Shilo Jeyaraj

437-245-2243 — stjeyara@uwaterloo.ca — www.linkedin.com/in/shilo-jeyaraj — shilojeyaraj.com

Education

University of Waterloo **Waterloo, ON**
- *Bachelor of Applied Science in Mechatronics Engineering* Expected April 2028
- **Relevant Courses:** Data Structures and Algorithms, Algorithm Design and Analysis, Digital Computation, Circuits, SolidWorks/CAD

Technical Skills

- **Languages:** Python, TypeScript, JavaScript, SQL, Java, C/C++, HTML/CSS, Bash
- **Frameworks/Libs:** React, Next.js, Node.js, FastAPI, Flask, Pandas, NumPy, Tailwind CSS, Framer Motion, ROS 2
- **AI/ML/CV:** OpenAI API, LangChain, Google Gemini, TensorFlow, PyTorch, OpenCV, MediaPipe, OCR, Pgvector, Vertex AI SDKs, Claude Code, Cursor, Vector Embeddings
- **Cloud & Infrastructure:** Google Cloud Platform (Vertex AI, BigQuery, Cloud Run), AWS, Docker, PostgreSQL, Supabase, Git, Linux, Elasticsearch, MongoDB, WebSockets, Cloudflare, Search Engine Optimization, Devtools

Experience

Friedmann AI (FinTech Startup) **Sept 2025 – Present**
Software Engineering Intern
- Founding Engineer as a part of Canada's first AI financial planning software, designed and deployed a scalable **event-driven realtime system** using **Pub/Sub** and **WebSockets**, eliminating heavy polling cycles and enabling low-latency delivery for **10k+ concurrent advisors and clients**.
- Refactored a distributed **LLM orchestration pipeline** coordinating **multi-agent systems** through asynchronous message passing, cutting response latency by **40%**, lowering memory usage by **36%**, and reducing codebase size by **37%** through modularization and context caching.
- Optimized **PostgreSQL** query performance from **500 ms** to **50 ms** (**10× faster**) by implementing composite and covering indexes, eliminating full table scans, and enabling **Row-Level Security (RLS)** for multi-tenant data isolation within a **Dockerized CI/CD pipeline**.
- Architected a production-grade **Retrieval-Augmented Generation (RAG)** system with **vector embeddings** and **cosine similarity** search across **100,000+ document chunks**, parallelizing ingestion with background job queues to reduce retrieval latency by **40%**.
- Built secure and fault-tolerant integrations using **OAuth 2.0** across **Zoom, Google, and Microsoft APIs**, automating token refresh, implementing **webhook signature verification**, idempotent retry logic, and **exponential backoff** for reliable enterprise performance.

Normative (AI Consulting & Product Development) **Jan 2025 – Apr 2025**
Full-Stack Engineering Intern
- Built and deployed a full-stack internal web platform using **React.js**, **Flask**, and **PostgreSQL**, containerized with **Docker** and integrated into a **CI/CD pipeline**, improving reliability and performance for 30+ team members.
- Implemented secure authentication and **authorization middleware** using **JWT tokens**, **bcrypt hashing**, and **CORS configuration**, ensuring protected user sessions and preventing unauthorized access across internal APIs.
- Built an end-to-end **OCR + LLM data pipeline** integrating **Tesseract**, **OpenAI GPT APIs**, and **FastAPI microservices** to extract and summarize financial data from PDFs, improving recognition accuracy by **75%** and reducing manual entry by **60%**.
- Automated environment setup and testing using **Docker Compose**, **pytest**, and **Jest**, reducing developer onboarding time by **50%** and ensuring consistent deployments across environments.

Waterloo Aerial Robotics Group **Aug 2025 – Present**
Autonomy Subteam Engineer
- Developed computer vision pipelines using **Python**, **OpenCV**, and **machine learning frameworks** (**PyTorch/TensorFlow**), enhancing drone navigation accuracy through **Flask** backend integration.
- Optimized autonomy algorithms for object detection and flight control using **Python**, **PyMAVLink**, and **pyserial**, refactoring critical-path code to improve landing precision and flight stability through algorithmic enhancements and control loop optimization.
- Architected system integration using **React**, **TypeScript**, **Redux**, and **Socket.io**, developing **RESTful APIs** and **WebSocket** protocols to merge autonomy modules into UAV platforms through collaborative, full-stack development.

Projects

Brain Battle - AI-Powered Multiplayer Study Platform [🔗] [GitHub]
Next.js 15, TypeScript, Supabase, PostgreSQL (pgvector), OpenAI API, LangChain, WebSockets, Framer Motion
- Architected a real-time multiplayer education platform with WebSocket-based state synchronization supporting up to **20 concurrent users** per room, implementing anti-cheat detection through focus monitoring and timing analysis for competitive integrity.
- Engineered an AI document processing pipeline using OpenAI API and **LangChain** to generate contextual study notes and quiz questions from uploaded PDFs, implementing semantic search with PostgreSQL pgvector using **1,536-dimensional embeddings**.
- Developed a comprehensive gamification system with **20+ progression levels**, multi-factor XP calculation engine (accuracy × speed × difficulty), and real-time leaderboards synchronized across participants through **12 core database tables**.

DermaLens - AI Skincare Analysis Platform (Google Cloud AI Hackathon) [🔗] [GitHub]
Python, FastAPI, PyTorch, OpenCV, Google Gemini, Elasticsearch, BigQuery, Fivetran, Docker
- Developed a dual-model skincare analysis system training a custom PyTorch CNN on **500+ labeled skin images** collected and preprocessed with OpenCV, integrated with **Google Gemini API** for medical-grade condition detection, deployed on Google Cloud Run.
- Built a hybrid product search engine using Elasticsearch with vector embeddings to deliver personalized skincare recommendations in under **100ms**, achieving **10x faster** query performance than traditional PostgreSQL searches.
- Engineered an automated data pipeline with Fivetran to ingest **1,000+ skincare products** into BigQuery for real-time search indexing, reducing manual data entry by **90%** and enabling continuous product catalog updates.

Coursley (Semantic Search & RAG Elective Planning Platform) [🔗] [GitHub]
Next.js, TypeScript, Supabase, PostgreSQL (pgvector), LangChain, OpenAI API, Python
- Developed a **Retrieval-Augmented Generation (RAG)** system with **550+ courses** ingested into a pgvector database, enabling sub-second semantic search and context-aware recommendations.
- Built and deployed a full-stack platform with Next.js and Supabase, integrating JSONB filtering and multi-stage retrieval pipelines to improve course discovery efficiency by **40%** compared to manual catalog browsing.
- Enhanced student course selection through a natural language chat interface powered by **OpenAI** and LangChain memory.