PREETHI KONDURU
+1(557) 228-1175 | preethikonduru38@gmail.com |
LinkedIn | GitHub | Portfolio

Summary

Senior Software Developer with 8+ years of experience delivering enterprise-grade, cloud-native applications across insurance, healthcare, and banking domains. Strong expertise in Java 8–17, Spring Boot microservices, and event-driven architectures (Kafka) with secure REST/GraphQL APIs. Proven ability to deliver scalable UI solutions using Angular and React, while driving backend reliability through API versioning, validation contracts, caching strategies, and production-ready CI/CD. Hands-on with Kubernetes deployments, IaC, OAuth2/JWT security, and operational readiness in regulated environments.

Technical Skills

JAVA/J2EE Technologies	Spring Boot (2.x/3.x), Spring MVC, Spring WebFlux, Spring Security, Spring Data JPA, Hibernate, Spring Batch, Spring Cloud (Config, Gateway)
Programming Languages	Java (8/11/17), TypeScript, JavaScript, SQL, Python, PL/SQL, C, C++
Frontend:	Angular 16, NgRx, RxJS, Angular Material, Angular Router, React, HTML5, CSS3, Bootstrap
Backend& Frameworks:	Microservices, REST APIs, GraphQL, OpenAPI/Swagger, Resilience4j
Database	PostgreSQL, Oracle, MySQL, MongoDB, Redis
Messaging Servers	Apache Kafka, RabbitMQ, Apache Tomcat
Cloud & DevOps:	AWS (EKS, ECS, RDS/Aurora, S3, Lambda, CloudWatch, IAM), GCP (GKE, Cloud SQL, Cloud Storage), Docker, Kubernetes, Helm, Terraform, Jenkins, GitHub Actions, ArgoCD
Web / App Servers:	Kubernetes (EKS/GKE), ECS, Apache Tomcat
Web Services:	REST, GraphQL, OAuth2/JWT Security
Development Tools:	IntelliJ IDEA, VS Code, Eclipse, Maven, Gradle, npm, JIRA
Testing:	Junit (4/5), Mockito, Jest, Cypress, TestContainers
SDLC & Design Patterns:	Agile/Scrum, TDD, CI/CD, SDLC, SOLID, Design Patterns, API Versioning
AI & Productivity Tools:	AI/LLM-based summarization workflows, Prompt templates, Structured JSON validation

Professional Experience

Role: Senior Software Developer **May 2024 – Present**
Client: Geico, Maryland, USA

Project description:
Led development of a modernized Policy Administration and Claims Processing platform supporting underwriting, premium billing, claim intake, adjudication, and regulatory workflows. The solution is built on Java 17 and Spring Boot microservices with event-driven processing using Kafka, and deployed on AWS EKS for scalable, secure operations. Delivered responsive user experiences through Angular-based portals integrated via secure APIs, supporting both customer self-service and internal adjuster workflows.

Frontend:

- Designed and built responsive insurance portals using Angular 16 + TypeScript, integrating secure REST APIs for policy servicing and claims workflows.
- Implemented predictable state handling using NgRx + RxJS, improving workflow stability across multi-step user journeys.
- Developed reusable UI components and Angular Material form patterns to standardize validations and enterprise UX across modules.
- Enabled centralized error handling and request lifecycle management using Angular interceptors for consistent API-driven behavior.
- Maintained unit test coverage using Jest, validating UI workflows dependent on backend-driven rules.

Backend:

- Established microservice boundaries for underwriting, billing, and claims domains, enabling independent deployments and clear ownership.
- Designed secure REST and GraphQL APIs using Spring Boot, publishing OpenAPI/Swagger documentation and supporting API versioning (v1/v2).
- Implemented authentication and authorization using Spring Security (OAuth2/JWT, RBAC) to support regulated insurance workflows.
- Optimized transactional workflows using PostgreSQL tuning and implemented Redis caching for high frequency reads and stable response times.
- Implemented Kafka-driven workflows for policy and claim lifecycle processing with consumer group scaling and fault-tolerant handling.
- Applied reliability controls including idempotency, retry/backoff, and DLQ handling to prevent duplicate processing and isolate failures.
- Tuned PostgreSQL queries and implemented Redis caching for high frequency reads to improve throughput and response stability.

