**Name: S RAKESH**

 **Email :** **sana.rakesh5112@gmail.com**

 **Phone No: 980-320-0662**



**Professional Summary:**

* **Certified Java programmer** with **11** years of experience in design and development of applications in the field of **Finance, Banking, Health Care and Retail domains**.
* Proven expertise in **OOA**D of high-performance, high volume, distributed ‘n’ tier applications with **Java / J2EE** technologies and applied j2EE design patterns, UML methodologies effectively.
* Extensive Experience in **JAVA/J2EE** technologies using Core Java (**multi-threading, collections, IO, socket interface**s), **Servlets, JSP, JDBC, JNDI, MDB, JMS, MQ-Series, JAX-B, RFML, Java/J2ee** Design Patterns, Web Services.
* Experienced in Actimize **AML** Solutions **SAM**, Strong Knowledge on **CDD** and **WLF**.
* Experienced in open source technologies and frameworks like **Hibernate**, spring, **Struts**,.
* Experienced in project life cycle activities like Analysis (**High and Low level design**), Design, Development, Deployment & Installation, testing, documentation, maintenance of application and Technical support.
* Experienced in **BEA WebLogic server, Tomcat with Apache integration, WebSphere**.
* Experience in using databases like **Sybase, Oracle and SQL Server** and created **Stored Procedures, triggers, functions using T-SQL and PL/SQL**.
* Possessing strong analytical skills, a good team player with good leadership qualities and strong oral and written communication skills.
* Have hands on experience in Configuration Managements tools like Sonar, Fortify, RTC for Code Review.
* Excellent Knowledge of Object-Oriented Analysis &amp; Design using **UML**.
* Expertise in **Parallel and Distributed Computing, Designing** and implementing complex algorithms.
* Experience implementing **SOA, Webservice** applications.
* Experience in Micro Services and **Spring Boot** application development.
* Implemented and tune Actimize Customer Due Diligence(**AML-CDD**) models.
* Experienced in **object oriented, data structures, collections framework** API for framework design in Java including API development experience.
* Experienced in Microservices development using spring boot and development in **Pivotal Cloud Foundry (PCF).**
* Along with experience in **Data Structures, Multithreading** and have worked on search techniques and algorithms.
* Experience in handling messaging services using **Apache Kafka**.
* Worked on Data Structures**, multi-threading concepts in Core Java**
* Excellent understanding and experience in **OOPS principles inheritance, encapsulation, polymorphism**, and Core Java principles **collections, multithreading, synchronization, exception handling** and GUI design using Swings. Experienced in developing complexity-based codes related to data structures and algorithm design.
* Used search and encryption algorithms for better searching methods and security improvisation
* Develop **AWS Lambda** based Micro Services interacting with Dynamo DB &amp; SNS/SQS using AWS SDK.
* Designed and developed multiple **J2EE Model 2 MVC** based Web Application using J2EE Technologies, Patterns and popular open-source framework **Struts, Spring MVC, spring, JPA and Hibernate**.
* Deployed Spring Boot based **Micro Services Docker container** using **Amazon EC2** container services and using **AWS admin console**.
* Used JMS for the asynchronous exchange of critical business data and events among J2EE components and legacy system.
* Speeding up Existing Application the a Cache services using Redis Cache, Gemfire.
* Experience in designing and development of distribution scalable application using n memory data Grid(Coherence) and computing grids (GridGain, Datasynpse).
* Developed enterprise applications using **Scala**.
* Developed application using **AngularJS and NodeJS** connecting to Oracle on the backend.
* Experience in writing test-cases in Java environment using **Junit, Mockito and Power mock, JIRA**.
* Experience in managing the build files using **ANT and Maven**.
* Experience in using the **Log4j framework for Logging** Java Applications
* Strong experience with Software methodologies like **Agile, Waterfall and Test-Driven Development.**
* Experienced in UI frame works such as **HTML5, CSS3, Bootstrap, jQuery, Ajax, Google Web Toolkit (GWT), JavaScript, AngularJS, Tag Libraries, Custom Tags.**
* Experience working with **Java 1.8 Lambdas, Functional Interfaces and integrated Stream API** into the Collections API, to perform bulk operations on collections.
* Managed Amazon Web Services like **EC2, S3 bucket, ELB, Auto-Scaling, Dynamo DB, Elastic search, Virtual Private Cloud (VPC) through AWS Console and API Integration.**
* Hands on experience with Amazon web services (AWS) and Amazon cloud technologies such as **Amazon EC2 (virtual servers) and Amazon Cloud Watch (monitoring).**
* Extensive experience working in **Spring framework, Struts framework, Object Relational Mapping Hibernate framework and** web **services.**
* Implementing and supporting **AML** SAM and Actimize fraud model.
* Effectively worked on Real Time Trading application and Order Management Systems using financial industry standard/protocols like **SWIFT**.
* Experience using **Spring MVC, Spring Boot, Spring DAO, Spring Data, Spring IOC, Spring Annotations, Spring AOP, Spring Transactions and Spring Security.**
* Strong work experience in application integration and communicating using SOA, Web Services such as **JAX-RPC, JAX-RS, JAX-WS, SOAP, WSDL, XML and RESTFUL web services.**
* Extensive work experience in web technologies like **JSP, Servlets, JavaScript.**
* Extensively worked on developing UI components using **HTML5, CSS3 and JSON** to interact with **Restful web services**
* Experienced in numerous Design Patterns like **Singleton, Factory, Session Facade, MVC, Data Access Object, UML and Enterprise Application Integration.**
* Expert in Core Java with strong understanding of **Collections, Multithreading, Event handling, Exception handling and Generics.**
* Strong Experience in database design using **PL/SQL to write Stored Procedures, Functions, Triggers, Indexers and proficiency in** writing **complex queries, using Oracle, SQL Server and MySQL.**
* Experience in working with the NoSQL Database like **Cassandra DB and Mongo DB** to manage extremely large data sets.
* Experience in managing MongoDB environment from availably, performance and scalability perspectives.
* Experienced in JMS over messaging to exchange the information in more reliable and asynchronous way in enterprise Applications. Used **Apache Active MQ and Apache Camel.**
* Experience in version control system tools like **GIT, CVS and SVN.**
* Experience in unit testing the applications using **JUnit, TDD Framework.**
* Worked on implementing full life cycle of software development process (SDLC) in using version control tools like **Rational** Clear Case, Tortoise SVN, CVS (Concurrent Version System).
* Hands-on experience working with **Continuous Integration (CI) build-automation** tools such as **Maven, SVN, CVS, Jenkins** and **Apache Ant**.
* Significant experience in web development tools like **Spring Tool Suite (STS), Eclipse and IntelliJ.**
* Hands on experience with Windows, **UNIX and Linux** Environments.
* Having Experience on Shell Scripting and Deployment of Applications in Server.
* Experience in designing, developing, and deploying J2EE applications on **Web sphere, Web logic and Apache Tomcat** application Servers.
* Excellent written and verbal communication skills, Analytical, Problem-Solving skills, strict attention to detail and ability to work independently, lead/work within a team environment.

