

Alex Xu

👤 me.alex-xu.site — 📞 647-674-3251 — ✉ alex.zihao.xu@gmail.com — 🔗 linkedin.com/in/alex-xu-912bb528b — 🏠 github.com/AlexZihaoXu

Education

University of Waterloo

2023 – 2028

Computer Engineering, Honours, Co-operative Program

Skills

Languages: JavaScript, TypeScript, Python, Java, C++

Fullstack: Next.js, Electron, Bun, Hono, Flask, Nginx

Cloud & DBs: AWS, GCP, Cloudflare, PostgreSQL, Prisma

Web: Docker, Socket.IO, Tailwind, Bootstrap

AI & Agents: Claude Code, OpenAI, Ollama, llama.cpp

Dev Tools: VS Code, JetBrains IDEs, GitHub, Linear

Work Experience

Good Start Labs – New York, NY, USA (Remote)

Jan 2026 – Apr 2026

Backend Engineering FastAPI, Celery, RabbitMQ, PostgreSQL, Docker, Inspect AI, React, shadcn/ui, Python, TypeScript

- Built backend services for an internal **LLM benchmark platform** used to evaluate frontier models using **Inspect AI** and the **Inspect Harbor** framework. Integrated OpenRouter, Anthropic, and LM Studio as model providers.
- For xAI's evaluation, added **BattleCode 2025** and **BattleSnake** arenas to Good Start Labs' **Harbor RL library** by wrapping their official upstream engines with Docker-isolated execution and Elo-ranked opponent pools.
- Integrated **4 benchmarks** (GDPval, AgentDojo, AgentHarm, Terminal-Bench 2.0) into a **queued evaluation pipeline** (FastAPI, Celery, RabbitMQ, Docker) with **live worker log streaming** for long-running tasks.

BitGo Inc – Palo Alto, California, USA (Remote)

May 2025 – Aug 2025

Software Engineering Grafana, Mongoose, TypeORM, Node.js, TypeScript, JavaScript, Jira

- Migrated core components of BitGo's institutional cryptocurrency **wallet platform** from JavaScript to TypeScript using an internal AI-assisted tool, resolving **200+ build errors** and **50+ CI/CD failures**.
- Upgraded **Mongoose from v6 to v8** and optimized slow queries via **Grafana/Loki** with **zero production incidents**.

Apollo Intelligence Inc – Toronto, Ontario, Canada (Remote)

Sept 2024 – Dec 2024

Fullstack Developer & Designer Next.js, Supabase, Flask, React, Node, Python, NextUI, MUI, RESTful

- Co-founded Apollo Agent, which reached the **Y Combinator interview stage**.
- Developed Apollo Learn, an **AI-powered education platform** in Next.js/TS with a Claude + ChatGPT auto-grading engine, deployed for use in a **Queen's University course** during Fall 2024.
- Built a **browser-automation system** with a **Blockly** visual editor, integrating OpenAI + Claude models.

Projects

Agent Swarm – Autonomous Agent Platform

May 2026 – Present

github.com/AlexZihaoXu/agent-swarm

TypeScript, Next.js, Docker, dockerode, noVNC, MCP

- Built a **self-hostable platform** for autonomous agents, each an isolated **Docker** container running a real **Claude Code** session on its own Ubuntu desktop. Agents spin up and tear down on demand, grouped into teams by model tier.
- Applied the platform to Lobbify while a full-time student across **~700 tickets over 3 months** and **six versioned releases, including patches, within 16 days**: agents triage and resolve Linear tickets, run **end-to-end GUI tests**, and open PRs across web, app, backend, infra and **code signing**.
- Designed the **Node/TypeScript gateway**, which exposes the fleet's single port, drives **Docker** via **dockerode**, reverse-proxies agents' terminal & desktop streams, and applies ordered migrations to upgrade running agents in place.
- Designed a **computer-use MCP server** with **no MCP SDK**, **JSON-RPC 2.0** over stdio, giving agents mouse, keyboard and vision on their own X desktop through **python-xlib** and **XTEST**.

Lobbify (Multiplayer Platform)

Sep 2024 – Present

https://lobbify.games/ – Founder

Electron, Next.js, Prisma, PostgreSQL, WebRTC, Cloudflare, Docker

- Founded and shipped Lobbify, a cross-platform Electron app for hosting private multiplayer game sessions (currently Minecraft) with **automatic modpack sync** and **no port forwarding**. Public beta, **30+ early adopters**.
- Integrated **Microsoft authentication** for Minecraft accounts and built launcher-style workflows to browse, download, and configure **Minecraft versions and plugins**, delivering an experience comparable to CurseForge and Modrinth.
- Shipped **signed cross-platform desktop builds** through a CI/CD pipeline; the application uses a Bun/Hono backend with Prisma/PostgreSQL in **Docker on DigitalOcean**, fronted by Cloudflare with rate limiting and auth.
- Built **peer-to-peer networking over WebRTC** with **Cloudflare TURN relay fallback** for NAT traversal, and a **Brotli-compressed custom protocol** that port-forwards LAN games.

