

Keanu Liwongan

[Website](#) | kliwonga@uwaterloo.ca | github.com/kliwongan | linkedin.com/in/kliwongan

Education

University of Waterloo | Computational Mathematics 2020 - 2025

Relevant courses: Computer design and organization, data structures and algorithms, object oriented programming, functional programming, linear programming, computational discrete optimization, graph theory, scheduling theory

Skills

- **Languages:** Python, Rust, C++, C, Java, SQL, TypeScript, JavaScript, HTML/CSS
- **Tools:** AWS, Docker, Docker Compose, Terraform, Serverless Framework, Redis, Ansible, Datadog, MongoDB, DynamoDB, SQLite, Linux
- **Frameworks:** React, Node, Express, FastAPI, Flask, pandas, maturin, PyO3, Tokio.rs, Prost, gRPC, OpenAPI, nginx

Experience

Milk Moovement | *Software Developer Intern* 2024.09 - 2025.04

- Predictive analytics and tools for milk supply chain for farmers, producers, and milk cooperatives across the US and Canada

Milk Moovement | *Platform Development Intern* 2023.09 - 2024.05

- Implemented company-wide internal SSO solution for application dashboard using **AWS Cognito**, **React** and **TypeScript** to speed up developer login process by **100%**
- Automated entire developer onboarding process using **Ansible** and **Bash** scripting, including installing packages, **Docker** container configuration and automated **MongoDB** database loading, reducing onboarding time by **36x**
- Overhauled logging system of main application backend using a custom wrapper around **Winston** to speed up average log debugging time by **25%**
- Prototyped and launched new data generation service for business logic testing purposes using **TypeScript**, **DynamoDB**, **EventBridge**, **Terraform** and **Serverless Framework**

eSentire | *Software Developer Intern* 2022.09 - 2022.12

- Built integral microservices including authentication systems, multilayered database access and permission/rule managers using **Python** and **AWS Lambda**
- Worked on database abstraction layers built using **SQLAlchemy** and **Python** with sharding, repartitioning and custom cache population/invalidation properties
- Increased testing productivity by **30%** by automating deployment of complete end-to-end testing pipeline to sandbox using **pytest**, **Bash**, **AWS**, **Terraform**, and **Makefile utilities** as part of the **CI/CD** pipeline
- Fixed **50%** of hanging calls in **AWS Lambda** functions by creating an asynchronous timer interrupt framework using **asyncio** to await IO-blocking processes up to a certain time limit

Genellipse | *Team Lead and AI/ML/Data Scientist* 2021.09 - 2022.04

- Led team of four to successfully migrate existing POC code to working production code, per industry standards using **Python**, **AWS Lambda**, **Github Actions**, and **Terraform** by implementing **REST** microservices.

Projects

Robyn Asynchronous web server written in Rust for Python | *Open source* 2022.12 - 2024.04

- Prototyped standalone request validation schema mechanisms using smart caching techniques and OOP techniques in Python, while reducing cross-language latency and performance gaps on the Rust backend