Anoop Reddy Repaka

Available to relocate nationwide | Anooprepakaz@gmail.com | +14692746536 | Linkedin

Professional Summary:

- **Around 8 years of comprehensive experience** in the **Software Development Life Cycle (SDLC)** with a strong emphasis on **backend technologies** and **Java/J2EE**.
- **Proficient in Core Java SE 8**, including **Generics, Streams, Lambda expressions, Collection API**, and **Exception Handling**.
- **Expertise in Java frameworks**: Spring Boot, Spring MVC, Spring Security, Spring Data, Project Reactor, RxJava, and Akka for building scalable and resilient backend systems.
- **Skilled in event-driven architectures** and **real-time data processing** using **Kafka**, **RxJava**, **Project Reactor**, and **Akka**, ensuring efficient asynchronous communication and data integration.
- **Experienced in designing and implementing microservices architectures** with **Spring Boot** and **Kotlin**, leveraging **dependency injection**, **configuration management**, and both **synchronous and asynchronous inter-service communication**.
- **Proficient in creating and managing RESTful APIs** and **GraphQL endpoints** for flexible and secure data interactions.
- **Advanced knowledge of ORM frameworks**: Hibernate and Spring Data JPA for efficient data persistence and complex relational mappings across **relational** and **NoSQL** databases like **MySQL, PostgreSQL, Oracle, MongoDB**, and **Cassandra**.
- **Cloud computing expertise**: AWS (EC2, S3, RDS, Lambda, ECS, EKS), **Google Cloud Platform (GCP)**, and **Azure**, with strong skills in **infrastructure as code** using **Terraform** and **CloudFormation**.
- **CI/CD pipeline automation** using **Jenkins**, **GitLab**, **Azure DevOps**, along with **containerization** using **Docker**and **orchestration** with **Kubernetes**, ensuring efficient and reliable deployment processes.
- **Test-Driven Development (TDD)** practitioner, utilizing **JUnit** and **Jest** to maintain high code quality and application reliability.
- Skilled in testing frameworks **Selenium** and **Cucumber**, using them to automate UI testing and implement Behavior-Driven Development (BDD) practices, ensuring reliable and user-focused application behavior.
- **Agile methodologies** experience, including participation in daily scrum meetings, sprint planning, and delivering high-quality software within dynamic environments.
- **Version control proficiency** with **Git** and **SVN**, and **build automation** using **Maven** and **Gradle** for streamlined project management and continuous integration.
- **Strong problem-solving skills** and a deep understanding of **data structures** and **algorithms**, enabling the development of optimized and high-performance software solutions.
- **Proven leadership** in both independent and cross-functional team settings, driving the successful completion of critical projects and enhancing overall system performance and scalability.
- Possesses in-depth knowledge of **design patterns** and **microservices patterns**, utilizing them to develop maintainable and scalable codebases.

Technical Skills:

Languages	C, Python, Java, Kotlin
Java Technologies	JDBC, Servlets, JSP
Java Frameworks and Testing	Spring, Project Reactor, Akka Spring Boot, Spring MVC, RxJava, Spring Data, Spring Security, Spring Rest, Selenium, Cucumber
Web Technologies	HTML5, CSS3, JavaScript, AJAX, Bootstrap, GraphQL, REST and Soap Apis

Web Frameworks	Angular, React.js, Express.js
Open Source	JUNIT, log4j, GIT, CVS, SVN.
Operating Systems	Windows, UNIX, LINUX
Cloud	AWS, GCP, AZURE
Databases	Oracle, SQL Server, My-SQL, MongoDB, Cassandra, IBM Db2
CI/CD Tools	Jenkins, Gitlab, AWS Code Pipeline, Azure DevOps, Maven, Gradle
Containerization Tool	Docker, AWS Fargate
Orchestration Tools	Kubernetes, AWS ECS, AWS EKS.
Certifications	AWS Certified Solutions Architect-Associate, AWS Certified Cloud Practitioner, Microsoft Certified Azure Platform Developer Associate Microsoft Certified Azure Fundamentals

Professional Experience:

Client: NextGen, Atlanta, GA, USA **September 2022 - Present Role:**
Sr. Full stack Developer

Project Description:

I contributed to the development of a comprehensive healthcare management platform aimed at enhancing patient care, clinical workflows, and data integration across healthcare facilities. This platform is designed to streamline Electronic Health Record (EHR) systems, billing management, and patient communication while ensuring compliance with healthcare regulations and data security standards (HIPAA). The system is highly scalable and supports real-time data access, making it possible for healthcare providers to deliver efficient, informed care.

