Haozhe Li

Third Year Computer Engineering · University of Toronto

37 Grosvenor St, Toronto, Ontario, Canada

(+1) 437-661-7680 | lihaozhe013@gmail.com | My Website | GitHub | LinkedIn

Summary

Currently a third-year student in the Computer Engineering program at the University of Toronto. Possesses a strong interest in software development, DevOps, firmware, and embedded development.

Education

University of Toronto — *Toronto, Ontario, Canada*

B.A.Sc in Computer Engineering — *Sep. 2023 - Apr. 2028 (Expected)*

Work Experience

Front-End Developer (Intern) — *Hangzhou, Zhejiang, China*

Hangzhou EagleCloud Security Technology Inc. — *May. 2025 – Aug. 2025*

- Contributed to a 7-person team in developing a new front-end feature for an enterprise cybersecurity desktop app, and 7 front-end features for the admin Web Console, utilizing TypeScript, React.js, Ant Design, and Electron Framework.
- Applied advanced prompt engineering techniques and use Cursor AI IDE to enhance the quality and efficiency of coding, mastering prompt engineering methodologies.
- Proficient in utilizing GitHub as a remote repository platform, encompassing work with personal feature branches, forks, and pull requests. Demonstrates expertise in Git workflow and collaboration on large projects.
- Utilized DevOps and CI/CD pipelines for code deployment, self-tested code in test and pre-release environments to ensure compliance with company standards, and successfully contributed to a SaaS product release.

AI Lab Research Assistant (Intern) — *Shenzhen, Guangdong, China*

Shenzhen Research Institute of Big Data — *Jun. 2024 – Aug. 2024*

- Automated the one-time cleanup and reinstallation of all necessary Conda environments and specific packages, including PyTorch and TensorFlow, by writing Bash scripts, enabling research teams to quickly run fine-tuned new models on idle computers.
- 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, repaired 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 their initial setup time from days to just hours and allowing them to focus on research instead of configuration.

Projects

TradeFlow System — *Node.js, SQLite, React.js, AntD*

Personal Project — *July. 2025 - PRESENT*

- Developed and deployed a tradeflow system with stateless authentication, RBAC, i18n, and Excel export, which reduced order errors, minimized losses, and streamlined processes to cut time spent on order management and revenue calculation at an integrating circuit sales company.
- Configured GitHub Actions and deployed the application across AWS (pre-production) and Alibaba Cloud (production), ensuring high availability and scalability while saving the business significant costs by eliminating the need for a full-time labour position.
- 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, Video.js*

Personal Project — *Jan. 2025 - PRESENT*

- Developed an 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, unified 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.
- Open-sourced the project on GitHub with comprehensive documentation and structured commit history.

GIS Route Optimization Application

C++, GTK, Git, A, Dijkstra*

University of Toronto — *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.
- Utilized A* algorithm for shortest path finding, Dijkstra, multi-start greedy method and simulated anualling for multi-stop path finding, ultimately achieved 95% of the technical course grade.
- Maintained a clear Git branching strategy utilizing feature branches, which resulted in timely merges and effective conflict resolution, leading to each milestone being completed ahead of schedule.
- Coordinated team process: weekly wiki progress reports, meeting facilitation, task tracking tables, and authored GUI design proposal outlining architecture, task division, and timeline.

Runner Game (FPGA Board Game)

C, RISC-V Assembly, CPUlator

University of Toronto — *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)

Verilog, FPGA Board, ModelSim

University of Toronto — *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 simple video components, and delivered a playable game in 3 weeks.
- Proactively reminded the partner to participate in the collaboration, resolved conflicts caused by their prolonged absence and lack of cooperation, reported the situation to the course teaching assistant, swiftly adjusted the project plan, and accelerated progress. Ultimately completed the project in the final week and achieved a score of 70%.
- 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

Personal Project — *Jun. 2024*

- Developed a Rust based git snapshot tool that can auto fetch, pull, commit and push markdown notes to github with one click, enabled 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, simplified the operational complexity, enhancing the user experience for non-professional users.

Technical Skills

Programming Languages	C, C++, Go, JavaScript/TypeScript, HTML, CSS, Python, RISC-V Assembly, Verilog
Developer Tools	Git, Bash, AWS, VS Code, Cursor, Figma, 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, Docker, LaTeX, Google Workspace
Hardware Tools	FPGA board, Protoboard, Multimeter, Oscilloscope