Yangzoom Pun Magar

Senior Backend Engineer – FinTech

Email | Phone | LinkedIn | GitHub

Professional Summary

Senior Backend Engineer with 6+ years of experience designing and maintaining Spring Boot–based microservices and event-driven systems in cloud-native environments. Specialized in Kafka, AWS ECS, and CI/CD automation using Jenkins. Experienced in large-scale migrations, security remediation, and acceptance test–driven development (ATDD). Strong collaboration and mentoring skills with a track record of improving system reliability and reducing deployment failure rates in FinTech ecosystems.

Technical Skills

Languages:	Java (8/17), Python
Frameworks:	Spring Boot, Spring Security (OAuth2), Spring Data JPA, Hibernate
Messaging:	Kafka
Databases:	PostgreSQL, Oracle, MySQL, MongoDB
Cloud & DevOps:	AWS (ECS, S3, CloudWatch, IAM), Docker, Jenkins, OpenShift
Testing:	JUnit, Mockito, Cucumber, ATDD
Build & SCM:	Maven, Gradle, Git, GitHub
Tools:	Splunk, Excel Tracker

Professional Experience

Software Engineer → Senior Backend Engineer | Company Name | Jan 2018 – Nov 2025

- Worked on Spring Boot–based microservices integrated with Kafka for internal data streaming, ensuring backward compatibility and stability during platform migrations.
- Led the Spring Boot upgrade from 1.5.x to 2.7.x for 80+ services, addressing dependency and transitive library vulnerabilities (reduced from over 2000 to fewer than 150) and ensuring compatibility with the latest framework versions.
- Contributed to the Kafka Streaming Data Platform (SDP) migration from v3 to v4, implementing required code changes, configuration updates, and testing strategies to align with new internal standards.
- Performed schema transformation and dataset registration using internal tools, carefully flattening complex legacy data structures to ensure compatibility with the new schema format and downstream processing.
- Wrote and maintained unit, functional, and acceptance tests using JUnit, Mockito, and Cucumber, following Acceptance Test–Driven Development (ATDD) practices to improve code reliability and regression detection.
- Enhanced CI/CD pipelines in Jenkins, configuring build and deployment stages, adding quality gates, and ensuring proper integration testing before deployment.
- Supported multiple stages of the release lifecycle: performing non-prod testing, validating results, and ensuring production deployments met internal readiness standards.
- Monitored application health and system logs using Splunk and AWS CloudWatch, identifying and resolving issues during and after releases.
- Collaborated with repository owner teams for PR reviews and release approvals, ensuring compliance with internal quality and security guidelines.
- Conducted knowledge transfer (KT) sessions to onboard team members, guiding them through SDP migration, testing, release verification, and dataset registration workflows.

- Used Excel-based trackers to record release progress, PR statuses, test results, and validation outcomes across multiple microservices.
- Implemented IAM role pre-checks and validation in deployment flows, improving release reliability and reducing failure rates from ~9% to below 5%.
- Developed and enhanced error-handling mechanisms in deployment pipelines to detect and isolate issues like health check failures and AWS configuration mismatches, improving troubleshooting speed and developer productivity.
- Worked with SREs to validate Route 53 regional failover between East and West AWS regions for ECS services, ensuring continuity and high availability.
- Maintained and enhanced the Gear application, which automated cloud deployments via Jenkins pipelines and AWS ECS, including:
- • Adding IAM role existence and permission validation before deployment.
- • Integrating automatic retrieval of CloudWatch log URLs for failed deployments to simplify debugging.
- • Incorporating Docker image vulnerability scans within PR checks for proactive security validation.
- Participated in end-to-end release testing and post-deployment verification, ensuring successful integration of new components into existing distributed systems.
- Applied Spring Security (OAuth2) for securing APIs and implementing access token validation for internal authentication.
- Collaborated with cross-functional teams to troubleshoot environment issues, test pipeline failures, and coordinate deployments across multiple regions and services.

Java Full-Stack Developer | Previous Company | Jan 2020 – May 2021

- Designed and developed microservices and REST APIs using Spring Boot and Spring Cloud, integrating them with Angular 8 front-end components for full-stack delivery.
- Implemented Spring Data JPA for database persistence and optimized queries to improve response time across PostgreSQL and MongoDB.
- Developed and maintained Kafka producers and consumers for asynchronous message exchange between microservices.
- Automated builds and deployments using Jenkins pipelines and containerized services via Docker.
- Deployed and managed applications on AWS EC2 and S3, configuring IAM roles and policies for controlled access.
- Performed load, stress, and performance testing using JMeter, identifying performance bottlenecks and improving throughput.
- Implemented Spring Security with OAuth2.0 for secure authentication and authorization of REST endpoints.
- Collaborated in Agile/Scrum ceremonies, participating in design discussions, sprint reviews, and peer code reviews.
- Worked closely with QA and DevOps engineers to streamline release automation and resolve environment-related issues.

Java Developer | Earlier Company | Jan 2018 – Dec 2019

- Developed backend modules using Spring MVC, Spring JDBC, and Hibernate, implementing DAO and service layers following clean architecture principles.
- Integrated Kafka for message streaming, including logic for failed message reprocessing based on offset tracking.
- Implemented RESTful and SOAP web services, ensuring reliable integration with internal and external systems.
- Used JUnit and Mockito for automated unit testing and integrated builds via Maven and Jenkins.
- Leveraged AWS SDK for S3 data storage and SNS notifications to integrate cloud-based communication within legacy systems.
- Implemented OAuth2-based authentication to secure REST APIs and manage access control.
- Performed debugging and production issue resolution using logs and monitoring tools, ensuring minimal downtime.