**Technical Skills:**

|  |  |
| --- | --- |
| **Languages:** | Java, J2EE, C++, SQL, PL /SQL,RUST |
| **Application Servers:** | Apache Tomcat, WebSphere |
| **IDE-s/Text editors:** | Eclipse, NetBeans, Visual Studio, Any point Studio, Sublime Text, WinSCP, putty |
| **Operating Systems** | Windows, Linux |
| **Web Technologies:** | HTML, CSS, AJAX, JavaScript, jQuery, Angular, Bootstrap |
| **Middleware tools:** | JBoss. |
| **XML Web Services:** | XML Schema, SOAP, REST |
| **Frameworks:** | Spring MVC, Spring Boot, Spring Cloud, Struts, Hibernate, AngularJS, Angular2, Bootstrap |
| **Databases**: | Oracle 11g, SQL Server, MySQL, NoSQL (MongoDB) |
| **Development tools:** | Maven, SVN, Git, JIRA  |
| **Testing Tools:** | Apache JMeter, Soap UI, Junit, Selenium |
| **Scripting languages:** | Unix-shell scripting |
|  |  |

**PROFESSIONAL EXPERIENCE**

**Wells Fargo, Charlotte , NC. (June’ 2020 - Present )**

**Senior Java Developer**

**Description**: **Wells Trade** is an investment platform which allows investors to trade stocks, ETFs, options, mutual funds and bonds. It provides helpful tools, including the Stock Screener, the ETF Screener, and the Mutual Fund Screener to help investors to build a portfolio.