Cloud:

- Deployed services on AWS EKS using Docker and Helm, enabling scalable rollouts and stable production deployments.
- Managed CI/CD pipelines using Jenkins + GitHub Actions and GitOps deployments via ArgoCD to standardize release execution.
- Provisioned AWS infrastructure using Terraform, improving repeatability and auditability across environments.
- Implemented CloudWatch monitoring and alerting for operational visibility and faster issue detection.
- Supported production readiness through deployment validation, release coordination, and incident triage support.

Environment: Java 17, Spring Boot 3.x, Spring Security, Spring Data JPA, Hibernate, REST/GraphQL APIs, OpenAPI/Swagger, Kafka, Resilience4j, PostgreSQL, Redis, Angular 16, TypeScript, NgRx, RxJS, Angular Material, HTML5, CSS3, AWS (EKS, RDS/Aurora, S3, Lambda, CloudWatch, IAM, VPC, Secrets Manager), Docker, Kubernetes, Helm, Terraform, Jenkins, GitHub Actions, ArgoCD, Prometheus, Grafana, ELK Stack, JUnit 5, Mockito, TestContainers, Jest, Cypress, Git, GitHub, Maven, npm, IntelliJ IDEA, JIRA

Role: Software Developer **Nov 2021 -Dec 2023**

Client: DSS Inc, Florida, USA

Project description:

Contributed to delivering a HIPAA-aligned healthcare claims platform supporting adjudication workflows, eligibility validation, provider operations, and compliance reporting. Built full-stack features using React-based dashboards and Spring Boot services supporting secure claims processing and medical document workflows. Enabled event-driven processing using Kafka Streams and improved scalability through microservice-based design. Supported AWS container deployments and worked in Agile teams to deliver enhancements with strong compliance and operational discipline.

Frontend:

- Built claims workflow dashboards using React + TypeScript, enabling document actions, claim review screens, and approval flows through secure API integrations.
- Developed reusable React components for forms, grids, and workflow modules to standardize UI behavior across claims processing features.
- Implemented role-based access and guarded routing to align UI workflows with healthcare authorization requirements.
- Improved usability and responsiveness for large datasets by optimizing API-driven screens with efficient rendering patterns.
- Supported accessibility practices aligned with WCAG standards for compliance-driven healthcare workflows.

Backend:

- Designed REST and GraphQL APIs using Spring Boot, publishing OpenAPI/Swagger specifications and standardized error contracts for integrations.
- Supported decomposition of legacy modules into microservices, enabling independent scaling and service ownership boundaries.
- Implemented Kafka Streams workflows for claims lifecycle processing and asynchronous event-driven transitions across distributed services.
- Built idempotent event handlers using claimId + event versioning and implemented retry/backoff + DLQ handling to isolate failures.
- Optimized Oracle persistence and implemented Redis caching strategies to reduce database contention and improve API responsiveness.
- Implemented secure access controls using Spring Security (OAuth2/JWT) for HIPAA-aligned healthcare workflows.

Cloud:

- Supported AWS EKS deployments for containerized services using Docker and Kubernetes rollout strategies.
- Managed S3-based storage integration for healthcare documents and enabled CloudWatch monitoring for operational visibility.
- Maintained CI/CD pipelines using Jenkins and GitHub Actions with automated builds and test execution.
- Supported IAM access controls aligned with HIPAA security standards.
- Participated in production support for incident triage and stabilization fixes.

