Wei Yang Hongj

http://weiyangwiki.live/

Email : hongwy@u.nus.edu
Mobile : +65-97293587

Education

- **National University of Singapore** — Singapore
  Bachelors in Engineering(Honours) Computer Engineering; GPA: 4.20/5.00 — *Aug. 2018 - May 2022*

Experience

- **YipitData** — Mountain View, CA
  Software Engineer Intern — *Jan 2021 - Dec 2021*
  - **React/React Native**: Worked alongside QA engineers and UI/UX designers to develop an iOS/Android email application. Wrote the UI components needed via React Native and built upon existing/open-sourced native libraries written in Java/Swift for specific feature requests from users.
  - **Golang**: Designed and developed part of the back-end payment API for custom domain hosting via Godaddy and Stripe API
- **NUS Information Technology Winter Program** — Singapore
  Software Engineer Intern — *May 2020 - Dec 2020*
  - **Django/React**: Responsible for developing an automated IoT platform Frameworks/tech stacks used: React for front-end development, Django for back-end. MySQL database. Cisco's Identity Service engine API and Postman for fetching the requests. Microsoft ADFS single sign on authentication for NUS users.
- **NUS Information Technology Summer Program** — Singapore
  Software Engineer Intern — *Dec 2019 - Apr 2020*
  - **Python/React**: Responsible for developing a full-stack Wi-Fi analytics dashboard using IoT sensors. Wi-Fi data gathered is streamed to MySQL database. Linux bash commands for initial networking test on a Raspberry Pi. An in-depth technical article written by me, curated by "The Startup" on Medium explaining the tech stacks used as well as the developmental process:
    https://medium.com/swlh/building-an-automated-wi-fi-sensor-around-the-nus-campus-7f187524f58e

Projects

- **Implementation of concurrency and software transactional memory on top of a javascript-like language**: Made use of Antlr's grammarV4 to create the necessary grammar required for parsing of interpreted langugage. Thereafter, concurrency logic is implemented after parsing to enable multi-threading and context switching of block statements in the interpreter. Software transactional memory via variable snapshots are used to prevent race conditions and unsafe read/writes instead of traditional lock approach via semaphores.
- **NUS module timetable group scheduler in telegram**: Developed a telegram bot to automatically schedule groups of timetables and find free timeslots available for a group meetup from just their weblink. Made used of API from nusmods and telegrambot. Check it out: https://github.com/MeLoveCarbs/NUSMODS-group-scheduler
- **Computer engineer capstone project**: This project is a culmination of what we have learned throughout our coursework as a computer engineering undergraduate and is usually taken in the final year. This project requires us to develop from scratch the hardware and software components needed and integrate them together. The project's goal is to develop a wearable device that predicts the dance move of the user using machine learning models and streams the data collected using socket programming into a web dashboard. My role for this project is a full-stack developer, and the architecture I used for front-end, back-end, and database were ReactJS, NodeJS, and MySQL. My main task includes an iterative design of a front-end dashboard consisting of user testing and feedback, a back-end server for HTTP APIs, designing the database schema, and socket programming via Socket.io library to provide real-time data from the database to the dashboard.

Programming Skills

- **Languages**: Golang, Python, Javascript, C++, SQL, Java **Technologies**: AWS, React/React native, Echo