**Sai Pavan Kumar Gundapaneni**

Milwaukee, WI | +1 414-436-5713 | [saipavankumar2703@gmail.com](mailto:saipavankumar2703@gmail.com)

LinkedIn | GitHub | Portfolio | Medium

**PROFESSIONAL SUMMARY:**

* Over **8 years** of hands-on experience as a **Full Stack Java Developer** across **healthcare**, **banking**, and **retail domains**.
* Proficient in **Java**, including Streams API, Functional Interfaces, Lambda expressions, Exception Handling, and Collections.
* Expertise in **Spring Boot**, **Spring MVC**, **Spring Security**, **Spring Batch**, **Spring AOP**, and **Spring Data JPA** for backend development.
* Strong command over building and consuming **RESTful** and **SOAP** web services with secure integrations using **OAuth2** and **JWT**.
* In-depth knowledge of microservices architecture, containerization (**Docker**), and orchestration using **Kubernetes** and **AWS** **EKS**.
* Solid experience with messaging systems including **Apache Kafka**, **RabbitMQ**, and **AWS SQS** for real-time, decoupled communication.
* Skilled in front-end development using **ReactJS** with functional components, hooks, Context API, Redux Toolkit, and React Router.
* Experience in building SPAs with responsive UI using **HTML5**, **CSS3**, **Bootstrap**, **Tailwind** **CSS**, and **Material UI**.
* Expertise in cloud computing services including **AWS EC2**, **Lambda**, **S3**, **RDS**, **CloudWatch**, **ECS**, **ELB**, **SNS**, and **CloudFormation**.
* Well-versed in **CI/CD pipelines** using **Jenkins**, **AWS CodePipeline**, **GitLab** **CI**, and **Bitbucket** Pipelines.
* Strong understanding of **relational** databases (**PostgreSQL**, **MySQL**, **Oracle**) and **NoSQL** databases (**MongoDB**, **Cassandra**, **Redis**).
* Capable of designing optimized schemas, writing complex **SQL**, and tuning queries for performance.
* Knowledgeable in using **Hibernate** **ORM** for advanced mappings, projections, lazy loading, and batch processing.
* Hands-on experience using **JUnit**, **Mockito**, and **Jest** for unit testing, integration testing, and **TDD**-based development.
* Experience in designing and documenting APIs using **Swagger**, **Postman**, and **OpenAPI**.
* Applied design patterns such as Singleton, Factory, DAO, MVC, and Repository in enterprise-grade systems.
* Integrated DevOps practices into development life cycle including Dockerized builds, version control via **Git**, and automated deployments.
* Used Prometheus and Grafana for metrics collection and visualization; **Log4J** and **ELK** Stack for centralized logging.
* Familiar with security implementations such as **HTTPS**, **CORS**, **CSRF**, **SSO**, **RBAC**, and encryption standards.
* Excellent problem-solving and debugging skills, with experience resolving critical production issues.
* Continuous learner committed to mastering new tools and frameworks for enhanced software quality and team productivity.

**TECHNICAL SKILLS:**

* **Languages:** Java, SQL, JavaScript, HTML5, CSS3, TypeScript
* **Frameworks:** Spring Boot, Spring MVC, Hibernate, ReactJS, Redux, Node.js, Angular
* **Web Technologies:** REST APIs, SOAP, GraphQL, JSP, Servlets, Bootstrap
* **Databases:** Oracle, MySQL, PostgreSQL, MongoDB, Cassandra
* **Cloud & DevOps:** AWS (EC2, S3, RDS, Lambda),Azure, Jenkins, Docker, Kubernetes, GitLab
* **Tools:** GitHub, Maven, Gradle, Postman, JUnit, Mockito, Log4J, Swagger, ELK Stack, Kafka, RabbitMQ
* **Others:** Microservices Architecture, TDD, Agile/Scrum, CI/CD Pipelines, OAuth2, JWT

**PROFESSIONAL EXPERIENCE:**

**Client: NextGen Health Care, Atlanta, GA  
Role: Full Stack Java Developer  
Duration: April 2023 – Present**

**Project Description:**

Developed and enhanced a comprehensive healthcare platform designed to manage patient records, scheduling, prescriptions, and provider communications. The system is used across hospitals and clinics to improve care coordination, compliance, and digital engagement with patients.

