

Lakshmi Reddy Sanikommu

Senior Java Full Stack Developer

Unitedstates

502-509-6866

srilakshmireddysanikommu@gmail.com

[linkedin.com/in/laxmireddy-s](https://www.linkedin.com/in/laxmireddy-s)

PROFESSIONAL SUMMARY

- IT professional with **11+** years of experience as a Full Stack Developer specializing in the design, development, analysis, testing, and implementation of distributed web-based and client/server applications using **Java/J2EE** technologies.
- Proficient in **UI** design, having developed front-end applications using **JSP, HTML5, CSS3, JavaScript, AJAX, jQuery, ReactJS, Angular 11, Typescript, and ECMAScript**.
- Extensive experience with **ReactJS**, including **React Flux** architecture, **React-Router** for dynamic routing, and **Redux/Redux Toolkit** for state management.
- Integrated **PostgreSQL, MySQL, and MongoDB** with Golang using **GORM** and native database drivers.
- Expert in cloud-based deployment on **Amazon AWS**, with hand-on experience with **EC2, VPC, S3, EBS, ELB, CloudFront, IAM, RDS, Elastic Cache, SNS, SQS, and CloudWatch**.
- Hands-on experience with **Docker** for containerization, managing **Docker** images, push/pull operations with **Spring Boot**, and **CI/CD** pipelines using **Jenkins**.
- Skilled in building robust **APIs** using **REST** and **SOAP** technologies, leveraging **JAX-WS, JAX-RS, JAXB, Apache CXF** for service-oriented architectures.
- Integrated **Kafka** with microservices, **Spark**, and databases (**PostgreSQL, MongoDB, Cassandra**) to enable seamless data flow across distributed systems.
- Proficient in microservices architecture, working with **Spring Boot, Netflix Eureka**, and implementation using **NodeJS**.
- Extensive knowledge of **Spring** Framework, including **Spring MVC, Spring Boot, Spring Microservices, Spring Batch, and Spring Eureka**.
- Experience with **Java** versions **8** and **17**, using features like **lambda** expressions, method references, **Stream APIs**, and multithreading to improve code readability, performance, and parallel processing.
- Expertise in design patterns such as **Singleton, Business Delegate, DAO, DTO, MVC, and Template** for enterprise application development.
- Hands-on with testing frameworks such as **JUnit, Mockito, and PowerMockito** for unit testing, and **Jasmine, Protractor, and Karma** for front-end testing in **Angular** applications.
- Enabled **SSL/TLS** on WebLogic Server for encrypted communication, ensuring data security.
- Expert in working with databases, including **Oracle, MySQL, SQL Server, DB2, and NoSQL** databases like **MongoDB and Cassandra**, writing complex **SQL** queries, triggers, views, and stored procedures.
- Experience with authentication and authorization, using **OAuth** and **JSON Web Tokens (JWT)** for securing **RESTful** services.
- Familiarity with version control systems such as **Git**, using **GitHub** and **Bitbucket** for source code management, and build tools like **Maven, Ant, and Log4J** for automating processes.
- Used **Azure Resource Manager (ARM)** to deploy, update, or delete all the resources for solution in a single, coordinated operation.
- Used **JIRA** and **Rally** to Bug/Issue tracking and project management.
- Proficient in Linux/Unix/Mac OS X environments, efficiently using command line tools (CLI) for system management and troubleshooting.
- Engineering cloud-based AI/ML pipelines using **AWS SageMaker, TensorFlow, PyTorch, OpenAI APIs, and Step Functions** to enable predictive analytics, intelligent automation, and fraud detection in production
- Extended and enhanced existing test automation frameworks (e.g., **Selenium WebDriver**)

TECHNICAL SKILLS

Programming Languages

Java 8 / 11 / 17, PL/SQL, Scala, Go, Python, Node.js, Servlets, JSP, JSTL, JDBC, JMS, JNDI, EJB, JPA, Struts, Spring, Hibernate, JSF, Spring Boot, Spring Cloud

Web Technologies & Services

JavaScript, jQuery, XML, HTML, DHTML, CSS, Angular 12, Node.js, ReactJS, JAX-WS, JAX-RPC, JAX-RS, SOAP, REST, JAXB, Apache Axis

Testing Tools & Methodologies

JUnit, JIRA, Jasmine, Karma, Jest, Agile, Scrum, TDD (Test-Driven Development), OOAD, SDLC, BDD, UML, Rational Rose

