FEIYANG WANG

(323) 677-9366 | fyangwong@gmail.com | linkedin.com/in/fy-wang | github.com/fy916 | Los Angeles, CA 90010

EDUCATION

University of Southern California (GPA: 4.0/4.0) — Los Angeles, California
Master of Science, Computer Science — *January 2024 - May 2026 (Expected)*

University of Nottingham (GPA: 3.98/4.0, Top 1%) — Nottingham, United Kingdom
Bachelor of Science, Computer Science — *September 2019 - July 2023*

TECHNICAL SKILLS

- **Relevant Courses** (All Courses Completed with **A** Grades)**:**
  Algorithms & Complexity Analysis | Web Development & User Interfaces | Software Engineering & Object-Oriented Programming
  Machine Learning & Computer Vision | Databases & Data Structures | Operating Systems & Concurrency (C++/Linux/Ubuntu/Kali)
- **Web Dev**: Python (Flask, Django), Java (Spring Boot), TypeScript, JavaScript, Angular, React, Next.js, Node.js, Express, HTML, CSS, AJAX, RESTful API, jQuery, Bootstrap, Redux, Postman, Google Cloud Platform (GCP), Amazon Web Services (AWS), Microsoft Azure
- **Backend**: Unix/Shell Scripting, Python (PyTorch, TensorFlow, Scikit-Learn, OpenCV, Plotly, Pandas, PyQt, Beautiful Soup), Haskell, R, C, C++, C# (.NET, Unity), Java (Android Studio, Maven, Gradle), Distributed Systems, Relational Database (MySQL), MongoDB
- **Others**: Git, GitHub, JSON, XML, Jenkins, Docker, Spark, Kafka, Hibernate, CUDA, Conda, Power BI, Tableau, Wireshark, OpenGL

PROFESSIONAL EXPERIENCE

Sustainability Software and Data Intern | University of Southern California — Los Angeles, California
(TypeScript, Next.js, React, JavaScript, Node.js, jQuery, MongoDB, AJAX, GitHub, AWS, Agile Method) — *September 2024 - Present*

- Developed an interactive dashboard for visualizing sustainability data using **Next.js**, **React**, **Node.js**, **Bootstrap**, and **MongoDB**. Utilized **Agile** methodologies for rapid iterations. Used **GitHub** for version control and deployed the website on **AWS EC2**.
- Enhanced frontend experience using Promise-based **fetch** for **AJAX** API requests. Improved search handling with debouncing. Analyzed and optimized runtime efficiency using **Lighthouse** and **Chrome DevTools**. Reduced page load times by 75%.
- Designed comprehensive **Unit Tests**, Integration Tests, and Postman API Tests. Achieved high code coverage for core components. Proactively resolved critical bugs before deployment and maintained a **Lighthouse** performance score above 90.

Software Engineer | Whale Tech — Hangzhou, China
(Java, Spring Boot, REST API, Gradle, Jenkins, CI/CD, Kafka, Hibernate, MySQL, Microservices) — *July 2023 - November 2023*

- Contributed to an AI-driven conversational Retail Analysis Engine for thousands of partners using **Java Spring Boot**. Built **REST APIs** with **OAuth 2.0** protocol for seamless front-end and back-end integration. Utilized **Gradle** and **Jenkins** for **CI/CD** pipelines.
- Integrated Apache **Kafka** to support real-time, high-throughput data streaming from multiple retail and operational data sources. Used **Hibernate ORM** to optimize JDBC interactions with **MySQL** database. Delivered components one week ahead of schedule.
- Developed **microservices** to prefilter and aggregate user profiles using tokenized extraction, data normalization, and deduplication. Delivered preprocessed data to cloud **machine learning engines** via **REST APIs**. Reduced overall response time by 30%.

Software Engineering Intern | BGI Genomics (Top-Tier Bioinformatics Company) — Shenzhen, China
(Python, PyTorch, Flask, Docker, Jenkins, CI/CD, DevOps, Distributed System, C++, React, PowerBI) — *June 2022 - September 2022*

- Contributed to a machine learning-based single-cell analysis platform using **Python, PyTorch,** and **Flask**. Designed and implemented **30+ REST APIs** for interactive graph generation and advanced biological analysis algorithms.
- Applied **DevOps** principles to the application. **Containerized** the system with **Docker**, implemented the **CI/CD** pipeline using **Jenkins**, and automated the testing and deployment to the production server.
- Engineered a dynamic **work scheduling system** with **Preemptive Scheduling** to maximize **concurrent** performance across **distributed systems** by evaluating resources and real-time machine loads. Achieved a 50% reduction in average task wait times.

SOFTWARE ENGINEERING PROJECTS

Machine Learning-Based Video Analysis Toolbox | *(Python, PyQt, OpenCV, PyTorch, Docker, GitLab, Agile, Multi-Processing)*

- **Led** a **team of six** to develop computer vision software for container tracking and management at Ningbo Zhoushan Port, the **world's third-largest** port. Implemented algorithms including Person Re-Identification, Face Anti-Spoofing, Object Tracking, etc.
- Adhered to **Agile** methodologies and utilized **Python**, **PyQt**, **OpenCV**, and **PyTorch** for development. Designed modular and reusable code structures following **SOLID** principles. Utilized **GitLab** for version control and **Docker** for containerization.
- Developed a **PyQt**-based **GUI** using **MVC** architecture for real-time monitoring and analysis of multiple camera feeds. Utilized **QThreads** to manage **concurrent GPU-accelerated** analysis. Used **multiprocessing pipelines** for inter-process communication.

Game Bubble Bobble | *(C#, Java, JavaFX, Maven, Javadoc, GitHub, Unity, WebGL, MVC Architecture, Design Patterns)*

- Reconstructed a 2D Java game using **JavaFX** and **MVC** architecture. Applied **design patterns** such as Facade, Decorator, Template, Strategy, State, and Factory methods. Managed the project using **Maven** and documented updates using **Javadoc**.
- Migrated the game to **Unity**. Upgraded to a 3D multiplayer version with improved graphics and user experience. Used **C#** for development, deployed it as a **WebGL** build, and managed version control through **GitHub**.

Android Fitness Tracker | *(Java, Android Studio, Google Cloud API)*

- Developed an **Android app** for tracking workouts. Integrated **Google Maps API** for real-time trajectory mapping. Used **Android Services** for background data recording. Implemented **Content Providers** for storing and optimizing workout data points.