**Responsibilities:**

* Developed modular microservices using **Spring Boot** and **Node.js**, following **microservices** best practices for scalability, fault isolation, and maintainability.
* Designed and optimized **Java**-based service components with an emphasis on modularity, reusability, and scalability.
* Leveraged Java collections, concurrency utilities, and performance profiling tools (**JVisualVM**, **GC logs**) to optimize back-end processing.
* Conducted unit and integration testing with **JUnit** and **Mockito**, ensuring code reliability and code coverage metrics.
* Implemented robust exception handling and logging strategies using custom exception hierarchies and **Log4J**.
* Created dynamic and reusable **UI components** with **ReactJS** and **Redux** to deliver rich user experiences and efficient state management.
* Designed and documented **RESTful** **APIs** with proper authentication (**OAuth2, JWT**) and error handling to support complex business workflows.
* Implemented event-driven communication with **Apache** **Kafka** to process real-time patient notifications and system events.
* Wrote custom **Kafka** serializers and deserializers to standardize message formats across microservices.
* Managed transactional and unstructured data using **MySQL** (AWS RDS) and **MongoDB**, respectively, optimizing database access and data integrity.
* Wrote advanced **SQL** queries, performed database indexing, normalization, and query optimization for faster data retrieval.
* Utilized **Hibernate** **ORM** for efficient data persistence and implemented one-to-many and many-to-many relationships.
* Automated infrastructure provisioning on **AWS** with Terraform, ensuring consistency and quick environment setup for development, testing, and production.
* Configured and maintained **CI/CD pipelines** using **AWS CodePipeline**, integrating automated testing, security scanning, and deployment strategies.
* Designed and deployed microservices to **AWS ECS** and used **EC2** for hosting backend systems with load balancing via **AWS ELB**.
* Used **AWS CloudWatch** for centralized logging and monitoring; set up alarms and dashboards to track service health and usage metrics.
* Followed **Test Driven Development** (TDD) practices with **JUnit** and **Mockito** to improve code reliability and reduce bugs.
* Analyzed application logs and metrics via **ELK Stack** to proactively monitor system health and troubleshoot production issues.
* Contributed to improving project documentation, including API specs, system architecture diagrams, and runbooks.
* Integrated third-party healthcare **APIs** (FHIR, HL7) to exchange patient data with external systems.
* Developed middleware services for processing and transforming inbound/outbound EHR data.
* Containerized legacy services using **Docker** and refactored them to meet microservices patterns.
* Used **AWS Lambda** to trigger asynchronous backend tasks like email and report generation.

**Environment:** Java, Spring Boot, Node.js, ReactJS, Redux, MySQL (AWS RDS), MongoDB, Hibernate, Apache Kafka, AWS (EC2, S3, Lambda, RDS, ECS, ELB, CloudWatch), Terraform, AWS CodePipeline, JUnit, Mockito, ELK Stack, Agile Scrum.

**Client: BMO Harris Bank, Chicago, IL  
Role: Full Stack Java Developer  
Duration: February 2021 – March 2023**

**Project Description**:

Worked on the development and modernization of the bank’s digital banking platform, focusing on improving performance, transaction transparency, and fraud detection for both retail and corporate banking clients.

**Responsibilities:**

* Developed **RESTful microservices** using **Spring Boot**, designing APIs adhering to best practices with proper versioning, error handling, and security considerations.
* Built rich client-side web applications using **Angular**, focusing on performance optimization through lazy loading and efficient change detection strategies.
* Implemented dynamic form rendering and reactive form controls in **Angular**, enabling customizable workflows and better UX.
* Developed **Angular** services to interface with secure **backend** APIs and standardize data access across components.
* Utilized **Apache Kafka** to implement event-driven architecture supporting real-time transaction streams and fraud detection workflows.
* Created custom **Kafka** consumer groups with manual offset management for transactional processing.
* Designed database schemas and wrote optimized queries for **PostgreSQL**, ensuring ACID compliance and consistency; leveraged **Cassandra** for high-throughput event logging.
* Automated cloud infrastructure provisioning on **Azure** using Terraform, enabling scalable and repeatable deployments across environments.
* Integrated CI/CD pipelines with **Jenkins**, automating builds, tests, security scans, and deployments to Azure **Kubernetes** Service.
* Used **Bitbucket** for version control, enforcing branching strategies and conducting code reviews to maintain code quality.
* Wrote unit and integration tests with **JUnit** and **Mockito**, improving software reliability and facilitating continuous integration.
* Conducted performance tuning by analyzing slow queries and optimizing service response times.
* Configured monitoring and alerting systems using **Log4J** and **Kibana** to detect anomalies and reduce downtime.
* Used Spring Security with **OAuth2** and custom filters to protect **APIs** and integrate with **SSO**.
* Migrated legacy **XML**-based services to **JSON**/**REST** **APIs** to streamline integrations with fintech partners.
* Set up distributed caching using Hazelcast to reduce load on **PostgreSQL** for frequently accessed data.
* Configured and maintained multiple Helm charts for **Kubernetes**-based microservices deployments.
* Worked with **Azure** Service Bus to handle **asynchronous** transaction alerts and batch job notifications.
* Built reusable **Angular** directives and services to standardize form validations and error handling.
* Tuned **JVM** parameters and garbage collection strategy for microservices running in containerized environment.
* Collaborated within **Agile** Scrum teams, participating in sprint planning, backlog grooming, and daily standups.
* Worked closely with security teams to ensure compliance with banking regulations, applying role-based access controls and encryption techniques.

