

Rohith Rama

Sr. Java Full Stack Developer | Spring Boot | Angular | React | AWS | Kafka

Email: rohith.rama2589@gmail.com

Phone: +19373148902

LinkedIn: www.linkedin.com/in/rohith-rama-580193210

SHORT SUMMARY:

Accomplished Sr. Java Full Stack Developer with 11+ years of experience delivering enterprise-scale applications using Java 8/11/17/21, Spring Boot 3.x, Spring Boot Microservices, Spring Boot REST APIs, Angular 15+, and React.js. Expert in Apache Kafka, Kafka Streams, AWS cloud, Docker, Docker Kubernetes, Kubernetes EKS, Jenkins CI/CD, Jenkins pipelines, GitHub Actions CI, GitHub Actions CD workflows, and Terraform automation. Proficient in REST APIs, REST APIs OAuth 2.0, OAuth 2.0 JWT, TypeScript, TypeScript Angular, gRPC, GraphQL, and Spring Security across Banking, Financial Services, Healthcare, Insurance, Retail, Payments, and Cards domains. Skilled in BDD Cucumber, BDD Gherkin, TDD practices, Agile/Scrum sprints, Agile/Scrum ceremonies, System Design, Technical Leadership, Code reviews, SOLID principles with GitHub Copilot and Claude Code AI tools.

PROFESSIONAL SUMMARY:

- 11+ years of professional experience as a **Sr. Java Full Stack Developer** specializing in **Java 8/11/17/21, Core Java, J2EE, Spring Boot 3.x, Spring Boot** Microservices, **Spring Boot** REST APIs, **Spring MVC, Spring Cloud, Spring Security, Spring WebFlux, Spring Batch, Hibernate, JPA, and Microservices** architecture across Banking, Financial Services, Healthcare, Insurance, Retail, Payments, and Cards domains.
- Strong expertise in **REST APIs, gRPC, SOAP, GraphQL, and Apache Kafka** event-driven **Microservices**. Hands-on experience with **Kafka Streams, Kafka Connect, IBM MQ, and JMS** for scalable, fault-tolerant distributed messaging systems.
- Proficient in **Angular 15+** SPAs, **Angular 15+** component architecture, **Angular 15+** lazy loading, **React.js, TypeScript** advanced patterns, **TypeScript** generics, **JavaScript ES6+, HTML5, CSS3, Redux, NgRx, RxJS, Bootstrap, and Material UI**. Built 15+ enterprise **Angular 15+** and **React.js** applications with **TypeScript** for Banking and Healthcare clients.
- Extensive **AWS** expertise including **EC2, S3, Lambda, ECS, EKS, RDS, DynamoDB, API Gateway, CloudWatch, CloudFormation, and CodePipeline**, plus **Azure** (AKS, Azure DevOps) and **GCP** (Cloud Run, GKE, Pub/Sub, BigQuery, Cloud Functions).
- Expert **CI/CD** engineer with **Jenkins** pipelines, **Jenkins** build automation, **GitHub Actions** CI workflows, **GitHub Actions** CD deployments, **GitHub Actions** matrix builds, **AWS CodePipeline, Docker** image builds, **Docker** Compose, **Docker** ECR registries, **Kubernetes** orchestration, **Kubernetes** HPA, **Kubernetes** Helm charts, and **Kubernetes** EKS deployments.
- Expert in **Oracle PL/SQL, PostgreSQL, MySQL, MongoDB, DynamoDB, Cassandra, Redis, and MS SQL Server** database design, query optimization, stored procedures, and indexing strategies.
- Expert in **OAuth 2.0** authorization flows, **OAuth 2.0** JWT tokens, **OAuth 2.0** PKCE, **REST APIs** design, **REST APIs** versioning, **REST APIs** security, **TypeScript** API clients. Applied **BDD** Cucumber scenarios, **BDD** Gherkin, **BDD** step definitions, **TDD** red-green-refactor, and **Agile/Scrum** sprint ceremonies across all enterprise projects.
- Active daily user of **GitHub Copilot, Claude Code, and Cursor** AI tools accelerating **Angular 15+, Spring Boot, TypeScript, Docker, Kubernetes, Jenkins, GitHub Actions, OAuth 2.0, BDD, and Agile/Scrum** delivery.

