Kyle Su

604-616-6188 | kyle.su@uwaterloo.ca | linkedin.com/in/kyle-su143 | github.com/kyle-su1

Education

University of Waterloo Sept 2025 – Present

Computer Engineering, Bachelor of Applied Science

- **Average: 94%** — Term Distinction

Projects

Agentic AI Shopping Assistant | *FastAPI, React, Tailwind, LangGraph, Snowflake, Docker, Gemini API*

- Architected an autonomous **Multi-Agent System** using **LangGraph** with containerized **Docker** services, orchestrating specialized AI nodes (Vision, Research, Skeptic) to compile shopping reviews and recommendations
- Built a **React** SPA with **Auth0** that translates Gemini's normalized [y, x] 0–1000 bounding-box coordinates to CSS percentage overlays from the agent's structured JSON response
- Developed a hybrid vision pipeline (Gemini 2.0 Flash + Google Lens), increasing exact model-level identification accuracy from **35%** to **88%** over a single-LLM baseline
- Implemented an async microservice with **FastAPI** and **Python asyncio** to enable parallel agent execution
- Integrated a **Snowflake** vector database and warehouse cache to reduce redundant external API calls by $\sim$ **50**% (awarded Best Use of Snowflake API at CxC 2026)

Social Book Tracking App | *Node.js, React, PostgreSQL, AWS (EC2, RDS, Cognito), Redis, Socket.io*

- Built and deployed a **Node.js/Express** and **React** social platform to an **AWS EC2** instance, establishing an automated **CI/CD** pipeline via **GitHub Actions**
- Implemented **Socket.io**-based updates for friend requests and recommendations, achieving **<120 ms** latency while engineering idempotent event handlers to prevent duplicate transactions during network reconnects
- Optimized **PostgreSQL** query performance on **AWS RDS** by replacing sequential scans with **B-tree indexes** on high-frequency lookup paths, reducing book retrieval latency by **95%** across **100k**-row benchmarks
- Designed a **Redis**-backed TTL cache with deterministic ISBN keys to eliminate redundant Google Books API calls

Vector Database System | *C++, CMake, gRPC/Protobuf, xsimd, Google Benchmark*

- Built a vector database from scratch in **C++** supporting brute-force (O(n)) and HNSW graph-based (O(log n)) approximate nearest neighbor search, achieving a **7.8x** speedup on **100k** vectors
- Architected a **gRPC API layer** exposing CRUD operations, binary persistence, and remote query processing to the C++ vector database with **< 90μs** overhead per request
- Abstracted math kernels into a dedicated compute library utilizing unaligned **SIMD** batch instructions and scalar tail fallbacks, enabling portable, high-throughput distance calculations
- Designed a **memory-mapped (mmap)** storage engine, enabling zero-copy queries on datasets exceeding RAM
- Validated system correctness and performance with **35** unit tests and **4** Google Benchmark suites

Experience

Ground Station Developer Sept 2025 – Present

UW Orbital Design Team *Waterloo, Ontario*

- Developed REST endpoints for high-frequency telemetry queries with **fastapi-cache** 5-minute windows, cutting repeated database reads for static reference data
- Implemented a **Python-based orbital propagation algorithm** (SGP4) to compute satellite position data based on orbital elements (TLEs), using custom **dataclasses** and JSON outputs
- Developed **20+** unit tests with **pytest** to validate TLE parsing and orbital propagation accuracy

Coding Instructor June 2024 – Aug 2024

UBC Geering Up *Surrey, British Columbia*

- Taught **JavaScript** to **80+** elementary and middle-school students across multiple cohorts
- Designed and delivered **10+** structured lessons covering core programming concepts and hands-on coding exercises

Technical Skills

Languages: Python, JavaScript/TypeScript, C/C++, SQL, HTML/CSS, Bash
Full-stack: Node.js, React, FastAPI, REST APIs, gRPC, TailwindCSS, LangGraph
Databases & Cloud: PostgreSQL, AWS (EC2, RDS, Cognito), Redis, Snowflake, Firebase (Firestore/Auth)
Devops: Docker, Git, GitHub Actions, CI/CD, pytest, CMake, Postman