(DING-HAN) SEAN YAN

COMPUTER SCIENCE MAJOR

 linkedin.com/in/dhsyan   dhsyan@student.ubc.ca   +1-(604)-977-6799   DHSYan

SKILLS

Languages: Javascript/Typescript, C/C++, Java, Python, Rust, Bash, Scheme, Assembly, HTML/CSS, LaTeX
Tools: NodeJS, Jest, Docker, ExpressJS, MongoDB, Pandas, Git/Github (Action), Flask, Nix, Linux.

EXPERIENCES

Mathematics & English Tutor Taichung, Taiwan   05/2023 - 07/2023
Meiko School

- Tutored students from Years 1–12 in Mathematics and English, adapting methods to individual learning styles.
- Raised students' Mathematics test scores from 46% to 96% through a strong pedagogical approach focused on clarity, intuition, and active problem-solving.
- Fostered student confidence in mathematics by creating an engaging and supportive learning environment.
- Wrote detailed session reports to inform parents of student progress and offer actionable feedback for continuous improvement.

PROJECTS

Job Tracker API

- Built to solve the chaos of unorganized job hunting, this project was born out of firsthand frustration with tracking multiple job applications across spreadsheets and notes — transforming the process into a centralized, scalable, and developer-friendly backend system.
- Designed and developed a RESTful Job Tracker API using Node.js, Express.js, and MongoDB, enabling users to efficiently track job applications with CRUD operations, notes, and status management.
- Implemented robust data modeling and modular architecture with Mongoose ODM, improving code maintainability and reducing onboarding time for new developers by 40%.
- Containerized the entire application with Docker and Docker Compose, simulating a production-grade environment and enhancing deployment reliability across different systems.
- Achieved 95%+ unit test coverage using Jest, ensuring stable CI/CD integration and significantly reducing regression bugs during feature updates.

Nix Config; Infrastructure as Code (No fear in reinstalling your OS!)

- Developed declarative, reproducible Nix configurations to ensure identical OS setups across systems.
- Implemented infrastructure as code to automate system rebuilds, eliminating manual provisioning.
- Acquired deep understanding of Linux systems by daily driving NixOS, and increased efficiency by 100%

.dotfiles - Personalized Software Developement Workflow

- Engineered declarative NixOS configs, ensuring consistent environments across all systems.
- Automated system provisioning with IaC, enabling rebuilds, featuring rollbacks and removing manual setup.
- Daily drove NixOS, mastering Linux internals and 10x development workflow efficiency.

Terminal Chat App (built 100% in C)

- Built a fully multi-threaded Terminal Chat Application in C using TCP/UDP sockets and POSIX threads, enabling concurrent handling of multiple users and real-time communication.
- Designed thread-safe mechanisms to support features such as private messaging, broadcast messaging, and peer-to-peer media transfer without blocking.
- Mastered low-level network and concurrency programming by implementing per-client thread management, synchronization primitives, and custom communication protocols.

PRIMPL Assembler, SIMPL Compiler (School Project)

- Built a compiler and assembler for custom imperative and assembly languages, simulating a full toolchain.
- Gained strong proficiency in manual memory management, low-level architecture, and algorithms.
- Applied deep understanding of stack organization, including frames, pointers, and heap allocation.

EDUCATION

The University of British Columbia   09/2023 - Present

Program: Bachelor of Science
Major: Computer Science
GPA: 4.33/4.33
Standing: Dean's Scholar