MINGHAO DU

New York, NY | 917-657-3210 | md965@cornell.edu
minghaodu.com

EDUCATION

Cornell University, **Cornell Tech**, New York, NY — Expected May 2024
M.S. in Computer Science and Information Systems
Honors and Awards: Cornell Tech Merit-Based Scholarship

University of Liverpool, Liverpool, UK — Sep 2018-July 2022
B.S. in Information and Computing Science| GPA: 3.92/4.00 (First Class)
Relevant Coursework: Data Structure and Algorithms, Operating System, Computer Network, Database, Software Engineering, Advanced Object-Oriented Programming, Computer Architecture, Machine Learning
Honors and Awards: University Academic Excellence Award (top 5%)

TECHNICAL SKILLS

Coding Languages: Python, Java, C/C++, C#, HTML, CSS, JavaScript, SQL, PHP, LaTex, Shell
Frameworks: Flask, SpringBoot, TensorFlow, PyTorch, React
Other: Git, Linux, Microsoft Azure, Mongo DB, DevOps, Postman

EXPERIENCE

Emerson Electric Holding Co. Ltd | ***Software Engineer Intern*** | Remote — Aug 2020-Nov 2020

- Developed a product repair application for post-sale services, allowing customers to submit repair orders and communicate with the Emerson Repair Service Team; outcompeted 3 other **agile development** teams and produced a revenue increase of over **12%**
- Built **RESTful APIs** in **C#** and **.NET** framework, designed responsive layout web page utilizing **React.js** on the **client side**, implemented features such as search orders and filters
- Extended the application to the **mobile side** by building the WeChat Mini program from scratch, designed the **UI/UX** of the app and implemented the prototype with a React-like **Model-View-ViewModel (MVVM)** render engine

Zhejiang Songbai Information Technology Co. Ltd | ***Software Engineer Intern*** | Hangzhou, China — Jun 2020-Aug 2020

- Collaborated with a team to develop financial software in **Java** and the **SpringMVC+Spring+Mybatis (SSM)** framework that provides comprehensive financial asset transaction information and consulting-related services for customers
- Improved system robustness by **15%**, solved data inconsistencies by building a **thread lock** with Synchronized blocks

PROJECTS

Machine Learning Engineering Project – MiniTorch (Python, CUDA, Parallel Programming, CI/CD) — Fall 2022
A Comprehensive Machine Learning Framework

- Built a comprehensive machine learning framework from scratch, involving auto-differentiation, back-propagation, and tensor broadcasting mechanism, allowing users to run Torch code and train deep learning models efficiently
- Implemented an **object-oriented** tensor backend via higher-order functions (zip, map, reduce), leading to the system's scalability
- Adopted parallel and **multi-thread computation**, turned **1000+** lines of the code to **CUDA programming** version and deployed it on GPU, improving the computation performance by **41%**
- Followed **CI/CD** pipeline and wrote **200+** unit tests via **Pytest** framework, making the system work under **10 more** scenarios

Group Management System (RESTful APIs, Python, Flask, SQLAlchemy, Pytest, Agile, Git) — Spring 2021
A Group Management System for Automatic Grouping

- Designed and implemented a system for facilitating students' grouping and managing group requests, serving over 200 users simultaneously. Published the system as open source, saving over **14,000** Chinese Yuan for XJTLU IT Department
- Built **RESTful APIs** using the **Python Flask backend** and adopted **SQLAlchemy** toolkit for Object-Relational Mapping; utilized **Pytest** framework to design fifty test cases for six main user scenarios, cooperated with a team to process **30+** tickets
- Worked in an **agile development** manner (**Scrum process**), applied **Microsoft Azure Boards** to maintain **PBIs** and sprints inspection, used **Git** for repository **version control** and management

File Synchronization Application (Python, Socket Network Programming, UDP/TCP, Multi-thread) — Fall 2020
A Large Efficient Fast Trusty File Sharing System

- Developed an end-to-end automatic file synchronization application via **Python Socket network programming**, allowing users to synchronize files across different devices
- Innovated a novel file transfer mechanism and protocol that adopted **UDP** for hosts' communication and **TCP** for file transfer
- Addressed packet overlapping issues in transmission by flow control and manually adjusting the receiver buffer size; applied **multithread** techniques to overcome concurrency issues, improving the transmission rate by **24%**