Database Servers & Cloud

Oracle 8i/9i/10g/11g/12c, MS SQL Server 2000/2005, MySQL 4.1/5.1, DB2 7.2/8.1/9.1, MongoDB, Cassandra, Redis, Couchbase, DynamoDB, AWS (Amazon Web Services), Azure, GCP (Google Cloud Platform)

Version Control & Build Tools

CVS, SVN, VSS, Rational Clear Case, Stash, Git, SourceTree, GitHub, Git Desktop, ANT, Maven, Gradle, WebSphere, Jenkins, JBoss, Tomcat, IBM WebSphere, Jenkins, GitLab

EDUCATION

University of Central Missouri

Master's Degree in Computer Science

Aug 2011 – Dec 2012

Warrensburg, MO

Jawaharlal Nehru Technological University

Bachelor's Degree in Computer Science and Engineering

Jul 2007 – Jun 2011

Kakinada, IN

PROFESSIONAL EXPERIENCE

Client: State of Maine, Augusta, ME

Oct 2022 to Present

Role: Sr. Java Full Stack Developer

Responsibilities:

- Developed and maintained a web-based case management system using **Java 17**, **Spring Boot**, and **Angular 11**, automating legal document workflows, case tracking, and compliance reporting. Integrated **RESTful APIs**, **PostgreSQL**, and **AWS** services to enhance security, scalability, inter-departmental data exchange, and document version control.
- Utilized **Java 8** features like **Streams**, **Lambdas**, and **Completable Futures** to enhance backend processing efficiency, improving response times by 15%.
- Built and deployed scalable **RESTful APIs** using **Spring Boot**, enabling efficient microservices architecture across services.
- Integrated **GraphQL APIs** in **Angular** applications for efficient data fetching, reducing **API** response times by 30%.
- Designed complex database queries using **MongoDB** and **Cosmos DB**, improving query performance and scalability for high-traffic applications.
- Implemented **Apache Kafka** for real-time data streaming and event-driven architectures, improving system scalability and data processing efficiency.
- Automated testing with **JUnit** integrated into **Jenkins CI/CD** pipelines, achieving 90% test coverage and reducing manual testing efforts.
- Utilized **Kafka Streams API** for real-time data transformation, aggregation, and enrichment, enhancing analytics capabilities.
- Implemented unit and integration testing using **JMock** to mock dependencies and ensure isolated test execution.
- Integrated **JUnit**, **Mockito**, and **JMock** to achieve **high test coverage** in Java-based applications.
- Secured **Kafka communication** using **SSL/TLS encryption**, **SASL authentication**, and **ACL-based access control**, preventing unauthorized access.
- Implemented the technical transformation for the front end from **Angular 8** to **Angular 11**.
- Developed and maintained a **front-end** application using **Angular** following the **MVC** pattern, incorporating advanced **components**, **directives**, and **services** to enhance user interactivity and provide real-time updates.
- Worked with **Angular data binding** techniques like **string interpolation**, **event binding**, **property binding** and **two-way data binding** to enable communication between the view and the template.
- Utilized **Angular's HTTPClient module** to handle HTTP requests between the frontend Angular application and backend servers.
- Configured **Hazelcast** clustering for real-time data processing, enabling low-latency reads and writes in microservices architectures.
- Worked on implementing **AMQP** to enable reliable messaging and communication between distributed systems.
- Integrated, **Power BI** enables real-time data visualization, reporting, and analytics by connecting with backend services, databases, and **APIs**, providing interactive dashboards and secure access to key metrics.
- Managed **Jenkins CI/CD** pipelines, automating builds, tests, and deployments, reducing deployment time by 40%.
- Implemented **OAuth2** and **JWT** authentication in **Spring Boot** applications, ensuring secure user authentication and **API security**.
- Integrated **Hazelcast** with Java and Spring Boot to optimize session management and cache application data dynamically.
- Integrated **Web Flux** with **MongoDB** and other reactive data stores, enabling seamless reactive data retrieval and processing.
- Implemented serverless functions using **AWS Lambda** to efficiently execute backend logic.
- Implementing and optimizing **HTTP methods** (GET, POST, PUT, DELETE) for **RESTful APIs** in my project to ensure efficient client-server communication.
- Implemented RESTful Web services in Service Oriented Architecture (**SOA**) to retrieve data from client side and made **REST API** calls from Angular and parsed the data to the client.
- Integrated **Camunda** workflows with **microservices**, improving process automation and resolving workflow bottlenecks by 25%.
- Proficiency in utilizing **RabbitMQ APIs** to develop robust and scalable messaging solutions.
- Used **CQL** (Cassandra Query Language) to read from and write data into **Cassandra** database.
- worked on the **Alteryx** reporting workflow creation process, designing and automating data preparation, transformation, and reporting tasks for efficient analytics.
- Utilized **SonarQube** for continuous code quality analysis in **Jenkins**, reducing technical debt and ensuring code quality.
- Deployed Java EE applications on **JBoss/WildFly** application servers, ensuring high availability and load balancing.
- Tuned **JBoss performance** by optimizing thread pools, connection pooling, and JVM memory settings
- Designed, developed, and tested **HTML5**, **CSS3**, **Servlets**, **EJB**, **Bootstrap**, **JavaScript**, **jQuery**, and **Angular** that meet accessibility and web browser standards for the website.
- Applied **Spring Data JPA** to simplify the data access layer in enterprise applications, improving development efficiency and maintainability.
- Leveraged **GCP's Big Data** and analytics tools like **BigQuery**, **Dataflow**, and **Dataproc** to manage, process, and analyze large datasets, integrating these tools into **Java** applications.
- Integrated **Java** applications with various **GCP** services such as **Cloud Storage**, **Cloud SQL**, **Pub/Sub**, **Firestore**, **BigQuery**, and **Cloud Spanner** to manage data and ensure high availability.
- Wrote Controller classes in **Spring MVC** framework in Web layer of the application architecture.

