Pratik Chandlekar

Software Engineer

Mumbai, India | +91 9075467719 | pratikchandlekar280@gmail.com | https://pratik280.hashnode.dev/page/about |
https://www.linkedin.com/in/pratik280 | https://github.com/Pratik280 | https://pratik280.hashnode.dev/

Summary

Software Engineer with strong experience building production-grade microservices using Spring Boot and Spring Cloud primarily in the banking and payments domain. Designed secure, highly available systems with API Gateway security, peer-aware service discovery, and Nginx load balancing, and implemented schema-based multitenancy after evaluating database sharding trade-offs. Built real-time transaction monitoring systems backed by PostgreSQL, Redis, and Kafka, including Kafka KRaft cluster setup and producer/consumer integration. Contributed to CI/CD pipelines, containerized deployments on AKS, centralized logging, and observability. Actively participated in Agile delivery, mentoring juniors, conducting code reviews, and resolving production-critical issues. Actively publishes technical blogs on backend engineering and system design.

Experience

In-Solutions Global Ltd — **June 2022 - Present**

Software Engineer — Mumbai

Spring Boot API Development

- Developed production-grade Spring Boot REST APIs using Spring MVC, Spring Data JPA, Hibernate, and layered architecture, leveraging DTOs, ModelMapper, and Lombok for clean, maintainable code.
- Implemented consistent API response structures and global exception handling using @RestControllerAdvice, improving reliability, debuggability, and test coverage.
- Built transaction monitoring APIs and a React JS monitoring dashboard with real-time transaction status, using PostgreSQL → Redis sync via scheduled Spring Boot jobs to support low-latency dashboard APIs.
- Performed Kafka migration from ZooKeeper-based setup to KRaft mode, configuring a 3-node combined controller–broker cluster, tuning broker, controller, and server properties, and implementing Kafka producer and consumer APIs within Spring Boot microservices, including topic mirroring and cluster stretch scenarios.
- Designed and implemented payment-link generation APIs to route transactions dynamically across Lyra, TPSL, and PayU based on merchant configuration and routing rules.
- Created APIs handling Amex Target Transaction flows via SOCKS proxy, supporting POS and PG integrations for JPMC payment flows.
- Contributed to Merchant Dashboard development in Angular, enabling merchant-wise transaction views, transaction lookup, downloadable transaction reports, and visibility into active/inactive payment targets (Lyra, TPSL).

Multi-Tenancy & Database Architecture

- Evaluated database sharding for client data isolation, and after R&D and trade-off analysis, designed and implemented schema-based multitenancy using Hibernate JPA, enabling a single Spring Boot application to serve multiple clients with strict data isolation per schema and simplified operational management.
- Developed and optimized complex PostgreSQL join queries across multiple tables to efficiently retrieve transaction and payment data, working across development and production environments.

Spring Cloud Microservices Architecture & API Gateway

- Implemented a production-grade microservices architecture using Spring Cloud Gateway and a peer-aware Eureka service discovery cluster, with multiple Gateway instances deployed in a cluster and load-balanced via Nginx upstream configuration for high availability and fault tolerance.
- Resolved a critical production security issue by implementing an access-token validation filter in Spring Cloud Gateway, integrating with an RBAC authorization service (Spring Security), including bypass rules for internal and health endpoints, effectively preventing unauthorized public access to backend microservices.
- Configured Nginx as a reverse proxy with SSL termination, domain-based routing, rewrite rules, and upstream load balancing, routing external traffic securely to the Spring Cloud Gateway cluster and onward to internal microservices.

CI/CD, Containerization & Cloud Deployment (Jenkins, Docker, Helm, AKS)

- Built Jenkins CI/CD pipelines for Spring Boot microservices to compile JARs using Maven, build Docker images, and push versioned images to the Nexus Docker Registry, enabling a consistent and repeatable deployment workflow.
- Automated Helm chart versioning and deployments to Azure Kubernetes Service (AKS), and managed supporting infrastructure including Redis, Kafka, and PostgreSQL provisioned on Azure VMs to support microservice environments.
- Used Docker Compose to spin up isolated test environments with Kafka, Redis, databases, and application containers, enabling local and CI-level integration testing.
- Integrated centralized logging and observability using ELK (Elasticsearch, Logstash, Kibana) along with Prometheus and Grafana, streaming logs and metrics from multiple microservices to dashboards for client-specific monitoring and faster production issue analysis.

Skills

Core Java : Object-Oriented Programming (OOP), Collections, Exception Handling, Multithreading & Concurrency

Backend : Java, Spring Boot, Spring Cloud, Spring Security, Hibernate/JPA, REST API

Databases & Cache : PostgreSQL, Redis, SQL

DevOps & CI/CD : Jenkins, Docker, Kubernetes (AKS), Helm, Nexus, Docker Compose, Nginx

Cloud : Amazon Web Services (AWS), Azure

Messaging & Streaming : Apache Kafka (KRaft)

Observability : ELK Stack (Elasticsearch, Logstash, Kibana), Prometheus, Grafana

OS & Tools : Linux, Bash, Git, ChatGPT, Copilot

Practices : Agile/Scrum, Software Development Life Cycle (SDLC)

Frontend : Angular, React JS, Javascript, Typescript

Certifications

AWS Certified Cloud Practitioner — **June 30, 2024**

Amazon Web Services (AWS)

Education

Mumbai University — **June 2018 - May 2022**

Computer Science — Bachelor of Engineering

CGPA: 8.68