 

**Mokshith**

**Sr. Java Full Stack Developer**

**Email:** mokshith.javatechie@gmail.com

**Phone: 940-268-5989**

[](http://www.linkedin.com/in/mokshith-javatech)

**ABOUT ME:**

**Overall, 10+** *years of experience in Full Software Development Life Cycle (****SDLC****) which includes* ***Analysis****,* ***Design****,* ***Development****, Implementation, and Testing of web-based applications using* ***Java****/****JEE*** *technologies. Expertise in core* ***Java*** *concepts like* ***OOPs*** *concepts* ***Collections****,* ***Exception******Handling****, and* ***Multithreading****.*

**Java 8 | React | MySQL | Java Script| Node Js | Spring**

**PROFESSIONAL SUMMARY**

* Highly skilled and results-driven **Full Stack Java Developer** with **10+** years of expertise in **OOPs, Java/J2EE technologies, React, React, NodeJS,** and all phases of the **Software Development Life Cycle (SDLC**), including design, development, testing, and deployment of web-applications.
* Experienced in both **Waterfall** and **Agile methodologies**. Strong collaboration, Communication and Time Management skills, ensuring alignment between business requirements and technical implementations. Demonstrated ability to deliver **Efficient**, **Scalable**, and **Maintainable Code**.
* Experience in working with **MEAN Stack Development** using **Mongo DB**, **Express**, **Node,** and **React 4/7/8 Modules,** Services, Templates, directives, and **dependency injection to create a SPA** and worked with **React CLI**.
* Experience in using **React components**, Forms, Events, Keys, **Router**, **Redux architecture**, Flux concept & redesigning sites with everything from the contact page to the landing page.
* Hands-on experience with **Core Java** concepts such as OOP concepts, Collections, **Exception Handling**, Multi-Threading, and sound knowledge of **Data Structure and Algorithms**.
* Experienced in developing messaging modules by using **Spring JMS, Apache Camel, and Rabbit MQ.**
* Experience in **Spring framework** (**MVC/IOC/ORM/JDBC/AOP/BATCH/WEBFLOW/Context**) and **Struts** (MVC architecture, Interceptors, Action mappings, and resource bundles.). Good knowledge in **developing Micro Services using Spring boot**.
* Leveraged PostgreSQL’s full-text search capabilities to implement efficient search functionalities within applications, enhancing user experience.
* Managed user roles and permissions in PostgreSQL to enforce security policies and control access to sensitive data effectively.
* Integrated automated testing frameworks into CI/CD pipelines to ensure code quality and catch bugs early in the development process.
* Utilized tools like Git in CI/CD workflows to manage code changes, enabling seamless collaboration and version tracking among team members.
* Set up CI tools to automatically build and test applications upon code commits, ensuring immediate feedback on code quality.
* Proficiency in handling web and application servers in creating Domains, Configuration connection pools, and deploying applications on **Apache Tomcat, WebLogic**, WebSphere, and **JBoss**.
* I have extensive experience in **Core Java, J2EE, JSP, Servlets, JDBC, Spring, MuleSoft ESB**, **Maven**, **Hibernal Web Services,** and **SOAP**.
* Excellent Expertise in Amazon Web Services (**Amazon S3, Amazon EC2, Amazon SQS, Amazon Simple DB, Amazon Elastic Load Balancing, Amazon EBS, Amazon CloudFront, and Amazon Lambda**)**.**
* Implemented a server-less architecture using **API Gateway, Lambda, and DynamoDB and deployed AWS Lambda code**

from **S3 buckets**

* Created **Lambda deployment functions** and configured them to receive events from **S3 buckets**. Implemented security features Secure authentication with **LDAP, OAuth 2.0, D Security, SSL,** OAuth2.0 for creating tokens in the application using **Spring Security**, and JWT for providing tokens.
* Good experience in **J2EE Testing** in developing test cases with **JUnit**, **MOCKITO** for **Unit testing**, and logging using **Log4j** & Good Experience in **Mock Testing** Hands-on experience with build tools like **Ant**, **Maven**, **Gradle,** and Logging tools like **Log4J** and **Splunk**.
* Using **Jenkins** with **GitHub** to establish a **continuous integration (CI)** and **continuous deployment (CD) pipeline** using

