB), GCP, Kubernetes, Docker

Experience

Electronic Arts May 2025 – Aug 2025
Software Engineer Intern *Vancouver, BC*

- Shipped an agentic MCP pipeline for automated **GraphQL** querying on 1.2M+ records for 100+ producers using **Amazon Web Services (AWS), React.js, Python**.
- Delivered 50% reported time savings by implementing **retrieval-augmented generation (RAG)** using LangChain, **PostgreSQL** pgvector and the GPT 4.1 LLM.
- Engineered an InnerSource offering for FC's custom publish flow with **gRPC** and **Protocol Buffers**, reducing integration time by 60% for internal teams.

Dialpad Jan 2025 – Apr 2025
Software Engineer Intern *Vancouver, BC*

- Delivered Digital Scorecards [link] feature for EAP pilot, supporting grading flows for 5000+ conversations using **Vue.js**, **TypeScript** and **Django (Python)**.
- Engineered RBAC feature flags for Randstad, driving a 10% decrease in operational costs across 600K+ clients.
- Consolidated 20+ dashboards from **Datadog** to Observe, unifying logs & traces, reducing MTTR by 30%.

Projects

WasteWise | *AI based waste sorting tool* **Python, FastAPI, Neo4j, React Native**

- Architected a FAISS based RAG pipeline to sort waste into bins, utilizing OpenAI Embeddings and Neo4j, parsing 20k+ mappings of food items to bins.
- Built React Native (Expo) mobile app with react-native-vision-camera for real-time waste classification.
- Integrated YOLO model trained on TrashNet dataset with TensorFlow Lite for on-device image recognition.
- Ensured 100% test coverage through pytest, applying Test-Driven Development (TDD) for a robust design

HelpGetMeFit! | *fitness app built with microservice architecture* **Maven, Spring Boot, Java, JUnit5**

- Built Java Spring Boot microservices, implementing Saga, CQRS patterns for distributed transaction consistency.
- Sustained 8.5k RPS / 55k msgs/min throughput using RabbitMQ quorum queues with p99 latency of 260ms.
- Reduced MTTR to 12 minutes through comprehensive testing with JUnit5, Testcontainers, and k6.
- Automated zero-downtime deployments with blue/green rollouts and CI/CD pipelines.

Concurrent File System | *file system with multi-threading* **C++, Threads, Synchronization**

- Implemented a simplified POSIX-like file system in C++, supporting multi-threaded read/write operations with mutexes and semaphores for concurrency control.
- Designed a block allocation and caching mechanism to simulate storage performance tradeoffs under varying workloads.
- Integrated profiling hooks to analyze throughput, latency, and CPU utilization, experimenting with different scheduling policies (round-robin, priority-based).