**SAI RISHITHA CHANDA**

**Senior Java Full Stack Developer**

#### chandasairishitha1@gmail.com**|** **+1 314-626-0280**

**PROFESSIONAL SUMMARY**

**Senior Java Full Stack Developer around 10+ years** of software development experience in **Java/ J2EE, Spring Boot, Spring MVC, JSP, C, C++, JavaScript, SQL, PL-SQL, Struts, Hibernate, Web Services**. Involved in complete SDLC from inception, planning, design, coding and testing to project closure mainly in the intranet, Internet and standalone application design, development, maintenance and production support in n-tier architecture.

* Experience in all phases of **Software Development Life Cycle (SDLC)**, which includes Integration, Implementation, Interaction, Business Analysis/Modeling, Documentation, and Testing of all software applications.
* deployment of internet, e-commerce and client-server applications using **Java, Servlets, JSP, JavaBeans, JDBC, Spring, Hibernate, AJAX, JavaScript.**
* Involved in Application Design & Architecture using **Core Java 1.5, J2EE, EJB, Spring, Hibernate, JSP, Struts, Servlets, JMS, MDB, XML, Web Services (CXF), JAX - WS**.
* Experience in User Experience (**UX**) design and development, User Interactive (**UI**) Web Pages and visually appealing User Interfaces using **HTML 5, CSS3, Bootstrap, SASS, JavaScript, jQuery, Angular JS, Angular 4/8/9/11, React JS, Node JS and understanding in WCAG Guideline**.
* Experienced in using **React.JS components, Forms, Events, Keys, Router, Animations and Flux concept.**
* Expertise in various open-source frameworks like **Angular JS, Node JS, Struts 2.0/1.2, Spring 2.5/3.0, Spring MVC, IoC, Web and ORM Technology like Hibernate 3.0, JPA, EJB3.0, IBatis, MyBatis, JQuery.**
* Expertise in Web-Development technologies like **Servlets, JSP, JSTL, HTML5, JSON, JavaScript, CSS3, XML, XSL/XSLT, AJAX, JQuery.**
* Extensive experience in Amazon Web Services like **EC2, Simple DB, RDS, Elastic Load Balancing, SQS, SNS, AWS Identity and access management, AWS Cloud Watch, Elastic Beanstalk and Cloud Front, Redshift.**
* Designed and developed the **REST** based Microservices using the **Spring Boot, Spring Data with JPA (Java Persistence API) and deployed Microservices onto Docker.**
* Solid Experience in creating a Kubernetes Clusters in **AWS** and deploying a **Docker** Image on top of **Kubernetes Cluster**.
* Profound knowledge in configuring and deploying web applications in **Tomcat Server, Web Logic, Web Sphere and JBoss.**
* **Developed RESTful APIs and GraphQL endpoints** with MongoDB for efficient data handling in full-stack applications**.**
* Implemented authentication and authorization using JWT and OAuth with **MongoDB** for user management**.**
* Integrated **RabbitMQ** with Spring AMQP to enable reliable message publishing, acknowledgment, and error handling.
* Experienced in testing **Angular, Angular 2** applications using **Jasmine, Mocha, Karma, and Angular Test Bed.**
* RDBMS experience includes **Oracle, SQL Server**; database programming and connectivity using **JDBC, ODBC, SQL, PL/SQL, Stored Procedures, triggers, Queries.**
* **Implemented ORM frameworks** like Hibernate, JPA, and MyBatis for efficient database interaction, reducing boilerplate SQL code and improving maintainability.
* Improved data flow performance by 40% through optimized processors and parallel streaming in **Apache Camel** routes.
* Strong knowledge of **Java** and **Spring Boot** for building backend services integrated with Flink.
* Implemented efficient data retrieval and manipulation operations through **ORM frameworks** like Hibernate to ensure seamless database transactions and CRUD operations.
* Good experience in **AWS** services **like Elastic Compute Cloud(EC2), Elastic Load-balancers, Elastic Container Service (Docker Containers), S3, Cloud Watch, Cloud Trail, Cloud Formation.**
* Developed RESTful APIs that interacted with Couchbase for CRUD operations, leveraging N1QL for advanced query capabilities like SQL.
* Strong Experience in of **Software Development Life Cycles (SDLC)** **Agile, Waterfall Model, Rational Unified Process (RUP).**
* Developed **big data processing applications** using **Scala** and **Apache Spark** to handle large-scale datasets efficiently.
* Experience in **Test Driven Development (TDD), using Spring JUnit Runner, Mockito.**
* Experience working with various SDLC methodologies based on **Agile/SCRUM** like Test Driven Development (TTD), Incremental and Iteration methodology, Pair Programming, Agile Development.
* Good working knowledge on Version Control Systems like **GIT, SVN and CVS.**