**Responsibilities:**

* Designed, developed, and maintained various applications and microservices using **Java 11 and 17**,**Spring**, **Spring Boot, NoSQL, and SQL and AWS**.
* Used **Streams and Lambda expressions available as part of Java** to store and process the data.
* Used **Java 11** features like **Utility Methods, Nested Based Access Control, and Local Variable Syntax for lambda Parameters.**
* Develop quality code adhering to **Scala** coding Standards and best practices**.**
* Diligently worked on **SWIFT** message generation, consumption and their transformation using **JAXB** and **SAX** parser.
* Applied advanced **Core Java concepts and language** features to write clean, efficient, and maintainable code as part of the project implementation.
* Effectively worked on MDBs and JMS technology to exchange financial information with brokers using **SWIFT** protocol.
* Developed batch processing using **Spring Batch** in the persistence layer & improved performance of batch

processes using **Multithreading with Executor Framework** for control of asynchronous tasks in parallel threads.

* Implemented Java features like **Lambda, Streams, and Functional Interfaces** to reduce the line of code and better functionality in the Service Layer of the application.
* Applied best practices of agile development BDD, automated testing, continuous integration, etc. to allow for rapid delivery of high-quality software.
* Did database updates and load data in MongoDB.
* Utilized **Kafka** as a distributed streaming platform to build real-time data pipelines and implement publish-subscribe messaging patterns for seamless data integration and processing.
* Developed stream applications using **Kafka APIS and Kafka streams API**.
* Wrote producer and **consumer API (pub sub) to publish and consume data from topics**
* Implemented **Apachi camel** to process the **fix** messages with **Kafka cluster setup**.
* Implemented **Kafka producer and consumer applications on Kafka cluster** setup with help of **Zookeeper**.
* Developed REST web services using Spring REST and created unit and integration tests for the services using **Junit** and **Mockito**.
* Used JPA, **object/relational-mapping (ORM)** solution, the technique of mapping data representation from MVC model to Oracle Relational data model with SQL-based schema.
* Implemented **Spring AOP** module to implement the crosscutting concerns like logging, **security, Declarative Transaction Management.**
* **Implemented Microservices architecture with RESTFUL APL’s and oAuth2.**
* Developed Microservices with **Spring Eureka** to retrieve API routes for the entire cluster. Using this strategy each Microservice in a cluster can be load balanced and exposed through one API gateway.
* Used Microservice architecture with **SpringBoot** based services interacting through a combination of REST
* and **Apache Kafka message brokers**.
* Experience in budling Microservices by using SpringBoot.
* Used Spring Security for session fixation, clickjacking, cross site request for gery.
* Implemented **SpringMVC** flow to interact with different modules and extensively used **SpringAOP** and
* Developed several **REST webservices** which produces both **XML and JSON** to perform tasks, leveraged by both web and mobile applications.
* Implemented **AWS solutions using EC2, S3, RDS, EBS, Elastic Load Balancer, and EC2** instances.
* Worked with **AWS CLI and AWS API to manage resources on AWS**.
* Developed **AWS cloud formation templates and setting up Autoscaling for EC2** instances and involved in the automated provisioning of **AWS cloud environment using Jenkins**.
* Worked on **ROUTE53**,a scalable **cloud DNS webservice for effective routing, configuring DNS healthchecks**.
* Worked on **Multithreading** and optimized performance of applications.
* Created the **Docker containers** for managing the application lifecycle.
* Worked on creation of custom **Docker container images, tagging, pushing images, integration** of Springboot.
* Developed applications using **Angular11, Java11 and new features of java(lambdaexpressions).**
* Good working experience in **web Designs which includes HTML5, XHTML, XML, CSS3, JavaScript**.
* Developed UI components using **Angular dependency Injection, Models, databinding and controllers**.
* Implemented Angular Router to enable navigation from oneview to next as agent performs application tasks.