- ^ Implemented **GraphQL APIs** on the server-side, defining resolvers that handle queries, mutations, and subscriptions, ensuring accurate data retrieval and manipulation.
- ^ Implemented and maintained infrastructure using **GCP** tools like Cloud Deployment Manager or **Terraform** to provision and manage resources required for **Java**-based applications.
- ^ Leveraged **Spring Data JPA** repositories to streamline **CRUD** (Create, Read, Update, Delete) operations on database tables.
- ^ Integrated messaging services like **SNS/SQS** for asynchronous communication and event-driven workflows.
- ^ Proficiently used **Message Queues** to implement job queues, event-driven systems, and/or data streaming applications.
- ^ Enabled SSL/TLS on **WebLogic Server** for encrypted communication, ensuring data security.

Environment: Java 17, Spring Boot, Angular, Redux, GraphQL, REST APIs, MongoDB, Cosmos DB, WebSocket's, GCP, Jenkins, CI/CD, JUnit, SonarQube, OAuth2, JWT, Docker, Kubernetes, Redis, Kafka, RabbitMQ, HTML, CSS, JavaScript, Eclipse, App Insights, JMeter, Selenium, Log4j, GitHub.

Client: North winds, Roanoke, VA

Jan 2021 to Sep 2022

Role: Java Full Stack Developer

Responsibilities:

- ^ Developed a real-time claims processing engine using **Java 11**, **Spring Boot**, and **Microservices**, reducing adjudication time by 40%, with integration of distributed tracing (Zipkin), **OpenTelemetry**, and fault-tolerant **Kafka** consumers for accurate claims auditing and end-to-end data visibility.
- ^ Developed dynamic user interfaces with **React**, **TypeScript**, **ES6**, and **JavaScript**, improving front-end performance and user experience.
- ^ Built single-page applications (SPAs) using **ReactJS**, **React Router**, and **Redux** for efficient state management and navigation.
- ^ Simplified database interactions by integrating **Spring Boot** with **Hibernate**, **Spring Data JPA** for optimized data access.
- ^ Designed and implemented **RESTful APIs and gRPC services** with Go, improving data exchange efficiency between services.
- ^ Optimized **SQL queries and indexing strategies**, reducing database query execution time by **40%**.
- ^ Wrote unit and integration tests using **Go testing framework**, achieving **90%+ test coverage** for robust applications.
- ^ Designed and implemented **RESTful APIs** using **Java 17** and **Spring Boot**, ensuring scalable microservices architecture.
- ^ Leveraged **WebSocket** protocols in **React** applications to support real-time communication and dynamic user interactions.
- ^ Implemented Single Sign-On (SSO) and role-based access control (RBAC) using **Keycloak** to enhance authentication and authorization processes.
- ^ Deployed containerized applications using **Docker** and orchestrated them with **Kubernetes**, ensuring scalability and efficient microservice management.
- ^ Leveraged **AWS EC2**, **S3**, **Lambda**, and **DynamoDB** to build HIPAA-compliant, cost-effective infrastructure for claim record processing, enabling encryption-at-rest, access logging, lifecycle rules, and IAM role segregation.
- ^ Monitored application performance and system health using **CloudWatch** for detailed logs and metrics.
- ^ Created and maintained **CI/CD** pipelines using **Jenkins**, **GitLab**, and **AWS**, streamlining automated testing, builds, and deployments.
- ^ Leveraged **Spring Web Flux** to build reactive, non-blocking web applications, ensuring high performance and scalability in handling asynchronous data streams and real-time updates for user interfaces.
- ^ Utilized **Web Logic** for deploying and managing enterprise-level applications, ensuring scalability, high availability, and seamless integration with backend services in Java-based environments. Web Writing **Java** Code for fetching the data from **Cassandra** and Handling Terabytes of data and scheduling jobs on maintenance.
- ^ Deployed and scaled containerized applications using **AWS Fargate** for a high-traffic project.
- ^ Involved in designing user screens and validations using **HTML**, **jQuery**, **Ext JS** and **JSP** as per user requirements.
- ^ Utilized **Bitbucket** as a platform for version control and collaborative efforts, efficiently overseeing the codebase, enhancing team coordination, and maintaining version control accuracy throughout the project.
- ^ Developed static and dynamic Web Pages using **JSP**, **JBoss**, **DROOLS Eclipse IDE**, **Swing.**, **SPA**, **HTML5** and **CSS**.
- ^ Responsible for validation of Client interface **JSP** pages using **Struts** form validations. Integrated **Struts** with **Spring IOC**.
- ^ Utilized **Java 11+** features for improved exception handling, including try-with-resources, enhanced stack traces, and standardized exception classes.
- ^ Integrated **MongoDB** with **Java**-based microservices, providing scalable and efficient database solutions.
- ^ Developed reusable **React** components, enhancing code reusability and maintainability across projects.
- ^ Designed and documented **RESTful APIs** using **Swagger**, enabling clear communication and effective service integration.
- ^ Implemented unit and integration testing using **JUnit**, automated within **Jenkins** pipelines to ensure continuous integration.
- ^ Utilized **AWS CloudFormation** and **Terraform** for infrastructure as code (IaC), ensuring consistent resource provisioning and management.

- ⤴ Utilized **Message Queues** for integrating multiple applications and services within distributed systems.
- ⤴ Utilized Project Reactor with **Web Flux** to handle asynchronous data streams efficiently, reducing latency under high loads.
- ⤴ Designed and optimized database schemas in **MariaDB** to ensure efficient data storage, retrieval, and overall application performance.
- ⤴ Automated cloud service management tasks through **AWS CLI** for improved efficiency.
- ⤴ Created **JavaScript** unit tests using frameworks like Jest to validate web application components. Enhanced code reliability, development efficiency, and user experience through systematic testing.
- ⤴ Utilized **Spring Boot's** testing framework, including **JUnit** and **Mockito**, to write comprehensive unit tests and ensure the correctness and reliability of application components.
- ⤴ Developed reactive, non-blocking APIs using Spring **Web Flux**, enhancing scalability and responsiveness for real-time applications.
- ⤴ Leveraged **Kibana** to identify and troubleshoot application errors by visualizing log patterns and trends.
- ⤴ Developed comprehensive unit tests using **JUnit**, **Mockito**, and/or **Selenium** to ensure the quality and functionality of **microservices**.
- ⤴ Managed relational, NoSQL, and graph databases with Neptune to meet application requirements.
- ⤴ Used **Kibana** to visualize and analyze application logs for real-time issue detection.
- ⤴ Migrated **Camel** Routes from **Apac** Service Mix to **Solace** framework.
- ⤴ Exceeded **KPIs** through strategic application development, aligning solutions with organizational goals.
- ⤴ Skilled in **Jest** testing framework for robust unit and integration tests in web applications. Enforced code quality and bug prevention through continuous testing and test-driven development practices.
- ⤴ Used **Hibernate ORM framework** with **Spring** for data persistence and transaction management.
- ⤴ Used **Hibernate 3.0** object relational data mapping framework to persist and retrieve the data from database.
- ⤴ Wrote **SQL** queries, stored procedures, and **triggers** to perform back-end database operations.
- ⤴ Developed **ANT** Scripts to do compilation, packaging, and deployment in the **WebSphere server**.
- ⤴ Implemented the logging mechanism using **Log4j** and used for tracking the applications and **SVN** used for version control.
- ⤴ Implemented monitoring and alerting using **AWS CloudWatch** and Azure Monitor, ensuring high availability and operational efficiency.

