 

**Mokshith**

**Sr. Java Full Stack Developer**

**Email:** pavan.sigarevu@adventglobal.com

**Phone: 281-801-0073**

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

*Over* ***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****,* ***Multithreading****.*

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

**PROFESSIONAL SUMMARY**

* Around**10+ years**of**full stack development**experience in all stages of life cycle, referring to **Design, Development, Implementation,** and **testing** of **web-based applications**.
* Experience in Developing **User Interface (UI) Rich Web Applications** using Front End/User Interface (UI) Technologies using **HTML5, DHTML, CSS3 (Saas, LESS) frameworks, Bootstrap, Java script (ES5 & ES6), jQuery, AJAX** by managing all facets and pipelines of application.
* Good Understanding on using various **XML technologies** like **XPath, XQuery, XSLT, XSD, SAX, DTD/Schemas, DOM and JAXB.**
* Extensive experience in development and implementation of **ORM framework Hibernate** with **Spring Data Access.**
* Designed multi-threaded applications for real-time processing scenarios, such as data streaming and event handling, to meet strict performance requirements.
* Developed background tasks that run in separate threads, allowing the main application to remain responsive while performing time-consuming operations.
* Have good experience and exposure on most of the J2EE technologies like Spring3.0, Restful Web service, Hibernate, EJB, JMS, JDBC, Servlet, JSF, JSP, XML, XSLT, Java Script.
* Worked with **React.js Virtual DOM**and**React views**,java rendering using components.
* Implemented various screens for the front end using **React.js** and various predefined components from **NPM (Node Package Manager), bower and redux library for Dependency management.**
* Strong Experience in **JavaScript MVC** design pattern such as **Angular JS, Backbone JS and Ember JS, Knockout.js.**
* For UCM 10g we used Apache servers as admin servers and for WCC 11g weblogic 10.3.6.
* Provided end user support for all levels of Oracle ECM/WCC users.
* Developed unit and integration tests for existing **micro services** using **JUnit, Mockito** and **Docker** environments
* Having good Experience in **MVVM** design pattern such as **Angular 2.X and Angular 4.X.**
* Developed multiple **Angular components** which are compatible with latest versions of **typescript, Angular CLI.**
* Experience setting up Kafka cluster in AWS using MSK.
* Handled operations and maintenance support for AWS cloud resources which includes launching, maintaining, and troubleshooting **EC2** instances, **AWS IAM**, Elastic Load Balancers (**ELB**) and Relational Database Services (RDS).
* Developed and optimized SQL queries for performance in Oracle, MSSQL, and MySQL databases, ensuring efficient data retrieval and manipulation for various applications.
* Designed and implemented database schemas in relational databases, ensuring normalization and integrity constraints to maintain data consistency.
* Integrated the different pages to be a **single page** application using **Angular**  and **Typescript**.
* Created **AWS Security Groups**for deploying and configuring AWS EC2 instances.
* Defining and deploying monitoring metrics, and logging systems on AWS environment. Application and system log analysis using Cloudwatch, CloudTrail, Data Dog, Logic Monitor.
* Strong experience in developing **Single Page Application (SPA), using MEAN (Mongo, Express, Angular and Node) stack.**
* Implemented Kubernetes services and ingress controllers to manage external access and ensure seamless communication between microservices and users.
* Integrated Docker and Kubernetes into CI/CD pipelines using Jenkins, automating the build, test, and deployment processes for containerized applications, leading to continuous and reliable delivery.
* Used Kubernetes for efficient resource management, fine-tuning CPU and memory requests and limits to optimize application performance and scaling.
* Good understanding **Core Java SE 8** concepts like **Streams API, Time API, Functional Interfaces, Multithreading, Transaction Management, Exception Handling, Lambda functions, Generics and Collection API.**
* Good knowledge of working with **Relational database ORM frameworks like PostgreSQL, MySQL, DB2, Oracle 13c/11g, Ibatis and NoSQL DB (Mongo DB and Cassandra, Couch DB, Redis, Dynamo DB).**
* Experience with **Spring** **Kafka**, **ActiveMQ**, **RabbitMQ** and **zookeeper** apart from **JMS** as messaging services and message brokers.
* **Spring** **Kafka** template to process the messages smoothly on **Kafka** Cluster setup.
* Good knowledge of the **Java** based logging framework **Log4J** and **Java** based build automation tools **Apache** **Maven** and **Ant** Script

