Haozhe Li

+1 (437) 661-7680 | lihaozhe013@gmail.com | Portfolio | lihaozhe013 | lihaozhe013

Education

University of Toronto Toronto, ON, Canada
B.A.Sc in Computer Engineering *Sep. 2023 – Apr. 2028 (Expected)*

Experience

Frontend Developer (Intern) May. 2025 – Aug. 2025
Hangzhou EagleCloud Security Technology Inc. *Hangzhou, Zhejiang, China*

- Developed new front-end features for an enterprise cybersecurity desktop app and an admin web console as part of a 7-person front-end team, using TypeScript, React.js, Ant Design, and Electron Framework.
- Optimized coding workflows and enhanced software quality by applying advanced prompt engineering methodologies within Cursor (AI IDE), demonstrating expertise at the intersection of AI and modern coding practices.
- Mastered Git workflow and GitHub utilization on large projects, leveraging personal feature branches, forks, and pull requests to drive cross-functional collaboration and ensure successful, on-time code integration.
- Utilized DevOps and CI/CD pipelines for testing, self-tested code in development environment and pre-production environment to ensure the code complied with the company standards, successfully contributed code to a SaaS release.

AI Lab Research Assistant (Intern) Jun. 2024 – Aug. 2024
Shenzhen Research Institute of Big Data *Shenzhen, Guangdong, China*

- Automated research environment by scripting the one-time cleanup and reinstallation of Conda environments and key packages (PyTorch/TensorFlow), reducing setup time and enabling teams to immediately run new models on idle computing capacity.
- Reinstalled Ubuntu and Debian systems on lab computers to fix compromised software environments, and configured a seamless model deployment workflow by integrating SSH with the research team's web console, repairing numerous computers that the research team couldn't use for experiments.
- Developed a comprehensive guide and configured runtime environments for the research team to run open-source models from GitHub, reducing the time research teams spend on configuration.

Projects

TradeFlow System | *Node.js, SQLite, React.js, React Router, AntD* Jul. 2025 – Present

- Developed and deployed a tradeflow system at an integrating circuit sales company, incorporating JWT auth, RBAC, i18n, and Excel export features, which reduced order errors, minimized losses, and simplified processes to reduce time spent on order management and revenue calculation.
- Configured GitHub Actions and deployed the application across AWS (pre-production) and Alibaba Cloud (production), implemented PM2 for critical process and concurrency management, and configured Nginx for secure proxy and SSL hosting, establishing a robust platform that delivered significant labour cost savings.
- Open-sourced the public component of a tradeflow project on GitHub with comprehensive documentation, structured commit history, and a PR workflow to enable easy collaboration and maintainability for future contributors, while continuing to provide long-term, part-time maintenance, including regular upgrades and bug fixes for the full system.

StreamFile Server | *Go, Gin, Node.js, TypeScript, Express, Multer, Rust, Video.js* Jan. 2025 – Present

- Developed a lightweight, database-free static resource hosting server supporting Markdown rendering, video/audio playback, static webpage hosting, private link generation, file upload, and search functionality, enabling seamless content delivery without complex infrastructure.
- Refactored the backend from Node.js Express with a Rust-compiled native addon to Go using the Gin framework, unifying the technology stack, thereby simplifying and improving project performance and maintainability.
- Built a Markdown reader with React.js supporting LaTeX formulas and implemented all other frontend components using Tailwind CSS without UI frameworks, creating an ultra-lightweight, responsive interface that reduced bundle size and improved the load speed by 80% on low specification devices.

GIS Route Optimization Application – Course Project | *C++, GTK, Git, A*, Dijkstra* Jan. 2025 – Apr. 2025

- Developed a Geographic Information System (GIS) desktop application in C++ with GTK on Mate Desktop as a course project in a 3-person team, implemented map rendering, geographical name search, shortest path and multi-stop path finding features.
- Utilized A* algorithm for shortest path finding, Dijkstra, multi-start greedy method and simulated annealing for multi-stop path finding, ultimately achieved 90% of the technical grade.
- Maintained a clear Git branching strategy utilizing feature branches, which resulted in timely mergers and effective conflict resolution, leading to each milestone being completed ahead of schedule.

Runner Game (FPGA Board Game) – Course Project | *C, RISC-V Assembly, FPGA Board, CPUlator* Mar. 2025

- Developed a 2D runner game in C on a DE1-SoC FPGA board as part of a 2-person team, implementing core game logic, VGA display, and audio components, and delivered a fully functional game as the course project.
- Utilized Git for version control and CPUlator for simulation and debugging, optimized online collaboration, and reduced integration issues, which enabled the team to complete all development work in just 2 weeks.
- Compiled, deployed, and optimized the game on the FPGA board, ensuring stable performance during a 5-hour continuous run, demonstrating system reliability and robustness.

Greedy Mouse Game (FPGA Board Game) – Course Project | *Verilog, FPGA Board, ModelSim* Nov. 2024

- Developed a 2D Greedy Mouse game in Verilog on a DE1-SoC FPGA board, implementing core game logic, PS/2 keyboard, audio components, and video components, and delivered a playable game in 3 weeks.
- Proactively managed partner's extended absence and non-cooperation by escalating the situation to the TA and swiftly adjusting the project plan. Accelerated development to independently complete the project within the deadline, achieving a grade of 80% and demonstrating strong adaptability and problem-solving skills.
- Utilized ModelSim and DESim for game development and testing, compiled and deployed the game on the board using Quartus Prime, and presented a playable demo in the final presentation.

Git Snapshot | *Tauri, JavaScript, Rust* Jun. 2024

- Developed a Rust based git snapshot tool that can auto fetch, pull, commit and push markdown notes to GitHub with one click, enabling users unfamiliar with Git to quickly commit Markdown notes to GitHub with commit history.
- Implemented the GUI based on Tauri Framework, providing a big green submit button and error message pop-up display, simplifying the operational complexity, enhancing the user experience for non-professional users.

Technical Skills

Programming Languages: C, C++, Go, JavaScript/TypeScript, HTML, C#, Python, RISC-V Assembly
Developer Tools: Git, Bash, VS Code, Cursor, LTSpice, Quartus Prime, ModelSim, DESim
Frameworks: Node.js, Gin, React.js, Flask, NumPy, Tailwind CSS, GTK, Electron Framework
Tools & Other Skills: Nginx, AWS, GitHub Actions, Figma, LaTeX, Google Workspace
Hardware / Digital Design: FPGA board, Verilog