# **TECHNICAL SKILLS:**

|  |  |
| --- | --- |
| **Languages** | Core Java, Java 1.x to 1.8, Groovy, SQL, PL/SQL, JavaScript, TypeScript, Python, Shell Script |
| **Cloud Services** | AWS, Azure |
| **Web/App Servers** | Apache Tomcat 8, WebLogic12, JBoss4.5, IBM WebSphere, Jetty, Nginx  |
| **Web Service Framework** | Apache Axios, Jersey, AWS, and Apache CXF  |
| **Mark-up/ XML Technologies**  | HTML, HTML5, XHTML, JavaScript, CSS, AppML, XML, XSD, XSL/XSLT, SAX/DOM, AJAX, JQuery, JSON, Xpath and XLink |
| **Databases** | Oracle 12 /11g, SQL Server 2016, Mongo DB, MySQL, IBM DB2 and PL/SQL procedures,Cassandra, MongoDB, PostgreSQL, Amazon RDS |
| **Tools & Framework** | Angular 4/6/7/9,Struts, Spring 4.x, Hibernate4/3, JPA, Spring MVC, Spring WebFlow, Spring WebFlux, Spring DAO, Spring Batch, Spring Integration, JPA, JSF, Log4J, Junit, SOAP UI, Gradle, Apache Ant and Apache Maven, AWS |
| **Tools** | GWT, Eclipse, Hudson, Sonar& Jenkins, JIRA, Nexus, FileZilla, Maven and Gradle |
| **Testing** | Junit, Mockito, Power Mock, Test NG, Karma, Protractor, Selenium, Cucumber |
| **Web Service Protocols** | SOAP, REST, JAX-RPC, JAX-WS, JAXB, WSDL and UDDI |
| **Version Control Tools** | CVS, SVN, GIT,CM Synergy, and Clear Case, Bitbucket |
| **Operating Systems** | UNIX, Linux, Ubuntu, Windows XP/2000/VISTA/7/10/Server |

