



SwimmingLiu (刘永杰)

Intention: Java Backend / AI Application Engineer

📞 +86 15982617349 📩 SwimmingLiu@outlook.com 🌐 https://swimmingliu.cn

👤 SwimmingLiu

📧 SwimmingLiu

𝕏 SwimmingLiudev

Summary

Possesses a conscientious and practical work attitude with a strong sense of responsibility and teamwork skills. Proficient in reading English technical documents (**CET4 512 / CET6 513**). Passionate about **vibe coding** and efficient **prompt engineering**, achieving **87%** of AIGC coverage. Active open-source contributor with **14k+ stars** in Github / Gitee.

Education

ZheJiang Sci-Tech University (Computer Science) Master

2023.09 ~ 2026.06

Published one SCI Q2 paper, participated in **6 national research funds**, received the **National Third Prize** in the 'Huawei Cup' Graduate Mathematical Modeling Competition, and awarded a **First-Class Academic Scholarship**.

Wuhan University of Engineering Science (BigData) Bachelor (4.21 GPA)

2019.09 ~ 2023.06

11th LanQiao Cup, Hubei Provincial Second Prize; National College Student English Competition, Second Prize; School-Level Outstanding Student

Experience

Gildata Information Technology Co., Ltd. (Hundsun Electronics Group)

2025.05 ~ 2025.11

AI Application Engineer (Java)

Contributed to the development of **DataCopilot (DC)**, an **AI platform** that automates the processing of financial data (public opinions, bonds, wealth management). This achieved a **fully automated AI production workflow**, **improving efficiency by 37%** over the original process. The platform features **AI workflow orchestration**, **prompt management**, **dataset management**, entity linking, and manual data correction. It also integrates support services, including log monitoring, traffic monitoring, and disaster recovery. The project uses **CI/CD** for automated deployment on Kubernetes clusters, enabling dynamic service scaling.

Tech Stack: SpringCloud, Gateway, Nacos, Dubbo, Sentinel, Redis, MySQL, MongoDB, LiteFlow, Nginx, Kafka, Docker, XXL-JOB, Spring-AI-Alibaba-Graph

- AI Workflow Orchestration:** Designed and implemented an AI workflow orchestration engine. The engine supports dynamic compilation of a front-end Canvas into an executable **StateGraph**. It provides adaptation for standardized nodes (**LLM/RAG**), **interactive nodes**, and composite nodes (e.g., **Loop**, **Batch**, **Sub-Workflow**). The system includes validity checks for workflow links, such as **connection relationships**, **loop dependencies**, and **nested process validation**. It supports both synchronous and asynchronous execution modes. A **persistent global state** machine ensures **reliable tracking** and **fault recovery** for complex workflows. The system enables **dynamic service discovery**, **registration**, and multi-version management, processing approximately **800,000** public opinions data entries daily.
- Asynchronous Dataset Import:** Designed and implemented a Kafka-based **asynchronous data parsing** system. Used **sharded uploads** to decouple file upload and parsing tasks, ensuring the orderly processing of file shards. Created a system task table to track processing status, which ensures **idempotent check** and provides **self-healing capabilities**.
- Prompt Management:** Designed and developed a **full-lifecycle prompt management** system. Established a version control process covering prompt template development, testing, and review. This system supports both **offline** and **online evaluation** modes, reducing the iteration cycle from **one week** to **same-day** deployment while ensuring no impact on the production workflow.

HyperChain Technology Co., Ltd.

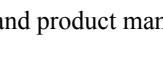
2025.03 ~ 2025.05

Java Backend Engineer

Developed and maintained the data dashboard, mini-programs (Alipay/WeChat), and management backend for the Carbon Inclusive project (Zhejiang/Nanjing/Gansu/Ningxia). This project uses a carbon credit system to quantify low-carbon behaviors and incentivize public participation. Managed the end-to-end development and deployment of the Gansu project, ensuring system stability and data consistency for the enterprise platform.

