Sai Sravanthi Ponigate

Senior Software Engineer

ponigatesai@gmail.com
(614) 636-1316
https://github.com/pssravanthi

Professional Summary:

9+ years of experience as a Software Engineer building scalable web applications using Java/Spring Boot (backend) and Angular/React (frontend). Skilled in debugging and troubleshooting complex Angular applications in large-scale enterprise environments. Proven track record in microservices, cloud-native development (AWS/Azure), and CI/CD pipelines. Experienced in all phases of SDLC, from requirement analysis to production deployment and support.

- Strong expertise in developing multi-tier web applications using **Java 8–17, Spring Boot (REST APIs, Security, Cloud, Batch), Scala, Node.js, gRPC, Microservices, Hibernate, JPA, JDBC, JMS, Java Multithreading.**
- Hands-on experience in frontend development using **Angular 12–17, React 18, JavaScript, TypeScript, HTML5, CSS3, Bootstrap**, and **RxJS** to create responsive and dynamic user interfaces.
- Experienced with application and web servers, including **Apache Tomcat, JBoss**, and **Nginx.**
- Proficient in building scalable **RESTful APIs, GraphQL** using **Spring Boot** and **JAX-RS** with robust input validation, exception handling, caching, and **Swagger** documentation.
- Implemented real-time event-driven systems using **Apache Kafka** and **ActiveMQ**, and managed asynchronous communication between services.
- Strong experience with cloud platforms such as **Azure (AKS, Azure Functions, App Services, Azure SQL, Blob Storage, Key Vault, Monitor)**, **AWS (Lambda, S3, EC2, CloudWatch, IAM, EKS, Aurora PostgreSQL, MSK, SNS, and SQS),** for deploying containerized microservices.
- Built and deployed containerized applications using **Docker 24.0** and **Kubernetes 1.28** with **CI/CD** pipelines in **Jenkins** and **GitHub** Actions.
- Proficient in **RDBMS** like **PostgreSQL, MySQL, Oracle**, and **DB2**, with expertise in query optimization, indexing, and stored procedures.
- Knowledge of **NoSQL** databases like **MongoDB** and **DynamoDB** for managing semi-structured and distributed data.
- Hands-on experience designing caching using **Redis.**
- Experienced in **Agile/Scrum methodologies, TDD, BDD** using **JUnit/Mockito, Jest,** and tools like **Git, Bitbucket, Jira, Confluence,** and **Maven.**
- Familiar with monitoring and observability tools such as **Prometheus, Grafana, ELK Stack**, and **Splunk** for maintaining application health and performance.
- Strong understanding of Data Structures and Algorithms, debugging, and performance tuning in distributed concurrent systems.

Technical Skills:

Backend	Java 8–17, Spring Boot 3.5, Spring Cloud, Node.js, REST/GraphQL APIs, Kafka, Hibernate
Front End	Angular 12–17, Angular CLI, React 18, TypeScript 5.4, HTML5/CSS3, Bootstrap, RxJS, .NET, Redux
Web/App Servers	Apache Tomcat 10, JBoss EAP 7.4, Nginx
Databases	MySQL 8.4, Oracle 12c/19c, DB2 11.5, Microsoft SQL Server 2022, MongoDB 7.0, DynamoDB, Redis, Elasticsearch 7.17, Cassandra
Cloud Platform Services	Microsoft Azure (AKS, Azure Functions, App Services, Azure SQL, Blob Storage, Key Vault, Monitor), AWS (IAM, S3, EC2, ECS, Lambda, RDS, CloudWatch, EKS, Aurora PostgreSQL, DynamoDB, SNS, SQS, Route 53, ELB, MS), Salesforce.com CRM
Messaging Tools	Apache Kafka 3.2–3.7, Kafka Streams, Kafka Connect, Avro/Protobuf, ActiveMQ 5.18