**Environment:** Java, Spring Boot, Angular, PostgreSQL, Cassandra, Apache Kafka, Azure, Terraform, Jenkins, Bitbucket, JUnit, Mockito, Log4J, Kibana, Agile Scrum.

**Client: Starbucks, Seattle, WA  
Role: Java Developer  
Duration: September 2019 – January 2021**

**Project Description**:

Revamped Starbucks’ inventory and supply chain system by transitioning from monolithic to microservices-based architecture. The new platform supports real-time stock updates, logistics tracking, and predictive restocking algorithms across stores.

**Responsibilities:**

* Designed and implemented backend microservices with **Spring Boot**, focusing on modularity, scalability, and ease of maintenance.
* Developed user-facing dashboards using **Angular**, enhancing usability with responsive design and efficient state management.
* Refactored **Angular** components using reactive forms and **RxJS** to improve scalability and maintainability of **UI** workflows.
* Migrated legacy **Java** components to modern **Spring Boot microservices**, reducing code complexity and increasing test coverage.
* Created reusable utility classes and configuration modules in **Java** for standardizing service integrations and logging across the platform.
* Configured health checks, circuit breakers, and fallback mechanisms to increase fault tolerance and system resilience.
* Built custom caching layers using **Redis** to minimize database load and optimize high-frequency data access patterns.
* Integrated asynchronous communication patterns using **RabbitMQ**, decoupling services and improving system resilience.
* Developed and optimized database interactions with **MySQL**, including query optimization and schema design; implemented **Redis** caching for improved read performance.
* Automated cloud provisioning using **AWS CloudFormation** and CLI, enabling repeatable and reliable infrastructure setup across environments.
* Deployed containerized microservices on **AWS ECS** with load balancing and auto-scaling using **AWS ALB** and CloudWatch alarms.
* Implemented centralized logging and alerting using **AWS CloudWatch** Logs and metrics dashboards.
* Utilized **AWS S3** for storing deployment artifacts, configuration files, and static web assets.
* Created and maintained **CI/CD pipelines** using **GitLab CI**, automating build, testing, and deployment processes to improve delivery speed and quality.
* Authored comprehensive unit and integration tests with **JUnit** and **Mockito**, driving a culture of testing and early defect detection.
* Collaborated with QA and DevOps teams to triage production issues and deploy hotfixes as necessary.
* Utilized version control with **Git**, managing feature branches, pull requests, and code merges to support collaborative development.
* Used **Log4J** integrated with **ELK** Stack for centralized logging and operational monitoring to quickly identify and resolve issues.
* Improved application performance by profiling **JVM** memory usage and tuning garbage collection parameters.
* Followed Agile Scrum methodology, actively contributing to sprint planning, backlog refinement, and retrospectives.

**Environment:** Java, Spring Boot, Angular, MySQL, Redis, RabbitMQ, AWS, CloudFormation, GitLab CI, Git, JUnit, Mockito, Log4J, ELK Stack, Agile Scrum.

**Client: HashTag Technologies, Coimbatore, India  
Role: Java Developer  
Duration: July 2017 – August 2019**

**Project Description:**

Worked on the development of multiple enterprise solutions for clients in logistics, CRM, and HRMS domains. These systems were used by small and medium-sized enterprises to streamline operations, track performance, and automate workflows.

**Responsibilities:**

* Assisted in developing backend **RESTful APIs** using **Spring Boot**, focusing on core CRUD operations and business logic implementation.
* Collaborated with frontend developers to integrate UI components built with **ReactJS**, ensuring seamless communication between frontend and backend.
* Implemented asynchronous messaging using **RabbitMQ** to improve application responsiveness and fault tolerance.
* Developed and optimized **SQL** queries in **Oracle DB**, designed relational schema and maintained data integrity; worked with **MongoDB** for flexible document storage needs.
* Participated in code reviews, identifying bugs and suggesting improvements to maintain high code quality standards.
* Wrote unit tests with **JUnit** to verify functionality and helped integrate automated testing in CI pipelines.
* Assisted in troubleshooting and debugging issues in development and staging environments using log analysis and stack traces.
* Automated deployments and infrastructure provisioning on AWS using CloudFormation, improving deployment consistency and reducing manual errors.
* Gained experience with version control using **GitHub**, practicing effective branching, pull requests, and conflict resolution.
* Participated actively in **Agile** Scrum ceremonies such as daily standups, sprint planning, and retrospectives to support iterative delivery.
* Maintained application logging with **Log4J** and supported the creation of monitoring dashboards to aid in production support.
* Developed understanding of secure coding principles by validating inputs, handling exceptions, and protecting sensitive data.

**Environment:** Java, Spring Boot, ReactJS, Oracle DB, MongoDB, RabbitMQ, AWS, CloudFormation, GitHub, JUnit, Log4J, Agile Scrum.

**EDUCATION:**

**Master of Science in Information Technology and Management**  
University of Wisconsin-Milwaukee | Milwaukee, WI

**Bachelor of Technology in Computer Science**  
Sree Dattha Group of Institutions | JNTU Hyderabad