Jiayu Liu

▪ Jersey City, NJ   ▪ jl13683@nyu.edu   ▪ https://www.linkedin.com/in/jiayu-liu-023902204/   ▪ https://www.github.com/jLiucoder
▪ https://www.jiayujerryliu.com

EDUCATION

New York University   *Sept. 2022-May 2024*
Master of Science in Computer Engineering, GPA 3.7/4.0   *New York City, NY*
- **Courses**: Internet Architecture and Protocols, Data Structure and Algorithms, Computer Systems Architecture, Machine Learning, Real Time Embedded Systems, Principle of Database Systems

Boise State University   *Aug. 2018-May 2022*
Bachelor of Science in Computer Science, GPA 3.7/4.0   *Boise, ID*
- **Courses**: Data Structures, Design and Analysis of Algorithms, Agile Development, Operating Systems, Linear Algebra

TECHNICAL SKILLS

- **Languages**: Java, JavaScript/TypeScript, Python, SQL, HTML/CSS
- **Frameworks**: Spring Boot, Node.js, Django, React.js
- **Tools &Platforms**: Docker, Terraform, AWS, MySQL, PostgreSQL, Redis, Kafka, Postman, Selenium WebDriver, Git, GitHub Actions Continuous Integration (CI)
- **Software Development Practices**: REST API Development, Object-Oriented Programming, Software Development Life Cycle, Agile Development
- **Miscellaneous**: IntelliJ IDEA, Visual Studio Code, MySQL Workbench, Oracle Data Modeler, Jira/Atlassian

EXPERIENCE

Ra Labs   *May 2023-Aug. 2023*
Software Engineer Intern   *New York City, NY*
- Collaborated closely with the project manager to decipher product requirements, and played a pivotal role in identifying and implementing essential backend and blockchain features within **a 6-person development team**.
- Executed end-to-end development of **Solidity** smart contracts for user listing, transactions, and Ethereum payments in a data trading SaaS product.
- Designed and built **Nest.js REST API** with TypeORM for CRUD operations on off-chain user information stored persistently in a PostgreSQL **Docker** image.
- Conducted unit tests for smart contracts and the Nest.js backend services with **Jest**, achieving 90% and 87% coverage respectively.
- Integrated **Swagger** documentation accessed over 100+ times by the team for in-browser API testing.
- Implemented **React** front-end with **Redux** for state management across 20+ components, improving the overall usability of the UI.

PROJECTS

Vercel Clone   *Dec. 2023-Feb. 2024*
GitHub: https://github.com/jLiucoder/nova-vercelClone
- Built a Vercel clone using **Express.js**, automating static site deployments by allowing users to input their GitHub repository link, thus streamlining the build and deployment processes.
- Utilized **Terraform** to automate the setup of AWS services and Upstash services, ensuring infrastructure reproducibility.
- Leveraged **AWS ECS** with **Fargate** for serverless container execution, integrated **S3** for storage, used **Redis** for pub/sub log streaming from container to backend, and **WebSocket** for real-time log delivery from backend to frontend.

One-Room Chat App   *Oct. 2023-Dec. 2023*
GitHub: https://github.com/jLiucoder/oneRoomChat-remake
- Implemented a multi-user, single room real-time chat app using React with Redux, Styled with Tailwind CSS, and enabled real-time communication via WebSocket.
- Utilized **Spring Boot** with **Spring Data JPA** to ensure efficient server-side operations, implemented **JWT** for user authorization.
- Integrated **Redis** as a caching layer to cache messages and used ZSet data structure to rank top active users.
- Incorporated **Apache Kafka** as the message queue for asynchronous data processing, to ensure scalability and reliability of the application.
- Crafted **Docker Compose** scripts to streamline the orchestration of services with a single command.

Amusement Park Web App   *Jan. 2023-May 2023*
GitHub: https://github.com/jLiucoder/amusementpark
- Designed a 10-entity database schema using **Oracle Database Modeler** for an amusement park e-commerce website.
- Enhanced data operations with SQL **triggers** to automate aspects of business logic, created **indexes** and **transactions** for search efficiency and ensure data consistency.
- Built **Django** web app with Django's ORM enabling essential CRUD functionalities such as user and shopping cart management.