TECHNICAL SKILLS:

Backend (Primary)	Java 8/11/17/21, Core Java, J2EE, Spring Boot 3.x, Spring MVC, Spring Cloud, Spring Security, Spring WebFlux, Spring Batch, Spring Data, Hibernate, JPA, Microservices, REST APIs, SOAP, GraphQL, gRPC, Drools
Frontend (Secondary)	Angular 15+, React.js, TypeScript, JavaScript ES6+, HTML5, CSS3, Redux, NgRx, RxJS, Bootstrap, Material UI, Node.js, Express.js
Messaging	Apache Kafka, Kafka Streams, Kafka Connect, IBM MQ, JMS, AWS SQS/SNS
Cloud	AWS (EC2, S3, Lambda, ECS, EKS, RDS, DynamoDB, API Gateway, CloudWatch, CloudFormation, CodePipeline), Azure (AKS, Azure DevOps), GCP (Cloud Run, GKE, Pub/Sub, BigQuery, Cloud Functions)
DevOps / CI-CD	Docker, Docker Compose, Docker ECR, Kubernetes, Kubernetes EKS, Kubernetes AKS,

	Kubernetes Helm, Jenkins, Jenkins CI/CD, GitHub Actions, GitHub Actions CI, GitHub Actions CD, Terraform, Maven, SonarQube, Git
Databases	Oracle, Oracle PL/SQL, PostgreSQL, MySQL, MongoDB, DynamoDB, Cassandra, Redis, DB2, MS SQL Server
Testing	JUnit 5, Mockito, Selenium, Jest, Cucumber
Security	OAuth 2.0, OAuth 2.0 JWT, OAuth 2.0 PKCE, JWT, SAML, SSO (Okta), SSL/TLS, Spring Security
AI Tools	GitHub Copilot, Claude Code, Cursor
Domains	Banking, Financial Services, Healthcare, Insurance, Retail, Payments, Cards
Soft Skills	Code reviews, Mentoring, Agile/Scrum, Agile/Scrum Sprints, TDD, TDD practices, BDD, BDD Cucumber, BDD Gherkin, SOLID principles, System Design, Technical Leadership

EDUCATION:

- **Masters in Computer Science** — Wright State University, Dec 2013
- **Bachelors in Computer Science** — Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology, India, May 2012

PROFESSIONAL EXPERIENCE:

Client: TD Bank, Cherry Hill, NJ

Mar 2024 – Present

Role: Sr. Java Full Stack Developer

Project Title: Digital Loan Origination and Credit Processing

Project Description: Built an enterprise-grade digital banking platform using Spring Boot, Kafka, Docker, Kubernetes, GitHub Actions, OAuth 2.0, BDD, and Agile/Scrum that streamlined the loan origination process end to end. The Spring Boot Microservices and Angular frontend brought together multiple internal workflows into a coordinated Spring Boot REST APIs experience, reducing manual processing using Docker Kubernetes deployments.

Responsibilities:

- Designed and implemented production-grade **Java 8/11/17/21** backend **Microservices** using **Spring Boot 3.x**, **Spring Boot** REST services, **Spring MVC**, **Spring Cloud**, **Spring Security**, and **Spring WebFlux** for reactive non-blocking **REST APIs**. Containerized all **Spring Boot** services using **Docker** and orchestrated with **Kubernetes** on **AWS EKS**. Improved system throughput by 25% adhering to **SOLID principles** and **System Design** best practices.
- Architected event-driven **Microservices** using **Apache Kafka**, **Kafka Streams**, and **Kafka Connect** for high-throughput financial transaction processing. Designed **Kafka** topic partitioning, **Kafka** consumer group rebalancing, and **Kafka** exactly-once semantics ensuring fault-tolerant real-time **Kafka**-driven data pipelines across distributed banking services.
- Built and deployed **REST APIs** and **gRPC** services using **Spring Boot 3.x** and **Spring Cloud Gateway** with standardized error handling, circuit breakers, and resiliency patterns. Enabled secure, scalable API contracts across 12+ banking **Microservices**.
- Implemented **Hibernate**, **JPA**, and **Spring Data** ORM frameworks for **Oracle**, **PostgreSQL**, and **MongoDB** data access. Applied advanced **Oracle** query optimization, indexing strategies, and **Oracle** stored procedures for high-throughput banking workloads.
- Built serverless functions using **AWS Lambda** integrated with **API Gateway** and **AWS SQS/SNS**. Reduced operational costs by \$25K annually through event-driven financial data processing workflows.
- Implemented **Drools** rule engine for complex credit decisioning and eligibility workflows. Automated business rule execution for loan origination, reducing manual underwriting effort significantly.
- Engineered robust **CI/CD pipelines** using **Jenkins**, **GitHub Actions**, and **AWS CodePipeline** with **SonarQube** quality gates and automated **JUnit 5** regression suites. Improved release reliability and reduced manual deployment steps across all financial application modules.
- Integrated **OAuth 2.0**, **OAuth 2.0** JWT tokens, and **Spring Security** for enterprise authentication across all banking **Microservices**. Applied **OAuth 2.0** authorization code flow, **OAuth 2.0** PKCE patterns, and **OAuth 2.0** client credentials enforcing role-based access control and PCI-DSS compliance.

