Samuel Kim

(778) 953-2568 samuelyongkim@gmail.com linkedin.com/in/samuelkim019 daedal00

Summary

Computer Science undergrad specializing in backend systems. Built APIs, data pipelines, and real-time services used by 1,700+ users, with hands-on experience in Typescript, Python, and AWS. Strong foundation in reliability (testing, migrations, performance tuning) and eager to grow as a backend engineer tackling distributed systems challenges.

Experience

ConnectPie – POS/E-Commerce Startup **July. 2025 – Present**
Software Engineer | *Python, FastAPI, AWS* Port Coquitlam, BC

- Developed a multi-tenant order delivery platform linking Ecwid e-commerce with Windows POS agents, enabling near real-time handoff of takeout orders for dozens of restaurants.
- Developed webhook ingestion and POS polling APIs in FastAPI (ASGI), deployed on AWS Lambda via API Gateway, with delivery state tracking in DynamoDB and guaranteed at-least-once order delivery through SQS.
- Optimized infrastructure costs by introducing a Pulumi-provisioned serverless stack (Lambda, DynamoDB, SQS, API Gateway), lowering monthly expenses by over 50% compared to RDS-based design.
- Introduced gap detection and backfill with EventBridge schedulers, ensuring no order loss during webhook outages and strengthening reliability guarantees.
- Collaborated with founders to refine technical specs, model DynamoDB schemas, and document failure-recovery workflows in Confluence and UML diagrams for scaling and onboarding.

Savi Finance – Fintech startup (1.7k users) **Jan. 2024 – Aug. 2024**
Software Engineer Intern | *TypeScript, GraphQL, Node.js, Nx* Remote

- Launched a core income/expense categorization feature in 7 days, which enabled the product team to ship ahead of schedule and meet customer expectations.
- Automated transaction-email ingestion pipeline with AWS (SES,SNS,SQS) and OpenAI 4o-mini, improving data consistency for 400+ users and eliminating manual categorization steps (∼35 sec saved per email).
- Led an 8-step, zero-downtime database migration, ensuring uninterrupted service across all financial platforms.
- Verified GraphQL microservice endpoints using Playground, curl & MongoDB Compass, and implemented Jest smoke tests to catch regressions before production, boosting release confidence.
- Authored three Agile-ready technical specifications (API design, workflow automation, categorization logic) alongside PMs, designers, and consultants to accelerate releases.

Projects

Muse | *Go, GraphQL, React, PostgreSQL, Redis, Docker*

- Designed & implemented a GraphQL API (30+ types/resolvers) in Go using cursor-based pagination and connection patterns, enabling efficient, scalable music discovery.
- Engineered a Redis-based caching layer for Spotify API calls, slashing album search latency by 87% (665→81 ms).
- Integrated Spotify Web API via OAuth2 to ingest real-time song and album metadata and user authentication.
- Implemented a GitHub Actions–based CI/CD pipeline with automated unit and integration tests, benchmarking, linting, and security scans—achieving 68% overall code coverage.

Education

The University of British Columbia **December 2025 (Expected)**
Bachelor of Science, Combined Major in Science Computer Science & Physics Vancouver, BC

Relevant Coursework: Data Structures and Algorithms, Database Systems, Computational Data Science, Intro. Artificial Intelligence, Computer Networking, Reliability and Security, Computer Systems and Hardware

Skills

Languages: C/C++, Python, Go/Golang, TypeScript, Java, HTML, CSS, SQL
Frameworks: Node.js, Express, gqlgen, Spring Boot, FastAPI, JUnit
Libraries & APIs: GraphQL, REST API, MongoDB, Mongoose, React, JSON-RPC, Nx
Tools: Git, Linux, Postman, Docker, Jira, Confluence, GitHub, Redis, AWS