

# JONATHAN MACOCO

github.com/jtmacoco | jonathan.macoco@gmail.com | 510-828-3291 | linkedin.com/in/jonathan-macoco

## EDUCATION

---

### Masters of Science, Computer Science

May 2026

San José State University, San José, California

- GPA: 3.57

### Bachelor of Science, Computer Science

May 2023

Chico State University, Chico, California

- GPA: 3.29

## TECHNICAL SKILLS

---

Languages & Frameworks: C, C++, Python, JavaScript/Typescript (Node.js, React, Next.js), Golang

Cloud/DevOps: AWS (ECS, EC2 S3, SES), GCP, Docker, Terraform

ML/AI: PyTorch, LLMs, RAG, Vector Databases, GANs

Database: MongoDB, PostgreSQL, Pinecone

Tools/Other: Git, OpenGL, CUDA, REST APIs, Agile, System Design, Distributed Systems, Multi-Threading

## EXPERIENCE

---

### Full Stack Software Engineer Intern, Coinable, Remote

Feb 2024 - June 2024

- Built secure email verification system with AWS SES, reducing onboarding friction for new users
- Designed and deployed RESTful APIs (Node.js + MongoDB) supporting scalable CRUD operations
- Architected MongoDB schemas that improved query efficiency and reduced data duplication
- Delivered responsive frontend login and signup flows in Flutter, streamlining user authentication

## PROJECT EXPERIENCE

---

### Distributed Collaborative Code Editor

- Built distributed editor with Golang backend, Node.js WebSocket server, Next.js frontend, deployed on AWS ECS
- Scaled to 10,000 requests and 1,000 concurrent sessions with zero failures, sustaining 1,170 requests per second
- Implemented CRDT-based conflict resolution, achieving less than 100ms median edit latency
- Ensured fault tolerance and even load distribution under reduced ECS capacity

### LLM Codebase Navigator

- Engineered LLM-powered tool indexing GitHub repos by files/functions with embeddings stored in Pinecone
- Achieved 83% accuracy on 60+ context aware code queries
- Designed a RAG pipeline with metadata filtering, multi-vector search, and FastAPI based responses

### N-Bodies Simulation

- Wrote CUDA kernels to parallelize gravitational simulation of 10,000+ particles in real time achieving 10x speed up
- Parallelized particle generation using C++ with OpenMP across 16 CPU cores achieving 3x speed up
- Visualized interactions using OpenGL + GLSL shaders

### Synthetic Malware Images

- Built Conditional GAN in PyTorch generating synthetic malware images from 9,300+ samples across 25 families
- Designed CNN classifier to validate GAN quality, improving class distinguishability
- Applied one-hot encoding + dataset balancing to enhance model generalization

### Deep Reinforcement Learning Agent

- Implemented a Deep-Q Network in PyTorch, enabling RL agent to achieve competitive performance in the Hex game
- Engineered custom Hex environment in PettingZoo, optimizing state representation and reward mechanics
- Improved training efficiency with batch updates and stabilized learning via epsilon decay exploration strategy