

Suleyman Kiani

+1 (289) 788-8260 | kianis4@mcmaster.ca | linkedin.com/in/suleyman-kiani | github.com/kianis4 | suleyman.io

EDUCATION

McMaster University

Master of Engineering in Computing and Software

Hamilton, Ontario

Sept. 2025 - Present

McMaster University

Bachelor of Applied Science in Computer Science

Hamilton, Ontario

Sept. 2018 - Nov. 2024

EXPERIENCE

Sales Analyst - Equipment Finance

Mitsubishi HC Capital Canada

Sept. 2025 - Present

Burlington, ON

- Facilitated end-to-end equipment financing transactions totaling **\$20M+** to date, supporting deal execution from intake and structuring through credit submission, documentation, and funding.
- Performed underwriting and spread analysis for mid-market borrowers, evaluating profitability (EBITDA/margins), coverage (FCCR), liquidity (working capital/current ratio/LOC), and leverage (debt-to-equity); produced clear, audit-ready write-ups with documented questions, answers, risks, and mitigants.
- Assessed transaction risk drivers—including customer concentration, contract duration/renewal terms, residual exposure, and related-party complexity—and collaborated with Credit to structure terms aligned to underlying contracts to reduce downside risk.
- Selected to support a high-visibility Expected Loss / Bad Debt initiative: translating legacy Excel/VBA logic into an explainable **Python** workflow and mapping how PD stress testing integrates with LGD/depreciation assumptions to improve forecasting for finance leadership.

Junior Web Developer

Giftcash Inc

May 2022 - Jul. 2023

Hamilton, ON

- Engineered migration of a **Python/Django** monolith into a **Node.js AWS Lambda** serverless stack, improving horizontal scalability by **15%** and trimming infrastructure costs by **20%** via right-sizing.
- Refactored backend services and optimized **PostgreSQL** via strategic indexing, caching layers, and query tuning, shrinking median API response times by **20%** and raising session retention **12%**.
- Automated **Puppeteer** and **Axios** workflows to scrape, normalize, and validate gift card balances, slashing manual verification work by **25%** and cutting related processing defects by **18%**.
- Orchestrated **Jenkins** and **GitHub Actions** CI/CD pipelines with automated unit tests, linting, and blue-green deploys, shrinking average release time by **30%** and rollback incidents by **20%**.

PROJECTS

Solstice Pilates Studio Management Platform

| *Next.js, PostgreSQL, Prisma*

Jul. 2025 - Feb. 2026

- Architected and shipped a Next.js/TypeScript studio-management SaaS (75k+ LOC) for a live Pilates studio, **scaling to 500+ users** and supporting daily operations.
- Implemented **Square** billing (payments, subscriptions, refunds, webhooks) **processing \$20k+ CAD** with auditable payment state; designed a PostgreSQL/Prisma layer (26 models, 31 migrations) and **80+ server routes** for scheduling, attendance, lead tracking, and an online store.
- Built an AWS SES/SNS/S3 comms stack for email campaigns with delivery tracking, transactional SMS with opt-out, and secure presigned uploads; added **Sentry** monitoring and **Playwright** E2E tests to harden production reliability.

Applify AI Resume Tailoring Platform

| *Next.js, MongoDB, Inngest*

Feb. 2025 - Jan. 2026

- Engineered a production-grade AI resume tailoring platform with a 6-step customization pipeline (job parsing, scoring, bullet rewrites, project prioritization, skills gap analysis, summary generation) to deliver job-specific resume outputs end-to-end.
- Architected event-driven background processing with **Inngest** and **Redis**-backed job state to run long-running AI workflows (20–60+ seconds) beyond serverless request limits, with step-based checkpointing and automatic retries for resilient execution.
- Implemented real-time job progress and session restoration UX using **SSE**-based updates plus resilient client polling, resolving race conditions via client-generated job identifiers and improving perceived responsiveness during multi-minute optimization runs.

- Optimized cost and latency by consolidating resume scoring into a single AI request and leveraging prompt caching for static instructions, while enforcing subscription-gated access and payments through **Stripe** and **MongoDB**-backed authentication.