**TECHNICAL SKILLS**

|  |  |
| --- | --- |
| **Languages/Utilities** | 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. |
| **Web Services** | WSDL, SOAP & REST. |
| **Web Technologies** | Applets, JSP, HTML, CSS3, JavaScript, JSON, AJAX, jQuery, Angular, React. |
| **Databases** | MYSQL, Oracle, SQL server, MongoDB, Couchbase. |
| **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. |

**PROFESSIONAL EXPERIENCE**

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

**Java Full Stack Developer**

**Project:** Office of Vital records

As a Java Full Stack Developer at the Office of Vital Records in the State of Utah, I manage and streamline processes related to death records and certificate issuance. My role encompasses the development and maintenance of the death portal, overseeing systems like EDEN (Electronic Death Entry) and UMED (Medical Examiner). I collaborate with various roles, including funeral directors, medical clerks, medical certifiers, local health departments (LHD), field investigators, and state administrators. My work ensures the seamless handling of death records from initial field investigation to final certificate issuance, integrating inputs from funeral homes, medical certifiers, pathologists, and other stakeholders

**Responsibilities:**

* Developed and maintained microservices using **Java Spring Boot**, ensuring modular, scalable, and efficient processing of death records.
* Implemented **Spring Security** for robust **authentication** and **authorization** mechanisms, ensuring secure access to the death portal and protecting sensitive data.
* Utilized **React** with **TypeScript** to develop dynamic, responsive, and user-friendly **single-page applications (SPA)** for various user roles, enhancing user experience and performance.
* Used **React’s lazy** loading and **React.lazy** to split code and load components only when needed, improving initial load times and performance.
* Developed applications that utilize multi-threading to perform multiple tasks simultaneously, improving responsiveness and resource utilization.
* Implemented synchronization mechanisms (e.g., mutexes, semaphores) to prevent race conditions and ensure data integrity when multiple threads access shared resources.
* Utilized thread pools to manage a fixed number of threads for executing tasks, reducing the overhead of creating and destroying threads for each operation.
* Developed ETL (Extract, Transform, Load) processes to integrate data from various sources into SQL and NoSQL databases, enhancing data accessibility for analysis.
* Provided training and support to team members on SQL and NoSQL best practices, fostering a deeper understanding of data management principles.
* Participated in design discussions to choose the appropriate database technology based on application requirements, workload characteristics, and scalability needs.
* Managed and optimized **MariaDB** **databases** for efficient storage, retrieval, and management of records, ensuring data integrity and high performance.
* Integrated **Bootstrap 4.0** into the death portal application to ensure a consistent and responsive user interface for various user roles, improving overall user experience and accessibility on different devices.
* Set up and maintained **CI/CD** pipelines using **Jenkins**, **automating** the build, test, and deployment processes to ensure seamless and continuous integration and delivery.
* Leveraged **AWS** services such as **EC2**, **S3, RDS, IAM, ECS,** and **Elastic** **Beanstalk** for scalable, secure, and efficient **cloud** infrastructure, facilitating reliable application deployment and operations.
