

# Ali Shariatmadari

📧 alish2001 | 📞 647-898-8532 | 🌐 alish.se | ✉ a3sharia@uwaterloo.ca | 📱 alish2001

## Education

**University of Waterloo** – Bachelor of Software Engineering (BSE)

Sep 2019 – April 2025

- Courses: Algorithms, Data Structures, Concurrency, Operating Systems, Numerical Computation, Compilers, Graphics

## Experience

**Bitgo** – Software Engineering Intern

Palo Alto, CA | May 2024 – Aug 2024

- Implemented instant ACH deposit/withdrawals for retail customers in a purely functional code-base with **fp-ts**
- Built notification system for off-exchange asset settlements between partners using **TypeScript** and **fp-ts**
- Increased test coverage for settlement stack by **20%** and migrated internal libraries to monorepo to eliminate tech debt

**PostGrid** – Software Engineering Intern

Toronto, ON | Jan 2024 – Apr 2024

- Created a tailored tracking feature for physical mail to increase customer conversions for clients using **React** and **Typescript**
- Sped up build and test time in CI by **2x** by implementing incremental caching strategies using **Github Actions** and **Docker**
- Strengthened the robustness of internal address verification and sanitation by introducing fallbacks on local data

**Bloq** – Software Engineering Intern

Chicago, IL | Sep 2022 – Dec 2022

- Developed and deployed containerized infrastructure using **Kubernetes**, **Docker** and **Node** for cryptocurrency staking on **AWS**
- Implemented **4+ JRPC** and **IPC** proxy services for authorized execution and monitoring using **Docker-Compose** and **Express.js**
- Wrote tooling for real-time monitoring of **AWS EC2** services with multi-layer authentication on **AWS Virtual Private Cloud**

**Capsule** – Software Engineering Intern

New York, NY | Sep 2022 – Dec 2022

- Spearheaded the development and release of a web3 NFT application using **React**, **Tailwind**, and **Typescript** deployed on **IPFS**
- Built **20+** responsive **React** components and setup CI/CD pipelines using **GitHub Actions** and **Husky**
- Deployed **Solidity** smart contracts with **100%** test coverage through automated integration and unit testing with **Hardhat**

**Bloq** – Software Engineering Intern

Chicago, IL | Jan 2022 – Apr 2022

- Developed real-time profiling and automated backup tooling for microservices using **Node.js**, **Docker**, and **AWS**, expanding profiler coverage by **30%**, and increasing the efficiency of system-wide backup generation by an average of **200%**
- Redesigned the API architecture to support asynchronous request processing and dynamic service routing using an **NGINX** ingress controller on a **Kubernetes** cluster, enabling rolling updates and improving scalability
- Integrated various blockchain protocols into existing cloud infrastructure for use in institutional cryptocurrency staking

**TD Bank** – Software Engineering Intern

Toronto, ON | May 2021 – Aug 2021

- Led the development of a banking dashboard prototype with a focus on user customization and predictive content using **React**
- Built an ingestion pipeline for sanitation and predictive analysis of financial data, and visualized future predictions using **Chart.js**

## Projects

**SVG-gen** 📄 – Graphics with LLMs | **Cohere RAG Hackathon Winner**

**Python, Cohere**

- Generate scalable vector graphics using Retrieval-Augmented Generation (RAG) alongside Cohere's Large Language Models

**DiVA** 📄 – Decentralized Voting | **Hack the North 2021 Winner**

**Solidity, web3.py, Flask**

- Decentralized, anonymous, and fraud-resistant election platform powered by **Ethereum** Smart Contracts
- Won out of **1,700+** participants, received a developer grant from 1517Fund

**CtrlAirSpace** 📄 – Air Mouse | **Hack the North 2020 Winner**

**Python, Mediapipe, OpenCV**

- A gesture-based HCI enabling hands-free mouse and keyboard control using video-based multi-finger tracking and voice input
- Won out of **2,200+** participants

**Vortex** 📄 – Game Engine

**Java**

- A modular 2D game engine with adherence to OOP principles, display buffering, collision detection, and sprite map parsing
- Utilized the engine to build a two-player 2D shooter game based on the Portal series with support for user-created levels

## Skills

**Languages** TypeScript/JavaScript, Python, C/C++, Java, Kotlin, Racket, Solidity, ARM & MIPS Assembly, SQL

**Frontend** React, React Native, Tailwind, HTML/CSS, Swing, JavaFX, Figma, Photoshop, Illustrator

**Backend** Node, fp-ts, Docker, Kubernetes, AWS, Hardhat, MongoDB, PostgreSQL, MySQL, Flask, FastAPI, Django