Responsibilities:

- Enhanced system responsiveness and scalability by utilizing **RxJava** and **Project Reactor**, maintaining HIPAA compliance.
- Engineered reactive pipelines enabling instantaneous updates to patient records, scheduling, and treatment plans, ensuring secure data handling.
- Built and maintained modular and scalable microservices using **Java** and **Kotlin** with **Spring Boot**, facilitating a 50% increase in deployment frequency and improved system maintainability.
- Leveraged **Akka** to develop concurrent, distributed, and resilient message-driven services, enhancing the platform's ability to handle high-throughput data streams securely.
- Developed **RESTful APIs** and **GraphQL** endpoints with **Spring Boot** and **Project Reactor**, enabling efficient, secure, and flexible data retrieval and manipulation.
- Seamlessly integrated third-party services such as insurance billing and laboratory systems through secure, high-performance APIs, ensuring interoperability. Also created an **Angular** single-page application to facilitate user interactions and streamline workflows.
- Utilized **Cassandra** for distributed database solutions to manage large-scale patient data, ensuring high availability and fault tolerance with 99.99% uptime.
- Enhanced data access layers using **Spring Data JPA** and **Hibernate**, enabling efficient CRUD operations and batch processing for bulk patient data transactions, ensuring data integrity.
- Deployed **Apache Kafka** for high-throughput, low-latency messaging, supporting real-time data ingestion and processing across the platform securely.
- Set up **Kafka Streams** and **Akka Streams** for real-time analytics and monitoring of patient and operational data, ensuring data confidentiality and integrity.
- Containerized microservices using **Docker** and orchestrated with **Kubernetes** on **AWS EKS**, ensuring scalable and resilient deployments.
- Streamlined continuous integration and delivery by automating **CI/CD pipelines** with **Jenkins** and **Maven**, integrating **Git** for version control, enabling faster deployment cycles while maintaining HIPAA standards.
- Ensured secure authentication and authorization across all services using **Spring Security** and **OAuth 2.0**, enforcing Role-Based Access Control (**RBAC**) to restrict access to sensitive patient data, maintaining HIPAA compliance.

- Leveraged **Spring Boot Actuator** and **Akka Monitoring Tools** to monitor application health, performance metrics, and system logs in real-time, facilitating proactive issue resolution.
- Utilized **Splunk** for centralized logging and advanced analytics, enabling proactive identification and resolution of performance bottlenecks while maintaining data security.
- Employed **Test-Driven Development (TDD)** methodologies, writing comprehensive **JUnit** test cases to ensure the reliability and robustness of backend services in a HIPAA-compliant environment.
- Performed integration and performance testing using **Postman** and custom frameworks to validate API functionality and system scalability, ensuring HIPAA compliance.
- Refactored and migrated legacy **Java** microservices to **Kotlin** with reactive extensions, improving code maintainability and system performance.
- Integrated reactive programming paradigms into existing Electronic Health Record (**EHR**) modules, reducing latency, increasing throughput, and ensuring secure data handling.

Environment: Java, RESTful Web Services, Project Reactor, Akka, Micro Services, Angular , Spring Boot Actuator, Spring Data JPA , Akka Monitoring Tools,Spring Core, Spring MVC, Spring Batch, Spring Boot, Kafka, Linux, Cassandra, Kotlin, Maven, Jenkins, GIT, Python, Splunk, JUNIT, JSON, Docker, EKS, AWS, Postman, GraphQL, RxJava.

Client: Western Union, Austin, TX, USA. **December 2020 – August 2022**
Role: Full Stack Developer
Project Description

I was part of the settlements team at Western Union, a financial services company, where I contributed to the development of their Global Transaction platform. This cutting-edge platform was created to improve the speed, reliability, and security of money transfers across Western Union bank accounts worldwide. My focus was on ensuring accurate and timely settlements for various financial transactions, such as ACH transfers, overdrafts, bill payments, and instant funding. The system was designed to accommodate user preferences for both speed and cost-effectiveness, enhancing the customer experience while ensuring compliance with financial regulations.

