Tianrui Zhang

Providence, United States | tianrui_zhang@brown.edu | 401-201-8410

Education

Brown University — Providence, United States
MS in Computer Science — *Aug 2025 – May 2027*

Shanghai Jiao Tong University — Shanghai, China
BS in Computer Science — *Sept 2021 – June 2025*

Experience

Software Development Intern — June 2024 – Sept 2024
Meituan — *Beijing, China*

- Developed a performance tracking component using **TypeScript** to collect scroll jank rate and rendering speed on Meituan's app, enabling in-app performance profiling and benchmarking across multiple pages
- Implemented a navigation hook that timestamps route transitions and page render completion, to accurately calculate render time and report it with the current page ID and metadata to the server for performance monitoring
- Implemented a gesture hook that timestamps the start of user scroll interactions, to calculate the proportion of janky frames during the scroll and report the actual scroll duration to the hive cluster upon completion
- Designed and integrated a debouncing mechanism to prevent redundant reporting during high-frequency scroll rendering
- Encapsulated component logic into a class and initialized it at a point where all dependencies were available
- Enabled a **23% reduction** in average scroll jank rate across key pages through data-driven optimizations

Research Assistant — Nov 2024 – May 2025
Shanghai Jiao Tong University — *Shanghai, China*

- Developed a **debugging agent** that utilizes the debugger to debug a failed program and locate the root cause of error
- Collected the error information and call stack during target program's execution, and extracted the location of failed test case from error information to let debugger stop at the failed test case when executed again under debugging mode
- Established a socket-based connection between the agent and the target program's JVM to capture debug command history and results during interactive debugging, enabling future use of this data by an LLM to assist reasoning
- Designed and implemented two tool functions that enable LLM to inspect the source code of a given function and invoke debugger commands on whitelist to safely learn about program's runtime state and set breakpoints
- Designed a prompt that informs LLM of target program's error information and call stack, along with operations allowed or encouraged, and requests LLM to return the location of program failure's root cause on method level

Projects

MiniDiscord | *Java, Spring Boot, MyBatis, MySQL, Redis, Kafka, JavaScript, React* — Apr 2025 – July 2025

- Developed a scalable chat application that supports instant messaging in both private and group conversations
- Designed and implemented a microservices-based backend composed of five services, including push notification, storage, user management, channel management and message handling. Leveraged **Kafka** for inter-service communication
- Enabled real-time message delivery by establishing a websocket-based connection between the server and the application when the user enters the chat interface. Designed an offline message queue to ensure message delivery for offline users
- Accelerated data display on client side by creating database index based on user id and channel id, and caching frequently requested data including user status, channel list and recent messages in **Redis** on server side
- Supported keyword search across all joined channels by integrating **ElasticSearch** service into project

GlobalMuseum | *Python, Flask, JavaScript, React, MongoDB, Leaflet, Echarts, Antd* — May 2024 – June 2024

- Developed a data visualization platform that displays collections and chart-based statistics from museums worldwide
- Scraped data about collections from museums' official websites, including images, description and metadata
- Designed and implemented an interactive dashboard where, by clicking on a museum's logo on the world map, and user can zoom in the map on the museum's area and browse all collections and statistics from the selected museum
- Displayed statistics of collections from the selected museum via different types of charts created using **ECharts**, including proportion of materials, frequency of associated keywords, production period and country distribution
- Designed and implemented a paginated UI where user can browse collection images and click for details

Technical Skills

Languages: Java, Python, C/C++, SQL, JavaScript, HTML/CSS
Frameworks: React, Flask, Django, JUnit, Spring Boot
Developer Tools: Git, Docker, Google Cloud, MySQL, Redis, Kafka, ElasticSearch
Libraries: pandas, NumPy, Matplotlib, MyBatis, Lombok, Tornado