Sr. Java Full Stack Developer

Name: Revanth.Dirisipamu
Mail Id: Revanth.dirisipam92@gmail.com
Contact No: 4793170202

Professional Summary:

- Over 10 +years of experience as a Full Stack Java Developer in designing, developing, and deploying enterprise web applications.
- Strong hands-on expertise in Java (8–17), Spring Boot, and JPA for backend microservice development.
- Built and integrated ReactJS front-end applications with RESTful Java services.
- Developed and deployed AWS-based solutions using ECS, EC2, S3, Aurora, API Gateway, ALB, and Route 53.
- Strong understanding and use of Kafka, Kubernetes, and CI/CD pipelines using Jenkins, GitHub, Maven, and Terraform.
- Strong experience in implementing API Management using Apigee, including onboarding APIs, managing API proxies, and configuring OAuth2 token validation policies.
- Hands-on expertise in encryption and decryption mechanisms using PGP and industry-standard cryptographic algorithms (AES/RSA) to secure sensitive financial data in distributed systems.
- Experience designing resilient microservices using Circuit Breaker (Resilience4j) and Saga patterns to handle distributed transactions and ensure fault tolerance.
- Proven ability to build cloud-native applications using Azure Event Hubs for event streaming alongside Kafka for real-time data processing.
- Strong experience in Domain-Driven Design (DDD) and implementing scalable, loosely coupled microservices architectures.
- Expertise in API lifecycle management, including versioning, throttling, rate limiting, and API security using API Gateway solutions.
- Hands-on experience implementing zero-trust security models and secure token-based authentication flows across distributed microservices.
- Experience working with IBM TRIRIGA platform including workflow customization, business object configuration, REST/SOAP integrations, and reporting.
- Proficient in writing unit and integration tests using JUnit and Cucumber, with experience in Splunk and ELK for log monitoring.
- Strong hands-on experience designing and implementing event-driven microservices using Apache Kafka with topic partitioning, consumer groups, offset management, and fault-tolerant message processing.
- Expertise in building high-throughput, low-latency messaging pipelines using Kafka Producers, Consumers, and Kafka Streams for real-time trade and transaction processing.
- Experience configuring Kafka clusters, managing brokers, replication factors, and ensuring high availability in distributed cloud environments (AWS/Azure).
- Integrated Kafka with Spring Boot using Spring Kafka for asynchronous communication, retry mechanisms, and dead-letter topic handling in production systems.
- Hands-on experience with Kafka Schema Registry, Avro serialization, and monitoring Kafka performance using ELK, CloudWatch, and Splunk.
- Worked in Agile/Scrum environments using tools like JIRA and Confluence.
- Strongly associated with usage of Selenium, Jenkins, Junit and Curl for Unit and System Testing.
- Designed and developed Micro service business components using Java /J2EE, Spring Boot and Implemented REST web services based on RESTful APIs.
- Experience in writing PL/SQL scripts, Triggers, Stored Procedures, views, and materialized views.
- Working knowledge of build technologies like Maven, ANT, and Gradle, as well as versioning tools like GIT, SVN, and Clear Case.
- Experience in using React JS components, Forms, Events, Keys, Router, plus Redux, Animations and Flux concept.
- Experience in developing web applications on MVC-based frameworks like Spring, Spring IOC, Spring Security, Spring Boot, Spring Cloud, and Spring MVC.