Responsibilities:

- Developed and maintained secure, high-performance Java applications using **Spring Boot** and **Project Reac**tor for Western Union Bank, ensuring robust, reactive, and reliable software solutions.
- Led the design and development of a comprehensive banking application utilizing the **Hibernate** framework and reactive technologies, optimizing data persistence, retrieval, and complex relational modeling with **IBM DB2** as the backend database.
- Architected and implemented infrastructure as code using **AWS CloudFormation** and **Terraform**, automating the provisioning and scaling of AWS resources.
- Developed **batch processing system**s for automated account reconciliation, reducing manual effort by 40% and streamlining operations.
- Integrated **Elasticsearch** into the banking tech stack for real-time data synchronization and seamless updates across applications.
- Spearheaded efforts to identify and automate recurring issues, enhancing operational stability and reducing system downtime.
- Utilized **Amazon SQS** for real-time processing, enabling efficient message queuing and enhancing the responsiveness of transaction workflows.
- Championed technology communities of practice, fostering innovation and the adoption of cutting-edge software engineering tools and practices, including **microservices**, **reactive design patterns**, and **Angular** for front-end development.
- Ensured strict adherence to Java coding standards and industry best practices, specifically tailored to high-stakes banking applications by using **SonarQube** and integrating these tools into the **CI/CD** pipeline to ensure all code is analyzed before merging to the main branch in **GIT**.

- Written unit tests using the **JUnit** framework for unit and integration testing.
- Maintained thorough documentation of the code using **JavaDoc**, including API specifications and architectural decisions to promote understanding and adherence to standards.
- Led cross-functional teams through the **full software development lifecycle**, from design and development to deployment, delivering multiple software products on time.
- Employed **Apache Spark Core** to process high-volume financial transaction logs, enabling accurate and efficient reconciliation for settlements.
- Integrated **Spark Streaming** with **Kafka** to monitor real-time transaction data, identifying anomalies and detecting potential fraud patterns.
- Implemented robust automated testing frameworks using **Selenium** for UI testing and **Cucumber** for Behavior-Driven Development (BDD), achieving 95% test coverage, improving collaboration, and significantly reducing production defects.
- Used **Spark SQL** to generate reports from multi-terabyte transaction datasets stored in **Cassandra**, ensuring timely compliance with financial regulations.
- Processed and transformed financial data for analytical models, leveraging Spark's distributed computing capabilities to deliver actionable insights for business operations.

Environment: Spring, Spring Boot, Project Reactor, Angular, Spring Data JPA, SonarQube, Java, Spring MVC, Kafka, Agile Methodology, JIRA, GIT, Docker, AWS(Lambda, SQS, SNS, S3,Cloud Formation) , Linux, Cassandra, Kotlin, Junit, Selenium, Gradle, Splunk, IntelliJ, Maven, Apache Spark.

Client: Nordstrom, Seattle, WA, USA. **December 2018 - November 2020**
Role: Software Developer

Project Description: As a Java Developer at Nordstrom Technology, I was part of a collaborative team tasked with enhancing the online shopping experience by developing a seamless and efficient checkout process. This project involved integrating multiple services, such as payment gateways, inventory management, and order processing systems, to create a cohesive user journey. We focused on enabling real-time updates to ensure customers received instant feedback on their orders and cart status, which significantly improved user engagement. Additionally, our team implemented solutions that allowed the application to scale effectively during peak shopping seasons, ensuring reliable performance even with increased traffic. Together, we aimed to provide a frictionless shopping experience that not only met but exceeded customer expectations.

Responsibilities:

- Designed and developed a RESTful API using **Java 11** and **Spring Boot** to facilitate communication between the front-end and back-end services. This API handled requests such as retrieving product details, processing payments, and managing user carts.
- The API adhered to REST principles, ensuring stateless communication and leveraging HTTP methods effectively for CRUD operations.
- To improve performance and user experience, I implemented **Apache Kafka** for asynchronous communication between microservices. This decoupled our services, allowing them to operate independently and handle spikes in traffic without performance degradation.
- Utilizing **Hibernate** for ORM, I created mapping files based on business logic and object relationships, facilitating efficient data management.
- I employed **Java 8** features such as Streams and Lambda expressions to perform operations like filtering and mapping on collections of data, enhancing code readability and performance.
- Collaborating with the front-end team, I ensured that the **React JS** components consumed the REST API effectively. The components displayed real-time updates for the shopping cart and order status, improving the user interface and experience.
- By implementing state management using the **Flux architecture**, I ensured that our application handled data flow predictably, which was critical for features like real-time cart updates.
- To manage deployment and scalability, I utilized **AWS ECS (Elastic Container Service)** to orchestrate Docker containers, ensuring our services were efficiently deployed and maintained.
- I also implemented **AWS Lambda** functions for serverless processing of tasks like sending confirmation emails or updating analytics, further enhancing system efficiency.

