Jun Xia

(949)-678-5255 | jackyxia.uci@gmail.com | LinkedIn | Github | Website

Education

University of California, Irvine **Sep 2021 – Mar 2025**

Bachelor of Science in Computer Science | GPA: 3.71 *Irvine, CA*

Relevant Coursework: Data Structure Implementation and Analysis, Database Management, Machine Learning and Data-Mining, Searching Systems, Human Computer Interaction, System Design, Test-driven Development

Honors: Dean's Honor List (6 quarters) **Sep 2021 - Jun 2024**

Technical Skills

Programming Languages: Python, HTML, CSS, JavaScript, TypeScript, Java, Scala, SQL, C++
Frameworks/Libraries: Streamlit.py, React.js, Node.js, Angular.js, TensorFlow, PyTorch, Pandas, Numpy, Keras
Technologies: Git, CI/CD, Linux, REST API, Docker, Kubernetes, MySQL, PostgreSQL, SQLite, MongoDB, WordPress, Google Cloud Platform Computing Service, AWS EC2

Experience

Software Developer (GitHub) **May 2024 – Present**

Google Summer of Code *Remote*

- Led the design and implementation of an interactive web interface for an open-source Python library Selector using **Streamlit**, enabling medical chemists with minimal programming experience to efficiently perform data analysis.
- Set up a **CI/CD** pipeline with **GitHub Actions** to automate the process of building **Docker** images and deploying them to DockerHub and **HuggingFace**, enabling thousands of external users to navigate chemical space with support for various file formats like SDF, SMILES, and InChi.
- Conducted comprehensive design and code reviews to align with organization best practices and architecture.

Undergraduate Researcher (Demo) **Jul 2023 – Present**

Information Systems Group, UC Irvine *Irvine, CA*

- Directed a team of **4** developers in building **SQLRewriter**, a web platform enhancing the QueryBooster framework by facilitating community-driven SQL query optimization and discussion.
- Designed and developed responsive user interfaces using **Next.js**, resulting in a **20%** increase in user engagement.
- Built and managed back-end services with **Flask** to ensure reliable data storage and retrieval.
- Conducted data migrations and database schema changes to optimize application performance and maintain compatibility with changing business requirements.
- Deployed the application using **Docker** and **Kubernetes**, ensuring high availability and scalability across production environments, supporting thousands of active users with minimal downtime.

Teaching Assistant **Sep 2023 – Jun 2024**

UCI Course, ICS 33: Intermediate Programming with Python *Irvine, CA*

UCI Course, ICS 31: Introduction to Python Programming

- Co-hosted 80-minute lab sessions three times a week, providing coding and theoretical assignment support to over **150** students in an engaging and inspiring way.
- Initiated innovative teaching methods to assist students with advanced topics such as **Asymptotic Analysis**, **Databases**, and **Functional Programming**, resulting in **93%** of students passing with a C or higher.
- Demonstrated excellent communication and teamwork skills by analyzing student learning gaps with other staff members, boosting the instructor's RateMyProfessor ratings by **20%**.

Projects

Fablix | *Java, JavaScript, HTML, CSS* **May 2024**

- Developed a full-stack web application to facilitate online movie browsing and transactions, utilizing a **MySQL** database and **Apache Tomcat** server hosted on **AWS EC2**.
- Enhanced security through **HTTPS**, **reCAPTCHA** integration, and use of **PreparedStatement** to prevent SQL injections, ensuring robust data protection.
- Input **360,000+** movie information through **SAX XML parsing** and developed advanced search capabilities, including full-text search, autocomplete, and fuzzy search, improving search speed by **40%**.
- Set up JDBC **connection pooling**, and configured **load balancing** to distribute database load, increasing system availability by **50%**.
- Deployed the application in **Docker** containers, tested on AWS for seamless containerization and scalability, and configured a **Kubernetes** cluster with automated **master-slave** MySQL setup on **10** instances, utilizing **AWS S3** bucket for persistent state storage, reducing deployment downtime by **80%**.
- Utilized **JMeter** to conduct performance testing on the Kubernetes cluster, measuring the system's throughput under different configurations, optimizing the application's responsiveness and scalability by **30%**.