

Sahil Alamgir

sahilalamgir.com | [sahilalamgir](https://github.com/sahilalamgir) | [in.sahilalamgir](https://in.sahilalamgir.com) | sahil.alamgir@uwaterloo.ca | [+1-437-688-2885](tel:+14376882885)

SKILLS

Languages: Python, SQL, JavaScript, TypeScript, Java, Kotlin, Go, C, C++, C#, HTML, CSS
Libraries/Frameworks: React, Next.js, Node.js, Flask, FastAPI, LangChain, TensorFlow, PyTorch, Pandas
Tools/Technologies: PostgreSQL, MongoDB, Redis, AWS, Azure, Docker, Snowflake, GraphQL, Linux, Git

EDUCATION

University of Waterloo – Bachelor of Computer Science GPA: 3.7/4.0
Wilfrid Laurier University – Bachelor of Business Administration *Expected Aug. 2028*

EXPERIENCE

Faire Jan. 2026 – Apr. 2026
Software Engineer Intern Waterloo, ON

- Led development and A/B test of unpaid Insider experience using Eppo, driving **3.4% segment GMV lift**
- Spearheaded redesign and A/B test of Insider benefits modal with PostHog, increasing **enrollments by 12.8%**
- Shipped admin tool to search failed checkout attempts with Next.js, reducing **triage time from 5 min to 10 sec**
- Engineered Kotlin API endpoints and data model to flag liquidation-eligible inventory, accelerating cash recovery
- Wrote unit, integration, and E2E test suites in Jest, achieving **80% coverage** to prevent production regressions

GenText May 2025 – Aug. 2025
Full Stack Software Developer Intern Toronto, ON

- Built semantic search system using Azure AI Search and vector indexing, reducing **retrieval latency by 60%**
- Shipped Microsoft Word add-in via Angular and .NET, allowing **1,000+ lawyers** to review and query documents
- Engineered replacement engine mapping AI suggestions to Word ranges, cutting **manual redlining time by 80%**

RemoteCasa Feb. 2025 – Apr. 2025
AI Full Stack Developer Intern Toronto, ON

- Architected RAG service in Go and LangChain, designing retrieval API to cut **support response times by 95%**
- Built mobile-responsive Next.js UIs using caching and lazy loading, reducing **page load times by 55%**
- Designed MongoDB schemas for listings and user profiles to enable search, and containerized backend with Docker

LEADERSHIP

Wat Street Jan. 2026 – Aug. 2026
Machine Learning Project Lead Waterloo, ON

- Led **8-person team** to build Python volatility contagion explorer, analyzing relationships across **500 stocks**
- Designed Pandas data pipeline to process **1.9M+ rows** of S&P 500 data into structured inputs for ML training
- Implemented PyTorch LSTM model to predict volatility, outperforming **HAR-RV baseline by 5.2%** on QLIKE
- Created interactive dashboard using React to visualize forecasted volatility and stock correlation patterns

PROJECTS

ForeverSober [!\[\]\(d3102649f02e825ddb76dc3de0190154_img.jpg\)](#) | *React Native, Expo, Supabase, Superwall, Figma* June 2025

- Launched sobriety mobile app via React Native and Expo, earning **\$160 MRR** from **200 users (15% retention)**
- Evaluated market trends and redesigned onboarding flow using Figma, increasing **paywall rate by 75%**

GenSheets (Data Science Tool) [!\[\]\(55ca3a38dbb940110628e54e3ea7505d_img.jpg\)](#) | *Next.js, TypeScript, Tailwind, Supabase, Danfo.js, Recharts* July 2025

- Deployed AI spreadsheet tool using Next.js and Supabase, enabling **100+ users** to edit data in natural language
- Created end-to-end data pipeline with Danfo.js and Recharts to analyze and visualize **100,000+ row datasets**

ShelfSmart (Youth Impact Challenge) [!\[\]\(1ad7b9dfa1e10e48660df5dd18a1b20c_img.jpg\)](#) | *Python, OpenCV, TensorFlow, Next.js, Supabase* Aug. 2025

- Won **\$2,400 prize (top 3/150 teams)** by creating platform that reduces food waste using Next.js and Supabase
- Developed computer vision model with OpenCV and TensorFlow to predict days-to-spoil with **MAE of 1.1 days**

AlzGuard (Youth Impact Challenge) [!\[\]\(7b7f78f3b14c2b344e3d1b2a79a760c9_img.jpg\)](#) | *Python, TensorFlow, Scikit-Learn, Flask, React* Aug. 2024

- Won **\$1,000 prize (top 10/150 teams)** by building AI Alzheimer's detection tool using React and Python
- Trained CNN on **2,000+ MRIs** using TensorFlow, served with Flask REST API, achieving **85.3% accuracy**