- Proficient in Elasticsearch, data modelling and querying using log aggregation, data extraction and reporting using Elasticsearch, Logstash, Kibana tools.
- Experience with large-scale distributed systems using JMS, Message Queues and Spring JMS Integration.
- Experience in Front - End UI technologies like HTML5, CSS3, JavaScript, jQuery, XML, JSON, AJAX, JSP, NodeJS, AngularJS, ReactJS and Bootstrap.
- Extensive AWS Experience including AWS Services such as EC2, VPC (NAT, VPC Peering and VPN), IAM (Identity and Access Management), Elastic Beanstalk, Lambda, S3, CloudFront, Glacier, SQS, SNS, RDS, DynamoDB, Route53, CloudWatch, CloudTrail, CloudFormation, Security Groups etc.
- Expert in working on Java 8 features Lambda Expressions, Functional interfaces Stream API's, Time API, and Improvements on Collection, Concurrency, and IO.
- Hands-on experience in using message brokers such as Active MQ and RabbitMQ.
- Proficiency in creating unit test cases using JUnit, as well as leveraging Splunk and Log4J for extensible logging, debugging, and error tracing.
- Strong experience collaborating with Product Owners, Scrum Masters, and cross-functional Agile teams to translate business requirements into scalable technical solutions and deliver high-quality software within sprint cycles.
- Hands-on experience working with UI/UX design tools like Figma to convert wireframes and design prototypes into responsive and user-friendly front-end applications using ReactJS and modern JavaScript frameworks.
- Proficient in developing automation and deployment scripts using Python and Unix Shell scripting to support CI/CD pipelines, environment setup, and batch processing activities.
- Solid understanding of UI/UX principles including responsive design, accessibility standards, and performance optimization to enhance user experience across enterprise applications.

Professional Skills:

Languages	Java, J2EE, SQL, PL/SQL, and Python
Web/XML Technologies	HTML5, CSS3, SCSS JavaScript, jQuery, ReactJS, AngularJS, Angular2/6, BackboneJS, ReactJS, XML, XSD, XSL/XSLT, SAX/DOM, AJAX, JSON, DOJO, and NodeJS.
Web services	SOAP, RESTful, UDDI, WSDL, JAX-RPC, JAX-RS, JAX-WS, JAX-B, Microservices, Apache Axis, and Apache CFX, GraphQL APIs, API Gateway, OAuth2 Authorization, JWT Token-based Security, API Management using Apigee, Event-driven Microservices
Enterprise Java	Java, J2EE, Swing, RMI, Sockets, JDBC, Servlets, JSP, JMS, Java Beans, JSTL, JSF, Struts, EJB, Spring, JTA, JNDI and JPA, Distributed Transaction Management (Saga Pattern), Reactive Programming (Spring WebFlux), Resilience Patterns (Circuit Breaker, Retry, Bulkhead), Secure API Design (OAuth2/JWT)
Web/App Servers	Web Sphere, WebLogic, Apache, Tomcat, and JBoss.
Tools & Framework	Struts, Hibernate, Spring MVC, Spring Web Flow, Spring IOC, Spring AOP, JMS, JSF, Log4J, SOA, Jasper reports, Spring Boot, Spring Batch, Spring Security, Spring Data, Jersey. IBM TRIRIGA, Jenkins, Git, Docker, Kubernetes, AWS, Apigee API Management, Resilience4j, Saga Pattern Implementation, Circuit Breaker Design Patterns, Azure Event Hubs, API Gateway (AWS/Azure), OAuth2 Authorization Server, JWT Token Services, PGP Encryption Tools, GraphQL (Apollo Server & Client), Domain Driven Design (DDD), OpenAPI/Swagger, SonarQube, Nexus IQ, Prometheus, Grafana, OpenTelemetry
O-R mapping	Hibernate, JPA and JTA
Database	Oracle, SQL-Server, MySQL server, MS SQL, IBM DB2, MongoDB, and NoSQL
Cloud Technologies	AWS (Amazon Web Services), Microsoft Azure, Azure Event Hubs, Azure API Management (APIM), Azure Monitor, Azure DevOps Pipelines, AWS API Gateway, AWS EventBridge
Version Control	GIT, CVS, SVN, Rational clear case, and Star Team
Testing-Tools/ Others	JUnit, Jasmine, SoapUI, Postman, Putty, Rational Clear Quest, RTC, Load UI, and JIRA
Development Tools	Eclipse, RAD, Spring Tool Suite (STS) and IntelliJ
Platforms	Windows, Win 2k Server, Sun Solaris, Mac OS, and UNIX

Professional Experience:

Client: Nationwide Mutual Insurance – Colombus, OH. **Jan 2024 – Till Date**
Sr. Java Full Stack Developer

Responsibilities:

- Developed and delivered scalable full-stack modules using ReactJS integrated with Spring Boot-based REST APIs, deploying containerized applications on AWS ECS to support high-availability enterprise systems.
- Containerized Java and React applications using Docker and deployed them on Microsoft Azure Kubernetes Service (AKS), integrating with Azure Key Vault for secure secrets management and Azure Blob Storage for distributed data handling.
- Designed, implemented, and maintained robust CI/CD pipelines using Azure DevOps and Jenkins, enabling automated build, test, and deployment processes across staging and production environments with minimal downtime.
- Implemented comprehensive API lifecycle management using Apigee, including API onboarding, versioning strategies, policy enforcement, security governance, and deprecation planning for enterprise-scale API ecosystems.
- Designed and implemented secure OAuth2 authentication mechanisms, including token generation, refresh token flows, and lifecycle management to ensure secure communication across distributed microservices.
- Managed PGP encryption key lifecycle, including key generation, rotation, storage, and usage to enable secure encryption and decryption of sensitive financial and customer data.
- Implemented Kafka exactly-once processing semantics (EOS) to ensure data integrity and eliminate duplicate processing in high-volume, real-time financial systems.
- Designed and optimized partitioning strategies for Apache Kafka and Azure Event Hubs to achieve high throughput, scalability, and fault tolerance in event-driven architectures.
- Implemented advanced deployment strategies such as blue-green and canary releases using Jenkins and Azure DevOps to minimize risk and ensure seamless production rollouts.
- Leveraged Helm charts for Kubernetes deployments, enabling version-controlled, repeatable, and scalable infrastructure provisioning and application rollout.
- Designed and implemented real-time monitoring and alerting systems using Prometheus AlertManager and Grafana dashboards to proactively detect and resolve production issues.
- Applied eventual consistency models using Saga orchestration patterns and event-driven microservices architecture to ensure data consistency across distributed systems.
- Performed JVM performance tuning, optimizing thread pools, connection pools, and memory management to improve application performance and reduce latency.
- Implemented secure API development practices aligned with OWASP Top 10 standards, including input validation, rate limiting, authentication, and threat protection.
- Utilized Azure services such as AKS, Azure Functions, App Services, and Application Insights for application deployment, observability, and performance optimization.
- Developed secure API proxies and enforced OAuth2-based policies using Apigee, ensuring secure exposure and governance of enterprise APIs.
- Designed and implemented PGP-based encryption and decryption workflows to ensure secure transmission of sensitive data across distributed systems.
- Applied Resilience4j-based Circuit Breaker, Retry, and fallback patterns to improve system reliability and prevent cascading failures in microservices.
- Implemented Saga-based distributed transaction management to ensure consistency and rollback capabilities across multiple microservices.
- Integrated Azure Event Hubs with Kafka to build real-time, high-throughput event streaming and ingestion pipelines.
- Designed and enforced API security policies including token validation, throttling, and rate limiting using API Gateway and Apigee for enterprise-grade security.
- Developed GraphQL APIs alongside RESTful services to optimize data retrieval, reduce over-fetching, and enhance front-end performance.

