**Sai Priya**

**Sr. Full Stack Java Developer**

Sales21@kteksoft.com| 4794375046

**PROFESSIONAL SUMMARY**

* 10+years of IT experience as a Java Full stack Developer with diversified exposure in **Software Process Engineering**, designing & building Web Applications using Java/J2EE and open-source technologies.
* Profound experience in **Software Development Life Cycle (SDLC)** phases such as analysis, planning, design, implementation, testing and maintenance.
* Extensive experience with iterative methodologies like **Agile (SCRUM)**, Waterfall, and Test-Driven Development (TDD). Utilized tools like **JIRA, Rally** to effectively manage epics, features and kanban boards.
* Experience in core java with strong understanding of **Object-Oriented Programming** (OOP) and concepts like **Collections**, **Multi-Threading**, **Exception Handling**, **Enums**, **Regex**, and **ArrayList**.
* Extensive programming experience in Java/J2EE and associated server-side technologies like **JDBC, Servlets, JSP, EJB, Spring, Hibernate, JRA, JPA, JTA, Java Mail, JMS, JNDI, JSON, XML, JAXP, JMX, RESTful** and **SOAP**.
* Proficient in building Web User Interface (UI) using **Angular 16, NodeJS, React, Bootstrap, JavaScript, Typescript, AJAX, JSON, XML, jQuery** that follows W3C Web Standards and are browser compatible.
* Expertise with backend programming using **SQL and PL/SQL Stored procedures, triggers,** and prepared statements on databases like **Oracle**, **SQL Server 2000**.
* Expertise in developing applications using **Object Relational Mapping** (ORM) frameworks like **Hibernate**, Eclipse Link, IBATIS, MVC frameworks like Struts, JSF, spring modules such as **Spring MVC, JDBC** and **Security**.
* Strong experience implementing **microservices** using **Spring Boot, Micronaut** and effectively leveraging messaging services technologies such as **JMS, Apache Kafka, RabbitMQ**, **IBM MQ** and **AWS Simple Queue Service (SQS)** for efficient and scalable communication between microservices.
* Experience with Application servers and Web servers such as **BEA WebLogic, IBM WebSphere, JBoss, Glassfish, Apache Tomcat**, **Apache HTTP Server**, and **Nginx** etc.
* Good experience in **NoSQL** databases **MongoDB, Cassandra**, and relational data modeling, object-relational integration (ORM) and physical design/tuning.
* Leveraged **Ant, Maven** and **Jenkins** for efficient project management, dependency management, and build automation, ensuring reliable and repeatable build processes.
* Strong working knowledge of maintaining/writing **SQL, PL/SQL, Triggers**, **Stored Procedures, NoSQL databases MongoDB** and **Cassandra**.
* Strong familiarity with cloud stacks **Azure**, **AWS**,and **GCP** cloud platforms to **deploy**, **maintain**, and optimize **scalable**, **secure**, and high-availability applications and services.
* Experience working with **XML** parsers like **JAXP** and **JAXB** for schema binding.
* Proficient understanding of code versioning tools, such as **Git, GitHub, SVN, CVS** and familiarity with **Rational ClearCase** (from IBM), **TFS** (Team Foundation Server from Microsoft).
* Hands on experience with build and deployment tools including **Maven, Gradle**.
* Logging and Debugging using **Log4j, Log back, JBuilder.** Unit and Integration testing using **JUnit, TestNG, Mockito, Cypress, Jasmine, Karma, Jest.**
* Skilled in integrating **SonarQube** into CI/CD pipelines, performing static code analysis, and ensuring adherence to coding standards**.**
* Proficient in utilizing Behavior-Driven Development (BDD) methodologies and frameworks like **Karate**, **Cucumber**.
* Integrated **Grafana** with **Prometheus** and other data sources to visualize critical metrics and **KPIs**, enhancing decision-making processes.
* Utilized **Kubernetes** Monitoring and Logging tools like **Prometheus, Grafana,** and **ELK** stack for monitoring and gaining insights.
* Implemented **CI/CD pipelines** using **Jenkins** and **GitLab** for automated deployment on **GCP** and **AWS** stacks.