* Expertise in using **RxJs features like Observable, MergeMap, ForkJoin, Map, filter and Angular Routing & Navigation and Angular Form**s used these features widely in Adopted Resource and Information Application (ARIA).
* Developed Application to assess JSON and XML from Restful webservice from consumer side using **Angular**.
* Utilizing the component-based architecture offered **by Angular 11 to build Angular11** components and UI
* services to consume REST services.
* Implemented **SpringBoot** **Microservices** to process the message into the Confluent Kafka cluster setup.
* With **Angular 11, I created a robust single-page application** and a dynamic, responsive web application.
* Develop the Client-side Services using **Angular Http to get data from Data Bases through back-end services** and to update or deleted the data in Data Bases and to retain the filtered data before navigating to another page for future use.
* Developed various generic JavaScript, **Angula**r functions to implement client-side validations.
* Used **Apache camel** integration for routing and transforming the data between two Applications using HTTP protocol.
* Implemented **REST and SOAP** web services Consuming and Exposing Servers with Camel Componets (**File ,Http/Https, MQ, Database**)
* Worked to secure our RESTful web services using OAuth 2.0 with spring security to authenticate anythird-party requests coming in.
* Performance tuning by Managing and Monitoring the **Jmv** performance by **Weblogic** Heap size, garbage collection **execute** threads, **EJB** **Cache** size, **JDBC** pools and **multi** pools.
* Secured API's by implementing **OAuth2.0** token-based authentication/authorization using **SpringSecurity**.
* Implement Continuous Integration and **Continuous Delivery (CI/CD**) pipelines **using Jenkins, Maven, and Git**.
* Used Jenkins as build management tool for continuous integration process.•
* Developed Maven scripts and developed builds using **Maven and continuous integration servers like Jenkins**.
* Developed test cases and performed unit testing using **JUnit**.
* Unique experiences with Pivotal Cloud Foundry and Open Shift/**Kubernetes** architecture and design troubleshooting issues with platform components (PCF), and developing global/multi-regional deployment models and patterns for large scale development/deployments on Cloud Foundry and Open Shift/**Kubernetes.**
* Used to test performance both on static and dynamic resources using **JMeter**.
* Used JUnit to write unit test cases and mocked the Service and controllers **using Mockito Framework**.

Using GIT for Source Control and Version Management and using **log4j for logging**.

**Environment:** Agile methodology, Java, J2EE, Hibernate, Spring, XML, , JAX-RS, REST API, AOP, Spring security, Oracle, MongoDB, J, HTML, AJAX, JavaScript, , Angular11, Docker, Kafka, Camel ,GIT, JIRA, PL/SQL, IntelliJ, JUnit, Log4J, Maven, Jenkins, , AWS IAM, EC2, S3, Micro Services, RDS, SQS, Jenkins, Jira, Swagger, Maven, Windows, Linux.

**Tradestation - New York, NY (Aug’ 2018– June’2020 )**

**Java Developer**

**Description: Tradestation** Global is a versatile trading platform that grants access to a broad spectrum of financial instruments like stocks, futures, forex, options, and bonds. It boasts features like real-time market updates, customizable graphing capabilities, and a variety of advanced trading tools.

Responsibilities:

* Used **Java 8 Method References feature to point to methods** by their names and used functional Interfaces.
* Maintained Interface compatibility and concurrency in the project using Java 1.8 new features like Lambda expressions, default, static methods and Concurrency API.
* Implemented React router for building the single page application.
* Oracle Coherence has been used to provide distributed Clustering architecture on WebLogic Clustered.
* Coherence cache is updated for dirty/clean reads by Trigger Message over JMS topic following user activity like add/update/delete.
* Used **Spring MVC framework for implementing Model-View- Controller (MVC)** architecture at the Web tier level to isolate each layer of the application so that complexity of integration will be reduced, and maintenance will be very easy.
* Used **Spring AOP** for solving crosscutting concerns like keeping logged data and transactions details etc.
* Used **Spring Batch for processing large amount** of data like transaction management, job processing, resource

management and logging.