- Implemented centralized authentication and authorization using OAuth2 Authorization Server for secure service-to-service communication.
- Documented APIs using Swagger/OpenAPI to standardize API contracts and improve cross-team collaboration and integration.
- Enabled distributed tracing and observability using OpenTelemetry integrated with Grafana and ELK stack for end-to-end performance monitoring.
- Developed high-throughput Kafka producers and consumers to support real-time event-driven processing pipelines.
- Implemented idempotent Kafka producers and managed offset commits effectively to prevent duplicate data processing in distributed systems.
- Tuned Kafka configurations including replication factor, retention policies, batching, and throughput optimizations to enhance system performance.
- Utilized Avro serialization and Schema Registry to ensure schema compatibility and data consistency across distributed services.
- Built real-time analytics dashboards by integrating Kafka Streams with Elasticsearch and Kibana for data visualization and monitoring.
- Developed responsive and dynamic UI components using ReactJS, Redux, and Bootstrap for enterprise dashboards and user interfaces.
- Collaborated with cross-functional and globally distributed teams to support cloud-based deployments and enterprise application delivery.
- Designed and implemented CI/CD workflows using AWS CodePipeline, Jenkins, and Git for automated build and release processes.
- Managed centralized logging and monitoring using Splunk and AWS CloudWatch to ensure system reliability and performance.
- Automated infrastructure provisioning using Terraform, creating reusable modules for AWS resources such as EC2, VPC, IAM, and S3.
- Containerized applications using Docker and deployed them across ECS and AKS clusters to ensure scalability and portability.
- Integrated ELK stack (Elasticsearch, Logstash, Kibana) for centralized logging, monitoring, and analytics.
- Enforced cloud security best practices using IAM roles, policies, and AWS Secrets Manager for secure configuration management.
- Performed front-end unit testing using Jest and React Testing Library to ensure UI reliability and performance.
- Configured Webpack for optimized front-end builds, module bundling, and performance improvements.
- Developed reactive and non-blocking microservices using Spring WebFlux to support high-concurrency systems.
- Implemented micro-frontend architecture to enable modular, scalable, and independently deployable UI components.
- Developed backend services using Node.js and Express for integration and supporting microservices architecture.
- Customized IBM TRIRIGA workflows, business objects, and integrations to support enterprise asset and facility management systems.
- Developed REST and SOAP integrations between TRIRIGA and external enterprise systems to enable seamless data exchange.
- Implemented TRIRIGA workflow automation, event triggers, and approval processes for business operations.
- Built backend microservices using Spring Boot to support TRIRIGA integrations and enterprise workflows.
- Developed BIRT reports for TRIRIGA dashboards to support operational and analytical reporting.
- Integrated TRIRIGA with enterprise authentication systems using SSO and role-based access control.
- Participated in Agile ceremonies including sprint planning, daily stand-ups, backlog grooming, and retrospectives.
- Developed automation scripts using Selenium, JUnit, Postman, and Jenkins to support testing and deployment validation.
- Implemented secure coding practices following OWASP standards, integrating OAuth2 and JWT-based authentication.
- Built reusable infrastructure-as-code modules using Terraform for scalable cloud environments.
- Used SonarQube for continuous code quality analysis, vulnerability detection, and compliance.

- Developed RESTful services and Elasticsearch-based search APIs for high-performance querying and analytics.
- Implemented Camunda workflows for business process automation, including event-driven and transaction-based workflows.

Environment: Java 11/17, J2EE, Spring Boot, Spring MVC, Spring Security, Spring WebFlux, Microservices, REST APIs, GraphQL, Kafka, Azure Event Hubs, JMS, ActiveMQ, Apigee, API Gateway, OAuth2, JWT, PGP, Resilience4j, Saga Pattern, Event-Driven Architecture, Docker, Kubernetes, Helm, AWS, Azure, Terraform, Jenkins, CodePipeline, Azure DevOps, Git, Maven, ReactJS, Redux, Bootstrap, HTML5, CSS3, JavaScript, Node.js, Express, Prometheus, AlertManager, Grafana, OpenTelemetry, Splunk, ELK, Oracle, PostgreSQL, MongoDB, JUnit, Selenium, Postman, Jest, Swagger, SonarQube, OWASP, Camunda, Blue-Green Deployment, Canary Deployment, JVM Tuning, Agile, JIRA, Confluence, Linux, Unix, MacOS.

Client: IDB Bank – New York City, NY. **Jul 2021 – Dec 2023**
Sr. Java Full Stack Developer

Responsibilities:

- Collaborated closely with Product Owners and Scrum teams to analyze requirements, refine user stories, and deliver scalable solutions aligned with business objectives in an Agile environment.
- Worked with UI/UX teams using Figma to translate design prototypes and wireframes into high-performance ReactJS-based front-end applications with consistent user experience.
- Applied UI/UX best practices including responsive design, accessibility, and cross-browser compatibility to enhance usability of enterprise applications.
- Developed and maintained automation scripts using Python and Shell scripting to support CI/CD processes, deployment automation, and operational tasks.
- Participated in technical design discussions and architecture reviews to ensure scalability, performance, and maintainability of microservices-based applications.
- Optimized front-end performance by implementing efficient rendering techniques, reducing load times, and improving client-side responsiveness.
- Built reusable and modular UI components to improve maintainability and scalability of front-end applications.
- Collaborated with DevOps teams to streamline deployment processes, environment configurations, and build automation using scripting and CI/CD tools.
- Ensured alignment between UI implementations and design specifications by validating layouts, styles, and responsiveness across multiple devices.
- Contributed to Agile process improvements by participating in retrospectives and identifying opportunities for technical and process optimization.
- Designed and implemented scalable backend microservices using Java and Spring Boot to support high-volume banking applications, enabling efficient trade processing, settlement, and reconciliation workflows.
- Built event-driven architectures using Apache Kafka and Azure Event Hubs to handle real-time financial transactions and ensure reliable data streaming across distributed systems.
- Implemented Kafka connectors, topic partitioning strategies, and consumer group configurations to handle high-throughput data ingestion with optimized performance.
- Ensured data consistency and transactional integrity by implementing Kafka exactly-once processing semantics (EOS) in trade settlement systems.
- Designed multi-region event streaming architecture using Kafka and Azure Event Hubs to support disaster recovery and high availability.
- Implemented secure API integrations using Apigee, including API onboarding, proxy configuration, and policy enforcement for banking APIs.
- Designed and implemented OAuth2 authentication mechanisms, including token generation, refresh token flows, and lifecycle management for secure API communication.
- Managed PGP encryption and decryption processes, including key exchange and secure data transmission for financial systems.
- Implemented API Gateway policies for request routing, authentication, rate limiting, and security enforcement across microservices.

- Applied Resilience4j patterns including Circuit Breaker, Retry, and Bulkhead to enhance system resilience in high-volume banking environments.
- Designed and implemented Saga pattern for distributed transaction management across payment, settlement, and reporting systems.
- Developed GraphQL APIs alongside REST services to optimize data retrieval for trading dashboards and reporting applications.
- Applied Domain Driven Design (DDD) principles to model complex banking domains such as trade lifecycle, settlements, and compliance workflows.
- Implemented centralized authentication using OAuth2 Authorization Server to enable secure service-to-service communication.
- Designed and implemented CI/CD pipelines using Jenkins, Azure DevOps, and Terraform to automate build, test, and deployment processes.
- Deployed containerized microservices using Docker and Kubernetes (EKS/AKS), ensuring scalability, reliability, and high availability.
- Utilized Helm charts for Kubernetes deployments to enable controlled rollout and version management of applications.
- Implemented canary deployment and rollback strategies in CI/CD pipelines to ensure zero-downtime releases.
- Automated infrastructure provisioning using Terraform for AWS services including RDS, ECS, and CloudWatch.
- Developed and maintained backend services using Java, Spring Boot, and RESTful APIs for enterprise banking applications.
- Refactored legacy monolithic applications into microservices architecture deployed across AWS and Azure environments.
- Integrated Kafka and JMS messaging systems to enable asynchronous communication across distributed services.
- Implemented retry logic, dead-letter queues, and offset management strategies to ensure fault tolerance and reliability.
- Monitored Kafka cluster health, consumer lag, and broker performance using CloudWatch, Grafana, and ELK stack.
- Implemented centralized monitoring and alerting using Prometheus and Grafana dashboards for SLA compliance.
- Built and optimized SQL queries, stored procedures, and database operations on Oracle and PostgreSQL.
- Developed responsive front-end applications using ReactJS and Angular integrated with backend microservices.
- Improved UI responsiveness and state management using modern JavaScript frameworks and design patterns.
- Collaborated with business users and compliance teams for UAT, requirement gathering, and production rollouts.
- Delivered CI/CD pipelines for multi-region deployments across AWS environments.
- Implemented DevSecOps practices using SonarQube, OWASP standards, and static code analysis tools.
- Developed automation scripts using Selenium Grid, JUnit, and Postman for test automation and validation.
- Built REST APIs and integrated external systems using REST and SOAP-based services.
- Implemented message-driven architecture using JMS and ActiveMQ to support asynchronous processing.
- Developed backend services using Java and Scala for trade data processing and batch operations.
- Supported batch monitoring and optimization strategies to ensure smooth execution of financial batch jobs.
- Wrote Python and Unix shell scripts for automation, backup, and production support processes.
- Participated in Agile/Scrum ceremonies including sprint planning, stand-ups, and retrospectives using JIRA and Confluence.
- Performed code reviews and enforced code quality using SonarQube and Nexus IQ tools..