- Configuring **AWS CloudWatch** allowed me to monitor system health and performance in real time. I set up alarms for key metrics to proactively address any issues, ensuring a reliable shopping experience for our users.
- I managed **AWS IAM** roles to enforce strict access controls, ensuring that only authorized services could communicate with each other and access sensitive data.
- Leveraged **Apache Spark** to process and analyze large volumes of clickstream data, providing insights into customer shopping patterns and enabling personalized recommendations.
- Used **Spark Streaming** integrated with **Kafka** to handle real-time data streams, ensuring smooth updates to shopping carts and order statuses during high-traffic periods.
- Analyzed historical sales data using **Spark SQL** to predict demand trends, optimize inventory levels, and reduce stockouts during peak shopping seasons.
- Utilized **Spark Core** for ETL processes to clean and transform data from multiple sources, ensuring a unified data format for inventory management and business intelligence systems.

Environment: Java 11, RESTful Web Services, React, Kafka, Hibernate, Spring Batch, Spring, Spring MVC, Spring transaction, Spring Security, AWS(ECS, Lambda, CloudWatch, IAM), Apache Spark.

Client: DataDot Software Solutions, Hyderabad, India. June 2017–November 2018
Role: Java Developer
Project Description:
Spearheaded the development and maintenance of dynamic web applications using Java, Spring Boot, and AngularJS, collaborating closely on feature design and implementation. Engineered a high-performance ticketing system and an employee task management portal, driving efficiency through Agile methodologies, rigorous unit testing, and swift bug resolution. Leveraged Java 1.8 to enhance system performance, developed scalable microservices, and integrated RESTful APIs for seamless functionality. Streamlined workflows via CI/CD pipelines, managed the codebase using Git, and employed Docker for efficient containerization. Led performance testing with Apache JMeter, ensuring system reliability while aligning business objectives with technical execution through cross-team collaboration.
Responsibilities:
- Contributed to the development and maintenance of robust web applications using **Java**, **Spring Boot**, and **AngularJS**.
- Collaborated with cross-functional teams to design and implement new features, ensuring seamless integration with existing systems.
- Developed a **responsive ticketing system** to efficiently manage and organize user inquiries related to company products.
- Played a key role in creating an **employee portal**, delivering features such as weekly agendas, to-do lists, and messaging systems, enhancing internal communication.
- Conducted **unit testing** and **bug fixing** to ensure high-quality application functionality, adhering to **Agile** development methodologies.
- Developed **RESTful APIs** for seamless communication between system components.
- Managed the codebase using **Git** for version control, ensuring clean and efficient collaboration.
- Leveraged **Java 1.8** features to enhance application performance and code efficiency.
- Implemented **CI/CD pipelines** to automate build, testing, and deployment processes, significantly improving development workflows.
- Conducted **performance testing** and optimization to boost system responsiveness and efficiency.
- Worked closely with frontend teams and stakeholders to align technical solutions with business objectives, ensuring impactful project outcomes.
- Designed and implemented **APIs** for user ticket data handling, optimizing server and frontend interactions in the ticketing system.
- Managed cloud resources using **Azure Blob Storage** for secure data backup and archival.
- Participated in **Azure** training sessions to improve cloud service implementation and adherence to best practices.
- Ensured optimized database design and efficient data retrieval in the Employee Portal using **Hibernate**.
- Utilized **Azure DevOps** for issue tracking and project management, maintaining transparency and efficiency across projects.

- Contributed to the adoption of **microservices architecture**, breaking down complex systems into independent, scalable components.
- Applied **Java 1.8** features to streamline the performance of microservices, improving backend operations and maintainability.
- Employed **Mockito** and **JUnit** for rigorous unit and integration testing, ensuring microservices' stability and reliability.
- Integrated **Docker** containerization, simplifying the development and deployment of microservices for greater portability.

Environment: Java, Spring Core, Spring, Hibernate, RESTful Web Services, Docker, JUnit, Oracle, GIT, RESTful APIs, Azure (Blob Storage, DevOps), Microservices architecture, Mockito, JUnit, Agile methodologies.

Education:

Master’s in computer science, Arizona State University, Tempe, Arizona
Bachelor’s in computer science, JNTUH, Hyderabad, India