

Can Sahin

Middle Software Engineer | Backend and Full-Stack

SUMMARY

Proficient in backend and full-stack development using Python, Go, Java, TypeScript, and PostgreSQL, with hands-on experience building high-throughput API gateways and event-driven microservices.

TECHNICAL SKILLS

Main Technical Skills	Python (1 yr.), TypeScript (1 yr.), Java (1 yr.), Go (1 yr.), PostgreSQL (1 yr.)
Programming Languages	Go (1 yr.), Java (1 yr.), JavaScript, Python (1 yr.), TypeScript (1 yr.)
C++ Libraries and Tools	C/C++/C#
AI & Machine Learning	LLM, MRR, Transformer
JavaScript Libraries and Tools	Styled components, Vite
Security	BlackDuck
Databases & Management Systems / ORM	PostgreSQL (1 yr.), Redis (1 yr.), Sphinx, SQL
UI Frameworks, Libraries, and Browsers	HTML/CSS Preprocessors, Styled components
Cloud Platforms, Services & Computing	AWS, GCP
Industry Domain Experience	Banking
SDK / API and Integrations	API, JWT
Deployment, CI/CD & Administration	CI/CD
Virtualization, Containers and Orchestration	Docker (1 yr.), Kubernetes, Terraform
Third Party Tools / IDEs / SDK / Services	Excel
Version Control	Git
	JWT

Mail / Network Protocols / Data transfer	
Operating Systems	Linux
Logging and Monitoring	Prometheus (1 yr.), Sentry (1 yr.)
Message/Queue/Task Brokers	RabbitMQ (1 yr.)
Other Technical Skills	Qdrant, Queues

WORK EXPERIENCE

IT Analyst (Co-op) - University of Ottawa (University of Ottawa IT Initiatives Support)

Duration: Sept 2023 to Apr 2024

Summary:

- Supported IT architecture governance and improved operational efficiency at the University of Ottawa by automating processes and coordinating governance activities
- The role involved enhancing meeting distribution accuracy and exploring generative AI adoption opportunities

Responsibilities:

- Developed a Python script to eliminate manual corrections in meeting distribution lists by cross-referencing group memberships and flagging discrepancies.
- Reviewed business and technical architecture documents to support architecture governance.
- Coordinated weekly Architecture Working Group (AWG) and monthly Architecture Review Board (ARB) sessions with project managers.
- Researched and presented generative AI adoption opportunities to IT architects and stakeholders.

Technologies: Python

Software Developer (Co-op) - March Networks (Compliance Automation Tools)

Duration: Jan 2023 to Apr 2023

Summary: Developed automation tools to streamline compliance review processes for legal and R&D teams at March Networks, significantly reducing manual effort and improving efficiency in license and component review workflows.

Responsibilities:

- Automated Black Duck report generation and formatting, replacing manual downloads and Excel macros.
- Built a Python tool to compare component, version, and license IDs across release versions to identify medium and high risk licenses.
- Optimized matching logic to reduce comparison runtime by approximately 50% on large releases.

Technologies: Python



Software Engineer (Go API Gateway)

Summary:

- Developed a high-performance Go API gateway capable of sustaining 13
- 7k requests per second with features including Redis-backed atomic rate limiting, circuit breakers for upstream services, and per-route metrics monitoring
- The gateway was deployed live on Render and monitored with Prometheus and Grafana

Responsibilities:

- Implemented Redis-backed rate limiting enforced atomically via Lua scripts.
- Developed circuit breakers to handle failing upstream services gracefully.
- Instrumented per-route metrics and integrated monitoring with Prometheus and Grafana.
- Deployed and maintained the gateway on Render platform.

Technologies: Go, Redis, Lua, Prometheus, Grafana, Render

Software Engineer (Job Search RAG Pipeline)

Summary:

- Built a Retrieval-Augmented Generation (RAG) system over 32 live Canadian job postings to improve job search accuracy
- The system combined BM25 and dense vector search with Reciprocal Rank Fusion and cross-encoder reranking, achieving high citation accuracy and composite score

Responsibilities:

- Implemented combined BM25 and dense vector search techniques with Reciprocal Rank Fusion.
- Developed cross-encoder reranking to reduce candidates from 20 to 5.
- Integrated an LLM-as-judge to verify claims and flag hallucinations.

Technologies: Python, LLM, BM25, Dense Vector Search

Full-Stack Developer (Gym Progression Tracker)

Summary:

- Delivered a full-stack application to automate progressive overload decisions for gym users
- The app featured a domain engine that auto-increments load on success and applies deload after failures, with JWT authentication and error tracking, deployed on Render and Vercel

Responsibilities:

- Developed frontend using React and TypeScript.
- Built asynchronous backend APIs with FastAPI and PostgreSQL database.
- Implemented domain logic for progressive overload and deloading.
- Deployed application on Render and Vercel with JWT authentication and Sentry error tracking.

Technologies: React, TypeScript, FastAPI, PostgreSQL, JWT, Sentry, Render, Vercel



Backend Developer (Semantic Caching Layer for LLM APIs)

Summary:

- Created a FastAPI middleware proxy to reduce redundant LLM API calls and latency by serving cached responses for semantically similar requests
- The system used sentence-transformers for embedding, Qdrant for vector storage, and was instrumented for monitoring

Responsibilities:

- Built FastAPI middleware proxy for caching LLM API responses.
- Embedded queries using sentence-transformers.
- Stored vectors in Qdrant with cosine similarity thresholds and TTL expiration.
- Dockerized the service and integrated Prometheus and Grafana for monitoring.

Technologies: FastAPI, Python, sentence-transformers, Qdrant, Docker, Prometheus, Grafana

Backend Developer (Spring Boot RabbitMQ Order Service)

Summary:

- Developed a Java 21 order microservice using Spring Boot and RabbitMQ to ensure duplicate-free message processing with idempotent consumers, dead-letter queue, and exponential-backoff retry mechanisms
- The service was validated with unit and integration tests

Responsibilities:

- Implemented idempotent message consumers backed by a processed-events table.
- Configured dead-letter queue and exponential-backoff retry strategies.
- Conducted unit and integration testing against Dockerized PostgreSQL and RabbitMQ.

Technologies: Java 21, Spring Boot, RabbitMQ, PostgreSQL, Docker

EDUCATION

- **Honours BSc Computer Science, Co-op Option, University of Ottawa**
Graduated Cum Laude. Dean's Honour List (Winter 2025).
Sept 2020 to Apr 2025

CERTIFICATION

- **AWS Cloud Solutions Architect (Coursera)**

