

ZHIXI (JOY) XU

☎ 434-326-8383 ✉ zhixi.xu.joy@gmail.com 📍 San Jose, CA, 95113

Education

Northeastern University

Sep 2023 – Expected May 2026

MS in Computer Science, GPA: 4.0/4.0

San Jose, CA

Relevant Courses: Object Oriented Design, Algorithms, Data Structures, Cloud Computing, Discrete Structures, Linear Algebra

Technical Skills

Languages: Java, Go, Python, JavaScript/TypeScript, C, C++, Swift, HTML/CSS

Frameworks: Spring Boot, Spring Cloud, Node.js, Express, Django, React, Next.js, SwiftUI, MyBatis, TypeORM

Databases & DevOps: MySQL, PostgreSQL, MongoDB, Redis, Elasticsearch, Docker, Kubernetes, Jenkins, CI/CD

Cloud & Tools: AWS, Git, REST APIs, RabbitMQ, BullMQ, Redux, Maven, JUnit, Figma, Postman

Professional Experience

curaJOY

Apr 2024 - July 2024

Fullstack Software Engineer Intern

- **Delivered** a mental assessment scoring and triggering system across 3 microservices using **Express** and **Node.js**, enabling automatic triggering through a **BullMQ-driven message queue** and **reducing maintenance efforts by 80%** through dynamic scalability improvements.
- **Led the development and implementation** of **REST API** endpoints in **TypeScript** for a gamification feature, **enhancing user engagement** through dynamic gem accumulation and reward systems. Designed and implemented the database schema using **PostgreSQL** and **TypeORM**, managed with **Liquibase** for versioning.
- **Collaborated** in an agile team to implement a responsive frontend using **Next.js**, **Redux**, and **Tailwind CSS**, **improving user navigation** and feature accessibility with progress tracking, filtering, and sorting functionalities.
- **Optimized the team's onboarding pipeline** in a fast-paced environment by streamlining documentation for environment setup, test configuration, and maintenance, reducing onboarding time for new team members from one week to a few hours.

Cloud Native Computing Foundation(CNCF)

July 2024 - Present

Open Source Developer

- **Enhanced system security** within the **Knative Eventing** platform on **Kubernetes** by implementing robust ingress-side access control for brokers, applying OIDC-based authentication and authorization policies.

Selected Projects

Distributed Cache System | Go, Caching, Distributed System

- Implemented a distributed cache system in **Go**, modeled after **groupcache**, supporting **in-memory** and **HTTP-based caching** for faster data retrieval.
- Developed an **LRU cache eviction policy** to manage memory, ensuring high cache hit rates and optimized resource usage.
- Built a **consistent hashing mechanism** to **distribute cache keys evenly** across nodes, achieving load balancing and enhancing scalability and fault tolerance in a distributed environment.
- **Integrated Protocol Buffers (Protobuf)** to improve inter-node communication efficiency and reduce latency.
- Ensured system robustness through **unit testing**, validating functionality, preventing cache stampede, and maintaining reliable operation under **high concurrency**.

LittleRPC | Go, Remote Procedure Call

- Developed a **lightweight RPC framework** in Go, emulating Go's net/rpc with additional features like protocol exchange, registry, service discovery, load balancing, and timeout processing.
- **Enhanced data transmission efficiency** through message serialization using Gob and JSON
- **Built a high-performance client** with asynchronous and concurrent call support, integrating error handling and timeout mechanisms.
- Added **HTTP support** and a simple **service registry** for dynamic service discovery and robust load balancing across servers.

ShopFlex | Java, Full Stack Web Application, AWS

- **Refactored** a monolithic e-commerce platform into a scalable **microservice-based architecture** using **Java** and **Spring Cloud**, with separate services for user management, orders, payments, and more. Combined **relational (MySQL)** and **nonrelational (MongoDB)** databases to persist and manage data from different services
- **Integrated AWS services** for a robust, serverless architecture, utilizing **AWS Lambda** for real-time data processing, **S3** for scalable and secure storage of product images, and **CloudFormation** for automated infrastructure management.
- Enhanced system performance and resilience by implementing **Elasticsearch** for optimized search (**reducing query time by 80%**), **Redis** for caching frequently accessing data, and **RabbitMQ**, along with **AWS SNS** and **SQS**, for robust inter-service communication and fault tolerance.
- **Automated CI/CD pipelines** using **Jenkins** and **GitHub Actions**, enabling seamless deployment to AWS environments, and ensuring high availability and scalability with **AWS EKS** and **Docker** containerized services.