- Championed engineering excellence through structured **Code reviews**, pair programming, and **Technical Leadership** mentoring 4 senior engineers on **System Design** patterns, **SOLID principles**, **BDD** test automation, and **Agile/Scrum** sprint delivery practices.

Environment: Java 8/11/17/21, Core Java, J2EE, Spring Boot 3.x, Spring Boot Microservices, Spring Boot REST APIs, Spring MVC, Spring Cloud, Spring Security, Spring WebFlux, Spring Batch, Spring Data, Hibernate, JPA, Microservices, REST APIs, gRPC, SOAP, Drools, Apache Kafka, Kafka Streams, Kafka Connect, IBM MQ, JMS, AWS (EC2, S3, Lambda, ECS, EKS, RDS, DynamoDB, API Gateway, CloudWatch, CloudFormation, CodePipeline), Oracle, Oracle PL/SQL, PostgreSQL, MongoDB, Redis, Docker, Docker Compose, Kubernetes, Kubernetes EKS, Kubernetes Helm, Jenkins, Jenkins CI/CD, GitHub Actions, GitHub Actions CI, GitHub Actions CD, Terraform, Maven, SonarQube, JUnit 5, Mockito, Cucumber, Git, OAuth 2.0, OAuth 2.0 JWT, TDD, BDD, BDD Cucumber, Agile/Scrum.

Client: Cigna Health, Plano, TX

Aug 2021 – Feb 2024

Role: Sr. Java Full Stack Developer

Project Title: Healthcare Claims and Benefits Processing Platform

Project Description: Built a HIPAA-compliant healthcare platform using Spring Boot, Angular, TypeScript, Docker, Kubernetes, OAuth 2.0, BDD, Agile/Scrum, Jenkins, and GitHub Actions. The Spring Boot Microservices integrated multiple backend services into an event-driven architecture supporting high-volume healthcare data processing.

Responsibilities:

- Designed scalable **Java 8/11** healthcare **Microservices** using **Spring Boot**, **Spring MVC**, **Spring Cloud**, **Spring Security**, and **Hibernate**. Achieved 99.3% uptime for healthcare claims processing through resilient **Microservices** design and **System Design** optimizations.
- Designed and implemented **Apache Kafka** topics, **Kafka Streams** processors, and **Kafka Connect** sink/source connectors for real-time healthcare claims event streaming. Reduced batch processing time by 28% through **Kafka**-based exactly-once event-driven architecture.
- Developed 30+ **REST APIs** and **GraphQL** services using **Spring Boot 3.x** and **JAX-RS** for healthcare member portals. Built **REST APIs** following OpenAPI/Swagger standards with **TypeScript** clients, rate limiting, versioning, and **OAuth 2.0** secured **REST APIs** for legacy **SOAP** integrations.
- Implemented **Hibernate**, **JPA**, and **Spring Data** for **Oracle**, **PostgreSQL**, and **MongoDB** data access. Applied **Drools** rule engine for complex HIPAA-compliant claims adjudication and eligibility decisioning.
- Integrated **OAuth 2.0**, **Okta SSO**, and **Spring Security** for HIPAA-compliant authentication. Enforced PHI data encryption, **JWT** token management, and strict RBAC across all healthcare **Microservices**.
- Containerized healthcare **Microservices** using **Docker** and orchestrated **Docker** containers with **Kubernetes** on **AWS EKS** and **AWS ECS**. Deployed using **AWS Lambda**, **RDS**, **DynamoDB**, and **API Gateway** with **Jenkins** and **GitHub Actions CI/CD pipelines** following **TDD** practices.
- Applied **BDD** Cucumber scenarios, **BDD** Gherkin acceptance criteria, and **Selenium** end-to-end testing for healthcare workflow validation. Maintained comprehensive code coverage through structured **Code reviews** and **SOLID principles** enforcement.