Environment: Java 17, Maven, Hibernate, React JS, HTML, Kafka, RabbitMQ, Apache Tomcat, Json, Java Script, Spring, Spring ribbon, Oracle, Mongo DB, JDBC, Eclipse, JUnit/Mockito, JMeter, Selenium, Jenkins, GitHub, Kubernetes, Docker, Log4j, AWS EC2.

Client: LPL Financial, San Diego, CA

Oct 2018 to Dec 2020

Role: Java Full Stack Developer

Responsibilities:

- ⤴ Designed and developed a high-performance financial transaction processing platform using **Java 11**, **Spring Boot**, **Hibernate**, and **Angular** with **TypeScript**, integrating **OAuth2** and **JWT** security protocols, handling millions of transactions daily with sub-millisecond latency, and ensuring scalability, resilience, and fault tolerance
- ⤴ Developed dynamic user interfaces using **Angular 8**, **TypeScript**, **ES6**, and **JavaScript**, creating seamless and responsive front-end experiences.
- ⤴ Built and optimized single-page applications with **Angular**, implementing state management with **NgRx** and routing for efficient navigation.
- ⤴ Ensured consistency across environments by packaging applications into **Docker** containers and achieved scalability and efficient resource management through **Kubernetes** orchestration.
- ⤴ Developed and maintained **RESTful APIs** using **Spring Boot** and **Java 11**, enabling microservices-based architecture.
- ⤴ Skilled in software development and version control using **GIT**, **GitLab**, **GitHub**, and **Bit bucket**, proficient in build automation with **Maven** and **Gradle**.
- ⤴ Automated trade processing workflows for FX, and Fixed Income products, improving operational efficiency and reducing manual intervention in trade reconciliation and settlement processes.
- ⤴ Designed, implemented, and maintained large-scale data processing applications using **Hadoop** ecosystem components such as **HDFS**, **MapReduce**, **YARN**, and **HBase**.
- ⤴ Developed Automation scripts for new enhancements using **Selenium WebDriver** and **TestNG**.
- ⤴ Analyzed user interactions and transaction trends using **Kibana** for data-driven decisions.
- ⤴ Experienced in using **Cucumber for BDD**, developing feature files and step definitions to ensure clear understanding and fulfillment of business requirements through automated tests.
- ⤴ Expertise in **Graph QL** for efficient data queries, **Graph DB** for complex data relationships, and **Solr** for high-performance search functionality.
- ⤴ Automated performance testing using **JMeter** and **Postman**, ensuring stability under high load and monitoring applications with **Azure** Monitor.
- ⤴ Experience with **gRPC** and **RMI** for **Java** and other languages for remote method invocation and remote code execution.