**TECHNICAL SKILLS**

|  |  |
| --- | --- |
| **Programming Languages** | Java/J2EE, SQL, PL/SQL |
| **Java Technologies** | Core Java 17, J2EE, JSP, Servlets, JDBC, Struts, Java Reflection, Java Persistence API (JPA), Spring Boot, Spring MVC, EJB, Hibernate |
| **Web Technologies** | HTML5, CSS3, XML, Spring Boot, JavaScript, jQuery, Typescript, NodeJS, Angular 16.0, Vue.js, React, AJAX, SOAP, JSON, JSTL |
| **Web/Application Servers** | WebLogic, Apache Tomcat, WebSphere |
| **IDE’s** | Eclipse, WebStorm, IntelliJ Idea, NetBeans, Microsoft VS code |
| **Databases** | MySQL, PostgreSQL, Oracle, SQL Server, Cassandra, MongoDB |
| **Cloud Computing** | AWS, GCP, Azure |
| **Other Tools** | DORA, Apache Kafka, Terraform, Log4J, JUnit, TestNG, Maven, Gradle, Cypress, Jenkins, Jira, Git, GitHub, Kubernetes, SonarQube, Rabbit MQ. |
| **XML Technologies** | XML, SAX, DOM, XSL, WSDL, SOAP |