* Documented and tested **APIs** using **Swagger**, providing clear, comprehensive, and interactive API documentation for seamless integration and communication between system components.
* Conducted automated end-to-end testing using **Cypress**, ensuring the reliability, performance, and quality of web applications.
* Managed database security and user access controls in SQL databases, implementing best practices for data protection and compliance.
* Designed and implemented robust CI/CD pipelines using tools like Jenkins and GitLab CI, automating the build, test, and deployment processes, which reduced deployment times by over 50%.
* Integrated automated testing into the CI/CD workflow, utilizing frameworks such as JUnit and Selenium, to ensure comprehensive test coverage and maintain high code quality before production releases.
* Participated in **Agile (Scrum) methodology**, actively involved in sprint planning, daily stand-ups, development, system testing, and user acceptance testing to ensure timely and high-quality delivery of features.
* Designed and implemented modern **microservices** **architectures** on **AWS**, using services such as **ECS**, **ECR, ELB, MSK**, **Auto Scaling Groups**, and **EC2** to build scalable and resilient systems.
* **Optimized** **application** **performance** by implementing efficient coding practices and performance tuning.
* Developed and integrated **RESTful** **APIs** with **Spring Boot**, ensuring seamless communication between **microservices** and **front-end applications.**
* Configured and maintained **Jenkins** for continuous integration and **deployment**, ensuring automated, reliable, and consistent application delivery.
* Implemented caching solutions using **Elastic** **Search** to enhance application performance and ensure quick **data retrieval**.
* Developed data access layers using **Hibernate** and **Spring Framework** for efficient and **robust** data persistence and **backend** processing.
* Worked with **JMS** and **Spring Boot**, providing an instance embedded with Kafka for efficient message processing and integration.
* Implemented **OAuth 2.0** for secure and authorized access to **RESTful web services**, enhancing **API security**.
* Integrated monitoring and logging using tools such as **Spring Boot Actuator** and **Logback**, providing visibility into application performance and simplifying **troubleshooting**.
* Used **dependencies** like **Spring** **Cloud** and **Spring** **Data** **JPA** to enhance the scalability and data management capabilities of **microservices**.
* Regularly updated **dependencies** and **Spring Boot** versions to leverage new features, security patches, and performance **improvements**, ensuring the application remains **up-to-date** and optimized.
* Used **Kafka** to **decouple microservices** and facilitate asynchronous processing, improving system responsiveness and scalability.
* Developed and maintained **Dockized** microservices using **Java** **Spring Boot**, **ensuring modular**, **scalable**, and **efficient** processing of death records within containerized environments.
* Implemented Docker containerization for all microservices, **creating Docker files** and managing **images** for consistent deployment across development, testing, and production stages.
* Configured and managed Kubernetes clusters to **orchestrate** **containerized** **microservices**, enabling automated scaling, **load** **balancing**, and **high** **availability**.

**Environment:** React, Node JS, HTML5, CSS3-SASS, Twitter Bootstrap4.0, ES6 JavaScript, AJAX HTTP Service, TypeScript, Spring boot, JPA, jQuery, Apache Kafka, Jenkins, Log4j, Bootstrap, MY SQL, MariaDB, Elastic search, Redis, NPM, Web pack, RESTful, EC2, Swagger, Java-17, Spring Boot 3.2.5, Kafka.