Visualization & BI	Power BI, Tableau
Source/Version Control	GIT 2.46.0, GitHub, Bitbucket, Jira 9.17
Monitoring Tools	Prometheus 2.48, Grafana, Splunk 9.2, DataDog, ELK Stack
Operating Systems	Windows (XP,7,8,10,11), Ubuntu 24.04 LTS, Linux (RHEL, CentOS 8), UNIX
Development Methodologies	Agile/Scrum, TDD, BDD
IDE	Eclipse, NetBeans 20, IntelliJ IDEA 2024.1, Visual Studio 2022, Spring Tool Suite 4.24.0
Testing	JUnit 4.13,5.10, Mockito 5.10, REST Assured, Selenium 4.19, Cucumber, Jasmine, Karma, Jest, React Testing Library
REST API Tools	Postman 10, SoapUI 5
CI/CD & DevOps tools	Jenkins, Docker, Kubernetes (EKS, AKS, OpenShift), Terraform, Chef, Ansible

Certifications:

- Microsoft Certified: Azure Fundamentals
- Oracle Autonomous Database Cloud 2021 Certified Specialist
- Oracle Cloud Infrastructure 2021 Certified Architect Associate
- Oracle Cloud Infrastructure Developer 2021 Certified Associate
- Oracle Cloud Infrastructure Foundations 2021 Certified Associate
- Aviatrix Multi-Cloud Networking Associate

Professional Experience:

Client: Fidelity Investments, Dallas, TX.
Project: FHB Participant Services Portal
Role: Senior Software Engineer **August 2024 to Present**

Description: This application is developed for Phone Associates to serve Fidelity Health & Wellness participants. The primary entry point is another internal application, which allows navigating to the Participant overview page. This application also features manual search, where associates can search for individual participants using multiple search parameters.

Responsibilities:

- Collaborated with UI/UX designers to translate Figma wireframes into pixel-perfect interfaces using HTML5, CSS3, Bootstrap, and Angular Material.
- Developed responsive UI components using Angular 17, RxJS, and TypeScript signals to manage reactive state, improving frontend performance.
- Integrated Server-Sent Events (SSE) with Angular to stream real-time participant eligibility decisions and backend scoring updates, reducing refresh latency.
- Developed performance-tuned visual components using Highcharts to render participant health scores and utilization patterns dynamically, optimizing render cycles on large datasets.
- Resolved Angular performance bottlenecks by refactoring legacy components and lazy-loading routes, cutting initial load time by 40% and enabling smoother user transitions on low-bandwidth networks.
- Designed role-based dashboards (Angular 17) for 500+ associates, reducing search time.
- Integrated RESTful APIs using Angular HTTP Client with centralized error handling, retry logic, and caching to reduce request failures.
- Designed and enforced RBAC (Role-Based Access Control) using Angular route guards and Spring Security, ensuring data privacy compliance.
- Architected 15+ RESTful APIs (Java 17/Spring Boot 3.5.0) with OpenAPI specs, reducing latency.
- Implemented backend microservices using Node.js to handle API gateway routing and server-side rendering tasks, ensuring faster response times for storefront pages.
- Built microservices architecture using Spring Cloud and Apache Kafka for decoupled inter-service communication.
- Integrated Cucumber BDD framework for end-to-end acceptance testing of microservices workflows, ensuring business scenarios align with implementation and reducing regression defects by 40%.

- Integrated Cassandra as a secondary NoSQL store for high-throughput event data, enabling sub-second retrieval for API audit and rollback features.
- Wrote and maintained unit and integration tests using Jasmine/Karma (frontend), Jest, and JUnit 5/Mockito (backend), achieving high code coverage.
- Deployed containerized microservices on Azure AKS (Azure Kubernetes Services) using Terraform, reducing cloud costs.
- Designed and deployed container workloads on Amazon ECS in parallel with EKS clusters for specific high-throughput APIs, optimizing compute costs and enabling fine-grained auto-scaling.
- Integrated AWS ELB and Route 53 for service discovery and traffic routing across microservices, improving system uptime and regional failover support.
- Instrumented UI with custom telemetry hooks using DataDog RUM and configured system-level monitoring and alerts via DataDog and Splunk to reduce MTTR.
- Documented technical specifications and API contracts using Confluence and Swagger UI, streamlining onboarding and knowledge transfer.
- Participated in cross-functional working groups with downstream application teams to review and improve shared GraphQL schema evolution and service compatibility.