Tech Stack: SpringCloud, Gateway, Nacos, Dubbo, Sentinel, Redis, MySQL, Nginx, Kafka, Docker, XXL-JOB

- Implemented **scheduled data synchronization** using the XXL-JOB framework. This system performed **batch incremental synchronization** of carbon credit and emission data from scenario sources (e.g., Ant Forest, Ele.me, Cainiao), processing an average daily volume for **1.6 million** users.
- Optimized response speed** by implementing a distributed cache architecture (Redis + DB) to resolve high-concurrency query bottlenecks for the dashboard. This approach also solved **data consistency issues**. The optimization increased overall interface QPS from **482** to **2.32k** and maintained an average response time under **60ms**.
- Developed a user-side **low-carbon leaderboard** using **ZSet** structure in Reids. This system supports **QR code invitations**, allowing new users to contribute to city rankings. The leaderboards are segmented by city and district, which effectively resolves potential **Big Key** issues in Redis.



Projects

OrangeMust (20.2K for maximum Daily active users)

2023.08 ~ now

Project Introduction : **OrangeMust** is an AI-powered tool mini-program modeled after Tencent's **Bang Xiao Mang**. The program provides login verification, **AI task assistance** (AI chatbot, image generation, document conversion), important date reminders, key purchases, and feedback. The admin portal supports order refunds, log recording, and product management. Achieved over **173.9k cumulative users** and **800+** average **daily active users**.

Tech Stack: SpringBoot, SpringMVC, Redis, MyBatis-Plus, MySQL, Nginx, RabbitMQ, Redisson, Spring-AI, Spring-AI-Alibaba

Work Content

- Implemented an **asynchronous seckill sale** for system currency using Redisson **distributed locks**, preventing **overselling** and enforcing a **one-order-per-user** limit; achieved **concurrent WeChat payment ordering** (QPS of **1.76k**).
- Built a dialogue streaming channel using SSE. Designed a **dynamic API Key pool** using Redis (Hash + ZSet) with **fair priority scheduling**. Used Lua scripts to atomically manage key status, concurrency, and quotas under **high traffic**. Developed **health detection**, **dynamic cooling**, and **alert systems** to enable **automatic key isolation** and **manual replenishment**. Combined **global** and **user rate limiting** with dedicated thread pools to handle peak traffic.
- Developed core AI chatbot functions, implementing **RAG** (Retrieval-Augmented Generation), intelligent **multi-format attachment parsing**, and **multi-turn conversation management**. Optimized the AI dialogue API response time to approximately **5.5 seconds** using **context engineering** and **system prompt optimization**, improving response speed by **26%**.

Skills

- Familiar with Java; proficient in Collections and JUC concurrency. Understands the underlying principles of **HashMap**, **TreeMap**, and **ConcurrentHashMap**.
- Proficient in MySQL; understands mechanisms such as **transactions**, **indexes**, **locks**, and **MVCC**. Familiar with **master-slave synchronization** and index optimization.
- Skilled in common Redis data structures; proficient in **cache update strategies**. Understands and can resolve issues such as **Cache penetration**, **Cache avalanche**, and **Cache Hotspot Invalid**. Knowledgeable of **distributed locks**, **master-slave synchronization**, and split-brain problems.
- Proficient in frameworks such as SpringMVC, SpringBoot, and MyBatis-Plus; understands core principles like IoC and AOP. Experienced with middleware including **Redis**, **Kafka**, and Nginx. Capable of independently developing and deploying projects (**Docker**).
- Proficient in common **Linux commands** and basic Shell programming. Familiar with backend service deployment and go-live processes. Served as the **laboratory server administrator**, responsible for daily server maintenance and management.
- Familiar with the process of building, training, and deploying open-source LLM models (e.g., gpt-oss, DeepSeek-V3.2-Exp). Understands the MCP protocol. Experienced in building Agent applications using platforms such as Dify, Coze, and Agent Builder.