# Yue (Elsie) Wu

+1 347-277-2558 | wuyue403@gmail.com | 8775 costa Verde Blvd, San Diego, CA, 92122 | wuyue403.vercel.app

## Education

**University of California San Diego** San Diego, CA
*Master of Science in Computer Science* *Sept. 2024 – June 2026(expected)*

**New York University** New York, NY
*Bachelor of Arts in Computer Science — GPA: 3.73* *Sept. 2020 – May 2024*

## WORK Experience

**GYFR** Jun. 2024 – Aug.2024
*Incoming Software Engineer Intern* *Chicago, IL*

- Design and develop user interface components for the GYM app using **React.js** and **TypeScript**, including user login, payment method, and session completion pages.
- Implement real-time features such as video calls and session feedback and use **Firebase** for backend services, including authentication, data storage, and payment processing.

**JD.com, Inc** Jun. 2023 – Aug. 2023
*Software Engineer Intern* *Shanghai, China*

- Designed and developed data visualization modules for logistic warehouses used by 20+ internal teams, focusing on generating charts and connections as a BI tools, utilizing **Spring Boot** and **Vue.js**.
- Designed and implemented **RESTful APIs**, mapping HTTP methods for efficient and reusable routes, with CRUD actions on **MySQL** using **Spring Boot**.
- Designed and implemented efficient data model and operations using **MySQL** for user info, chart info, and parameters, optimized the data retrieval with indexing and connection pool.
- Collaborated with UI designers to implement interfaces including input control and navigational components for chart displays, and data-driven search lists feature with **Vue.js**, enabling decision making with this tool.

## Projects

**Bank Application** | *Kubernetes, PostgreSQL, Golang, gRPC, Redis* June. 2024 – Present

- Designed and developed a **Golang-based** bank web service, tailored for operating the **Gin** framework, facilitating account management, real-time balance monitoring, and secure money transfers.
- Developed **gPRC API** with Go for CRUD operation of user data on **PostgreSQL** and implemented protobuf for defining service interfaces.
- Designed asynchronous processing in Golang using background workers and **Redis** as its message queue.
- Designed and created data schemas with ER diagrams. Optimize **PostgreSQL** query with rollback functionality to make atomic transactions.
- Enhanced the web application's 2Auth by implementing an SMS one-time-password authentication with PASETO.
- Built a **CI/CD** pipeline using **GitHub Actions** to automate the testing process. Containerized the application with **Docker**, and utilized **AWS EKS (Kubernetes)** for seamless deployment on **AWS EC2** instances.

**WuniChat - Real-Time Chat Application** | *Express.js, Node.js, Socket.IO, JavaScript* Mar. 2024 – Apr. 2024

- Developed a real-time chat application with individual chat and group chat using **Node.js**, **Express.js**, and Socket.IO, enabling users to join and communicate in specific chat rooms.
- Engineered the real-time chat feature using **Socket.IO**, facilitating instant message exchanges among users while minimizing message delivery latency.
- Implemented **TCP** and **UDP** protocols for robust network communication, applied **AIMD** control algorithm to manage network congestion effectively, utilized the Link State Algorithm to determine optimal routing paths.

**Face Recognition Web Application** | *React.js, PostgreSQL, Express.js, Node.js* Sept. 2023 – Dec. 2023

- Designed and developed a web application capable of recognizing faces in user-uploaded images with integrated AI using **React** and **Node.js**.
- Implemented **RESTful API** using **Node.js** and integrated with **PostgreSQL** to handle CRUD operations.
- Implemented **JWT**-based user authentication and secured validation for client-submitted data, and addressed password encryption for user registration under **Node.js**.
- Utilized **React Hooks** to manage component state and lifecycle in functional components; optimized rendering performance with useCallback, reduce response time by 8%.
- Achieving face recognition through images sourced by integrating Clarifai face-detection API and utilized **JavaScript** to implement dynamic client-side pages with reusable components.

## Technical Skills

**Languages**: Python, Java, C/C++, C#, JavaScript, HTML/CSS, Golang, SQL
**Technologies/Framework**: React.js, Node.js, Spring Boot, Bootstrap, MongoDB, Express.js, Docker, Kubernetes, Redis, gRPC, PASETO
**Tools**: AWS(EC2, EKS), PostgreSQL, MySQL, Socket.IO, Wireshark, ER Diagrams, JWT, Protobuf, Linux, Git