YIMIAN LIU

i@yimian.xyz | https://iotcat.me | 607-391-5915

EDUCATION

Cornell University **Ithaca, NY, USA**
M.Eng. in Electrical and Computer Engineering; GPA: **3.85/4.00** *Aug. 2022 – Dec. 2023 (Expected)*
Relevant coursework: Distributed Computing Principles, Fundamentals of Machine Learning, Embedded Operating System, Computer Networks, UNIX Tools and Scripting

University of Liverpool **Liverpool, UK**
B.Eng. in Electrical and Electronics (Honor Degree); GPA: **3.83/4.00** *Sep. 2017 – July 2021*
Relevant coursework: C/C++ Programming, Software Engineering, Image Processing, Neural Networks

SKILLS

Languages: Python, C/C++, TypeScript/JavaScript, Java, HTML/CSS, PHP, Lua, Linux Bash, SQL, Assembly
Technologies: Kubernetes, Docker, Nginx, Node.js, React, Sass, MongoDB, MySQL, Git, CI/CD, RESTful, AWS

WORK EXPERIENCE

Software Engineer Intern **New York, NY, USA**
Guzman Energy *July 2023 – Aug. 2023*
•Developed a responsive web application for a trading index portal using **ReactJS**, **TypeScript**, and **Material-UI**.
•Implemented Microsoft Azure 2FA login for the trading portal by utilizing **OAuth2**, OpenID Connect, and **JWT**.
•Leveraged **Docker** to containerize the applications and deployed them to **AWS EC2**.
•Developed an **iOS** and **Android** AI scribing mobile APP from scratch using **React Native**, based on the UI/UX team's design on Figma.
•Collaborated with the backend team through the **RESTful API** to handle the application's data and user management.

PROJECTS

Deployment and Orchestration of a Kubernetes Infrastructure **Ithaca, NY, USA**
Individual Project *Jan. 2023 – Present*
•Engineered a scalable Kubernetes infrastructure with **Kubeadm** and Helm, efficiently managing high-traffic web services.
•Orchestrated **Docker** containers using **Kubernetes** for effective load balancing and auto-scaling.
•Deployed **Nginx Ingress** to manage external service access, enhancing security and load distribution.
•Utilized **Prometheus** and Grafana to monitor system performance and health, optimizing resource utilization and stability.
•Established a centralized logging infrastructure using **Elasticsearch**, Logstash, Kibana (**ELK** Stack), and **Kafka**, thereby facilitating real-time log processing, comprehensive system monitoring, and efficient troubleshooting.
•Integrated DroneCI and Github for **CI/CD**, enabling streamlined updates and efficient bug fixes.

Full-Stack Development and Management of High-Traffic Web Services **Ithaca, NY, USA**
Individual Project *April 2018 – Present*
•Conceptualized, designed, and deployed various self-developed, purpose-specific websites and web services.
•Utilized cutting-edge technologies like Gatsby, **ReactJS**, **Sass**, Bootstrap, WebSocket, and **JQuery** for building interactive and responsive front-ends for websites, using GA4 for data-driven performance tracking and user interaction analysis.
•Efficiently handled back-end services using a mix of Node.js, **Express**, Python **Flask**, and **PHP**, showing flexibility and adaptability across different programming environments and requirements.
•Created a highly popular public Random Image API using **PHP**, **MySQL**, and Redis, which has **over 100,000 visits** per day, demonstrating the ability to develop and manage high-performance web services.

Distributed, Linearizable, Sharded Key-Value Database **Ithaca, NY, USA**
Course Project advised by Prof. Lorenzo Alvisi *March 2023 – May 2023*
•Designed and built a robust, sharded key-value store in **Java**, optimized for performance in distributed settings.
•Implemented and modified the **Multi-Paxos** protocol, a crucial algorithm to guarantee the database system's resilience and continued operation, given a majority of servers remain functional.
•Enhanced the system's functionality by streamlining the Multi-Paxos protocol into a simpler, more efficient **Raft**-like protocol, resulting in improved system performance and reduced complexity.
•Employed a strategic approach to optimize the key-value store by incorporating sharding across server groups. Each group was designed to execute the Paxos consensus algorithm, thus ensuring system-wide consistency and reliability.

AWARDS

•**Machine Learning:** Secured **2nd rank** out of 55 in an in-class Kaggle competition at Cornell by architecting a highly accurate CNN model using **Keras**, with data preprocessing and ensembling techniques applied to a noisy MNIST dataset.