**PROFESSIONAL EXPERIENCE**

**Client: Huntington Bank, Columbus, OH April 2024 – Present**

**Sr. Java Full Stack Developer**

**Responsibilities:**

* Actively contributed to the full software development life cycle within an **Agile (Scrum)** environment, participating in design, development, system testing, and user acceptance testing phases.
* Developed high-performance **Single Page Applications (SPAs)** following a Micro Service Oriented Architecture, leveraging **Angular** 17.0 framework, **Cassandra DB**, and **MySQL** database.
* Implemented advanced **Angular** features including directives, expressions, routing capabilities, bookmarking, filters, validations, as well as **Angular UI** components, ng-Modules, Providers, Services, and navigation classes.
* Utilized **AngularCLI** and **TypeScript** to create Controllers and Angular components for web pages, optimizing performance through **Angular** Lazy Loading techniques.
* Implemented comprehensive unit tests for **Angular** using **Karma** and **Jasmine**, **Jest** ensuring code quality, and enabling test-driven development.
* Developed scalable trading platform with microservices using **Spring Boot**, **Angular**, enabling real-time trade execution and order management.
* Implemented payment transactions system with **Angular** front-end, **Spring Boot** microservices, and **Apache Kafka** for event-driven communication.
* Created reactive user interfaces using **RxJs**, **Angular**, and **WebSockets**, providing a seamless user experience.
* Incorporated smooth and visually appealing **Angular** animations, transitions, and effects to enhance the user experience and deliver polished interfaces.
* Leveraged **Angular**'s built-in **AJAX HTTP Service** with observables for efficient and incremental updates to the user interface.
* Integrated **Node.js** modules with **NPM** for streamlined dependency management, leveraging tools such as **Grunt** for task automation and **Express** for building fast, scalable APIs.
* Implemented a microservices architecture with **Spring Boot** based services, facilitating communication through **Apache Kafka** message brokers for seamless, scalable, and real-time **data** exchange.
* Extensively utilized **JPA 2.1**, **Hibernate** named queries, criteria queries, **Hibernate Query Language (HQL),** **PL/SQL,** optimistic locking, and caching for effective data processing from databases.
* Integrated business modules leveraging **Spring Boot** with **Cassandra DB** for scalable, and data management.
* Deployed **Spring Boot** based microservices on **Docker** containers using **Amazon ECS** (Elastic Container Service).
* Deployed **Docker**-contained **Spring Boot** microservices into **AWS EC2** container service using the **AWS** admin console and utilized **AWS Lambda** to run applications in the **AWS console**.
* Deployed Java-based microservices on **AWS Fargate**, leveraging containerization for scalable, serverless application hosting and eliminating the need for manual server management.
* Integrated **AWS Fargate** with **CI/CD** pipelines to automate the deployment and scaling of **Java** applications, improving deployment speed and reducing operational overhead.
* Developed and deployed serverless applications using **AWS Lambda** with **Java**, automating backend processes, and enabling scalable, event-driven architectures integrated with **DynamoDB**.
* Designed and optimized data access patterns in **DynamoDB**, using Java-based Lambda functions to efficiently handle high-throughput queries and ensure low-latency data retrieval.
* Extensively utilized **JPA 2.1**, **Hibernate** named queries, criteria queries, **Hibernate Query Language (HQL),** **PL/SQL,** optimistic locking, and caching for effective data processing from databases.
* Utilized **Cassandra Node Connector** for database persistence and interaction with **Cassandra** databases.
* Deployed front-end applications using **AWS S3** and configured **CORS** for consuming RESTful services across domains.
* Developed and maintained **Infrastructure as Code (IaC)** using **Terraform** for multi-cloud environments, including **AWS, Azure**, and **GCP**.
* Implemented modular **Terraform** **code** structures to improve reusability and maintainability across multiple projects.
* Utilized **Terraform** workspaces to manage environments (staging, production) with the same code base.
* Integrated **Terraform** with **CI/CD** pipelines for automated infrastructure provisioning and updates.
* Implemented state management best practices using remote backends like **AWS** **S3** and **AWS** **DynamoDB** for **Terraform** state files.
* Utilized **OpenShift**'s built-in continuous integration and continuous deployment (CI/CD) pipelines, integrating with tools like **Jenkins** and **Git**, **GitHub** for automated build, testing, and deployment of CDP pipelines.
* Utilized **Selenium WebDriver** to interact with web browsers and perform cross browser testing.
* Performed performance and load testing using **Selenium Grid** to simulate real world user behavior.
* Designed and implemented data models in **Cassandra DB** to support scalable and high-performance Java applications, ensuring efficient data storage and retrieval.
* Developed Java-based applications leveraging **Cassandra’s** distributed architecture for handling large volumes of data with low-latency read and write operations.
* Utilized the **DataStax** Java Driver for **Apache Cassandra** to integrate and interact with Cassandra clusters, optimizing data access and manipulation in Java projects.
* Implemented data consistency and replication strategies in **Cassandra** to maintain data integrity and availability across distributed environments.
* Integrated **SonarQube** into CI/CD pipelines for continuous code quality assessment, enabling automated static code analysis and real-time feedback on code vulnerabilities.
* Conducted in-depth code reviews using **SonarQube**, identifying critical bugs, code smells, and security vulnerabilities in Java applications, leading to more robust and maintainable codebases.
* Configured and customized **SonarQube** rules and profiles to align with project-specific coding standards, ensuring consistent code quality across development teams.
* Experienced in writing and optimizing complex **SQL** queries in **PostgreSQL**, analyzing execution plans, identifying bottlenecks, and implementing indexing strategies.
* Implemented **Apache Kafka** for building scalable, fault-tolerant, and real-time data pipelines, enabling efficient data streaming and message distribution across distributed systems.
* Designed and deployed **Kafka** topics, producers, and consumers, leveraging **Kafka**'s partitioning and replication mechanisms to ensure high throughput, low latency, and reliable message delivery.
* Developed scalable, fault-tolerant systems using **Apache Kafka** for efficient data streaming, implementing pub-sub messaging, event-driven architectures, and real-time data pipelines.
* Implemented user authentication and security for RESTful web services using the **OAuth 2.0** protocol.
* Utilized the **Swagger** Framework for implementing and testing RESTful APIs with a rich UI.
* Developed comprehensive unit test cases, test automation using testing frameworks such as **Karma, Jasmine, Jest, JUnit**, **Cypress**, **Mockito**.
* Familiar with designing and maintaining infrastructure as code using **Kubernetes** manifests for seamless application deployment on **OpenShift** platforms.
* Utilized **Kubernetes** monitoring and logging tools like **Prometheus, Grafana,** and the **ELK (Elasticsearch, Logstash, Kibana)** stack to monitor.
* Orchestrated containerized payments transactions applications using **Kubernetes** for automated deployment, scaling, and management.
* Configured and maintained **Jenkins** CDP pipelines for implementing continuous integration and continuous deployment (CI/CD) processes.