**PROFESSIONAL EXPERIENCE**

# **Client: State Street Bank, New York, NY OCT 2023 till now**

# **Role: Sr. Java Full Stack Developer**

#### **Responsibilities:**

* Implemented **Java 17** to use asynchronous HTTP client, var keyword for **lambda parameters**, improvise garbage collector and to access control in nest based.
* Developed and modernized microservices using Java 17, Spring Boot, Spring Cloud, and Spring Batch to enhance scalability and performance.
* Developed RESTful and GraphQL APIs using Java 17, leveraging its concise syntax and runtime improvements.
* Built cloud-native microservices with Java 17 and deployed them on Kubernetes and AWS.
* Experimented with GraalVM to build native executables using Java 17 for reduced memory consumption and startup times
* Implemented **Java 17** features, such **as lambda expressions, the Stream API, and functional interfaces.**
* Designed dynamic and cross-browser compatible pages using **HTML, CSS, JavaScript, Angular 16 and Typescript.**
* Utilized **Angular 16** modules, services, templates, directives, dependency injection, lazy loading, and differential loading for SPA development.
* Implemented reactive programming patterns using **RxJS** observables, allowing for efficient data flow management and real-time updates within the application.
* Used **Bootstrap** **JavaScript** plug-ins to implement Tabs, Tooltips, Pop Overs, Buttons and Alerts.
* Integrated **Angular’s** HTTP client module to interact with **RESTful** **APIs**, handling data retrieval, submission, and error handling effectively.
* Migrated legacy SQL-based systems to **Couchbase**, improving scalability and reducing query latency by over 40%.
* Used **Angular** **16** to parse the data fetched from APIs and used backbone.js to develop an application to perform **CRUD** operations on the database server.
* Optimized **Spark jobs** by tuning **RDD transformations, DataFrames, and Datasets** to improve performance.
* Integrated **Apache Spark** with **Hadoop, HDFS, Hive, and Kafka** for data ingestion and processing.
* Implemented and managed CI/CD pipelines using **Jenkins**, **GitLab**, and **AWS CodePipeline** to automate build, test, and deployment processes, improving release efficiency by **30%**.
* Integrated Hibernate **ORM with Spring Boot applications**, using Spring Data JPA for data access layer abstraction and repository pattern implementation.
* Developed **scalable and high-performance applications** using **Scala** in a distributed computing environment.
* Implemented **functional programming paradigms** in Scala to build efficient and maintainable code.
* Designed and developed **microservices** using **Scala, Akka, and Play Framework** for distributed systems.
* Designed and maintained **ORM mappings**, relationships (One-to-One, One-to-Many, Many-to-Many), and inheritance hierarchies to represent complex business logic in the relational database.
* Developed **RESTful** **APIs** with **Spring** **Boot**, utilizing **IoC** and **Dependency** **Injection** for modular design.
* Integrated Spring Security with **OAuth2**/**Open** **ID** Connect TUF CWS Security Wrapper for robust authentication and authorization, ensuring application security.
* Hands-on experience in developing integration with **Elastic search** in any of the programming languages like **JAVA and JavaScript.**
* Leveraged **Spring Data JPA** for efficient database operations, and integrated **Swagger** for **API** documentation.
* Used **Maven** for project management and build automation and **Continuous Integration** is done using **Jenkins**.
* Designed **Microservices** **architecture** with **Eureka** for service registration and discovery.
* Implemented inter-service communication using **RESTful APIs**, **message queues (Apache Kafka),** or event-driven architectures (e.g., **Apache Kafka, AWS SNS/SQS**).
* Migrating existing applications into **Microservices architecture** using **Rest APIs** and **Spring Boot.**
* Proficient in writing comprehensive unit tests with **JUnit, practicing TDD**, utilizing annotations, and integrating mocking (**Mockito**) for reliable codebase.
* Used **Git** as **version control** to maintain versions of the application.
* Utilized **MongoDB** with **Node.js** to implement database interactions, performing **CRUD** operations and complex queries to store and retrieve data.
* **Designed and optimized MongoDB schemas** for high-performance and scalable applications in Java/Spring Boot environments.
* Created and maintained automation scripts using **PL/SQL** and shell scripting to automate routine database tasks, such as data extraction, transformation, and loading (**ETL).**
* Implemented authentication and authorization using frameworks like **Passport.js or JSON Web Tokens (JWT),** ensuring secure access to protected resources and user authentication.
* Developing and maintaining **Kafka** producers and consumers to interact with the **Apache Kafka** cluster by integrating other systems and tools such as databases, message brokers, and data storage systems.
* Developed **AWS Lambda** functions using Java to implement serverless and event-driven architecture to handle specific business logic and tasks, such as data processing, file handling, or **API integrations.**
* Integrated **AWS Lambda functions** with other **AWS services**, such **as Amazon S3, DynamoDB, or API Gateway,** to create scalable and resilient serverless applications.
* Utilized **AWS SDK**s and APIs to interact with various **AWS services** within the **Lambda functions.**
* Experienced in implementing Microservices, **Service Oriented Architecture (SOA)**with **XML-based Web Services (SOAP/WSDL)** using Top Down and Bottom-Up approach.
* Used **AWS SDK** for connection to **Amazon S3 buckets** as it is used as the object storage service to store and retrieve the media files related to the application.
* Deployment **(CI/CD**), automating the build, testing, and deployment processes.
* Used **Terraform** to define and manage infrastructure as code (**IaC**) for provisioning, managing, and scaling cloud resources and infrastructure.