PodcastHub Distributed Recording Platform | *RabbitMQ, PostgreSQL, FastAPI* Oct. 2025 - Jan. 2026

- Architected a distributed podcast recording platform using hexagonal architecture and domain-driven design, separating domain logic from infrastructure to enable independent evolution of polyglot **Node.js** and **Python** microservices.
- Engineered an event-driven workflow with **RabbitMQ** to decouple real-time recording from asynchronous media mastering, improving system resilience under back-pressure and keeping the UI responsive during long-running processing.
- Built a scalable ingestion and processing pipeline combining **FastAPI** services with object storage and background workers to handle chunked audio uploads, multi-track stitching, and format conversion without blocking user sessions.
- Containerized and orchestrated the full stack with **Docker Compose** and production deployments across **Vercel** and **Railway**, resolving cross-origin and networking constraints to support reliable end-to-end recording sessions in hosted environments.

Mike Ross AI Legal Associate | *LangGraph, FastAPI, MongoDB Atlas* Jan. 2026

- Engineered an agentic legal research assistant delivering instant, cited guidance across Canadian Criminal, Tax, and Tenancy law by combining multi-step reasoning with retrieval-augmented generation.
- Architected a **LangGraph** state-machine workflow (route → research → synthesize) to deterministically control tool use and verification loops, reducing hallucinations via structured output validation and claim checking against source documents.
- Built a high-precision semantic retrieval layer on **MongoDB Atlas Vector Search**, indexing **16,000+ official legal documents** with metadata-based jurisdiction filtering to keep results aligned to the correct province/federal context.
- Developed a full-stack, streaming Q&A experience by integrating a **FastAPI AI backend** with a **Next.js frontend** over **Server-Sent Events**, enabling responsive, token-by-token answers and reliable deployment for a live demo.

Mac Study Companion | *Next.js, FastAPI, Redis* Jan. 2026

- Architected an AI-powered lecture transcription and note-generation platform, converting student-uploaded audio/video into timestamped transcripts and structured study notes with source-linked citations.
- Designed an event-driven microservices system with a **FastAPI** API gateway and **Redis Streams** message bus, enabling loosely coupled services for upload, transcription, and note generation with independent scalability.
- Implemented real-time processing visibility via WebSocket-based status updates, improving UX for long-running transcription/summarization jobs and reducing perceived wait time through incremental progress reporting.
- Applied hexagonal architecture and domain-driven design to isolate core domain logic from external AI and storage providers, improving testability and enabling future RAG-based Q&A and semantic search extensions without major rewrites.

Safety Net NYC Crime Analytics Desktop App | *Java, JUnit, OpenCSV* Mar. 2023 - Apr. 2025

- Engineered a **Java** desktop analytics application to process and visualize **~4,000,000 NYPD arrest records**, enabling borough- and neighborhood-level crime distribution insights for travel and real estate decision-making.
- Designed a modular data model using custom abstract data types for coordinates, incidents, and geographic regions, supporting spatial containment checks and radius-based crime lookups around any NYC address.
- Implemented temporal crime exploration with date-sorted records and binary-search-driven queries, enabling fast retrieval of incidents across user-selected time windows on multi-million-row datasets.
- Built crime-aware route planning between NYC neighborhoods by modeling locations as a weighted directed graph and applying Dijkstra's algorithm, producing "safest path" recommendations based on crime-weighted edges; validated core logic with **JUnit** unit tests.

TECHNICAL SKILLS

Programming languages: TypeScript, Python, Swift, Java, C/C++

Frameworks: Next.js, FastAPI, React, Express, Django

Tools: Prisma, Docker, RabbitMQ, Vercel, Railway, Docker Compose, LangGraph, Sentry, Jupyter Notebook, Chakra UI, Jest, GCC

Databases: PostgreSQL, MongoDB, Redis, Supabase, S3

Cloud services: AWS, SES, SNS, AppSync, Vercel, Sentry