- ^ Demonstrated expertise in **Scala programming** and leveraging the **Spark** framework for scalable data processing and analytics solutions.
- ^ Extensively used various **Spring Framework modules** like **MVC**, **DI (IOC)**, **Auto Wiring**, **Spring Security**, **Spring Boot**, and **AOP**.
- ^ Extensively used **Hibernate mappings**, **HQL**, **Criteria**, **Lazy loading transactions** and locking for interacting with **Oracle** database.
- ^ Implemented **RESTful APIs**, ensuring seamless communication between front-end interfaces and back-end systems, promoting a cohesive and interconnected digital ecosystem.
- ^ Utilized **NgRx** and **RxJs** for state management, aligning with **Agile** and **Kanban** methodologies for rapid, responsive development.
- ^ Involved in writing **Spring Configuration XML**, file that contains declarations and business classes are wired-up to the frontend managed beans using **Spring IOC** pattern.
- ^ Utilized **Spring MVC** for implementing Model-View-Controller (**MVC**) architecture, improving code modularity and maintenance.
- ^ Managed **AWS EC2** for scalable web applications and **S3** for secure storage, integrating **AWS Lambda** with **API Gateway** for serverless **Python** and **Node.js** scripts
- ^ Improved API reliability and testing through effective use of HTTP status codes and tools like **Postman**.
- ^ Integrated **MariaDB** with the backend services using **Spring Boot** and **JPA/Hibernate** for seamless data access and CRUD operations in the application.
- ^ Utilized **AWS IAM** for access control and user management and implemented **AWS RDS** for database management.
- ^ leveraged **Spring Web Flux** to build reactive, non-blocking web applications, ensuring high performance and scalability in handling asynchronous data streams and real-time updates for user interfaces.
- ^ Executed integration tests with **JUnit** and **Mockito**, verifying the interactions between different modules and ensuring seamless integration.
- ^ Created **Kibana** dashboards to monitor **API** latency, resource usage, and system performance.
- ^ Used **Hibernate** Transaction Management, **Hibernate** Batch Transactions, and **cache** concepts.
- ^ Used **Spring Security** and **OAuth2** to secure **REST APIs** with token-based authentication and authorization.
- ^ Utilized **Docker** and **Kubernetes** for container orchestration and cloud deployment, ensuring scalability and high availability.
- ^ Managed cloud resources using **Terraform** and deployed applications on **AWS** services, including **EC2**, **Lambda**, and **S3**.
- ^ Worked with **NoSQL** databases like **MongoDB** and **DynamoDB** to design scalable, flexible database architectures.
- ^ Used **JUnit** and **Mockito** for unit testing, and integrated them into **Jenkins** pipelines for continuous integration.
- ^ Configured **WebLogic** Server security realms to manage users, groups, and roles effectively, enforcing role-based access control (RBAC).
- ^ Built **CI/CD** pipelines using **Jenkins** and **GitLab** for automated testing, builds, and deployments, improving team productivity.
- ^ Leveraged **WebSocket** protocols in **Angular** applications to enable real-time communication and dynamic user interactions.

Environment: Agile methodology, Java, J2EE, Hibernate, Spring, XML, JSP, JAX-RS, Spring AOP, Spring security, Oracle, MongoDB, JSP, HTML, AJAX, JavaScript, jQuery, Angular 8, Docker, Kubernetes, Kafka, GIT, JIRA, PL/SQL, STS, JUnit, Log4J, Maven, Jenkins, WebLogic Application Server, AWS IAM, EC2, S3, RDS, SQS, PHP

Client: Polaris, Minneapolis, MN

Jun 2016 - Sep 2018

Role: Java Full Stack Developer

Responsibilities:

- ^ Collaborated in an **Agile** development environment, participating in retrospectives and Scrum sessions to improve team communication and project outcomes.
- ^ Engaged in various phases of the Software Development Life Cycle (**SDLC**), including design, development, and testing, while creating use case, class, and sequence diagrams.
- ^ Utilized **Java 8 Lambda** expressions and **Streams** for efficient internal iteration and operations on collections.
- ^ Developed user interfaces using **HTML5**, **CSS3**, **JavaScript**, **jQuery**, **Bootstrap**, **Angular 6**, and **Node.js**, creating prototypes and layout designs.
- ^ Designed and implemented the persistence layer using **jQuery**, **AJAX**, and third-party libraries for enhanced user interaction.
- ^ Improved code quality and reliability by performing unit testing with the Test Framework **JUnit**.
- ^ Developed and maintained backend services using **Kotlin**, enhancing performance and scalability in a retail company's **microservices architecture**.
- ^ Implementing or exposing the **Microservices** architecture with **Spring Boot** based interacting through a combination of **RESTful** and **Apache Kafka** message brokers.