**Environment:** Java 17, Angular 16, Node.js, HTML5, CSS3 (SASS), Twitter Bootstrap, Kubernetes, ES6 JavaScript, GraphQL, OpenShift, AJAX HTTP Service, TypeScript, Spring Boot, JPA, PL/SQL, jQuery, OAuth 2.0, Apache Kafka, AWS, SonarQube, Cassandra DB, Terraform.

**Client: Highmark, Dallas, TX Jan 2024 – Mar 2024**

**Sr. Java Full Stack Developer**

**Responsibilities:**

* Developed complex user interfaces using **React.js**, harnessing its component-based architecture, virtual DOM, and declarative approach to build highly interactive and dynamic web applications powered by **React**.
* Leveraged **Redux** as a state management library in conjunction with **React**, ensuring a predictable and centralized data flow throughout the React application. Implemented actions, reducers, and selectors to manage the application's state seamlessly with **React** and **Redux**.
* Developed highly scalable real time applications using **RxJs** and **React**, optimizing performance and ensuring responsiveness.
* Leveraged **React** Native to develop cross-platform mobile applications, utilizing existing **React** expertise to create seamless and performant user interfaces for both iOS and Android platforms.
* Leveraged **React Hooks (use State, use Effect)** to manage state in functional components, enhancing code clarity and efficiency.
* Integrated **React** with **RESTful** **APIs** using **Axios/Fetch** for asynchronous data fetching, implementing error handling, and loading states.
* Integrated **React** with **Redux-Saga, React-Bootstrap/Material-UI,** and **React-DnD** for advanced state management**, UI design**, and drag-and-drop functionality.
* Implemented client-side interfaces using **React.js** and utilized middleware such as **Redux-Promise** to retrieve data from the backend and perform **RESTful services**.
* Developed **Micro Services** to provide Restful APIs using **Spring Boot** with **Spring MVC**, adhering to a Micro Services-based Cloud Architecture.
* Utilized **Spring Boot** and **REST** services extensively while developing web-based applications.
* Implemented **Spring Actuator**, **Spring Cloud**, and **Spring Sleuth** for all **Micro services** to monitor health metrics along with logging.
* Developed **micro services** to provide RESTful APIs using **Spring Boot** with **Spring MVC**, adhering to a **Micro services**-based Cloud Architecture, enabling modular and scalable application development.
* Designed and developed **Micro services** business components using **Spring Boot**, leveraging its opinionated conventions and embedded containers for streamlined development and deployment of micro services.
* Proficient in developing **RESTful APIs** using frameworks like **Spring Boot** and integrating them with **gRPC** and **RPC protocols** for efficient data transfer.
* Utilized **Java 1.8** features such as parallel streams and lambda expressions to handle search functionality in web applications developed using the **Spring MVC** framework.
* Implemented data access objects (DAO) and utilized **JPA-Hibernate** for the data access layer, performing CRUD operations.
* Created and maintained mapping files, persisted classes, and implemented transaction control in **Hibernate** for efficient database access.
* Utilized **Open Shift**'s monitoring and logging tools like **Prometheus** and **Elastic search** for application performance monitoring and troubleshooting in containerized environments.
* Implemented state management solutions leveraging **RxJs** observables and **Redux** principles.
* Leveraged **Node.js** as a client-side server, defined **XML** documents for Web Services input/output sources, and developed **XML schemas** for validation.
* Utilized **JMS** for asynchronous exchange of critical business data and events between **J2EE** components and legacy systems.
* Employed **Docker** for building **Docker images** and containers, enabling efficient management of code and dependencies.
* Worked with **GCP** services like **Google Compute Engine**, **Cloud Storage**, **Cloud SQL**, **Cloud Load Balancing**, and **Cloud Data store** for installation, configuration, and auto-scaling.
