Raju Kumar Sharma

Phone: +91-8210574728 | Email: rk.sharma7794@gmail.com |
LinkedIn: linkedin.com/in/raju-kumar-sharma-9a2737169 | GitHub: github.com/RAJUPREM

Backend Engineer — Java — Spring Boot — System Design
Microservices — Multithreading — API Security

Professional Summary

Results-oriented Backend Engineer with 5+ years of experience in designing, developing, and deploying scalable, secure, and high-performance backend systems. Proficient in Java, Spring Boot, RESTful APIs, and microservice architecture with a solid foundation in data structures, algorithms, and system design principles. Experienced in building distributed, event-driven systems using Kafka, implementing fault tolerance, load balancing, caching, and secure authentication (JWT). A proven problem solver with 200+ LeetCode problems solved, applying real-world algorithmic techniques such as DFS, BFS, recursion, and dynamic programming to optimize production code.

Technical Skills

Languages: Java, SQL
Backend: Spring Boot, REST APIs, Microservices, Kafka, Servlets/JSP, Event-Driven Architecture
Cloud & DevOps: Docker, Kubernate, GitHub Actions, AWS
Security: JWT, Secure API Design
Databases: Oracle ,Hibernate ORM
Tools: IntelliJ, Postman, SwaggerUI, Git, Jira, SonarQube
Concepts: OOP, Data Structures & Algorithms, Design Patterns, System Design, Distributed Systems , Stream API

Experience

Application Engineer May 2022 – Present
Oracle *Bangalore, India*

- Designed and implemented microservices-based architecture using Spring Boot and Hibernate, improving scalability and modularization across large enterprise systems.
- Applied HLD/LLD principles, caching strategies (Redis), and resilience patterns (circuit breakers, retries) to enhance performance and fault tolerance.
- Applied system design principles (SOLID) and design patterns (Factory, Singleton, Observer) to ensure maintainable and extensible codebases.
- Ensured application security and observability by integrating JWT authentication, Zipkin tracing, and Prometheus monitoring.
- Developed microservices using Spring Boot and Hibernate ORM to streamline data management and enable modular scalability.
- Built and optimized RESTful APIs, reducing response latency by 60% through Redis caching and SQL query optimization.
- Integrated Feign Client for seamless inter-service communication and Eureka for reliable service discovery.
- Configured centralized configuration servers to streamline deployment and configuration management.
- Enabled lazy loading in high-traffic modules, reducing memory footprint and improving response times by 20%.
- Configured load balancing and rate limiting mechanisms to handle concurrent requests efficiently.
- Enhanced SQL analytics performance by implementing indexes and materialized views.
- Monitored and patched vulnerable dependencies proactively to maintain security compliance.
- Leveraged Core Java multithreading for concurrent applications and applied Java 8 Stream API for functional data processing.
- Developed and deployed enterprise web applications using Java/J2EE, Spring Boot, and Apache Tomcat with Maven builds and Git-based CI/CD.
- Developed microservices using Spring Boot/Helidon with comprehensive JUnit and Mockito test coverage across layers.

- Applied advanced design patterns (Factory, Singleton, Adapter, Builder, Dependency Injection) to improve scalability and maintainability.
- Implemented Kafka-based streaming pipelines for real-time communication and event-driven systems.
- Applied modular system design, caching, and load-balancing strategies for scalable, resilient services.
- Migrated 100+ microservices to Oracle Cloud Infrastructure (OCI), achieving seamless, scalable deployments.
- Led full microservice delivery lifecycle including system design, caching, JWT security, and distributed tracing on OCI.
- Containerized and orchestrated microservices using Docker and Kubernetes (EKS), enabling efficient scaling and high availability.
- Designed and deployed scalable, secure applications on AWS using EC2, RDS, Lambda, and CloudFormation for infrastructure automation.
- Applied algorithmic techniques (DFS, BFS, recursion, DP) and core data structures (Trees, Graphs, HashMaps) in backend optimization; solved 200+ LeetCode problems demonstrating practical problem-solving skills.

Project Engineer Nov 2020 - May 2022
Wipro Technologies *Bangalore*

- Developed a Kafka + Spring Boot notification system capable of handling 1,000+ concurrent users, ensuring real-time delivery of alerts and messages.
- Reduced average system latency from 2 seconds to 200ms by implementing caching and database connection pooling.
- Designed modular backend services following microservice patterns, enabling easier maintenance and faster deployment cycles.
- Collaborated with cross-functional teams to deliver production-ready features under tight deadlines, ensuring system scalability and robustness.
- Integrated Spring Security and JWT to build secure user authentication and authorization across internal APIs.

Projects

School Management System *(Java,Spring Boot, Hibernate, Spring Security, Microservice,Spring Cloud, Transaction)*Jan 202

- This is a school-based application system built using Spring Boot and Hibernate. This innovative system empowers schools to streamline their operations, enhancing the student and faculty experience. Our solution simplifies administrative tasks, such as enrollment, attendance tracking, and grade management, tuition charge tracking while providing real-time communication between teachers and students. By leveraging the power of Spring Boot and Hibernate, we've created a reliable, secure, and efficient platform that is making a positive impact on education management.
- https://github.com/RAJUPREM/Spring-Boot/tree/master/SpringBootBegining5

Mentor on Demand Microservices (JWT) security including API gateway *(Java,Spring Boot, Hibernate, Spring Secur*

- Any student came and searched for a mentor according to their requirement. This service is totally secured as every endpoint)API) is secured with role-based authentication, user, mentor, and admin need tokens to access their respective endpoints. I have Created 4 services (UserService, MentorTrainingCourseService, MentorOnDemandAuthenticationService, AdminService) and MentorOnDemandEurekaServerRegistry is to register the services and mentorapigate is to avail gateway to provide a single port for every service. I have used RestTemplte to have communication between services.
- https://github.com/RAJUPREM/MentorOnDemandMicroserviceJWT
- Postman Collection
  https://warped-sunset-217651.postman.co/workspace/New-Team-Workspace 22c295c8-4321-48ee-84cf-5fadf554a6a7/collection/30347421-8aee6af7-6500-4aab-878d-ca7b99883369?action=share&creator=30347421

Achievements

- FSGIU Pacesetter Award for FY25 Q3
- Solved 200+ LeetCode problems, applying algorithmic thinking to real-world backend performance tuning.
- Successfully migrated multiple microservices from GitLab CI/CD to OCI pipelines, improving release reliability and deployment efficiency.

Education

JIS College Of Engineering ,Kalyani Kolkata, India
B.Tech in Computer Science Engineering — CGPA: 7.99/10 *July 2016 – June 2020*