Environment: Java 8/11, Spring Boot, Spring MVC, Spring Security, Microservices, REST APIs, GraphQL, Kafka, Azure Event Hubs, JMS, ActiveMQ, Apigee, API Gateway, OAuth2, JWT, PGP, Resilience4j, Saga Pattern, Docker, Kubernetes, Helm, AWS, Azure, Terraform, Jenkins, Azure DevOps, Git, Maven, ReactJS, Angular, NodeJS, HTML5, CSS3, JavaScript, Elasticsearch, Kibana, Grafana, Prometheus, Oracle, PostgreSQL, JUnit, Mockito, Selenium, Postman, Swagger, SonarQube, OWASP, Agile, JIRA, Confluence, Linux, Unix.

Client: Eversource – Boston, MA. **Jul 2020 – Apr 2021**
Java Full Stack Developer

Responsibilities:

- Developed backend applications using Java and Spring Framework, handling complex relational data operations using Oracle PL/SQL for enterprise utility systems.
- Collaborated with Product Owners and business teams to understand functional requirements and translate them into technical implementations within Agile development processes.
- Assisted in implementing responsive UI components based on design guidelines and improving user experience using HTML, CSS, and JavaScript frameworks.
- Applied basic UI/UX principles to improve usability, accessibility, and consistency across enterprise applications.
- Developed and maintained simple automation scripts using Shell scripting to support deployment and data processing tasks.
- Participated in Agile ceremonies including daily stand-ups, sprint planning, and retrospectives to support iterative development.
- Supported front-end and backend integration by working with REST APIs and ensuring seamless data flow between layers.
- Assisted in improving application performance by optimizing queries and enhancing UI responsiveness.
- Worked with development and operations teams to support application deployments and environment configurations.
- Ensured adherence to coding standards and best practices while contributing to development and testing activities.
- Supported documentation and knowledge sharing to improve team collaboration and project understanding.
- Designed and deployed Java-based applications following MVC architecture using Spring Boot and Spring MVC, enabling modular and scalable system design.
- Built and exposed RESTful web services to support integration with internal and external enterprise systems.
- Developed responsive user interfaces using AngularJS, HTML, CSS, and Bootstrap, improving user interaction and front-end performance.
- Implemented secure API communication using OAuth2 and JWT-based authentication mechanisms to protect enterprise services.
- Designed and optimized SQL queries, stored procedures, and database operations to improve performance of data-intensive applications.
- Developed integration scripts and automated processes for data warehouse operations and backend processing.
- Participated in production deployments and release management activities, ensuring smooth rollout of application updates.
- Assisted in implementing CI/CD practices using Jenkins and ANT-based build tools for automated build and deployment processes.
- Contributed to application monitoring and logging using Log4J, enabling efficient troubleshooting and issue resolution.
- Implemented exception handling using Spring AOP to centralize error management and improve maintainability.
- Utilized Spring IOC for dependency injection and modular application development.
- Worked with Hibernate ORM framework for database persistence and object-relational mapping.
- Implemented message-driven architecture using IBM MQ and Message Driven Beans (MDBs) for asynchronous communication.
- Developed dynamic web functionalities using AJAX and JavaScript to enhance user experience.
- Created and executed MongoDB scripts and supported NoSQL and relational database integrations.
- Participated in Agile development processes, including sprint planning, daily stand-ups, and retrospectives.
- Collaborated with cross-functional teams to gather requirements through JAD sessions and translate them into technical solutions.
- Developed and maintained technical documentation including use cases, sequence diagrams, and activity diagrams.
- Applied core Java concepts including multithreading, collections, exception handling, and concurrency for building robust applications.
- Supported application servers such as IBM WebSphere and Apache Tomcat for deployment and runtime environments.
- Participated in Test-Driven Development (TDD) using JMock and TestNG for unit and integration testing.
- Maintained version control using SVN and managed source code repositories using Eclipse-based tools.
- Assisted DevOps teams in configuring application servers and deployment environments across development and production systems.