**Docker container services** to deliver the applications.

* Maintained code repositories using various version control tools like **GIT**, and **Bitbucket**
* Adept at integrating Apache Kafka for real-time data streaming, enabling event-driven architectures and facilitating communication between microservices.
* Experienced in developing Kafka producers and consumers to publish and consume messages, ensuring reliable message delivery and processing.
* Familiar with using Kafka Connect to integrate various data sources and sinks, simplifying data ingestion and exporting processes.
* Implemented sharding in MongoDB to distribute data across multiple servers, enhancing scalability and ensuring high availability for large datasets.
* Developed aggregation pipelines using MongoDB’s aggregation framework to perform complex data transformations and analysis efficiently.
* Building and Deployment of **EAR**, **WAR**, and **JAR** files on test, stage and production servers, knowledge of functional usage and deployment applications in Apache Tomcat.
* Utilizing **Web** **Sphere** and **JBoss** to implement the **business** applications.
* **Education Details:**

I completed my bachelor’s from Karunya Institute of technology and Sciences from Coimbatore in the year 2015.

**TECHNICAL SKILLS**

|  |  |
| --- | --- |
| **Languages/Utilities** | Node, Java, C, C++, Python, PL/SQL, XML. |
| **JAVA Technologies** | Core Java, Multithreading, JDBC, JNDI, Java beans, J2EE, Junit, Maven,  Mockito, Ant, Log4j. |
| **Frame Works** | Spring, Spring Boot, Spring Batch Struts, Kafka, JSF and Hibernate, Sw. |
| **Web Services** | WSDL, SOAP & REST. |
| **Web Technologies** | Applets, JSP, HTML, CSS3, JavaScript, JSON, AJAX, jQuery, React, React, Typescript. |
| **Databases** | MYSQL, Oracle, SQL server, MongoDB, Couchbase, PostgreSQL. |
| **Middleware** | Servlets, Java Beans, RMI, EJB, JMS. |
| **IDEs and Tools** | Eclipse, NetBeans, Text pad, Maven, UML, Log4j, ANT. |
| **Version Control Tools** | Subversion, GitHub, CVS. |
| **Web/Application Servers** | Tomcat, IBM WebSphere, JBoss, Apache. |
| **Methodologies** | Waterfall, Agile. |
| **Operating Systems** | Windows 7/8/XP, Linux, UNIX. |
| **Cloud Technologies** | AWS Cloud Components S3, EC2, Lambda, Elastic Bean Stalk, ECS, RDS. |