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

**Java Full Stack Developer**

**Project:** Network Optimization System

**Responsibilities:**

* Participated in **Agile (Scrum) methodology** and Involved in **Design, development, System Testing and User Acceptance Testing.**
* The application is based on **Micro Service Oriented Architecture** using **Angular**and **Cassandra DB and MySQL**to develop**Single Page Applications (SPA).**
* Implemented **Angular directives, expressions, routing capabilities** for **bookmarking**, **filters, validations** and **Angular UI components, Components, ng-Modules, Providers, Services,** and **classes** for **navigation** respectively.
* Worked with **Angular CLI** and **Typescript** in **Angular** to create **Controllers** and **Angular components** for the Web Pages.
* Extensively used **Angular UI** (**Angular Bootstrap**) for **ng-grid**, and **UI components** and other directives such as **ng-repeat** and **ng-show**/**ng-hide.**
* Established monitoring and alerting mechanisms using tools like Prometheus and Grafana to track the health of applications post-deployment, enabling proactive issue detection and faster response times.
* Collaborated with cross-functional teams to refine CI/CD processes, introducing best practices for version control, branching strategies, and release management, which improved team efficiency and reduced deployment-related incidents.
* Conducted performance profiling to identify bottlenecks in multi-threaded applications, refining algorithms and thread management strategies for optimal performance.
* Implemented **Angular directives, expressions, routing capabilities** for **bookmarking**, **filters, validations** and **Angular UI components, Components, ng-Modules, Providers, Services,** and **classes** for **navigation** respectively.
* **Optimized performance** is achieved by **Angular Lazy Loading**.
* Used **Spring Boot** which is radically faster in building cloud **Micro Services** and develop spring based application with very less configuration.
* Implemented **Bootstrap 4.0** alongside Angular to design and develop a responsive Single Page Application (SPA) for the Network Optimization System, ensuring smooth user interactions and consistent styling throughout the application.
* Integrated NoSQL databases like Elasticsearch for full-text search capabilities, enabling rapid and flexible search functionalities across large datasets.
* Designed and implemented data models in Elasticsearch, optimizing indexing strategies to enhance search performance and reduce latency.
* Leveraged Elasticsearch's aggregation features to perform complex analytics and reporting on large volumes of unstructured data.
* Integrated **Bootstrap's UI** components such as navigation bars, buttons, and alerts within **Angular** applications to streamline development and enhance the visual appeal and functionality of the user interface.
* Used **Bootstrap’s** built-in styles and components to reduce custom **CSS** code and improve page load times, contributing to faster and more efficient application performance.
* Designing and implementing **modern micro services architectures on cloud** infrastructures using AWS Services such as ECS, ECR, ELB, AWS Elastic Search, MSK, Auto scaling Groups, EC2, etc.
* Used **Spring MVC** components to implement business layer and **Spring Boot** for **Micro-services**
* Implemented **Elastic Search** for caching to enhance the performance of application.
* Implemented **Node JS** as a server-side proxy for observing various response time.
* Implemented modules into **Node JS** to **integrate** with **designs and requirements**.
* Used **Node Package Manager**  to manage modules & used it to install useful tools such as **Grunt and Express**.
* Built and deployed all required service using **AWS Python CDK** for source control.
* Deployed frontend application using **AWS S3**and**configured CORS**to consume **RESTful services** from another domain.
* Expertise in integrating Terraform with **Ansible, Packer** to create and Version the AWS**Infrastructure**, designing, automating, implementing and sustainment of Amazon machine images (**AMI**) across the AWS Cloud environment.
* Used **Express JS** which is fast, assertive, essential, and moderate web framework of **Node.js** that helps manage a **server** and **routes**.
* Worked with **Express JS** which gives **MVC Structure** to the **Node JS.**
* Built-in **AJAX HTTP Service** with observables is used extensively for making quick and **incremental updates to UI** of the application.
* Developed Data Access Object (DAO) persistence layer using Hibernate 3.0.
* Applied to the **business modules integration** using **Spring Boot**, **Cassandra DB**.
* Worked with **Cassandra Node Connector** for database persistence to interact with Cassandra.
* Worked in integrating **JMS** with **SPRING BOOT** by providing an instance which is embedded with **Active MQ**.
* Deployed **Spring Boot based micro services**on **Docker containers**using **Amazon ECS.**
* Experience in introducing **Docker** utilizing Docker tool compartment. Condition provisioning arrangements **utilizing Docker**.
* Authenticating Users and to provide security to **Restful web services**using **OAUTH 2.0.**
* Using **Swagger Framework**, actualizing the **Restful API** and **Swagger** gives the rich UI to test API.
* Worked on unit test cases using **Karma** and **Jasmine** on client side.
* Wrote comprehensive Unit test cases using **JUnit** and **Spock**.
* Load data from various data sources into HDFS using **Kafka**.
* Worked with the configured and maintained **Jenkins** to implement the **CI/CD** process.

