William Zhang

630-890-8089 | zhangywu@umich.edu | linkedin.com/in/williamzhang04 | will-zhang.com

Education

University of Michigan Ann Arbor, MI
Bachelor of Science in Engineering - Data Science *May 2027*
Relevant Coursework: Data Structures and Algorithms, Web Systems, Software Engineering, Machine Learning

Experience

Amazon Web Services (AWS) May 2025 – August 2025
Software Developer Intern *Arlington, Virginia*

- **Reduced manual qualification effort by 80%** by designing event-driven workflows in **Go** with back-pressure, idempotent retries, and failure isolation to automate test orchestration
- Cut firmware qualification time from **1-2 weeks** to **1 day** by designing a no-touch firmware validation pipeline for **1,400+ Bering libraries**, enabling bi-monthly global release cycles
- Maintained **99.95%** data-path availability by automating drive thermal checks, DCO tests, and movement validation, and scaled fleet monitoring across **2,800+** Glacier libraries

Bumpups Jan. 2025 – April 2025
Software Engineer Intern *Remote*

- Increased video analysis throughput by **6x** (from **180 s** to **30 s**) by parallelizing preprocessing and feature extraction with bounded worker pools and zero-copy buffers in **Python**
- Built a distributed task pipeline to process **10,000+** videos/week using async I/O, multiprocessing, and compact serialization; reduced peak RSS and GC pauses on long runs
- Deployed containerized microservices with **Docker** and optimized CI/CD workflows to reduce build and rollout time from **10 minutes to under 3 minutes**, ensuring smooth scaling for **5,000+ users**

Avodah Sep. 2024 – Dec. 2024
AI Software Engineer Intern *Remote*

- Developed a low-latency **speech-to-text** engine using a fine-tuned **Soniox Transformer**, improving transcription accuracy to **98.3%** through audio denoising and voice normalization
- Implemented a real-time streaming pipeline in **Python and C++** with buffered concurrency and async inference, reducing end-to-end latency from **5s to 0.9s**
- Containerized the full pipeline on **AWS ECS** with autoscaling and fault-tolerant health checks, maintaining **99.9% uptime** across production deployments

Projects

Unite – AI-Powered Gaming Platform May 2025

- Scaled matchmaking platform to **10,000+ game sessions** by building distributed real-time infrastructure with **React, TypeScript, Node.js, and Socket.io**, ensuring sub-200 ms response latency
- Integrated **AWS Amplify, Lambda, and Step Functions** to orchestrate authentication, matchmaking, and analytics tasks asynchronously, maintaining **99.9% uptime** for **5,000 concurrent users**

MatchaBot August 2024

- Engineered a Discord bot in **Python** using **discord.py** that handled **20,000+ active users** and processed **5,000+ song requests/day** through efficient event-driven concurrency
- Enhanced community interaction by integrating **OpenAI GPT-4** for real-time conversational replies, achieving **150,000+ total chat sessions** while maintaining <200 ms command latency

High-Performance Data Pipeline (C++) Nov. 2024

- Built a templated C++ data-structure library (vector, priority queue, hash map) optimized with custom allocators and move semantics to process **200k+ log events/sec** in stress-test workloads
- Implemented multi-threaded graph search and knapsack heuristics using **std::thread** and fine-grained locks, reducing scheduling latency from **120 ms to 35 ms** under peak load

Technical Skills

Languages: Python, Go, C/C++, Java, JavaScript, TypeScript, SQL (PostgreSQL, MySQL), Swift/SwiftUI, C#
Frameworks/Libraries: React, Node.js, Flask, MongoDB, Redis, Bootstrap, Firebase, Supabase, Pandas, NumPy, scikit-learn, PyTorch, TensorFlow, Cypress, discord.py
Tools: Git, Docker, AWS (ECS, Lambda, S3, Step Functions, SNS), Google Cloud, Azure, Kubernetes, CI/CD (GitHub Actions, Jenkins), VS Code, FFmpeg