* Utilized **GCP**'s managed services for containerized applications, such as **Google Kubernetes Engine (GKE**), for orchestrating and managing containerized workloads.
* Leveraged **GCP**'s serverless capabilities like **Cloud Functions** and **Cloud Run** for deploying and scaling event-driven and stateless applications.
* Employed **GCP**'s machine learning services, such as **AI Platform** and **Vertex AI**, for developing and deploying machine learning models at scale.
* Implemented **Enterprise Java Beans (EJB)** for efficient interaction with databases, including insertion, updating, deletion, and retrieval of values.
* Experienced in data migration and **ETL (Extract, Transform, Load)** processes using **PostgreSQL**.
* Leveraged NoSQL databases (e.g., **MongoDB, Cosmos DB**) for data storage and processing in large scale **ETL** pipelines.
* Utilized cloud platforms (e.g., **AWS, Azure**) to provision and manage **ETL** infrastructure, ensuring high availability and scalability.
* Experienced in deploying, managing, and monitoring containerized applications using **AKS**.
* Skilled in leveraging **OpenShift**'s container orchestration capabilities for efficient application deployment.
* Utilized **log4j** for logging error messages and debugging purposes.
* Utilized **Junit, Cypress** and **Mockito** for comprehensive unit testing, built packages using **Maven**, **Gradle** and deployed applications on **WebLogic** **App** **Server** to ensure robust and reliable software performance.
* Automated build configuration and dependency management using **Gradle** for Java/Android projects
* Optimized build performance by implementing custom **Gradle** tasks and plugins
* Integrated **Gradle** with continuous integration/continuous deployment (CI/CD) pipelines
* Implemented **Rabbit MQ** for efficient message queuing, enabling reliable and scalable communication between distributed applications and services.
* Implemented **Rabbit MQ** exchanges, queues, and bindings for efficient messaging patterns including publish-subscribe and work queues.
* Utilized advanced **Rabbit MQ** features including acknowledgments and QoS settings for reliable, high-throughput message delivery.
* Utilized **Rabbit MQ**'s clustering and mirrored queues for high availability, fault tolerance, and uninterrupted message processing.
* Utilized **Jenkins** for continuous integration and continuous deployment (**CI/CD**) processes of CDP pipelines.
* Implemented **SOAP**, **WSDL**, and **UDDI** components for web services to interact with external systems.
* Integrated **WebSphere** with enterprise systems and databases like **DB2** and **Oracle**, ensuring seamless data flow and transaction management.
* Utilized **Oracle's** advanced features (partitioning, materialized views) for optimized performance, integrating **OAuth 2.0** for secure **API** access.
* Implemented **Oracle RMAN** and **Data Guard** for backup and disaster recovery, with **OAuth** authentication securing access to these processes.
* Deployed **Oracle RAC** for high availability, using **OAuth** tokens for secure inter-node and client authentication.
* Implemented **Ansible** playbooks to automate infrastructure provisioning, configuration management, and application deployment.
* Contributed to the design and maintenance of a robust **CI/CD** pipeline using **Ansible** for automated testing and deployment.
* Leveraged **Kubernetes** for orchestrating and managing containerized applications across distributed deployments.

Utilized **Kubernetes Config Maps** and **Secrets** for secure application configuration management, decoupled from container images and deployments.

**Environment:** Java 8, React.js, Redux, React Native, Spring, Spring Boot, Hibernate, HTML5, CSS3, Ansible, JavaScript, Micro services, Kubernetes, Rabbit MQ, OpenShift, Node.js, Oracle DB, OAuth, ETL, JMS, AJAX, XML, BDD, GCP, Docker, Maven, Gradle, EJB, log4j, JUnit, Cypress, Jenkins, RESTful, SQL.

