

Jeshwin William James

Software Engineer

United States | Open to Relocation | jeshwin.w.james@gmail.com | +1 (572) 276-1916 | [LinkedIn](#) | [GitHub](#) | [Publication](#)

SUMMARY:

Software Engineer with 3+ years of experience designing and building scalable distributed systems and low-latency backend applications using Java, Spring Boot, and Apache Kafka. Proven track record of developing fault-tolerant microservices, optimizing high-throughput transaction platforms, and delivering measurable performance improvements for applications serving millions of users. Experienced in building cloud-native distributed systems on AWS and GCP with expertise in event-driven architectures, scalable backend platforms, and AI-assisted engineering workflows supporting intelligent applications.

TECHNICAL SKILLS:

Programming Languages:	Java, Python, JavaScript, SQL, Go
Backend Development:	Spring Boot, Spring Security, Hibernate, JPA, Microservices, REST APIs, gRPC, Asynchronous Processing
AI & LLM:	AI-Assisted Development, LLM APIs, Prompt Engineering, Spring AI
Frontend Development:	React.js, Redux, Next.js
Messaging & Streaming:	Apache Kafka, JMS, Event-Driven Architecture
Databases & Data Storage:	Oracle, PostgreSQL, MySQL, Cassandra, Redis
Search & Indexing:	Elasticsearch
Big Data & Data Processing:	Apache Spark, Hadoop
Cloud Platforms:	AWS (EC2, RDS, S3, IAM, CloudWatch), Google Cloud Platform (GCP)
Containerization & Orchestration:	Docker, Kubernetes, Helm
Computer Science Fundamentals:	Data Structures, Algorithms, System Design, Distributed Systems, Operating Systems, Multithreading, Concurrency
Operating Systems & Networking:	Linux, TCP/IP, HTTP, Load Balancing
DevOps & CI/CD:	Jenkins, Maven, GitHub Actions, Git, Blue-Green Deployments
Testing & Quality Engineering:	JUnit, Mockito, Selenium, REST Assured
Monitoring & Observability:	Prometheus, Grafana, ELK Stack (Elasticsearch, Logstash, Kibana), Distributed Tracing
Performance & System Engineering:	Low-Latency Systems, High-Throughput Systems, Scalability, Fault Tolerance

CERTIFICATION:

[AWS Certified Solutions Architect](#) (SAA-C03) – Associate, Amazon Web Services

PROFESSIONAL EXPERIENCE:

Software Engineer, State Street, United States

January 2026 - Present

- Developed low-latency Java Spring Boot microservices for real-time fund transaction ingestion and reconciliation, processing daily trade and settlement records across global custody accounts while ensuring regulatory-compliant financial data handling.
- Owned the design and implementation of event-driven transaction processing pipelines using Apache Kafka and JMS, enabling asynchronous trade validation, settlement matching, and ledger updates while improving platform throughput by 40%.
- Drove database performance optimization initiatives across Oracle and PostgreSQL reconciliation workloads, reducing query latency from 3.2 seconds to under 1.1 seconds in production.
- Built React.js + Redux front-end dashboards integrated with REST APIs to visualize reconciliation exceptions and AI-assisted analysis workflows, enabling engineers and operations teams to investigate transaction failures and resolve breaks 35% faster.
- Implemented cloud-native deployment workflows using Docker, Kubernetes, Jenkins, and AWS (EC2, RDS, S3), enabling automated CI/CD pipelines and maintaining 99.9% service availability for critical trade-processing systems.

Software Engineer, Accenture

February 2022 - December 2024

- Developed backend services supporting high-traffic e-commerce checkout systems used by millions of users on the Flipkart platform, ensuring reliable order processing and payment workflows.
- Maintained high-availability microservices using Java and Spring Boot, ensuring seamless communication via gRPC and REST APIs to minimize service-to-service overhead.
- Owned the design and implementation of Kafka-driven order processing pipelines that decoupled inventory, payment, and logistics services, improving fault tolerance and operational scalability.
- Improved payment success rate by 12% through implementing smart retry logic and integrating multiple Indian payment gateways (UPI, NetBanking, and Wallets).
- Led performance optimization efforts across checkout microservices, reducing end-to-end API latency by 35% through Hibernate query tuning and Elasticsearch indexing strategies.
- Championed a "test-first" culture by maintaining 90%+ code coverage through JUnit, Mockito, Selenium, and REST Assured while leveraging AI-assisted development workflows to accelerate implementation, debugging, test generation, and code reviews.
- Managed containerized workloads using Docker and Kubernetes, implementing Helm charts and Jenkins CI/CD pipelines to automate blue-green deployments with zero downtime.
- Established comprehensive monitoring dashboards using Prometheus and Grafana, coupled with the ELK Stack for distributed tracing to identify and resolve production bottlenecks in real-time.
- Optimized Google Cloud Platform (GCP) resource allocation through horizontal pod autoscaling and spot instance utilization, resulting in a 22% reduction in annual infrastructure overhead.

PROJECTS:

AI-Powered Pull Request Intelligence Platform | Java, Spring Boot, Kafka, PostgreSQL, Docker

- Architected an AI-powered pull request intelligence platform using Spring Boot microservices, Apache Kafka, Spring AI, and LLM-based code analysis to automate review summaries and actionable code recommendations, reducing review turnaround time by 65%.
- Designed event-driven Kafka pipelines that asynchronously orchestrated AI analysis requests across distributed microservices, ensuring fault-tolerant processing of 10,000+ pull request events/day.
- Designed REST APIs for pull request ingestion, AI inference orchestration, review history management, and result delivery using PostgreSQL-backed microservices.
- Containerized services using Docker while implementing centralized monitoring, logging, and performance observability to improve AI workflow reliability and reduce troubleshooting effort by 40%.

GitHub: <https://github.com/jeshwinwilliam/pr-intelligence-platform-jeshwin>

Distributed Event Processing System | Java, Spring Boot, Kafka, Docker

- Built a distributed event-processing platform using Spring Boot microservices and Apache Kafka to process 50,000+ real-time events/day with low-latency asynchronous workflows.
- Implemented Kafka consumer groups, retry mechanisms, and dead-letter queues, improving fault tolerance and ensuring reliable event processing during service failures.
- Developed REST APIs for event publishing and asynchronous workflow orchestration, enabling integration with AI-powered event analysis services while reducing manual operational effort by 40%.
- Designed a horizontally scalable, containerized microservices architecture using Docker to support high-availability event processing and future AI-driven workflow orchestration.

GitHub: <https://github.com/jeshwinwilliam/distributed-event-processing-system>

EDUCATION:

Master of Science (MS) in Computer Science – Oklahoma City University, Oklahoma City, OK	2026
Bachelor of Technology (B.Tech) in Computer Science and Engineering – Hindustan University	2022