Jun Kai Zhang

416-786-1259 | jkai.zhang@mail.utoronto.ca | linkedin.com/in/jun-kai-zhang/

SKILLS

- Languages: JavaScript, TypeScript, HTML, CSS, Java, Python, C++, C, Bash, SQL
- Frameworks: React, React Native, Angular, Node.js, Express.js, Flask
- Databases: PostgreSQL, MongoDB, SQLite, Firebase
- Tools: Git, GitHub, Docker, VS Code, Android Studio

EDUCATION

Honours Bachelor of Science, Computer Science Sep 2022 - Present
University of Toronto Scarborough
Specialist in Computer Science Co-op
CGPA: 3.63/4.00

PROJECTS

CollabPen Oct 2024 - Dec 2024

- Engineered a code collaboration **React** web app utilising **Node.js**, **Express.js**, and **y-websocket** to enable real-time communication
- Engaged in meetings weekly to communicate progress and concerns, which allowed the team to adapt under pressure and still present a robust deliverable
- Wrote meaningful commit messages and pull request summaries on **GitHub** to concisely communicate changes to the team
- Resolved server hanging issues by utilising **Web Workers** and a **Redis** cache server to handle long tasks like invoking an LLM
- Streamlined the production and development environments by leveraging **Docker** and automated deployment by using **GitHub Actions**

Stock Social Network Jul 2024 - Aug 2024

- Built a stock social networking app with **React**, integrating **PostgreSQL** for capturing relational representations of data
- Applied **normalisation** techniques to ensure the database is in BCNF, minimising data redundancy and anomalies
- Reduced the number of joins needed for complex query operations by **denormalising** some tables, optimising the overall efficiency for such queries

GoHere App Jan 2024 - Apr 2024

- Wrote comprehensive unit and integration tests for **API** calls using **Jest** which ensured the reliability of the backend by 100%
- Crafted a **Shell/Bash** script with written documentations to easily configure default variables needed to launch the app

Dungeon Venture Nov 2023 - Present

- Developed a 2.5D dungeon crawler game purely in **C++**, featuring a built-in level editor that integrates techniques like binary space partitioning to optimise rendering
- Fixed iterator bugs by using comparisons to an inorder traversal function for occurrences of mismatches; ultimately pinpointing the error that was to be fixed
- Applied **OOP** and **SOLID** principles to abstract the code for improved modularity and maintenance

Instamall May 2023 - Aug 2023

- Constructed an **Android** e-commerce app, employing **Java** for backend interactions and **Firebase** for an easy-to-use object-oriented interface for data storage and retrieval