Environment: JDK 1.7, J2EE, Eclipse, Spring, Hibernate, Web-Services (REST), HTML, Java Script, jQuery, AJAX, Oracle 11g, Test NG, J Mock, ANT, SVN, IBM Web Sphere Application, Spring MVC, TOAD, Windows.

Client: Qualcomm – Hyderabad, India. **Apr 2017 – Dec 2019**
Java Developer

Responsibilities:

- Used Elasticsearch for powering not only Search but using ELK stack for logging and monitoring our systems end to end Using Beats
- Extensively worked with Servlets and Struts based multi-tier applications in developing J2EE Components.
- Spring Framework components including Spring IOC, Spring AOP, Spring MVC, and Spring ORM, as well as Spring JPA and Spring boot, were used to implement the application's different levels.
- Used a variety of Core Java principles to implement numerous features and improvements, including multi-threading, exception handling, serialization, I/O streams, garbage collection, and collection APIs.
- Used the AWS environment to plan and carry out development tasks that include AWS lambda as well.
- created and executed Mongo DB scripts to automate reporting and data updates in NoSQL databases.
- I was able to write SQL queries, Postgres, Oracle PL/SQL stored procedures and functions, as well as relational database triggers and views, with ease.
- Designed and coded application components in an Agile/TDD environment utilizing a test-driven development and SDET approach and pair-programming.
- Developed Message Handler Adapter, which converts the data objects into XML message and invoke an enterprise service and vice-versa using Java, JMS and MQ Series.
- Using the Junit framework, test cases were developed, and unit testing was carried out.
- I used Java 8 capabilities including the Stream API, Time API, Multithreaded Functional Interfaces, Transaction Management, Exception Handling, and Collection API. The test cases were created using the Groovy/Grails Tool Suite (GGTS), and manual testing was done.
- With the use of Restful Web Services, distributed systems and Schema design are developed with good exposure to and familiarity with Service Oriented Architecture (SOA) and the Microservices methodology.
- Created log files in development and production servers using the Log4J Framework for logging and debugging the application.

Environment: IntelliJ, CSS, HTML, Core Java, JPA, JUnit, JSP, Servlets, Mongo DB, Oracle 10g, Log4J, Windows, AWS and JBoss Application Server.

Client: Genpact – Hyderabad, India. **Jan 2014 – Mar 2017**
Software Developer
Responsibilities:

- Developed full-stack modules using ReactJS and Spring Boot in a microservices environment.
- Used Redux and React Router to manage state and navigation in enterprise web applications.
- Worked Closely with the Core Product team, planning the product and Schema design, and providing the inputs for product design and implementation.
- Worked with Java, Linux, Git, Microservices, SQL, Hibernate, Agile, Oracle, Spring Boot, React, and Spring.
- Worked on developing a WEB-based project which includes technologies such as O, Java, Spring, React, Git, Spring Boot, MongoDB, and Oracle Database.
- Worked on development, maintenance, and deployment in Linux servers.
- Here I stand out as a first developer in India Subsidiary helped me understand the product in my own way and helped developers that joined later in providing the Kt's and core understanding of the project.
- Developed test cases and performed unit testing using **Junit** framework.
- The project was developed using Java, Linux, Git, Microservices, Hibernate, Agile, Oracle, Spring Boot, React, and Spring.
- Implemented microservices based architecture using Spring Boot interacting through REST API.

Environment: IntelliJ, Java, Linux, Git, Microservices, Hibernate, Agile, Oracle, Spring Boot, React.

EDUCATION: -

Master of Science (M.S.) in Technology Management
Lindsey Wilson College, USA | 2021

Bachelor of Technology (B.Tech) in Computer Science
SRM University, Chennai, India | 2013