**PREFESSIONAL EXPERIENCE**

**State of Utah, Salt Lake City, Utah Jan 2024 to Present**

**Java Full Stack Developer**

**Responsibilities:**

* Participated in Agile (Scrum) methodology, engaging in design, development, system testing, and user acceptance testing to deliver high-quality web applications.
* Utilized React and TypeScript to develop interactive web components and controllers, enhancing user experience and application performance.
* Leveraged Node.js for server-side development, utilizing its event-driven, non-blocking I/O model to build scalable, high-performance applications.
* Built RESTful APIs and web services using Express.js, providing a robust backend architecture for modern web applications.
* Utilized Amazon DynamoDB, a fully managed NoSQL database service, to handle high-traffic workloads with fast, predictable performance.
* Implemented TypeScript to introduce static typing to JavaScript, allowing for early error detection and improved code quality during development.
* Employed AWS Lambda, a serverless compute service, to run code in response to events, automatically scaling to manage varying workloads.
* Handled asynchronous operations in Node.js effectively using promises and async/await, enhancing code readability and performance.
* Developed React components with routing capabilities for dynamic navigation, including features like bookmarking, filters, and validations.
* Achieved performance optimization through React Lazy Loading, improving application load times and user engagement.
* Integrated Node.js with various frontend frameworks like React to create full-stack JavaScript applications, ensuring seamless data exchange.
* Configured CI/CD pipelines to automate deployment processes using Jenkins and Docker, significantly reducing time to market for new features.
* Utilized Docker to create consistent development and testing environments, ensuring reliable application performance across different stages.
* Implemented infrastructure as code using tools like Terraform in CI/CD processes to manage provisioning and configuration, enabling reproducible environments.
* Designed normalized database schemas in PostgreSQL to maintain data integrity while using advanced features like foreign keys and constraints.
* Optimized complex SQL queries using window functions and common table expressions (CTEs) for enhanced analytical capabilities and reporting.
* Leveraged AWS S3 for scalable object storage, ensuring efficient data retrieval and management for applications.
* Implemented caching solutions with Redis to improve application performance and scalability, reducing server load.
* Developed Single Sign-On solutions to enable seamless authentication across multiple applications, enhancing user experience and security.
* Visualized RESTful APIs using Swagger/OpenAPI specifications for automated documentation, testing, and client generation.
* Integrated Node.js applications with various databases—including MySQL, PostgreSQL, and MongoDB—using ORMs like Mongoose for efficient data management.
* Utilized AWS CloudWatch to monitor and track metrics, logs, and events, ensuring optimal application performance and availability.
* Executed auto-scaling with AWS to dynamically adjust resources based on application demand, ensuring consistent performance during peak loads.
* Implemented JWT for secure user authentication and authorization in RESTful web services, ensuring robust security measures.
* Employed WebSockets and Socket.IO for real-time, bidirectional communication between clients and servers, enhancing user interaction.
* Utilized AWS Elastic Beanstalk for simplified application deployment and management, streamlining the deployment of web applications.
* Integrated performance monitoring and logging solutions to ensure high availability and reliability of applications in production.
* Applied React Router for efficient navigation in single-page applications, enhancing user experience with smooth transitions between views.
* Developed robust token management strategies, including secure storage and expiration mechanisms for access and refresh tokens, enhancing security.
* Used Jenkins with GitHub to establish a continuous integration and continuous deployment (CI/CD) pipeline, automating the delivery of applications through Docker container services.

**Environment:** React, Node JS, HTML5, CSS3-SASS, JSTL, Twitter Bootstrap4.0, TypeScript, ES6 JavaScript, EKS, HTTP Service, TypeScript, Spring boot, JSP, Struts, DevOps, DynamoDB, AWS, EC2, ECS, S3, CloudWatch, SQS, SNS, AWS Lambda, OAuth, Elastic Beanstalk, IAM, SSO, Elastic Search, Redis, NPM, Web pack, RESTful, EC2, JSTL, React, JavaScript, TypeScript.