**Client: Target, Minneapolis, MN Aug 2023 - Dec 2023**

**Sr. Java Full Stack Developer**

**Responsibilities:**

* Followed **Agile** software development practices including paired programming, **test-driven development**, and **Scrum** status meetings.
* Developed and implemented code for new requirements as a developer using **Spring MVC** and **Hibernate**, following the **Agile** methodology.
* Developed **APIs** to support **Angular** **UI** migration and **XSDs** to build web service data.
* Worked on front-end design and development using technologies like **HTML5**, **CSS**, **JavaScript**, **Angular**, and **Vue.js**.
* Utilized **Vuex** and **Vuetify** to manage complex application state and build reusable UI components.
* Optimized performance and accessibility of web applications using best practices in **Vue.js**, including virtual DOM and reactivity.
* Mentored junior developers and team members, sharing knowledge and best practices in **Vue.js** development.
* Integrated third party APIs into **Vue.js** applications, empowering data visualization and dynamic content display.
* Developed and deployed **Spring Boot** applications, leveraging its opinionated approach and embedded servers (e.g., Tomcat, Jetty, WebSphere) for simplified deployment and streamlined development workflows.
* Utilized **Spring Boot**'s auto-configuration and starter dependencies to rapidly bootstrap applications with minimal configuration, enhancing productivity and reducing boilerplate code.
* Took advantage of **Spring Boot**'s built-in support for externalized configuration, allowing seamless management of application properties across different environments using property files, environment variables.
* Developed the Controller and Business layers of the **MVC**, using **Spring**.
* Configured **XML** files for **Spring Security** injection to ensure back-end security by discriminating user roles and access authority for URIs and server-side assets.
* Used **Spring MVC** for implementing the web tier and developed web-based application service components. Configured beans using **Spring IoC**.
* Developed code to consume web services using **SOAP** and **REST**. Developed **RESTful** web services using **JAX-RS**.
* Identified and resolved memory issues by using **JVM** memory management tools.
* Implemented a content-based message router for outgoing **XML** web service requests, then processed the response messages to insert into an enterprise database.
* Used **Bindy**, **Stax**, and **JAXB** with **Camel** to work with **CSV** and **XML** files, and used **Groovy** to design and develop web-based applications.
* Generated comprehensive test reports using **BDD** frameworks, providing stakeholders with visibility into test results and facilitating effective communication.
* Integrated data across several applications using **JSON** documents with dynamic schemas and the **MongoDB** (**NoSQL**) database.
* Engineered scalable and high throughput backend systems using Java and **MongoDB**, handling millions of daily transactions.
* Contributions to open source projects related to **MongoDB** and Java development.
* Implemented data persistence, querying, and indexing strategies to optimize **MongoDB** database performance.
* Proficient in designing and implementing database schemas in **PostgreSQL**, including tables, views, and relationships.
* Implemented data validation and error handling mechanisms to maintain the integrity of the data throughout the **ETL** process.
* Architected and implemented **ETL** pipelines using **Airflow**, **SQL** for data extraction, transformation, and loading.
* Optimized **ETL** performance through code refactoring, parallel processing, and cloud-based resource scaling.
* Automated **ETL** processes using Airflow and Jenkins to schedule data ingestion and transformation, improving operational efficiency.
* Used **JIRA** to track project progress and wrote test cases. Used **JUnit** as a testing framework in a **Test-Driven Development** environment and **Selenium, Cypress** for Test Automation.
* Implemented messaging solutions using **IBM MQ**, leveraging its reliable and secure messaging capabilities for asynchronous communication between applications and services.
* Configured and managed **IBM MQ** queue managers, defining message queues, channels, and other MQ objects to facilitate efficient message routing and delivery.
* Developed applications integrating with **IBM MQ** using client libraries like **JMS** or **MQI** for reliable messaging.
* Monitored and administered **IBM MQ** environments using tools like IBM MQ Explorer for message flow.
* Used **Hibernate** in the data access layer to access and update information in the database.
* Led initiatives to improve overall code quality by organizing **SonarQube** training sessions for development teams, promoting best practices in code analysis and remediation.
* Collaborated with cross-functional teams to integrate **SonarQube** with other DevOps tools, such as Jenkins and Maven, enhancing the efficiency of the development and deployment processes.
* Implemented **Azure Functions** for serverless computing and event-driven architectures, enabling efficient and scalable execution of code without provisioning or managing servers.
* Utilized **Azure DevOps Pipelines** for continuous integration, continuous delivery, and automated deployment of applications to Azure environments.
* Engineered and deployed highly scalable applications on **Azure Container Instances**, optimizing resource utilization within budget constraints.
* Implemented **Role-Based Access Control (RBAC)** for secure access control and permission management within the Azure environment.
* Integrated **Jaeger** distributed tracing with Java micro services, allowing for end-to-end transaction monitoring and performance optimization across complex distributed systems.
* Developed custom **Grafana** plugins using Java to extend monitoring capabilities, tailoring dashboards to specific business needs and improving overall observability.
* Integrated **BDD** tests into the **CI/CD** pipeline using tools like **Jenkins** or **Bamboo**, enabling automated testing and early detection of issues.