**Environment**: Java 17, Spring Boot, Spring Security, Eureka, HTML/CSS/Typescript, Microservices, Material, Node JS, Web Services, IntelliJ, Postman, REST services, JUnit, JMeter, Gradle, GIT, Contrast, Angular 16, Accessibility, Tomcat, Apache Kafka, MongoDB, AWS, Kubernetes, Docker, Concourse, Cucumber, Protractor, Karma, Jasmine, Jenkins, Jira, Fargate.

# **Client: GE healthcare, Chicago, Illinois April 2021 to Sep 2023**

# **Role: Sr. Java Full Stack Developer**

# **Responsibilities:**

* Involved in various phases of Software Development Life Cycle (**SDLC**) such as requirements gathering, modeling, analysis, design, development, and testing.
* Developed the view-controller components using **JSPs, HTML5, CSS3, AJAX, JavaScript (jQuery).**
* Implemented routing logic and navigation from screen to screen and implemented login functionality on the client side in **AngularJS**.
* Hands-on experience in developing integration with **Elastic search** in any of the programming languages like **JAVA and JavaScript**.
* Implemented core features **of AngularJS** framework such as **dependency Injection, Databinding, Filters, Directives, Templates, Services and Deep Linking Designing.**
* **Worked with Docker and** Kubernetes to deploy full-stack applications with **MongoDB** in microservices architecture**.**
* **Integrated MongoDB with Azure services** like Lambda, S3, and DynamoDB for cloud-based full-stack solutions.
* Used **Java 8** features like **stream and Lambda expressions.**
* **Developed and automated CI/CD pipelines** using Jenkins, GitLab, and GitHub Actions for efficient build, test, and deployment processes.
* Integrated **Maven**, **Gradle**, and **npm** into CI/CD pipelines for automated dependency management and build automation.
* Enhanced code quality by integrating **SonarQube** into CI/CD pipelines for static code analysis and security scanning.
* Managed infrastructure provisioning using **Terraform** and **CloudFormation** integrated into CI/CD pipelines.
* Used **Java 8** Method References feature to point to methods by their names and used functional Interfaces.
* Used **Spring MVC** framework for implementing **Model-View- Controller (MVC)** architecture at the Web tier level to isolate each layer of the application.
* 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 **Oauth2** token-based authentication/authorization scheme using **spring security**.
* Worked on setting up **Maven** scripts to build, package, and deploy application code to the target **WebLogic Application** Servers and worked on **continuous integration** servers like **Jenkins**.
* Implemented **multithreading** and concurrency features in **Java 11+** to optimize performance and enhance responsiveness in a high-demand application.
* Monitored and maintained the **RabbitMQ** cluster, utilizing tools like Prometheus and Grafana for monitoring message throughput, queue depth, and system health.
* Utilized **RabbitMQ** for load balancing and scalability in high-traffic environments, reducing latency and improving the performance of distributed services.
* Integrated **Azure** Active Directory (**Azure AD)** for authentication and authorization in applications.
* Utilized **Azure Blob Storage**, **Azure SQL Database**, and **Azure Cosmos DB** for data storage and management in cloud-based applications.
* Deploying the applications into the **Azure** and extracting the data from the **Azure cloud services**.
* Experience with **Core JAVA** with focus on **multithreading** and **data structures.**
* Experience with leveraging **Terraform** providers and plugins to interact with various cloud services and **APIs**.
* Built **Terraform** code and Arm templates for resources to be provisioned in **Azure cloud.**
* Integrated **Spring Data JPA** with **Spring Boot** to simplify database operations and facilitate seamless communication with a relational database.
* Utilized **Spring Security** to implement authentication and authorization mechanisms, ensuring secure access control to the application's resources.
* Developed **RESTful web services** using **Spring MVC**, allowing clients to consume and interact with data via well-defined API endpoints.
* Integrated **messaging systems** such as **Kafka and RabbitMQ** to enable asynchronous communication and event-driven architecture between microservices.
* Closely worked with **Kafka Admin** team to set up **Kafka cluster** setup on the QA and Production environments.
* Had knowledge on **Kibana** and **Elastic search** to identify the **Kafka** message failure scenarios.
* Build prototype for various required services such as Scheduling, Logging and Notification Service using third party **Node JS** based **JavaScript** library.
* Utilized project management tools like **JIRA, Trello, or Asana** for effective project tracking, task management, and collaboration, promoting transparency and teamwork.
* Employed **Hibernate** annotations, such as **@Entity, @Table, @Column, and @OneToMany, to effectively map Java objects to corresponding database tables.**
* Successfully utilized relational databases such as **MySQL, PostgreSQL, and Oracle** in conjunction with **Spring Data JPA.**
* Configured **IBM MQ** clusters to ensure high availability and scalability, enabling seamless communication across distributed environments.
* Worked in Agile TDD environment using SDLC tools like **GIT, Jira, Jenkins, Maven** for build packaging and deployments of the application components.