Environment: Java 8/11, Core Java, J2EE, Spring Boot, Spring Boot Microservices, Spring MVC, Spring Cloud, Spring Security, Spring Batch, Spring Data, Hibernate, JPA, Microservices, REST APIs, GraphQL, SOAP, Drools, Apache Kafka, Kafka Streams, Kafka Connect, IBM MQ, JMS, AWS (Lambda, EC2, S3, ECS, EKS, RDS, DynamoDB, API Gateway, CodePipeline), Oracle, Oracle PL/SQL, PostgreSQL, MongoDB, Redis, Docker, Docker Compose, Docker ECR, Kubernetes, Kubernetes EKS, Kubernetes Helm, Jenkins, Jenkins CI/CD, GitHub Actions, GitHub Actions CI, Maven, SonarQube, TypeScript, Angular 15+, OAuth 2.0, OAuth 2.0 JWT, JWT, Git, TDD, BDD, BDD Cucumber, Agile/Scrum.

Client: MetLife, New York, NY

Jan 2020 – Jul 2021

Role: Java Full Stack Developer

Project Title: Insurance Operations and Policy Processing

Project Description: Developed a modernized Spring Boot and Angular platform to streamline policy administration. The Spring Boot Microservices solution with Angular TypeScript frontend, Docker Kubernetes deployment, Jenkins GitHub Actions CI/CD, OAuth 2.0 security, and BDD Agile/Scrum delivery simplified complex insurance workflows.

Responsibilities:

- Engineered high-throughput **Java 8** **Microservices** using **Spring Boot**, **Spring Batch** partitioned jobs, **Spring Security**, and **Hibernate** ORM for insurance policy processing. Processed 100K+ policy records daily achieving 22% throughput improvement through **JPA** query optimization and **Spring Data** repository tuning.

- Developed **Angular 8** reactive forms with **TypeScript**, **RxJS**, and **NgRx** state management for insurance policy workflows. Ensured accurate data capture and improved form reliability across enterprise insurance platforms.
- Implemented **CI/CD pipelines** using **Jenkins** and **GitHub Actions** with automated **JUnit 5** and **Mockito** test suites following **TDD** practices. Maintained high code quality through **SonarQube** gates and structured **Code reviews**.
- Containerized **Spring Boot Microservices** using **Docker** images and deployed on **Kubernetes** via **AWS ECS** with **Helm** charts. Applied **OAuth 2.0**, **JWT**, and **Spring Security** RBAC across all insurance service endpoints.
- Integrated **Apache Kafka** for fault-tolerant **IBM MQ**-compatible asynchronous messaging across distributed insurance **Microservices**. Applied **BDD** Cucumber testing and **Agile/Scrum** sprint delivery practices.

Environment: Java 8, Core Java, Spring Boot, Spring Boot Microservices, Spring Batch, Spring MVC, Spring Security, Spring Data, Hibernate, JPA, Microservices, REST APIs, Angular 8, TypeScript, JavaScript ES6+, NgRx, RxJS, HTML5, CSS3, Bootstrap, Apache Kafka, IBM MQ, JMS, AWS (EC2, S3, Lambda, IAM, RDS), Docker, Docker images, Kubernetes, Kubernetes Helm, Jenkins, Jenkins pipelines, GitHub Actions, Maven, SonarQube, JUnit 5, Mockito, OAuth 2.0, OAuth 2.0 JWT, JWT, Git, TDD, BDD, BDD Cucumber, Agile/Scrum.

Client: State of California, Sacramento, CA

Dec 2017 – Dec 2019

Role: Java Full Stack Developer

Project Title: Government Services Management System

Project Description: Developed a Spring Boot and Angular system to modernize government service delivery. The Spring Boot Microservices platform with Angular TypeScript SPAs, Docker Kubernetes deployments, Jenkins GitHub Actions CI/CD pipelines, OAuth 2.0 authentication, and BDD Agile/Scrum practices reduced processing delays for citizens.

Responsibilities:

- Designed scalable **Java 8/11 Microservices** using **Spring Boot 2.x**, **Spring Cloud**, **Spring Security**, **Hibernate**, and **JPA** for government service modules. Implemented **OAuth 2.0** authorization code flow and **JWT** token-based authentication across all government **Microservices** endpoints. Improved page load performance by 20% for public-facing government service portals.
- Built responsive **Angular 7** frontends with **TypeScript**, **RxJS**, **Bootstrap**, and **HTML5/CSS3** implementing lazy loading and WCAG compliance. Delivered accessible government service portals for citizens.
- Containerized **Spring Boot Microservices** using **Docker** and deployed on **Kubernetes** via **AWS ECS**. Implemented **CI/CD pipelines** using **Jenkins** and **GitHub Actions** automating **Docker** builds and **Kubernetes** rollouts with **SonarQube** quality gates.
- Integrated **Apache Kafka** for asynchronous event messaging and **PostgreSQL**, **MongoDB** for government data management. Applied **BDD** Cucumber testing, **Code reviews**, **SOLID principles**, and **Agile/Scrum** delivery.