**Simmons Bank, Atlanta, GA July 2021 to Dec 2023**

**Java Full Stack Developer**

**Responsibilities:**

* Designed and developed RESTful web services using Spring MVC and Spring Boot, exposing multiple endpoints with Hibernate integration for efficient data handling.
* Built the presentation layer for application modules using React, HTML5, and CSS, delivering responsive and engaging user interfaces.
* Implemented microservices-oriented architecture by integrating Spring Boot services with a React front end, enhancing modularity and scalability.
* Utilized TypeScript to provide static typing in JavaScript, improving collaboration in large codebases through interfaces and type annotations.
* Leveraged Amazon CloudFront as a CDN to deliver content with low latency and high transfer speeds, enhancing user experience.
* Integrated third-party APIs and libraries to enhance functionality, streamlining development processes and reducing implementation time.
* Managed OAuth scopes to control access levels for third-party applications, ensuring secure permissions for users.
* Optimized query performance in MongoDB using indexing strategies, including compound and text indexes, to enhance application responsiveness.
* Integrated MongoDB with Node.js applications using Mongoose for effective data modeling and validation, improving development efficiency.
* Conducted data migrations from SQL databases to MongoDB, ensuring data integrity and alignment with NoSQL storage requirements.
* Applied security best practices in OAuth implementations, including HTTPS usage, validating redirect URIs, and implementing PKCE for enhanced security.
* Built scalable and fault-tolerant Node.js applications by utilizing modern concurrency models like the event loop and worker threads.
* Created comprehensive REST API documentation using Swagger and utilized Docker for containerization, improving deployment efficiency.
* Developed Web APIs using Node.js to facilitate seamless data exchange between the client and server.
* Employed asynchronous programming patterns in Node.js, utilizing callbacks, promises, and async/await for efficient I/O operations.
* Automated project documentation generation using tools like Sphinx or MkDocs within the CI/CD pipeline to ensure up-to-date information.
* Implemented Python scripts to automate repetitive CI/CD tasks, such as environment setup and deployment processes.
* Developed Python applications that interact with CI/CD tools' APIs (e.g., Jenkins REST API) for programmatic management of builds and deployments.
* Utilized IAM for auditing, logging, and monitoring user activities and access events to ensure compliance and security.
* Focused on DevOps practices while developing RESTful web services and APIs using Spring Boot and Hibernate.
* Employed RxJS for reactive programming and asynchronous operations, while utilizing NgRx for state management in React applications.
* Utilized Express.js as a web application framework for Node.js, providing flexibility and minimalistic features for application development.
* Managed stored procedures and functions in PL/pgSQL to encapsulate business logic within PostgreSQL, improving performance and maintainability.
* Configured PostgreSQL for high availability using streaming replication and failover mechanisms, ensuring minimal downtime.
* Conducted performance tuning through indexing strategies and query optimization to enhance database responsiveness.
* Implemented fine-grained access control in DynamoDB with AWS IAM integration to secure data access.
* Utilized Amazon CloudFront to cache content at edge locations, improving load times and user experience.
* Enabled real-time communication using WebSockets and Socket.IO for applications requiring live updates, such as chat functionalities.
* Integrated Node.js with various databases, including MySQL and PostgreSQL, using ORMs like Sequelize and Mongoose for efficient data management.
* Achieved horizontal scalability by deploying multiple instances behind a load balancer, leveraging Node.js’s event-driven architecture.
* Structured logging in applications using key-value pairs and JSON format for efficient search and analysis with tools like Splunk.
* Utilized TypeScript to enhance maintainability in applications, allowing gradual integration into existing JavaScript projects.
* Configured DynamoDB with primary keys and secondary indexes for efficient data retrieval and management.
* Implemented lazy loading and eager loading strategies in Hibernate to optimize data fetching from the database.
* Optimized SQL queries for performance and scalability as part of DevOps practices.
* Developed complex AWS Lambda functions for asynchronous processing and integration with other AWS services.
* Fine-tuned API Gateway settings for optimal performance, enhancing request/response handling.
* Implemented caching strategies in Node.js applications using Redis or Memcached to enhance performance and responsiveness.
* Managed domain names and traffic routing using AWS Route 53, ensuring reliable access to resources.
* Created reusable modules and components in Node.js to extend core functionalities, promoting code reusability.
* Bundled React frontend code with build tools like Webpack or Create React App for efficient deployment.
* Implemented service meshes using Istio for advanced traffic management and security in microservices architectures.
* Utilized Quarkus for streamlined development tasks and integrations with technologies like Hibernate, RESTEasy, and Kafka.
* Gained expertise in version control systems like Git for collaborative development on both Windows and web application projects.
* Utilized JSP technology for developing interactive and data-driven web applications following the MVC architecture.
* Handled transactions with MySQL databases using JDBC in Java applications, ensuring data integrity and atomic operations.
* Developed REST services to expose processed data, enabling integration with various front-end applications.
* Implemented Spring Security and transaction management in applications for enhanced security and data consistency.
* Integrated multiple Spring Framework components, including Spring Core, AOP, MVC, and Security, into microservices architectures.
* Configured Jenkins for continuous integration, automating build and deployment processes for web applications.
* Utilized AWS services like Lambda, S3, and DynamoDB, creating integrations for critical application functionalities.
* Developed functions and roles in AWS Lambda to automate script execution and application processes.
* Secured Node.js applications by implementing robust authentication and authorization mechanisms, managing API keys, and integrating with identity providers.
* Gained proficiency in using Cassandra for scalable data storage, optimizing data models for application performance.
* Executed CRUD operations efficiently in Cassandra using the Java Driver and managed schema changes with CQL.
* Configured replication and partitioning strategies in Cassandra to ensure data consistency across distributed clusters.
* Collaborated with teams to configure backend state management, utilizing remote backends like Amazon S3 and DynamoDB for Terraform state files.
* Utilized NoSQL databases to handle large volumes of semi-structured data, optimizing for performance and scalability.
* Managed infrastructure as code using DevOps tools like Terraform and Ansible to streamline application deployments.
* Automated deployment processes using Docker and Kubernetes as part of DevOps best practices.
* Utilized Java and AWS Lambda for event-driven processing with S3 and SQS, enhancing application responsiveness.
* Troubleshot technical issues in both Windows forms and web applications to maintain optimal performance and user satisfaction.
* Deployed and managed Node.js applications in cloud environments such as AWS and Azure, using Docker and Kubernetes for container orchestration.
* Developed and tested REST APIs with JSON, utilizing Postman for testing and documentation of web services.
* Wrote comprehensive JUnit test cases to validate application functionality and ensure code reliability.
* These revised points should effectively highlight your relevant experience in Node.js, React, AWS, and CI/CD, aligning well with the requirements of the position.