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

# **Client: State Of Michigan, Michigan Aug 2018 to Mar 2021**

# **Role: Java Full Stack Developer**

# **Responsibilities:**

* Participated in various phases of **Software Development Life Cycle** including Requirements Analysis, Design, Implementation and Maintenance in **Agile and Daily Scrum environment.**
* Expertise in Object Oriented Software Development (**OOAD**), Gathering new requirements and understand/document business/technical requirements in **UML** using **IBM** Rational suite of products.
* Leveraged **Kibana's** capabilities to integrate with **Java 1.8** features such as Functional Interfaces, Parallel Streams, and Lambda expressions, enabling efficient data analysis and visualization.
* Implemented **UI features** for user interactivity and responsiveness **using HTML5, CSS3, and Bootstrap.**
* Used jQuery to make the **HTML, DHTML and CSS** code to interact with the **JavaScript** functions to add dynamism to the web pages at the client side. Working with **React Router** for developing SPA’s.
* Develop various screens for the front end using **ReactJS** and used various predefined components from **NPM.**
* Developing a native mobile app using **React Native, firebase, Node.js, and npm**
* Developing web applications and webpages using **React Js, firebase, jQuery, Bootsrap, and npm.**
* Used **SoapUI/Postman/ARC** for testing and developed **JUnit Test** cases to test the availability of the services.
* Designing and developing solutions for **CI/CD** and Release Management using **GIT, Jenkins, Maven, Docker, Artifactory, Kubernetes, Helm, and Amazon EKS** which propagates micro services from dev to production.
* Implemented **micro services** containers using **Docker and Kubernetes.**
* **Utilized Kafka and MongoDB** change streams for real-time data streaming and event-driven architectures.
* Deployed the applications in **Apache Tomcat Application Server 8.x.**
* Used **Spring Core** Annotations for **Dependency Injection Spring DI** and **for Restful web services APIs.**
* Involved in stage deployments using **Tomcat server.**
* Utilized **Terraform** for automatically managing resource dependencies, ensuring that resources are created or modified in the correct order to satisfy inter-resource dependencies.
* Converted a monolithic app to **microservices architecture** using **Spring Boot** using 12-factor app methodology.
* Utilized **Spring Boot** Actuator to monitor and manage the application, exposing metrics, health checks, and diagnostics endpoints for operational insights.
* Used **Cucumber** to implement Behavior Driven Design (**BDD**) for acceptance tests and **Unit testing.**
* Developed the backend using **Groovy and Grails, value object and DAO.**
* Deployed the Application into **Docker** container and made easily accessible at runtime using Cloud Foundry and other cloud services like **AWS**.
* Skilled in configuring and optimizing **AWS** resources to meet specific application and business requirements, resulting in cost-effective and efficient solutions.
* Implementing a real-time data streaming solution using **Apache Kafka.**
* Involved in designing and developing **Kafka** producers and consumers in Java to ingest and process high-volume data streams.
* Building a distributed system where microservices communicate with each other through **Kafka**.
* Utilized **Node.js** in microservices architectures, building lightweight and scalable services that can be independently deployed and communicated through **APIs or message queues.**
* Implemented security measures, such as **IAM** roles and policies, to restrict access and ensure the secure execution of **Lambda functions.**
* Implemented deployment automation using **AWS CloudFormation** or **Infrastructure as Code (IaC)** tools to enable consistent and repeatable deployments of Lambda functions.
* Developed **RESTful** web services using **Java Servlets JSON JAX-WS and JAX-RS** and persistence using **Hibernate and JPA framework.**
* Designed and developed **RESTful JSON/XML** web service for handling asynchronous **AJAX** requests.
* Utilized **Spring Boot's** testing framework, including **JUnit and Mockito**, for **unit testing and integration testing**, ensuring the reliability and quality of the application.
* Skilled in scripting and automating **API tests in Postman**, ensuring efficient and repeatable testing processes.
* Implemented **Test Driven development** using frameworks like **Junit, Mockito and Easy Mock**.
* Designed and implementing **Session Facade, MVC, Front Controller and DAO design patterns.**

