Wesley Sum

github.com/wesleysum wesleysum.com linkedin.com/in/wesleysum wesleysumsoftwaredev@gmail.com 657-720-9801

Summary

Software engineer skilled in building high-performance systems and scalable platforms. Proficient in full-stack development, cloud technologies, and optimizing system performance. Experienced in projects like distributed KV storage, gaming platforms, and AI-powered document processing.

Education

Virginia Tech, College of Engineering Expected: Spring 2025

Bachelor of Science in Computer Science ***Blacksburg, Virginia***

Courses: Data Structures and Algorithm, Computer Organization, Object-Oriented Programming, Software Design and Engineering, Computer System, Mobile Software Development, Introduction to Artificial Intelligence

Technical Skills

Languages: Java, Python, JavaScript/TypeScript, C, Kotlin
Frameworks & Tools: React, Vue.js, SpringBoot, Node.js/Express, Tailwind CSS, MyBatis, JUnit, Resilience4J, Jenkins, CI/CD, Git, TestNG
Cloud & Databases: AWS (RDS, S3, DMS), Docker, Kubernetes, Redis, RabbitMQ, Kafka, PostgreSQL, MySQL, MongoDB, Firebase

Experience

CodeDay Blacksburg, VA

Software Engineer Intern *June 2024 - Aug 2024*

- Developed a dynamic frontend using **Vue.js** and **Element UI**, creating an intuitive and visually engaging user interface for enhanced usability.
- Streamlined database operations with **Spring Data JPA**, reducing query times from **500ms to 40ms** by leveraging **Redis** for caching high-traffic data, significantly lowering **MySQL** query load.
- Architected and executed a seamless **MySQL-to-AWS RDS** migration plan using **AWS DMS**, achieving **99%** system availability and maintaining data integrity via a dual-write strategy.
- Slashed storage costs by **70%** through the migration of image and video data to **AWS S3**, implementing lifecycle policies and version control for automated management.
- Achieved **95%** unit test coverage and automated deployment pipelines using **Jenkins CI/CD**, ensuring continuous testing and deployment for increased system reliability.

Projects

Distributed KV Storage System GitHub

- Developed a high-performance system handling over 8,000 ops/sec with sub-millisecond latency, achieving 99.99% uptime through **Raft** consensus with leader election and log replication.
- Implement the **Raft** consensus algorithm, with core functions such as Leader election, log replication, and snapshot update.
- Based on the consistent hashing architecture, the data is partitioned into Shards and can be migrated in multiple **Raft** Groups.
- Integrated support for various storage engines (**RocksDB**, **B-tree**, **hash tables**) to optimize for different I/O models.
- Enhanced performance by 40% and efficiency by 35% through optimizations like **Asynchronous Apply**, **ReadIndex**, **FollowerRead**, and **Prevote**.

GameHub - High-Performance Gaming Platform GitHub

- Built a high-performance gaming web platform with a backend built on **SpringBoot**, **MyBatis**, **Redis**, **AWS RDS**, and **RabbitMQ**, and a frontend built with **React** and **Node.js**.
- Supported 6,000 concurrent users and handled 4,000 transactions/sec with a P99 latency of less than 1 second.
- Implemented database sharding and table partitioning with **Amazon RDS**, improving query performance through indexing and partitioning.
- Cached data in **Redis**, reducing query latency from 200ms to 30ms using TTL for data expiration management.
- Optimized **Kafka** partitioning and asynchronous message processing, increasing throughput from 300 to 1,300 messages/sec.
- Secured user authentication with **JWT**, supporting multiple login methods to ensure user data security.
- Built **CI/CD** pipelines with **Jenkins** and added load testing and performance validation for 3 services and 21 interfaces using **TestNG**.

DocuSync AI: Full Stack Intelligent Document Processing System Demo

- Developed a scalable, AI-powered document processing tool using **React**, **Tailwind CSS**, and **TypeScript**, reducing document handling time by 20% and improving workflow efficiency for knowledge workers.
- Engineered RESTful APIs via **Next.js** API routes for high-performance request handling, optimizing backend processes and increasing data retrieval speed by 40%.
- Integrated **OpenAI's GPT-3.5 API** and **Langchain** technologies, resulting in a sophisticated AI Agent capable of document loading, splitting, storage, retrieval, and output functionalities.

SandBox: Scalable CMS for Content Management Demo

- Designed and implemented backend architecture using **Node.js** and **MongoDB**, enabling efficient and secure CRUD operations for a scalable content management platform.
- Enhanced platform security by implementing user authentication with **Bcrypt** and **JWT**, ensuring data protection and reducing unauthorized access incidents.
- Leveraged **AWS S3** for cloud storage, optimizing data handling and reducing retrieval latency
- Devised advanced search, tagging, and **recommendation** algorithms, boosting user engagement and improving content discoverability.