**Environment:** Core Java, Vue.js, J2EE, JDK, Spring Boot, Spring MVC, Spring security, IBM MQ, Kafka, Hibernate, JSP, JavaScript, ETL, HTML5, CSS3, SQL, JUNIT, Cypress, Agile, RDBMS, Git, GitHub, Apache Tomcat, REST, RESTful, JVM, BDD, XML, Mongo-DB, JSON, SonarQube, Azure, Grafana, Jaeger, Jenkins, Chef.

**Client: Probus Technologies Pvt Ltd, Nagpur, India Jan 2020 – May 2023**

**Java Developer**

**Responsibilities:**

* Designed the front-end screens using **JSP**, **JSTL**, **Servlets**, HTML 5 and **JavaScript** and **jQuery**, **DOJO** struts tiles.
* Used **Spring Framework** extensively for dependency injection/**Inversion of control** (IOC), and integration of Controller, Service and **DAO layers**.
* Developed the presentation layer using **Spring MVC, Spring Core for Inversion of Control (IOC),** and application Context of **Spring Framework** in the project.
* Implemented entire application using **Core java, java Collections** and **Spring MVC** design framework.
* Designed and developed **Micro Services** business components using **Spring Boot**.
* Define the storage purpose of collections in **MongoDB**. Unstructured data was handled with the help of **MongoDB**.
* Used **log4j** for logging and **SVN** for version control. Developed **Maven and Gradle scripts** and developed builds for continuous integration servers like Jenkins for CDP pipelines.
* Implemented **SOAP Web Services** for other applications to communicate.
* Involved in consuming, producing **SOAP based Web services.**
* Implemented **AWS Cloud Watch** for comprehensive monitoring and logging of **AWS** resources.
* Enabled proactive detection and resolution of issues with **AWS Cloud Watch** alarms and **AWS Systems Manager**.
* Extensively integrated **AWS services** with enterprise and web applications.
* Leveraged **AWS Elastic Beanstalk** and **AWS Lambda** for application integration.
* Migrated the traditional on-premises environment to **AWS Cloud**.
* Utilized **Amazon EC2** instances, **Elastic Load Balancing (ELB)**, and **Amazon RDS** for **SQL Server** for migration.
* Utilized **Amazon S3** (Simple Storage Service) for secure and scalable object storage.
* Utilized **Amazon EFS** (Elastic File System) for secure and scalable file storage.
* Worked on mocking frameworks like using **Jasmine**.
* Development of internal dashboard applications using the **Apache HTTP Server**.
* Expertise in implementing applications using **Apache Kafka**.
* Used **Git, GitHub** as Version Control System for the application.

**Environment:** J2EE, Java 8, Core Java, JSP, JSTL, Servlets, HTML 5, JavaScript, jQuery, Ajax, Bootstrap, CSS, Git, GitHub, MongoDB, JAX - RS, log4j, SVN, Maven, Gradle, Jenkins, SOAP, AWS, Apache Kafka, Jasmine.

