Jing Wang

(919) 491-9302 jw844@duke.edu linkedin.com/in/jing-wang-34520628b https://github.com/OrangelineE

Education

Duke University, Durham, NC **May 2024**
Master of Engineering in Electrical and Computer Engineering *GPA: 3.9/4.0*

University of Glasgow, Glasgow, United Kingdom **June 2022**
Bachelor of Engineering in Electronics and Electrical Engineering *Honours of the First Class*

University of Electronic Science and Technology of China, Chengdu, China **June 2022**
Bachelor of Engineering in Electronics and Electrical Engineering *GPA: 3.74/4.0*

Technical Skills

Languages: C++, Java, C, SQL, Python, HTML/CSS, JavaScript

Development Tools: GitHub, GitLab, Jira, CI/CD, Docker, Bash, Perl, Maven, Gradle, Postman, Tomcat, Figma

Systems & Frameworks: Linux (Ubuntu), JUnit, Spring MVC, Mybatis, Springboot, MySQL, PostgreSQL, Flask, AWS (EC2, Lambda, Speech Recognition), Azure (Web App Service, Speech Recognition), RESTful API, VMware, WireShark

Experience

Software Backend Developer Intern | Trip.com Group **May 2023 – August 2023**

- Developed a **SpringMVC**-based project management reporting website, reducing reporting time from 30 to 5 minutes.
- Implemented employee query features, enhancing data retrieval speed by processing **HTTP requests** for **JSON** data.
- Integrated **Webhooks** to capture change event payloads and store them in **Mybatis** for real-time data updates.
- Conducted JUnit and Postman **API testing** for bug detection and resolution. Utilized Tomcat and Java for remote debugging with breakpoints and employed **LOGGER** for detailed problem tracking and analysis.

Software Test Intern | Chengdu Super Love Technology Co., Ltd **February 2022 – May 2022**

- Achieved a 100% on-time delivery rate through the management of tasks and timelines in agile development using **Jira**.
- Reported and resolved over 200 product defects by working closely with **cross-functional** teams including developers, product managers, and designers, actively participating in product reviews and crafting test cases.
- Utilized Postman and vConsole for thorough **troubleshooting** and assessment of product functionality.

Projects

Live AI Hackathon [Joint Duke - Havard]:Panacea Mind | *Python, React/Native* **March 2024**

- Directed the development of 'Panacea Mind', an AI-driven mental wellness app that analyzes user speech, providing sentiment charts, reports, and mental illness suggestions, using React/Native for front-end and Python for back-end.
- Utilized **AWS services** including speech recognition and NLP to analyze user speech tone and sentiment.
- Ensured 100% data security and compliance by implementing **ethical AI practices**, reducing potential issues.
- Managed the team and project timelines, contributing to the project winning awards for **Best AI** and **Best Design**.

Mini Amazon System Project | *Python, HTML/CSS* **March 2024**

- Designed a online marketplace using Python, Flask, and **PostgreSQL**, and HTML/CSS, replicating key functionalities of Amazon including user account management, product listings, and transaction processing.
- Implemented cart and order functionalities, managing real-time inventory checks and transaction processing.
- Designed the table schema and built a **large database** with realistic data, using pagination for extensive content.
- Coordinated updates and features using GitLab, hosting meetings, assigning tasks, and ensuring alignment with goals.

Really Interesting Strategic Conquest | *Java* **April 2023**

- Crafted a multi-player combat game with movement, upgrade, and attack features, employing **S.O.L.I.D** principles and design patterns such as **Abstract Factory** and **Chain of Responsibility** for diverse territories and game mechanics.
- Outlined project architecture and created UML diagrams, managing code via GitHub version control and peer reviews.
- Managed project in Jira, achieving 3 evolutions and ensuring timely delivery through **effective communication**.

Implementation of malloc and free | *Data Structure, C++* **February 2023**

- Designed a metadata structure and used the **sbrk** function for efficient **heap** memory allocation, implementing both standard and **thread-safe** versions of malloc and free for safe multi-threaded memory allocation.
- Implemented **Best-Fit** and **First-Fit** strategies to find suitable memory blocks, optimizing allocation.
- Realized a 75% fragmentation rate by splitting memory blocks from larger ones and merging adjacent available blocks.