**Environment:** Java, Spring Boot, REST, SQL, HTML5, CSS, JavaScript, jQuery, AJAX, JSON, React, Eclipse, JBoss, Maven, AWS, DB2, Micro Services, Postman, Spark, DynamoDB, Terraform, NodeJS, Couchbase UML, PostgreSQL,kafka.

**Lowe’s, Madison, WI Jan 2020 to July 2021 Java Full Stack Developer**

**Responsibilities:**

* Developed and Implemented **Micro Services using Java/J2EE**, **Spring Boot**, and **Spring Framework**.
* Used **JSF** & **RFID** framework in developing a user interface using React Components, Validators, Events, and Listeners for handling request processing.
* Handled the security and Authorization by **Spring Security** and **OAuth authentication Protocol**.
* Involved in writing the test cases for the application using **JUnit**.
* Developed RESTful APIs using Express.js, facilitating seamless communication between frontend applications and backend services.
* Utilized middleware in Express.js to handle requests and responses, enhancing application functionality through logging, authentication, and error handling.
* Implemented routing in Express.js to manage complex URL structures, enabling clean and organized code for multiple endpoints..
* Created custom plugins or scripts in Python to extend the functionality of CI/CD tools, tailoring the pipeline to specific project needs.
* Used Python-based tools like Ansible for configuration management, automating the setup of servers and application environments in the CI/CD pipeline.
* **AWS Amazon Elastic Container Service (ECS)** is a fully managed container orchestration service that allows you to run Docker containers on a scalable infrastructure.
* Deployed **Spring Boot** applications with microservice architecture using **REST** and **Apache Kafka**.
* Skilled in integrating Elasticsearch with applications to provide real-time search functionality, utilizing its RESTful API for data indexing and querying.
* Experienced in managing and optimizing Elasticsearch clusters, including setting up node configurations, shard management, and monitoring performance.
* **BPMN platforms** typically expose REST APIs for interacting with process instances, tasks, and variables programmatically.
* Implemented data replication using MongoDB’s replica sets to enhance fault tolerance and provide high availability for mission-critical applications.
* Integrated Express.js with MongoDB using Mongoose for efficient data modeling and validation, streamlining database interactions.
* Configured CORS (Cross-Origin Resource Sharing) in Express.js applications to manage cross-domain requests and enhance security.
* Leveraged Express.js to serve static files, optimizing performance and user experience for web applications.
* Used Kafka for actively tracking real-time data processing, messaging, and log aggregation.
* worked on creating **serverless Microservices** by integrating **AWS Lambda**, **S3**, **cloud** **watch**, and **API gateway**.
* Included security groups, **network ACLs, Internet Gateways**, and **Elastic IPs** to ensure a safe area for the organization in **AWS private cloud**.
* Message broker services are achieved in the application using Active MQ with the JMS framework.
* Implemented **Spring Dependency Injection (IOC)** feature of spring framework to inject beans into the User Interface and **AOP** for **Logging**.
* **AWS ECS** simplifies the process of deploying, managing, and scaling containerized applications using the **AWS infrastructure.**
* Developed SPA project using **React TypeScript for frontend, NodeJS**.
* Ability to collaborate effectively with cross-functional teams like **web applications** to **analyze requirements**, design solutions, and deliver successful projects.
* Used **Spring Security framework** for login authentication, and password hashing. Worked on **Java Message Service (JMS) API** for developing message-oriented middleware (MOM) layer for handling various asynchronous requests.
* **XStore’s** POS terminals can be deployed within warehouse facilities to facilitate sales transactions, order processing, and customer interactions.
* Creating custom, general use modules and components which extend the elements and modules of core **Node.JS.**
* Implemented **Quartz scheduler** to execute batch programming using **Spring Batch**.
* Configured Jenkins to handle application deployment in the **Pivotal Cloud Foundry (PCF) server** and to integrate with **Git Hub version control**.
* Build a **CI/CD pipeline** using **Jenkins automation** and **GitHub repositories** using the webhooks plugin.
* **JIRA** tool for project management used in the application to track bugs, issues, and project management functions
* Set up **Jenkins server** and build jobs to provide automated builds based on polling the **GIT source control system**.