- ^ Implemented pre-authentication and database security using **Spring Security** and securing application access and **Spring Batch** for scheduling and maintenance of batch jobs.
- ^ Implemented **Spring ORM** with **Hibernate** taking advantage of features like Annotation metadata, Auto wiring, and Collections to implement **DAO** layer with **Hibernate** Session Factory, **HQL**, and **SQL**.
- ^ Created Services to consume **REST API's** and to communicate between components using **Dependency Injection** provided by **Angular**.
- ^ Used **Spring Core** annotations for **Spring Dependency Injection**, **Spring MVC** for **Rest API's** and **Spring Boot** for microservices.
- ^ Implemented real-time trade data integration across sales and trading platforms, ensuring seamless communication between front, middle, and back office systems to support fast and accurate decision-making in the Derivatives domain.
- ^ Involved in development of **Restful** web services using **MERN** stack, **MongoDB**, **ExpressJS** and **NodeJS**.
- ^ Proficient in leveraging **Spark's** core functionalities such as **RDDs** (Resilient Distributed Datasets) and Data Frames.
- ^ Experienced with **RDBMS** implementation and development using **Oracle**, **My SQL**, **MSSQL** and **DB2**.
- ^ Designed and deployed scalable, enterprise-level applications using Oracle **WebLogic** Server, ensuring high availability, efficient load balancing, and seamless integration with backend services
- ^ Used **GIT** for version control and **JIRA** for bug tracking and project management.
- ^ Applied **PostgreSQL** indexing and partitioning strategies to enhance database scalability and reduce data retrieval times by 25%.
- ^ Managed database migrations and schema updates in **PostgreSQL**, ensuring data integrity and minimizing downtime during deployments.
- ^ Developed enterprise applications using **Hibernate**, **Spring IOC**, **Spring MVC**, **Spring Boot**, and **RESTful** web services for robust back-end functionality.
- ^ Implemented authentication and authorization mechanisms using **Spring Security** and **OAuth2** for secure application access.
- ^ Developed **RESTful** web services that produce both **XML** and **JSON** for diverse applications, utilizing Jersey for **JAX-RS** implementation.
- ^ Utilized **AWS Lambda** for data uploads to **S3** buckets and triggered functions, ensuring high availability and fault tolerance.
- ^ Executed **Docker** container implementations for enterprise build systems and managed containerized applications effectively.
- ^ Developed reusable **APIs** and connectors with **MuleSoft** Anypoint Platform to streamline integration processes and reduce redundancy by 25%.

Environment: JDK 1.8, Angular 6, Spring, Hibernate, Spring Boot, LINUX, Maven. RESTful Web Services, Jenkins, HTML5, CSS3, React JS, Bootstrap, JavaScript, jQuery, Ajax, Node.js, MongoDB, SoapUI, Postman, API Life Cycle, JUnit, JMeter, Git, JAX-RS, JIRA, Cucumber, Kafka, Putty, Agile.

Client: Kaiser Permanente, Seattle, WA

Mar 2013 to May 2016

Role: Java Developer

Responsibilities:

- ^ Developed a patient data integration platform using Java 8, Spring Boot, and REST APIs to enable seamless real-time exchange of insurance and clinical records across hospital systems, enhancing provider workflows.
- ^ Built interactive user interfaces with **Angular 2**, **HTML5**, **CSS3**, **JavaScript**, and **Bootstrap**.
- ^ Implemented **CI/CD** pipelines using Jenkins for automated builds, testing, and deployments.
- ^ Deployed and managed cloud-based applications on **AWS**, utilizing **EC2** and **S3** services.
- ^ Utilized **MySQL** and **MongoDB** for database management, creating complex queries and optimizing performance.
- ^ Built secure RESTful services using Spring MVC and JAX-RS with Swagger documentation and JWT-based authentication, improving interoperability and patient record access across legacy and cloud-based systems.
- ^ Employed design patterns like Singleton, Factory, Business Delegate, and DAO to build scalable solutions.
- ^ Managed source code with **Git** and **SVN**, coordinating multiple branches and resolving merge conflicts.
- ^ Handled session management and security with **Spring Security**, implementing **OAuth2** for authentication.
- ^ Used Node.js for dependency management in Angular projects and optimized build processes.
- ^ Worked on asynchronous operations using **Spring's** task execution framework to enhance performance.
- ^ Developed **Java** applications utilizing advanced multithreading techniques for improved concurrency.
- ^ Used **Hibernate** and **JPA** to handle **ORM** and database transactions, ensuring data consistency.
- ^ Designed and integrated services in a **Service-Oriented Architecture** (SOA) environment.
- ^ Conducted static code analysis using **SonarQube** to maintain high code quality and performance standards.
- ^ Integrated **Spring AOP** for cross-cutting concerns like logging and transaction management.
- ^ Ensured cross-browser compatibility and responsive design for web applications.
- ^ Developed **REST APIs** for external systems and ensured secure data exchange through **OAuth2** and **SSL**.

Environment: JDK1.8, JSP, Servlet, EJB, spring, JavaScript, Hibernate, Multithreading, Query, Struts, Design Patterns, HTML5, CSS3, JMS, XML, Eclipse, Apache, Oracle ECM, Java 8.0, Struts.