* Secured the API's by implementing **Oauth2token-based authentication/authorization** scheme using spring security.
* Used **Hibernate ORM** framework as persistence engine and wrote Hibernate Queries to automate the mapping between Oracle database and Objects in Java.
* Involved in writing **SQL and Stored Procedures** for handling complex queries with help of TOAD and access them through Java Programs from Oracle Database.
* Worked with **MongoDB** to store non-relational data into collection and retrieve them whenever required.
* Used **WebLogic server to route our JMS queue messages** to different business floors and configured routes in WebLogic and used JMS in e-mail notification.
* Implemented the associated business modules integration using **Spring and Hibernate data mapping**.
* Implemented Object-relation mapping using Hibernate framework to fetch the data from Oracle12c database.
* Used **Spring MVC Transaction Management, Spring/Hibernate Batch Transactions& Hibernate cache concepts**.
* Implemented **Hibernate Template** to great extent making use of **HQL, Named Parameters, Named Queries and interfaces such as Expression, Query and Criteria**.
* Apache camel, Kafka end point used to **publish and consuming trade messages**.
* Implemented Cloud capabilities **like Centralized Configuration, Load Balancing, Service-to-Service call using Feign, Service Registry and Discovery using Eureka in Spring Cloud**.
* Member of the **Power Virtual Agents Team**. Works in collaboration with design team to design and developed dynamic and browser compatible user interfaces using **React.js, Redux, TypeScript, HTML5** and **SASS** that extends **CSS** with **dynamic behavior**.
* Extensively involved to develop **graphical dialog tree editor** that allows users to define bot responses and overall conversation flow.
* Experienced in implementing **fallback routes, conversation editor canvas** and **Microsoft Flow connectors.**
* Implemented **WAI ARIA** to improve **accessibility and testing** with screen reader software.
* Designed and developed custom reusable **UI widgets** such as Modal, Spinner, Dropdown, etc using **office-ui-fabric-react** to use throughout the application.
* Implemented custom **grid system** using **fabric Details List** to be used across application.
* Involved in write application level code to interact with **APIs**, Web Services using **AJAX**, **JSON**  and **XML**.
* **Redis Cache** used to cache reference Data.
* Created the React JS components and triggered Angular code to render the React components using life cycle hooks.
* Designed and develop Common Foundation Classes(Property Manager and Log Manager) using Java, JMS, Abstract Factory Pattern, and Java Cryptography Extension (JCE) to readd/write encrypted/clear text property files and do auditing, logging and debugging.

**Environment:** Java 1.8, concurrent package Spring Boot, Micro services, Restful Services, ReactJS, CI/CD pipeline (DevOps), JPA/Hibernate, Tomcat, Maria DB, Oracle DB, Intellij, AWS, Kubernetes, Maven, Tomcat, Java 11, Kibana, Design Patterns, Multi-Threading, SQL, GITHUB, Agile Methodology, Jenkins, Junit, Postman, Angular, Spring Batch, JSON, Elastic Search, Kibana, XML, JSON,JIRA

**Nomura Securities, NY (Mar’ 2017– Aug’2018 )**

**Java Developer**

**Description: Neovest** - Electronic Trading Platform of the hedge fund designed to provide traders with a service to actively manage their portfolios, analyze the market and execute trades at their discretion. Traders can buy and sell such securities as stocks, bonds, options, futures, mutual funds, FX and exchange-traded funds.

