Yolande Yan

(424) 440-4429 | yjy@seas.upenn.edu | https://github.com/yjyolandeyan

EDUCATION

University of Pennsylvania, School of Engineering and Applied Science Philadelphia, PA
B.S. in Computer & Information Science and Cognitive Science Expected May 2025

- GPA: 4.0 / 4.0 | Honors: Dean's List

SKILLS

- Languages: **Java**, **Python**, C, C++,**JavaScript**, SQL, YAML, TypeScript, HTML/CSS.
- Frameworks: **React.js, React Native, Node.js, Express.js**, Spring MVC, **Spring Boot, Spring Cloud**, Django, JUnit, JavaFX, Flask.
- UI Frameworks: Ant Design, Material-UI, BootStrap.
- Databases & Cloud Platforms: **MySQL**, SQL Sever, MongoDB, Redis, Firebase, **AWS**, Azure, Google Cloud Platform.
- Tools & OS: Docker, CI/CD, Git, Postman, Linux, Dagster, Apache Tomcat.

INTERNSHIP EXPERIENCE

Bank of America Global Technology Summer Analyst New York, NY
Software Engineering Intern for Liquidity Management Technology June 2024 – August 2024

- Secure authentication through **OAuth** authentication or API keys and implement data encryption and access control measures to protect sensitive data.
- Improved database query performance and system stability. The average query response time was reduced from **300ms** to **150ms**, and the number of transactions processed per second increased from **200 to 300 TPS**.
- Reduce main thread blocking by handling background tasks with the asynchronous task queue **Celery**.

Datablau| Remote China
Software Development | Business Cloud Management Platform February 2024 – June 2024

- Based on **Spring Boot, MyBatis, Redis, MQ** and **React**, the commercial management cloud platform integrates project, listing, tenant, investment, contract and financial management functions to improve enterprise management efficiency.
- The login uses **JWT** token technology to achieve stateless single sign-on, replacing the original **session** sharing problem under the **cookie session** mechanism, and using **OAuth2** protocol to complete the third-party application authentication and authorization.
- Introduce **Redis** to cache frequently accessed data and build indexes for high-frequency query fields, reducing data query time from an average of **100ms** to **20ms**.
- Optimize **RabbitMQ** configuration, increase the number of consumer instances, and use batch processing technology. Increased throughput from 5000 messages per second to 10000 messages per second.
- Upgraded the project from **Webpack4** to **Webpack5**, increasing build speed by **12s**. To reduce **loader** and **plugin** syntax errors, use the check-updates upgrade package tool. Update merge, url-loader, chunkhash, file caching and other configurations afterward.

Rockwell Automation San Jose, CA
Software Engineering Intern May 2023 – July 2023

- Assisted in the rollout of the Enterprise Integration Center, integrating software into visual and low-code process editors to improve engineering efficiency.
- Enhanced the backend functionality of Datablau's metadata matching system by optimizing system integration and monitoring capabilities using **Java, Python**, Apache, ActiveMQ, Hawtio, and **Spring Boot** frameworks.
- Developed Redis client to control IP request rate, realize data persistence function, and complete product release 4 weeks ahead of schedule.

PROJECT

Oral Health Cloud Backend Management System

- A system based on **Spring Cloud** and **Spring Boot** and **React** developed to optimize data collection and quality management to form a complete health profile by establishing a full lifecycle oral health management platform for the community population.
- Optimized for **Slow loading on first screen**. By loading on demand, **CDN** introduction, and **Gzip** compression of resources, the loading time of the first screen was optimized from the initial **5200ms** to about **800ms**.
- Participated in the design and development of the privilege management module, combined with **SpringSecurity** to realize the security control of the system. Ensure that different user roles have appropriate permissions to protect the information security of the system.
- Upload large-size files (such as MP4 videos) using **Multipart Upload** to realize the functions of continuous uploading at breakpoints, retrying in case of errors, drawing progress bars, and pausing the upload. The upload time is reduced from **9-17s to 1-2s**.
- Use **Ant Design** and **ECharts** to encapsulate **15+** public components (Regular Checksum, Table, Pagination, Dialog, Search Component, etc.), and write related documents for other projects to use.
- The search service adopts **ElasticSearch** to realize full-text search function, which achieves the effect of accurate and fuzzy query.

E-commerce Shopping Site

- Combining **Vue 3** and **Django 4**, the aim is to build a complete e-commerce platform. The project will cover a number of key features including product categorization, shopping cart, order management, user authentication and payment system.
- Analyzed page key performance indicators with **Performance**. Page performance was optimized by **Anti-Shake and Throttling**, **Routing Lazy Loading**, and **Image Lazy Loading**. Page **FCP** time was reduced from **3 s** to within **1 s**.
- Use **vue-router** navigation guards to implement validation of user login status based on **token** and use **localStorage** to solve the problem of **vuex** non-persistent storage.
- Combined with **Mixins** attribute, extract code with high reuse rate as public methods, reducing project code redundancy by 3000 lines.
- Processed model data using **Django REST Framework (DRF)** serializer to ensure data formatting and API fit.
- Reduced query response time from **10s** to within **1s** by **SQL** execution plan analysis to achieve reasonable use of indexes to improve query speed.