**Environment:** Java, J2EE, Spring Boot, Express JS, Microservices, JSF Framework, Spring Security, React, NodeJS, Kafka, JavaScript, HTML5, CSS and Bootstrap, XStore, REST, RESTful, NoSQL Mongo DB, Vert.x, JUnit, Jenkins, Log4j, JIRA, Git, BPMN Amazon Web Services, JPA Framework, Spring Rest, Spring MVC, Spring Security, Hibernate, IAM**.**

**Scene Health, Brook landfilled**, **MD Mar 2019 to Dec 2019**

**Full Stack Developer**

**Responsibilities:**

* Involved in System Analysis and Design methodology as well as Object-Oriented Design and development using OOAD methodology to capture and model business requirements.
* Responsible for Object-Oriented Design using Enterprise Architect.
* Review and guide the application architects with Object-Oriented Design using Rational Unified Process (RUP) using Enterprise Architect.
* Expertise in version control systems like **Git** for collaborative development and code management in both **Windows forms and web application projects**.
* **Access Management (IAM)** in Java projects involves integrating **authentication, authorization, and other security-related functionalities**.
* Worked on developing applications, automated scripts leveraging MongoDB.
* Used **Spring Boot** for building applications with spring initializers and **bootstrapping** the build configuration and develop custom solutions that act upon **Jenkin's** information in authentic time.
* Recommend and develop a strategy for integrating the internal security model into new projects with **Spring** **Security and Spring Boot**.
* Implemented **React 8** Router to enable navigation from one view to the next as the customer performs application tasks. Experience with **React.js** workflows **Flux, Redux**.
* I used **AWS PySpark** and **AWS Glue**, specializing in architecting and implementing robust data processing solutions.
* Developing User Interaction screens using **HTML5, CSS3, JavaScript, JQuery, Bootstrap, AJAX, React.js, React.js** and JSON. Enhanced application performance by React 6 Component-based development because of future React 8 framework transitions.
* Extensively developed User-defined Custom JSP tags to separate presentation from application logic.
* Designed and developed **Web Services (SOAP, WSDL**). Created and compiled **XML Schema** to generate Java Bean classes using Axis.
* Created end to end integration tests with Cypress and unit tests with Jest.
* Developed user interface by using the **ReactJS for SPA development**.
* Implemented the business layer for MongoDB Service.
* Defined AWS Security groups which acted as virtual firewalls that controlled the traffic allowed to reach one or more EC2.
* Worked on Data Migration using **ETL** tools, we used **TOAD (Tool for Oracle Application Developers)** and intermediate proficiency in Oracle Enterprise Manager **(OEM)**.
* Used React flush to polish the data for single directional flow.
* Installation, Configuration & administration of **WebLogic application Server**.
* Involved in writing SQL, Stored procedure, and **PLSQL** for the **back end**. Used Views and Functions at the **Oracle Database** end. Developed the **PL/SQL** scripts for rebuilding the application Oracle Database.
* Developed **HTML and JavaScript.** Developed Scripts for **UNIX platform** deployment.
* Used the Mozilla Firefox extension, Firebug, to view and debug **HTML, DOM, and JavaScript**.
* Project coordination with other **Development teams, System managers**, and webmaster and developed a good working environment.