Responsibilities:

* Used Java 8 Lambda expressions and Stream API to support functional-style operations on streams of elements.
* Worked on Java functional programming concepts like Streams, Functional Interfaces, new date time API and lambda expressions.
* JProfiler is used to profile he application (Memory view and CPU view and Thread view are used ).
* Involved in developing responsive UI using HTML5, CSS3, JavaScript, Bootstrap and SPA (single page application) using Angular 6/8, Typescript, Angular-CLI, NGRX, Bootstrap and Web pack.
* Used Angular router and implemented module wide routing and created different forms using Angular data driven and template driven forms.
* Developed Angular Components, Services, and Filters and took part in developing unit test and integration test cases using Jasmine, Karma.
* Developed generic component for reading files and writing to files using Java IO and NIO.
* Generic component for reading from IBM MQ and Writing to MQ queues.
* Used Spring Boot which is radically faster in building cloud Micro Services and develop Spring based application with very less configuration.
* Developed Micro services & APIs using Spring Cloud, Spring Security, Spring Boot, Spring Integration.
* Design and implementation of different micro services and its Docker files.
* Created and maintained Docker images and Docker containers.
* Configuring the Docker containers and creating Docker files for different environments.
* created Docker machine in Amazon EC2 instance and created Docker containers in Docker Machine running in Amazon EC2
* Deploy a Spring Boot Application to **AWS** Beanstalk
* To send emails to users developed Java API to interact with the **Amazon SNS**.
* Involved in creating EC2 instances and installed required configurations and applications on it and interact with S3 buckets for storing object level data into it.
* Used AWS Cloud Watch to monitor the application and to store the logging information.
* Spring Data JPA used retrieve and update, inserting data in database.
* Secured the API's by implementing Oauth2 token-based authentication/authorization scheme using spring security.
* JSON request/response from UI is processed using Spring Boot, Restful features with micro services Architecture and Spring Data JPA framework is used to save/retrieve the data in the backend.
* Used Node.js CLI tool Node Package manager (NPM) to build Angular application.
* Used Apache Maven as a build tool for building the Restful web-service

**Environment:** Java 1.8, concurrent package, lambda expressions, streams, J2ee, webservices, JMS,XML(Dom, Sax, Xpath, Stax, XSLT),IBM MQ, Kafka 2.0, Angular6/8, Spring Boot, Apache Camel 2.2 , Spring Batch, Spring Data JPA, EHCache, Jprofiler, Eclipse, Jenkins, Confluence, Jira, SourceTree , Swagger, Maven, Windows, Linux,DB2,Sql Server, Coherence.

**Client: AIG, Jersey City, NJ Feb’2016 – Mar’ 2017**

**Role: Java Developer**

**Description:** The project was to automate the claims processing systems and enhance the CRM by providing real time monitoring for its Crop insurance division and determine the premium of the old and new customer.. consolidate their Claims data existing in multiple databases across the country.

**Responsibilities:**

* Involved in design, development and testing phases of the SDLC using Agile Methodology.
* Developed user interface using JSP Tag libraries, third party libraries and Java Script to simplify the complexities of the application and developed the User Interface using HTML, CSS, and Ajax.
* Developed the STP Interface subsystem using Message Driven Beans that pushes trades to the STP system and listens for any incoming trade confirmations, execution and settlement reports
* Used Bootstrap and Angular JS to create Controllers to handle events triggered by clients and send request to server.
* Integrate the dynamic pages with Angular JS .
* Used AJAX API for intensive user operations and client-side validations.
* Consumed RESTful web services using Angular JS/AJAX to fetch the server data and used JSON parsing to render the data in UI.
* Developed application using Struts MVC Framework wrote Configuration XML file that contains declarations and another dependent object declaration.
* Inserted Dependency Injection between components using Spring’s (IOC) concept.
* Used Multithreading in programming to improve overall performance using Singleton design pattern in Hibernate Utility class.
* Implemented Asynchronous messaging between components using JMS.
* Used Singleton design pattern in Hibernate Utility class.
* Developed various Java beans for the performance of business processes and effectively involved in Impact analysis.
* Worked as a back-end developer and developed several REST endpoints to process the XHTML documents to import them in the Oracle Database or fetch them using Grails framework
* Developed application service components and configured beans using Spring IOC, creation of Hibernate mapping files and generation of database schema.
* Deployed scalable services, securely and reliably, on a wide variety of platforms using Docker.
* Developed Web Logic container security components for adding vendor specific Security Realm to application using JMX.
* Implemented SOA architecture with Web Services using SOAP, WSDL, UDDI and XML using CXF framework tool/Apache Commons. Designed and developed Hibernate DAO is using spring framework’s Hibernate Dao Support and Hibernate Template.
* Configured and managed Apache Web Server on Linux for web based applications using REST/SOAP, Apache and WebSphere.
* Creating and managing user accounts, system security, performance tuning, change management, documents and procedures on Linux.
* Worked on Cassandra Querying language to load bulk of data and execute queries.
* Created Data Access Objects (DAO) to add, modify and deletion of records.
* Created tables, triggers, stored procedures, SQL queries, joins, integrity constraints and views for multiple databases, Oracle using TOAD tool.
* Involved in Bug fixing of various modules that were raised by the testing teams in the application during the Integration testing phase.
* Implemented Log4J for Logging Errors, debugging and tracking.
* Performed deployment of applications IBM WebSphere Application Server.