Environment: Java 17, Angular 17, TypeScript 5.4, CSS3, HTML5, Node.js 20, Spring Boot 3.5.0, Spring Security 6.2, Microservices, Maven 3.9, PostgreSQL 16, GitHub, Git 2.42, Jenkins, Visual Studio Code 1.89, IntelliJ IDEA 2024.1, DBeaver 24.0, Jira 9.11, JUnit 5.10, Mockito 5.10, Linux (Ubuntu 22.04), JSON, SOAP UI 5.7, Postman 11, AWS, Spring Cloud, Apache Kafka 3.7, Docker, Kubernetes, DataDog, Splunk 9.2, Swagger 3.1, Confluence 8.7

Client: Citi, Columbus, OH.
Project: Customer Experience Optimization Platform
Role: Senior Software Engineer **August 2023 to July 2024**

Description: Worked on a customer-centric digital transformation initiative to modernize Citi's platform by building cloud-native microservices, enhancing frontend user experience, and integrating observability to support real-time customer onboarding and account services.

Responsibilities:
- Developed 10+ reusable React.js 18 components (dynamic forms, data grids)
- Integrated Redux for state management across React components, ensuring consistent UI behavior and predictable state transitions in complex forms and dashboards.
- Incorporated interactive data visualizations using Chart.js and D3.js within React dashboards, enabling real-time insights for customers onboarding metrics and account trends.
- Integrated WebSocket-based push notifications to provide real-time status updates for account verification and fraud alerts, reducing user inquiry load by 35%.
- Developed scalable REST and GraphQL APIs using Java 17 and Spring Boot 3.2, adhering to SOLID principles and supporting modular customer onboarding features.
- Integrated gRPC services for internal microservice-to-microservice communication to optimize payload size and latency for high-volume transactions.
- Leveraged Kafka, including AWS MSK, for real-time event-driven processing of financial transactions, handling 5M+ daily events with improved durability and low latency.
- Wrote optimized PostgreSQL queries and managed schema versioning using Flyway, ensuring seamless DB migrations and query performance.
- Integrated Elasticsearch for indexing customer profiles and account metadata, enabling full-text search and boosting retrieval performance by 50%.
- Configured Oracle Database instances for legacy service compatibility, migrated selected schemas to PostgreSQL using Flyway, and optimized stored procedures for performance parity.

- Achieved 90% code coverage through rigorous unit and integration testing using JUnit 5 and Mockito, reducing post-release defects by 70%.
- Automated CI/CD pipelines with Jenkins and GitHub Actions for build, test, and deployment workflows, reducing release times by 60%.
- Implemented blue-green deployments (OpenShift/Kubernetes), achieving zero downtime during 6 major releases.
- Deployed microservices on AWS (Lambda, EC2) and integrated S3 for document storage; configured CloudWatch for automated monitoring and alerting.
- Integrated centralized logging and observability using Splunk, Prometheus, and Grafana, and performed RCA on production incidents via Splunk and CloudWatch, reducing MTTR by 40%.
- Designed a user-assisted AI interface that accepts human feedback for model outputs and pipelines it to Kafka for downstream training, enabling continuous model refinement.
- Conducted peer code reviews and documented APIs using Swagger and Confluence to support transparency and future maintainability.

Environment: Java 17, Spring Boot 3.2, React.js 18.2, Node.js 18.17, PostgreSQL 15, AWS (S3, EC2, Lambda, CloudWatch), Apache Kafka 3.5, Jenkins, GitHub Actions, Docker, Kubernetes, OpenShift, JUnit 5.9, Mockito 4.11, Postman 10, SOAP UI 5.7, Splunk, Prometheus, Grafana, Terraform, JSON, REST, GraphQL, Agile/Scrum, IntelliJ IDEA 2023.1, Visual Studio Code 1.81, Jira

Client: British Telecommunications, Hyderabad, India.
Project: BT Connect Platform
Role: Software Engineer **May 2022 to July 2023**

Description: The BT Connect Platform was designed to modernize British Telecom's service provisioning and network operations. The initiative focused on transforming legacy telecom systems into a scalable, cloud-native microservices architecture. It enabled automated service workflows, real-time monitoring of network KPIs, and enhanced customer experience by integrating modern UI dashboards, PostgreSQL-based analytics, and cloud infrastructure for performance and scalability.