**Environment:** Java EE 5,7, 8 , JSP, Servlets, API, JSF, Spring DI/IOC, Hibernate, Groovy, XML, HTML, JDBC, Web services, Rational Software Architect, Web Services(SOAP/WSDL and REST), React.js, Node, Jasper, ORM, Web Logic Application Server, Junit, Maven, Log4J, SVN, Oracle 10g, JavaScript, CSS, UNIX.

**Farmers Inc Group, California Jan 2018 to Dec 2018**

**Full Stack Developer**

**Responsibilities:**

* Used **Spring framework MVC** in the development of new modules based on requirements.
* Utilized various utilities like JSP, JSTL, JavaScript, HTML, & CSS, and jQuery.
* Refactored existing project to make it more **RESTful** and **thread safe**.
* Involved in the creation of dynamic web pages with the use of JSP and HTML.
* **Conﬁgured and set up Java Workspace** which included setting up server and theme installation and configuration. Became familiar with using and navigating through Java Workspace (Eclipse).
* **Debugged Java classes** used to modify web pages such as Business Objects and Data Access Objects.
* Implemented JDBC to perform the database calls from the Java layer.
* Used JDBC to persist Java Objects into the database.
* For **XML marshaling** and **unmarshalling used JAXB**. Worked with databases like **Oracle**.
* Tested many components of web applications and documented my observations.
* Used Servlets as an interface between the front and backend to handle **HTTP requests** and send the response back to the front end.
* Participated in analyzing the requirements and depicted them through use cases and scenarios.
* Participated in **unit testing**, **integration testing,** and installation of the system.
* Contributed to the decision-making of the software and hardware requirements and tool evaluations.
* Implemented logging for the whole application using **Log4j**.
* Involved in daily scrum meetings and worked on the sprint tasks in the **Agile Scrum development**.

**Environment:** Java, J2EE, Spring, Core Java, SOAP/Rest, Web Services, JavaScript, HTML5, CSS, SASS, JIRA, Linux, Selenium NOSQL(MongoDB), React JS, AJAX, Apache JMeter.

**Albertsons, SFO, CA July 2015 to Dec 2017**

**Associate Software Engineer**

**Responsibilities:**

* Developed UI screens using **JSP, HTML, CSS, JavaScript**.
* Worked on **JSP, Servlets, Struts framework**and production support issues of the existing applications.
* Development of **Action Forms, Action Servlets, Action, and Validating Action Forms in Struts framework.**
* Implemented **Struts Dispatch Action class** and form bean classes using **struts framework.**
* Worked on the JAVA **Collections API** for handling the data objects between the business layers and the front end.
* Implemented **Multithreading** for handling multiple requests and for high performance.
* Build and maintain **SQL scripts, indexes, and complex queries for data analysis and extraction.**
* Develop a business continuity plan for the SQL Server Databases using **JDBC drivers**.
* Created an XML configuration file for **Hibernate** to map to SQL DB.
* Developed **web services** for sending and getting data from different applications.
* Used **JDBC** to access **Oracle** database for accessing customer information.

**Environment:** Core Java, HTML, CSS, J2EE, JSP, HTML, JavaScript, Servlets, JMS, Hibernate, JDBC, SQL, DAO, Web Services, Oracle, ANT.