**Client: Ethnotech Solutions, India Feb 2017 – Dec 2019**

**Title: Java Developer**

**Responsibilities:**

* Familiar with designing and implementing Java/J2EE technologies, created design documents using Rational Rose.
* Used Spring framework with EJB 3.0 to create services which were able to handle millions of transactions.
* Implemented the MVC architecture using Apache Struts1.2 Framework and views using Struts tags and JSTL2.0.
* Developed Java applications using Core Java (Java SE 7), utilized JDBC for database interactions with MySQL 5.5.
* Created dynamic web applications with Servlets 3.0 and JSP 2.2 and managed project tasks, timelines using Asana.
* Utilized RabbitMQ for asynchronous messaging, MySQL for relational database management, and PostgreSQL for advanced data storage and retrieval in our software development projects.
* Implemented backend logic and business processes using Spring Framework 3.2, Spring Core and Spring MVC.
* Integrated Hibernate 4.2 for ORM (Object-Relational Mapping) to streamline database operations.
* Implemented RESTful web services using JAX-RS with Jersey and SOAP web services using JAX-WS.
* Assisted in the development of enterprise applications using EJB 3.1 technology and implemented transaction management using JTA and managed application components using JMX and used Visual Studio Code (VS Code).
* Implemented Jenkins continuous integration server to automate build, test, and deployment processes.
* Utilized JUnit 4.1 for unit testing to ensure code quality and utilized Git for version control.
* Configured and managed Oracle WebLogic Server for Java EE application deployment.
* Used Jira for project management and JMS for asynchronous communication between components.
* Implemented RMI interfaces for seamless communication and used Java Applets to enhance user interactivity.
* Collaborated with UI/UX designers to implement frontends using HTML5, CSS3, and JavaScript, leveraging libraries like jQuery 1.10.

**Environment:** J2EE, Rational Rose, Spring framework, EJB, Apache Struts, JSTL, Core Java (Java SE 7), JDBC, MySQL, Servlets, JSP, Asana, RabbitMQ, PostgreSQL, Spring Framework, Spring Core, Spring MVC, Hibernate, JAX-RS, Jersey, JAX-WS, JMS, EJB, JTA, JMX, Visual Studio (VS), Jenkins, JUnit, Git, RMI, Java Applets, HTML5, CSS3, JavaScript, jQuery.

**WebCoreStone Software Solutions, Pune - India Jul 2014 – Jan 2017**

**Java Developer**

**Responsibilities:**

* Collaborated closely with clients to gather and analyze requirements for new features, employing **Agile** methodologies and tools to foster effective communication and iterative development.
* Utilized **JIRA** and **Confluence** for efficient project documentation and task tracking, collaborating with clients and stakeholders to maintain transparency and effective communication throughout the project lifecycle.
* Actively participated in **Agile** ceremonies, including daily stand-ups, sprint planning, and retrospectives, to ensure seamless teamwork, identify potential roadblocks, and continuously improve processes.
* Contributed to the development of robust and scalable web applications utilizing **Java**, **JSP**, **Spring MVC**, and **Hibernate** frameworks.
* Leveraged the **Spring MVC** framework to establish a robust application architecture and streamline the development process, promoting code maintainability and reusability.
* Implemented the **Hibernate** Object-Relational Mapping **(ORM)** layer to facilitate efficient database interactions and simplify complex **SQL** queries, enhancing overall application performance.
* Assisted in database design and management processes, working with **MySQL**, **PostgreSQL**, and **Oracle** databases to ensure data integrity and reliability.
* Designed and optimized **MySQL** relational database schemas to align with application requirements, ensuring efficient data storage, retrieval, and processing.
* Developed comprehensive unit tests using **JUnit** and **TestNG** frameworks to ensure code quality, reliability, and adherence to predefined standards.
* Utilized **Selenium** for automated functional testing to ensure the quality and performance of applications.
* Ensured code coverage and maintained high code quality through **continuous integration (CI)** practices, leveraging tools like **Jenkins** for automated builds and testing.

**Environment:** J2EE, Java 8, Core Java, JSTL, Servlets, Spring MVC, Hibernate, MySQL, PostgreSQL, Oracle, JUnit,

TestNG, Jenkins, JIRA, Confluence, Agile Methodologies.