Qiyuan(Vincent)Zhu

Tel: 2154303398 | Email: qiyuanzhu522@gmail.com | https://www.linkedin.com/in/qiyuan-zhu-6171b9249/

EDUCATION

Northeastern University
Master of Computer Science (GPA: 3.9/4.0) — Sep 2023-May 2026

University of Pennsylvania
Master of Public Administration (GPA: 3.8/4.0) — Sep 2022-May 2023

Beijing Foreign Studies University
Bachelor of Arts in Portuguese Studies — Sep 2017-Jun 2021

TECHNICAL SKILLS

Programming Languages: Java, JavaScript/TypeScript, Python

Frameworks/Tools: Backend - Springboot/Node.js, ORM – Hibernate/Mybatis; Micro Services; API-SOAP/REST/RPC, Converter - Jackson2; Dependency – Maven/Gradle; Git – GitHub, Bitbucket; Docker; Linux - Command, CronJob; CI/CD - AWS CodeCommit, CodeBuild, ECR, CodeDeploy; Message Queue - Kafka, AWS Kinesis/SNS/SQS; Relational Database - MySQL, PostgreSQL, AWS RDS； NoSQL - MongoDB, AWS DynamoDB; Serverless - AWS Lambda; Cache - Redis; Unit Test - Junit; Monitoring & Dashboard - Splunk

Cloud Platform: AWS(S3, EC2, RDS, DynamoDB, Lambda, Kinesis, SNS, SQS, RedShift, Glue, IAM, EMR, CodeCommit, CodeBuild, ECR, CodeDeploy etc.), Azure, GCP

Relevant Courses: CS5004：Object-Oriented Design and Analysis, CS5200: Database Management Systems, CS5610: Web Development, CS6620:Fundamentals Cloud Computing, CS 6650 Building Scalable Distributed Systems

PROFESSIONAL EXPERIENCES

Shawk Solution
Software Engineer Intern — Aug 2025-Present
- Built a full-stack education platform using **React**, **TypeScript**, and **Tailwind CSS**, delivering a responsive UI that improved page load speed by **30%** across devices.
- Implemented secure authentication with **JWT** and **bcrypt**, integrating **WeChat QR login**, which increased user sign-up rate by **20%**.
- Designed and deployed a **PostgreSQL** database schema, optimizing queries via **Spring Data JPA** and **Hibernate ORM**, which reduced average query latency by **25%**.
- Developed **Java Spring Boot** RESTful APIs with **90%** unit test coverage, integrated **GitHub Actions** CI/CD and **AWS Cloud Watch** monitoring, cutting regression bugs by **30%** and improving deployment efficiency.

Artisk Inc.
Software Engineer Intern — Feb 2025-Apr 2025
- Integrated **ChatGPT API** with advanced **prompt engineering** to create an interactive AI-powered chatbox, boosting user engagement by 35% in pilot testing.
- Built scalable web apps using **Java**, **Spring Boot**, and **Spring MVC**, managing data with **PostgreSQL** and **Spring Data JPA**, and reduced server response time by **20%**.
- Developed **JUnit test** coverage (**85%+**) and automated **CI/CD pipelines** with **GitHub Actions**, ensuring stable releases and faster iteration cycles.
- Engineered a **multimodal** chatbot platform by integrating **OpenAI**, **Claude**, and **Coze APIs**, enabling **image generation** and **logo design** in addition to text interactions.

PROJECT HIGHLIGHTS

Product Information Storage and Publisher System — May 2025-Aug 2025
- Built a new product publishing system connecting **Vue.js** with **MySQL**, **AWS SNS/SQS**, and **3 Spring Boot consumers**, automating data flow across systems.
- Developed an end-to-end **data pipeline** with scheduled **Cron jobs** and **fan-out messaging**, enabling daily synchronization to multiple consumers without manual intervention.
- Deployed **Spring Boot** apps on **AWS EC2**, configured **Spring AOP** logging, and integrated **Splunk Dashboards** with **AWS CloudWatch** for real-time monitoring.
- Eliminated manual nightly scripts, saving engineers **30 minutes** per day (including weekends) and improving operational efficiency.

Customer Service System — Dec 2024-Apr 2025
- Designed a backend system using **Java Spring Boot**, introducing modular **controllers**, **services**, and **repositories** layers to ensure scalability and maintainability.
- Implemented four modules: **Customer Info**, **Order Processing**, **Return/Refund/Reship**, and **Logging**, employing **Spring AOP** for centralized logging and performance monitoring.
- Deployed the application to AWS using **CodeBuild**, **ECR**, and **CodeDeploy**, practicing **CI/CD automation** and container-based deployment, reducing deployment from **8** manual steps to **a one-click release**, cutting deployment time by **80%** and minimizing errors.