Responsibilities:
- Developed real-time dashboards using React.js 18 and integrated backend services via secure REST APIs to visualize network KPIs and SLAs.
- Designed and implemented RESTful microservices using Java 11, Spring Boot, and Spring MVC to automate service provisioning workflows.
- Implemented Kafka-based asynchronous messaging and designed fault-tolerant consumers to handle 10K+ events/sec, improving throughput, system reliability, and reducing message loss by 99%.
- Integrated Postman into CI pipeline to validate API contracts and regressions; achieved 85% test coverage using JUnit 5 and Mockito.
- Managed builds with Maven 3.8 and implemented CI/CD pipelines in Jenkins 2.346, integrating Bitbucket and Git for version control.
- Enabled centralized logging with Logback and integrated with Splunk for real-time error tracking and RCA.
- Automated API documentation (Swagger/OpenAPI) for 15+ microservices, reducing onboarding time by 50%
- Used Prometheus 2.37 and Grafana 8 to monitor system health and custom metrics such as API throughput and Kafka lag.
- Deployed containerized services on AWS EC2 instances and stored application logs on S3 for audit and diagnostics.
- Developed post-deployment scripts in AWS Lambda to validate provisioning workflows using AI-based rules.

Environment: Java 11, Spring Boot 2.7, React.js 18, Apache Kafka 3.2, PostgreSQL 13, Maven 3.8, Jenkins 2.346, Git 2.35, Bitbucket, JUnit 5, Mockito, Postman, AWS EC2/S3/Lambda, Splunk 9, Prometheus 2.37, Grafana 8, Logback, Jira.

Client: Tata Consultancy Services, Hyderabad, India.
Project: Digital Integration Spine Framework Development
Role: Software Engineer **September 2021 to April 2022**

Description: The Digital Integration Spine Framework (DISF) in TCS is a scalable, flexible platform that integrates digital technologies, data, and processes to drive enterprise transformation. It aims to provide a reliable, secure, and efficient foundation for adopting AI, IoT, cloud, and analytics, enabling agile operations and personalized customer experiences across industries.

Responsibilities:

- Followed Agile Scrum methodology, contributing to sprint planning, backlog grooming, and daily standups to ensure delivery milestones.
- Gathered requirements and collaborated with business users for SIT (System Integration Testing) coordination across multiple stakeholders.
- Designed and implemented 25+ RESTful APIs using Java 8 and Spring Boot, enabling seamless data flow for AI, IoT, and analytics modules.
- Validated service endpoints using Postman and documented them with OpenAPI/Swagger for development and QA handover.
- Added input validation and global exception handling using @ControllerAdvice to strengthen service reliability and fault tolerance.
- Developed automated system health checks and daily report generation features using Spring Scheduler and Java Mail API.
- Troubleshot and resolved 50+ defects across microservices and data layers, improving system stability and performance.
- Used DB2 11.5 for transactional workflows, optimized stored procedures, and implemented complex joins for performance enhancement.
- Refactored DB2 stored procedures, slashing report generation time from 10min to 2min for 50+ daily users while reducing server CPU usage by 35% during peak loads.
- Managed Git repositories merged feature branches, and integrated releases to develop/master branches for monthly production deployment.
- Reviewed code submissions, enforced team coding standards, and supported documentation in Confluence for reuse and onboarding.

Environment: Java 8, Scala 2.13, Git, Log4j 2.14, IBM DB2 11.5, SQL, Apache Kafka 2.8, Jenkins, Postman 8, SOAP UI 5.5, Bitbucket, Jira, Eclipse 2021-06, IntelliJ IDEA 2021.2, Splunk 8.1, JUnit 4.13, Java Mail API 1.6, Linux (CentOS 7), JSON, OpenAPI

Client: NCR, Hyderabad, India.
Project: Loyalty Promotion Engine
Role: Software Engineer **July 2018 to August 2021**

Description: The Loyalty Promotion Engine at NCR Corporation is a robust platform designed to enhance customer engagement for retail and hospitality clients by managing loyalty programs, promotions, and rewards. The system processes customer transactions, tracks loyalty points, and applies promotional rules in real-time, integrating with point-of-sale (POS) systems and third-party services to deliver personalized offers and seamless user experiences.