Environment: Java, Html, CSS, JavaScript, JMS,JSP, J2EE, DOM/SAX PARSERS, JMS, Spring (Services, DAO, MVC, IOC) and Struts Framework, Hibernate, Spring, RESTful, SOAP, JMX, Linux, WSDL, UDDI, Oracle, PL/SQL, Log4J, Maven, Tortoise SVN, Web Sphere.

**Client: IBM- Bangalore India July’2013 – Dec’ 2015**

**Role: Java Developer**

**Description** Insight is an application that defines the ways for Service Providers to communicate with Vector which is a SMS gateway application that allows for the dissemination of SMS. It is designed with a robust architecture and technology to manage high volume through puts and have maximum up-time. It has many modules to help customers to have a secure path to send a bulk number of messages to a bulk number of mobile without hustle. Some of the modules among them are bulk messaging, and 2 FA. In this application, the 2-Face authentication helps in the security for the customers to have individual profile and individual sender-address. Bulk Messaging helps the customer to send a million messages at a time without an issue. This application takes the list of the mobile numbers and a customized SMS, appends the sender address to their message and to their customers in bulk. These two modules contain many work flows and many reports. Reports help the customer and service provider to pull the statistical analysis of the messaging report.

**Responsibilities:**

* Understanding user requirements participating in design discussions, implementation feasibility analysis on server-side
* backend levels, and documenting requirements.
* Coded Action classes, Java Beans, Service layers, and Business delegates to implement business logic with the latest
* features of JDK, such as Annotations and Generics.
* Applied J2EE design patterns including Business Delegate, Service Locator, Value Object (Transfer Object), Data Access
* Object, and Session Facade to architect and develop scalable, maintainable, and modular enterprise solutions.
* Used Validator framework to validate client-side data and exception handling using Struts exception handler.
* Worked on Web services using standard protocols such as XML, SOAP, and HTTP.
* Involved in the Struts upgrade to 2.x, which involved changes to the action class.
* Migrated JSPs from Struts1 to Struts2, removing custom tags and modernizing the application's user interface.
* Developed Java Web Services for different phases. Used Struts2 framework to handle application requests using SOAP
* web services. Implemented the data persistence using Hibernate.
* Designed and Implemented SOAP web services using JAX-WS for different operations provided in User Interface.
* Used JDBC driver to connect to the Oracle database for all data transactions.
* Proficiently developed Stored Procedures, Cursors, Triggers, and Views in a PL/SQL Database environment, facilitating
* efficient data storage, retrieval, and updates.
* Developed the application using IntelliJ IDE, used the WebLogic server to deploy the applications, worked on
* the JMS queues, and configured the WebLogic server.

**Environment:** UML, Java, , Struts, Axis , EJB2.0, JSP, Webservices, Eclipse 3.2 PDE JBOSS 4.0.2/, Jboss 4.0.5 Java, J2EE, Servlet’s, Struts, SOAP, XML, Hibernate, Oracle