**Environment**: J2EE, JDK, HTML, CSS, Ajax, Node JS, React JS, jQuery, Express’s, JavaScript, Node.js, Bootstrap, Oracle, Spring, Hibernate/SQL, AWS, Web Services (SOAP), JMS, RESTful, Multithreading, WebSphere, Apache, Maven.

# **Client: Bank of America, Florida, Nov 2016 to July 2018**

# **Role: Java Full Stack Developer**

**Responsibilities:**

* Defined and constructed layers, server-side objects and client-side interfaces based on **J2EE design patterns,** Developed and implemented **Swing, spring and J2EE based MVC (Model-View-Controller)** framework for the application.
* Extensively involved in developing Web interface using **JSP, JSP Standard Tag Libraries (JSTL) using Struts Framework.**
* Installation, configuration, tuning, backup, recovery, and upgrade of **IBM WebSphere Application Server V 6.x/7. Network Deployment, IBM WebSphere MQ V6.0/7.0 on AIX, Solaris, and Linux.**
* Configured WebSphere JMS settings like **Queues, Topics, Activation Specifications, Queue connection Factories and Topic connection Factories for WAS- MQ connectivity.**
* Configured virtual hosts and transports for **WebSphere application servers**.
* Actively involved in coding **using Java 1.8, spring 4.x, and Angular 4** to develop features from end to end.
* Developed application on spring framework by utilizing its features like **Spring Boot, Spring Dependency injection, Spring Beans, Spring Security, Spring JPA, and Spring Web flow using MVC.**
* Developed Microservices business components and **RESTful** service endpoints using **Spring Boot**.
* Utilized **Spring Boot's** internationalization support to create applications that can serve content in multiple languages and locales.
* Designed and documented **Kafka** topic schemas and message formats, ensuring data consistency and compatibility.
* Conducted performance tuning and optimization of **Kafka** configurations to improve message throughput and latency.
* Involved in the **Design UI Screen**, and created several mocks up Using **HTML5, CSS3, and JavaScript**.
* Hands-on experience **with JavaScript, Node JS, and Angular JS** as the development frameworks to build Single-page applications (SPA).
* Involved in the design and development of server-side layer using **XML, JSP, JDBC, JNDI, EJB, and DAO patterns using Eclipse IDE.**
* Implemented the Routing part of the application using **Angular 4 UI router module**, routing provider service with **HTML views/CSS styles.**
* Developed cross-browser and multi-browser compatible web pages using **HTML5, CSS3, and JavaScript.**
* Designed dynamic client-side **JavaScript**, codes to build web forms and simulate processes for web applications, page navigation, and form validation.
* Configured **Java objects** with backend database tables using **ORM** mapping in **Hibernate**.
* Configure and monitor numerous **MongoDB** instances and replica sets.
* **Worked with Docker and Kubernetes** to deploy full-stack applications with **MongoDB** in microservices architecture.
* **Implemented security best practices** in MongoDB, including authentication, authorization, and encryption.
* Build **REST** web service by building **Node.js** Server in the back end to handle requests sent from the front-end **jQuery Ajax calls.**
* Used **Micro-services** architecture, with **Spring Boot-based services** interacting through a combination of **REST and Spring Boot.**
* Development carried out under Eclipse Integrated Development Environment (**IDE**).
* Integrated **Spring DAO** for data access using **Hibernate**, used **HQL and SQL** for querying databases.
* Wrote **Hibernate** configuration and mappings **XML** files for database access.
* Worked on implementing **CURD** operations using **mark logic NOSQL REST API services.**
* Created and Designed **XML** schemas and transformed **the XML to XSLT style sheet.**
* Tested Restful web services using **SOAP UI Pro** and coded in groovy to develop data driven testing.
* Actively participated in the **OpenShift** and **Kubernetes** communities, contributing to forums, blogs, and open-source projects.
* Implemented various design patterns in the project such **as Data Access Object, Data Transfer Object and Singleton.**
* Created Stored Procedures in **PL/SQL**, Functions for the **DAO** layer by studying the required business objects and validating them using **Oracle database.**
* Experience in Creating and Updating Statistics to improve the **SQL** Server Performance.
* Successfully integrated various **AWS services, including EC2, S3, RDS, and Lambda**, to design and deploy robust and scalable applications.
* Utilized **AWS CloudWatch** to set up monitoring, alarms, and log analysis, ensuring proactive management and swift issue resolution.
* Proficiently worked with **AWS storage solutions like S3 and EBS**, as well as managed relational databases such as **RDS and NoSQL** databases like **DynamoDB**.
* Created and configured new **JIRA** projects and worked with departments to maintain existing **JIRA** projects.
* Used **log4j** for logging and **SVN** for version control.