Environment: Java 8/11, Core Java, Spring Boot 2.x, Spring Cloud, Spring Security, Spring MVC, Hibernate, JPA, Microservices, REST APIs, Angular 7, TypeScript, JavaScript ES6+, RxJS, HTML5, CSS3, Bootstrap, Apache Kafka, AWS (S3, EC2, ECS), PostgreSQL, MongoDB, Docker, Kubernetes, Jenkins, GitHub Actions, Maven, SonarQube, JUnit 5, Mockito, OAuth 2.0, OAuth 2.0 JWT, JWT, Git, TDD, BDD, BDD Cucumber, Agile/Scrum.

Client: Target, Chicago, IL

Oct 2016 – Nov 2017

Role: Java Developer

Project Title: Retail Inventory and Order Management System

Project Description: Built a Spring Boot and Angular 5 centralized retail system. The Spring Boot REST APIs backend with Angular TypeScript frontend, Docker containerization, Kubernetes orchestration, Jenkins pipelines, GitHub Actions CI/CD, OAuth 2.0 security, BDD Cucumber testing, and Agile/Scrum delivery replaced fragmented inventory workflows.

Responsibilities:

- Developed **Angular 5** UI screens with **TypeScript**, **HTML5**, **CSS3**, and **JavaScript ES6+** for retail operations. Built **Spring MVC** service layer using **Hibernate** and **JPA** DAO entities for inventory data access.
- Integrated **Apache Kafka** topics and **Kafka Streams** consumer groups for asynchronous order processing across retail **Microservices**. Reduced order processing latency by 20% through fault-tolerant **Kafka** dead-letter queue patterns.
- Built **Docker** images for retail **Microservices** and orchestrated with **Kubernetes** on **AWS ECS**. Implemented **CI/CD pipelines** with **Jenkins** automating **Docker** builds and **Kubernetes** rollouts. Applied **JUnit 5**, **Selenium**, **Mockito**, **TDD**, **BDD**, and **Code reviews** enforcing **SOLID principles**.

Environment: Java 8, Core Java, Spring Boot, Spring MVC, Hibernate, JPA, REST APIs, Angular 5, TypeScript, JavaScript ES6+, HTML5, CSS3, Bootstrap, Apache Kafka, Kafka Streams, AWS (S3, IAM, ECS, CloudFormation), Docker, Kubernetes, Kubernetes Helm, Jenkins, Maven, JUnit 5, Mockito, Selenium, Git, TDD, BDD, Agile/Scrum.

Client: JetBlue Airways, Long Island City, NY

Mar 2015 – Sep 2016

Role: Java Developer

Project Title: Airline Customer Booking System

Project Description: Created a Spring Boot and Java digital solution for airline booking using Docker Kubernetes deployment, Jenkins GitHub Actions CI/CD, OAuth 2.0 security, BDD Cucumber testing, BDD Gherkin scenarios, TDD practices, and Agile/Scrum delivery. Consolidated booking, ticketing, and records management into a reliable platform.

Responsibilities:

- Developed scalable **Java 7 J2EE** applications using Servlets, JSP, **Hibernate** ORM, and **JPA** for airline booking systems. Containerized booking services using **Docker** and deployed on **Kubernetes** clusters. Optimized **MySQL** and **PostgreSQL** queries and stored procedures for high-throughput booking workflows.
- Built **SOAP** web services using **JAX-WS** and **JAXB** and **REST APIs** for external airline partner integrations. Led automated **Selenium** UI testing with **JUnit 5** and **BDD** Cucumber scenarios achieving 84% code coverage and reducing cross-browser defects by 28%.
- Implemented **Jenkins** and **GitHub Actions CI/CD pipelines** with **Maven** build automation. Applied **Code reviews**, **SOLID principles**, **TDD**, **BDD**, and **Agile/Scrum** practices across all delivery cycles.

Environment: Java 7, Core Java, J2EE, Servlet, JSP, HTML5, CSS3, JavaScript, Hibernate ORM, JPA, SOAP, JAX-WS, JAXB, REST APIs, PL/SQL, JDBC, JBoss, MySQL, PostgreSQL, MongoDB, Jenkins, Jenkins CI/CD, GitHub Actions, JUnit 5, Selenium, Mockito, Maven, Git, SVN, Log4J, TDD, BDD, BDD Cucumber, Agile/Scrum.