Environment: Java 8/11, Spring Boot 2.x, Kafka Streams, RabbitMQ, Oracle, Redis, AWS EKS, Docker, Kubernetes, Jenkins, GitHub Actions, JUnit 5, Mockito, Git, GitLab, IntelliJ IDEA, Maven/Gradle, npm, JIRA, HIPAA

Role: Software Developer Feb 2020– Oct 2021
Client: Atlantic Union Bank

Project description:
Built and enhanced secure digital banking services supporting fund transfers, bill payments, account management, and transaction monitoring. Developed Angular workflows integrated with Spring Boot APIs and Kafka messaging, supporting secure access controls and container-based deployments on GCP.

Frontend:

- Developed customer-facing workflows using Angular + TypeScript, implementing reactive forms, validations, and reusable UI components.
- Built monitoring dashboards using RxJS patterns to display real-time updates across account and transaction flows.
- Implemented NgRx state handling to maintain consistent UI behavior across onboarding and payment journeys.
- Standardized reusable Angular components to improve maintainability across digital banking modules.

Backend:

- Developed Spring Boot REST APIs with standardized request validation, error handling, and API documentation for transaction services.
- Integrated Kafka messaging workflows for fraud alerts and transaction notifications with retry-safe consumer handling.
- Implemented Spring Security with OAuth2/JWT and RBAC enforcement for secured access patterns.
- Maintained unit test coverage using JUnit + Mockito and supported persistence across PostgreSQL and MongoDB.

Cloud:

- Supported deployments on GCP GKE, assisting with container rollout practices and operational monitoring.
- Worked with Cloud SQL and Cloud Storage to support persistence and secure document handling.
- Supported CI workflows to improve deployment consistency across development and staging environments.

Environment:
Java 8/11, Spring Boot, Kafka, PostgreSQL, MongoDB, Angular, TypeScript, NgRx, RxJS, GCP (GKE, CSQL, Cloud Storage), Docker, Kubernetes, JUnit, Mockito, Cypress, OAuth2/JWT, Git, Bitbucket, Eclipse/IntelliJ IDEA, Maven, npm, JIRA

Role: Software Developer **Jun 2018 – Jan 2020**
MindMade Technologies, Hyderabad, India

Project Description:
Developed enterprise web applications supporting inventory tracking, order processing, reporting workflows, and customer data management. Delivered Spring Boot REST APIs with MySQL persistence and Angular-based UI, with foundational AWS exposure in deployments and troubleshooting.

Frontend:

- Built Angular modules using routing, reusable components, and reactive forms to support business data entry workflows.
- Implemented responsive UI using HTML/CSS and Bootstrap for cross-browser compatibility.
- Built search, filtering, sorting, and pagination features for inventory and order datasets.
- Implemented validation and user-friendly error handling to improve data quality.

Backend:

- Developed REST APIs using Spring Boot for CRUD operations and workflow execution across inventory and order processing.

- Integrated Spring Security with OAuth2/JWT for secured endpoint access and protected data handling.
- Designed and optimized MySQL schemas and queries using indexing and tuning for improved performance.
- Built unit tests using JUnit 4 + Mockito and documented APIs using Swagger/OpenAPI.

Cloud:

- Gained hands-on exposure to AWS services including EC2, S3, RDS, and Elastic Beanstalk for deployment fundamentals.
- Supported build packaging and Elastic Beanstalk environment updates across development and QA stages.
- Assisted in deployment troubleshooting by validating application health and reviewing CloudWatch logs/metrics.

Environment: Java 8, Spring Boot, MySQL, Angular, TypeScript, AWS (EC2, S3, RDS, Elastic Beanstalk, CloudWatch), Maven, Git, Jenkins, JUnit 4, Mockito, Git, GitLab/Bitbucket, Eclipse, npm, JIRA, VS Code

Education

Masters in Computer and Information Sciences - Saint Louis University
Bachelors in Electronics and Communication Engineering - JNTUH, Hyderabad