**Environment:** Spring Core, Spring IOC, Spring MVC, Struts, JavaScript, Node JS, jQuery, JSON, Spring DAO, Spring Boot, Spring Batch, spring Security, Spring Integration, Vaadin, Hibernate, JSP, Oracle, PL/SQL, XML, spring web flow, Maven, JUnit, REST web service, HTML, CSS, SASS, AJAX, Jira, Angular, log4j, SVN.

# **Client: Third ware Solutions, INDIA. Jun 2013 to Jan 2016**

# **Role: Java Developer**

# **Responsibilities:**

* Responsible for developing Java components using Spring, Spring JDBC, and Spring Transaction Management
* Used Spring JDBC in the persistence layer capable of handling high-volume transactions.
* Drew sequence diagrams and Class diagrams using UML.
* Utilized Agile Methodologies to manage the full lifecycle development of the project.
* Used Maven and configured Jenkins to build and deploy the application.
* Used form classes of the Spring Framework to write the routing logic and to call different services.
* Used Spring DAO to connect with the database.
* Wrote PL/SQL stored procedures and did performance tuning of complex queries.
* Used the Spring framework for the middle-tier and Spring JDBC templates for data access.
* Developed SOAP/REST-based Web Services using both SOAP/WSDL and REST.
* Developed logging module using Log4J to create log files to debug as well as trace application.
* Express using **Node.js** and Implemented Back-end API using Node.js with the help of Express Framework.
* Involved in writing and modifying procedures, queries, views, and triggers and calling them from JavaScript using the local application framework
* Developed various test cases and performed unit testing using Junit.

Environment: JAVA EE 5, JUnit, AJAX, jQuery, JSP 2.1, Node JS, Mongo Db, SQL Server 2008, Web services, Apache Axis, XSLT, Servlets 2.5, Struts Tiles, XML, Spring 2.5, Spring Security 2.0, Spring MVC, Hibernate 3.0, Log4j, WebSphere 6.1, TOAD, Tortoise SVN.