Responsibilities:

- Participated in requirement gathering, design walkthroughs, and backlog grooming sessions with cross-functional teams.
- Built Angular 12-based web interfaces for configuring promotions and loyalty rules by business users.
- Leveraged deep Angular debugging to resolve change detection errors and memory leaks, improving SPA performance under high transaction loads.
- Used microservices and REST APIs to integrate loyalty rules engine with POS systems and external campaign platforms.
- Delivered pixel-perfect HTML5/CSS3 components compliant with responsive design and accessibility (WCAG) standards, significantly improving merchant user experience.
- Wrote business logic in Java and Scala to manage rule evaluation, thresholds, exclusions, and multi-tier rewards.
- Integrated Spring Security for role-based access control and encrypted data transmission between services.
- Implemented event-driven workflows with ActiveMQ to asynchronously process customer transactions and promotions.
- Designed MongoDB schemas for 1M+ loyalty transactions, enabling sub-100ms query performance.
- Created test suites using JUnit 4 and Mockito, achieving high test coverage and reducing bug leakage in QA.
- Implemented Redis caching for loyalty point calculations, reducing latency from 500ms to 50ms (90% faster).
- Automated builds and deployments using Jenkins with Bitbucket and Maven for version control and CI.
- Performed code reviews and tracked metrics to ensure a modular codebase, scalability, and high performance.

Environment: Java 8, Spring Boot 2.4, Scala 2.12, Microservices, Angular 12, ActiveMQ 5.16, JavaScript, HTML5, REST APIs, MongoDB 4.4, JUnit 4.12, Git, Eclipse 2020-06, Maven 3.6, Bitbucket, jQuery 3.6, Jenkins, Redis.

Client: Fraazo, Mumbai, India.
Project: Real-Time Supply Chain Optimization
Role: Software Engineer **April 2016 to June 2018**

Description: Developed a comprehensive real-time Supply Chain Management platform to streamline Fraazo's logistics and inventory operations. The project involved building a scalable dashboard to monitor key performance indicators (KPIs), automating operational workflows, and migrating legacy systems to a unified architecture. The platform integrated structured databases and real-time data processing to enhance operational efficiency, reduce manual intervention, and improve system reliability, resulting in significant performance gains and user engagement.

Responsibilities:

- Designed and developed a real-time Supply Chain Management dashboard using Java and Spring Boot for the backend
- Developed React.js for the front end, improving operational efficiency by 30%.
- Automated critical company operations by integrating MySQL and PostgreSQL databases into core workflows.
- Wrote optimized database queries and stored procedures in MySQL and PostgreSQL to support high-performance data operations.
- Integrated Apache Kafka for real-time data streaming and processing, improving communication workflows and operational efficiency by 25%.
- Handled 10,000+ daily transactions while reducing MySQL/PostgreSQL database load.
- Developed unit and integration tests using JUnit and React Testing Library to ensure robust functionality and system stability.
- Designed and implemented CI/CD pipelines using Jenkins and Docker, reducing deployment times by 50% and ensuring consistent application scaling.
- Led migration of a major legacy project to a modern architecture with zero downtime, utilizing Docker and Jenkins for seamless deployment, resulting in a 25% increase in user engagement.

- Utilized Agile/Scrum methodologies to ensure 100% on-time delivery of features and updates, collaborating with cross-functional teams.

Environment: Java 8, Spring Boot 2.2, React.js 16, MySQL 8.0, PostgreSQL 12, Apache Kafka 2.3, Redis 5.0, Jenkins, Docker, AWS, JavaScript, HTML5/CSS3, JUnit 4.12, Agile/Scrum, Bitbucket, REST API