

NIKHIL GARIGIPATI

Backend Software Engineer | Java · Spring Boot · Distributed Systems | Bangalore, India
nikhilgarigipati433@gmail.com | +91 86398 24414 | [LinkedIn](#) | [GitHub](#) | [LeetCode](#) | [Portfolio](#)

PROFESSIONAL SUMMARY

Backend engineer specializing in distributed Java systems — designed and engineered 2 production-grade microservice systems with **Kafka**, **Redis**, and **Spring Boot**. Experienced in **Data Structures & Algorithms**, **Object-Oriented Programming (OOP)**, **System Design**, **REST API design**, **JWT-based authentication**, **CI/CD pipelines**, and containerized deployment.

TECHNICAL SKILLS

Languages	Java 17, SQL
Frameworks	Spring Boot 3.x, Spring Security, Spring Data JPA, Hibernate
Databases	MySQL, PostgreSQL, Redis
Distributed Systems	Apache Kafka, Microservices
API & Security	REST APIs, JWT, Swagger/OpenAPI
DevOps & Testing	Docker, Docker Compose, Git, GitHub Actions, JUnit, Mockito, JMeter, TDD
Core Concepts	Data Structures & Algorithms, OOP, System Design, Multithreading, SOLID, Design Patterns

PROJECTS

Event-Driven Order Processing System | Spring Boot · Kafka · Redis · Microservices · Docker · MySQL

Oct 2025

github.com/nikhilg433/order-processing-system (README includes full architecture diagrams)

- Architected **3 independently deployable microservices** (Order, Inventory, Payment) processing **10,000+ order events** via **Apache Kafka** — zero REST calls; each service owns its own MySQL database
- Implemented **Choreography-based Saga** with full compensating transactions and **idempotent Kafka consumers** across 3 consumer groups with DB-level duplicate detection — zero duplicate processing across 10,000 simulated events, Kafka consumer lag = 0, and automatic rollback with no central coordinator on payment failures
- Applied **Redis Cache-Aside Pattern** on inventory lookups — **58% latency reduction** (12ms → 5ms); sustained **892 req/sec** with 0% error rate across 10,000 requests under 500 concurrent threads (Apache JMeter)
- Secured REST endpoints with **Spring Security** — stateless JWT authentication, BCrypt password hashing, RBAC (Admin/User roles); documented with Swagger/OpenAPI and live JWT authorization
- Automated with **GitHub Actions CI/CD** — automated build, test, and Docker Compose validation on every push; 19 Mockito unit tests covering all Saga failure paths across an **8-container** Docker stack

Distributed Job Scheduler | Spring Boot · Quartz · Redis · Kafka · MySQL · Docker

Jan 2026

github.com/nikhilg433/distributed-job-scheduler (README includes full architecture diagrams)

- Built to handle **1,000+ concurrent scheduled jobs** across 2 clustered instances with **Redis SETNX distributed locking** for exactly-once execution — load tested with Apache JMeter (1000 threads, 0% error rate); verified 0 duplicate executions under race conditions; fully recoverable after node crashes with zero data loss via Quartz JDBC persistence
- Built **Kafka event streaming pipeline** across 5 topics (scheduled, started, completed, failed, cancelled) — real-time job lifecycle visibility with full audit trail persisted to MySQL
- Engineered **exponential backoff retry** ($\text{delay} = \text{base} \times 2^n$) with configurable max attempts — prevents thundering herd on failure bursts
- Secured REST APIs with **Spring Security 6** — JWT authentication, RBAC (Admin/User roles), BCrypt password hashing; documented with Swagger/OpenAPI
- Containerized with **Docker Compose** (2 app instances + MySQL + Redis + Kafka); fully deployable via single `docker-compose up`

EDUCATION

Vellore Institute of Technology (VIT-AP)

B.Tech in Computer Science & Engineering, CGPA: **8.54/10**

Sep 2022 – May 2026

Amaravathi, India

Relevant Coursework: Operating Systems, Distributed Computing, DBMS, Computer Networks, Cloud Computing

CERTIFICATIONS & ACHIEVEMENTS

MongoDB Associate Database Administrator — MongoDB, Jul 2025 | Hack4Health Hackathon (Participant) — IIIT Delhi, Oct 2025 |
LeetCode: **250+** Problems Solved