WaterlooWorks Explorer (AI-assisted Job Platform)

Oct 2025

Personal Project

FastAPI, React, TypeScript, Ollama/OpenAI

- Engineered a full-stack platform with FastAPI and React, integrating tools (**Resume Analyzer**, **Cover Letter Generator**) powered by OpenAI and Ollama, with a UI/UX design.
- Designed an **LLM pipeline** to batch-process postings and infer attributes (salary ranges, job type, external requirements) through LLM reasoning—boosting job application speed by **5×** with improved relevance and quality.

Self-Balancing Robot

Aug 2025

Personal Project

ESP32, TMC2208, NEMA 17, MPU6050, PID, UDP, Sensor Fusion, Fusion 360

- Built a **two-wheeled self-balancing robot** powered by an **ESP32**, dual TMC2208 drivers, and **NEMA 17 stepper motors**, with motion sensing from an MPU6050. Modeled and printed the robot frame using Fusion 360.
- Implemented a **two-layer PID controller**: one for **angle stabilization**, another for **velocity control**.
- Added sensor fusion with **Kalman and complementary filters** for orientation estimation.
- Developed a custom **UDP protocol** for real-time **Wi-Fi remote control**.
- Integrated **fall and collision detection** with automatic motor cutoff to help protect hardware.

Party Physics (Game) (Grade 12 Computer Science Final Project)

Oct 2022 – Jan 2023

<https://github.com/AlexZihaoXu/PartyPhysics>

Java, Java2D, Dyn4J, TCP

- Developed a **multiplayer game** similar to Stick Fight from scratch, *cross-platform*.
- Built multiplayer functionality on a **custom TCP networking framework** with an abstraction layer for serialization and deserialization, supporting **300+ moving objects** with **sub-10 ms latency**.
- Engineered a **custom active rendering framework** in Java Swing with anti-aliasing, a **particle system**, and **interpolated animations** for smooth displayed motion at a lower tick rate.
- Integrated real-time physics simulation with the **Dyn4J** library, supporting **synchronized physics in multiplayer sessions**.

Minecraft Remake in C++

Oct 2022 – Oct 2023

github.com/AlexZihaoXu/minecraft_cpp

C++, OpenGL, GLSL, GLFW

- Built the **rendering engine** in C++ using **OpenGL**, **GLSL** shaders, and **GLFW**.
- Implemented a **meshing system** that only adds visible faces to the geometry, **reducing VRAM usage**.
- Tested with **Valgrind**, checking for memory leaks during runtime.

Blinky Face (Robot Pet)

Jan 2026

UofT Hacks

Raspberry Pi 5, ESP32, Python, TMC2209, NEMA 17, BNO085, UART, Fusion 360

- Built Blinky Face, a **Raspberry Pi 5 + ESP32 robot** with LCD face displays, a ToF sensor, TMC2209-driven NEMA 17 wheels, a BNO085 IMU, and a **2-DOF servo-actuated FPV head**. Created CAD models in Fusion 360.
- Wrote the **ESP32 control firmware** and a **Python library** encapsulating UART communication (motor commands, voltage, speed, gyro). Connected the Raspberry Pi 5 to a **local LLM** for reasoning and Wi-Fi FPV control.

Robotic Arm Project

Sept 2023 – Nov 2023

University of Waterloo – ECE 1A – ECE 198

- Developed a robotic arm for the ECE 198 project using the **STM32 Nucleo** board, programming the system and UI in C for control.
- Designed and refined the robotic arm using **Fusion 360**, optimizing the model through **5 iterations**.
- Created a **custom gear set** for stepper motor torque transmission. Designed a box-shaped control unit with an integrated wind tunnel intended for improved cooling.
- Implemented an **asynchronous I2C LCD driver** and configured **DMA-based ADC sampling for joystick input** to reduce blocking during peripheral I/O.
- Utilized **spring-based motion smoothing** for arm movement, reducing vibration and instability during operation.
- Implemented an **inverse-kinematics system**, mapping target coordinates to joint motion for simplified control.

Minecraft Remake in Java

Oct 2021 – Jan 2022

github.com/AlexZihaoXu/Minecraft_AE

Java, LWJGL, GLFW, OpenGL, GLSL

- Developed a remake of Minecraft from scratch with Java & GLFW. All aspects of the project were **self-taught** using resources like LearnOpenGL. No external code was copied from other repositories.
- Built a **custom game engine** with OpenGL and GLSL shaders using LWJGL. Implemented **multithreaded terrain generation** based on Perlin noise, **BFS-based lighting**, ray-casting-based world editing, and a collision system.
- Analyzed **decompiled source code** to study Minecraft's design patterns and inform an independent implementation.

Ice Knight (Game)

Feb 2025

<https://me.alex-xu.site/unity/ice-knight/>

Unity, C#

- Developed Ice Knight with 2 other developers for UWaterloo Game Dev Club W2025 Game Jam in **3 days**.
- Implemented **player controls**, **enemy AI**, and **projectiles**.
- Contributed to **VFX** including camera shake, particles, shadow casting, 2D lighting and post-processing effects such as glow. Drew and animated **all entities** in the game.