Rui Cao

Durham, NC 27705 | (919) 806-6496 | r.cao@duke.edu | LinkedIn | GitHub | Personal website

EDUCATION

Duke University, Durham, NC **Expected May 2024**

M.Eng in Electrical & Computer Engineering with Software Development concentration *GPA: 3.95/4.0*

Kean University, Union, NJ **Sep 2018 – Jul 2022**

B.S. in Finance, Minor in Computer Science and Mathematical Sciences *GPA:3.85/4.0*

SKILLS

- **Programming Languages:** Java, C/C++, Python, HTML, CSS, JavaScript, SQL, Swift, Verilog, R, Stata
- **Frameworks:** Django, Spring Boot, Angular
- **Databases & ORM:** MySQL, PostgreSQL, SQLAlchemy (Python), Hibernate (Java)
- **Tools:** Gradle, Maven, Docker, Git, AWS, Linux, CI/CD, Firebase, gdb, Valgrind

WORK EXPERIENCE

Duke University | Java, Gradle, Swift, Firebase, Spring Boot, Maven **May 2023 – Present**

Software Engineer Intern

- Designed, built, and deployed a cross-platform app using **Java** for Android and **Swift** for iOS, enabling Duke professors to post quizzes online for student review, with 500+ active users.
- Utilized **Firebase** as backend to create question bank module and implement question release feature.
- Enabled real-time notifications for students in Android and iOS versions by utilizing **Firebase Cloud Messaging** (FCM) topic messaging and sending network requests to the server developed using **Java** and **Spring Boot**.

PROJECTS

Gambling Game | Java, Spring Boot, Maven, Angular, Hibernate, MySQL **Jul 2023 – Present**

- Implemented a web game with features like character training, gambling, trading, chat, and alliances using **Spring Boot** for RESTful APIs, **Maven** for dependency management, and **Angular** for a dynamic user interface.
- Utilized **MySQL** and **Hibernate** for efficient data storage and management, enhancing performance and usability.
- Leveraged **Git** and GitLab for version control, collaboration, and issue tracking, incorporating Agile methodology with a **CI/CD** pipeline and UML diagrams, thereby improving team productivity and software quality.

Mini-Amazon | Python, Django, PostgreSQL, Socket, Multi-threading, Docker **Apr 2023 – Apr 2023**

- Developed a full-stack web app modeling Amazon system paired with warehouse and delivery app using **Django** and **PostgreSQL**, which simulates the entire process from purchasing to delivery.
- Used **Socket** and **Google Protocol Buffer** to communicate with world-simulator and Mini-UPS server. Enabled multi-threading to receive messages between servers.
- Realized UI for full-featured web pages using **JavaScript** and **HTML/CSS**, introducing features such as a search bar, shopping cart, account modification, email system, and BGM.

HTTP Caching Proxy | C++, TCP Socket, Multi-threading, Docker **Jan 2023 – Jan 2023**

- Developed an HTTP caching proxy server using **C++,** supporting GET, POST, and CONNECT requests, featuring concurrency for handling requests from multiple endpoints and socket-based packet transmission.
- Added RAII technique and strong exception safety guarantee to class modeling for robust server functionality.
- Enhanced server efficiency by implementing response caching following RFC7234 validation and expiration rules and facilitated deployment through **Docker**.

Thread Safe Malloc | C, Thread-Safe **Jan 2023 – Jan 2023**

- Implemented Malloc Library with First Fit and Best Fit allocation strategies and conducted performance experiments.
- Made it thread-safe, with locked version (pthread mutex) and non-locking version (Thread Local Storage).