**Environment:** Angular, Node JS, HTML5, Bootstrap, CSS3-SASS, Twitter Bootstrap4.0, ES6 JavaScript, AJAX HTTP Service, TypeScript, Spring boot, JPA, jQuery, Apache Kafka, Active MQ, Jenkins, Bamboo, Log4j, MY SQL, CassandraDB, Elastic search, Redis, NPM, Web pack, RESTful, EC2.

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

**Responsibilities:**

* Developed a **Micro Service Oriented application** with **React JS** in the front end and **Spring Boot services**.
* Developed and implemented views using **HTML5, CSS3/LESS**, **Bootstrap 4.0**, **AJAX**.
* Worked on responsive design and developed a single **ISOMORPHIC responsive website** that could be served to desktop, Tablets and mobile users using **React.js**.
* Worked with **React JS** to build Bootstrap components for responsive UI Objects and linked with API data to view render the virtual DOM dynamically.
* Worked with **Docker** to deploy **Micro-services** in modern containers to increase Isolation.
* Implemented various screens for the front end using **React.js** and used various predefined components from **NPM (Node Package Manager)** and **redux library**.
* Used React**Router**to create a **Single Page Application**(SPA) which can bind data to specific views and synchronize data with server.
* Design and develop application codes using Java/JEE, XML, SOAP, REST, Spring Framework, and Hibernate Framework.
* mplemented task parallelism by dividing large workloads into smaller, concurrent tasks that can be executed in parallel, improving overall execution time.
* Ensured thread safety in shared data structures and components, using appropriate locking mechanisms and atomic operations to protect critical sections of code.
* Utilized Java’s concurrency utilities (e.g., java.util.concurrent package) to simplify multi-threading implementation and improve code readability.
* Implemented data replication and sharding strategies in NoSQL systems to ensure high availability and scalability, accommodating growing data needs.
* Migrated data between SQL and NoSQL databases, ensuring data integrity and consistency during the transition while adapting to different data models.
* Collaborated with development teams to define and implement data access patterns, ensuring efficient data usage across applications.
* Monitored and maintained database health through regular backups, performance monitoring, and troubleshooting of issues as they arose.
* **Employed Bootstrap 4.0** to create a responsive and mobile-first design for the Lowe’s application, ensuring that the application’s interface adapts seamlessly to various screen sizes and orientations.
* Customized **Bootstrap** components, including modals, dropdowns, and carousels, to match the specific requirements of the application, enhancing **usability** and visual consistency.
* Utilized **Bootstrap’s** pre-defined classes and grid system to ensure cross-browser **compatibility**, delivering a uniform user experience across different web browsers.
* Manage and architect the Gateway Secure Token Service (**STS**) to achieve **SSO** and AWS integration across corporate vendors by **API Gateway**, SAML, OAuth 2.0, OpenID Connect, WS-Trust, WS-Federation, SiteMinder, **Kerberos**, PKI, AWS APIs, Temp Creds, XSLT.
* Performing Code reviews with spring boot, Docker, AWS Serverless Lambda, Node JS, S3, DynamoDB, CloudWatch, AWS Serverless Lambda, NextJS, Node JS, S3, Microservices, DynamoDB, CloudWatch, node.js, Logging - Grafana, Prometheus, AWS the team, design API reviews with the Architects.
* Utilized working knowledge of J2EE/Java and Spring, and Hibernate to troubleshoot technical issues (error messages, bugs with the software, availability).
* Implemented **flux pattern** by using **redux framework** as a core dependency.
* Used **Redux Framework architecture** (like **Reducers,** WebGl**, Saga**) for integrating the data with React-Native Application and created and used **Reducers** to modify **Store** State Tree.
* Used **Hibernate** as the **Object-Relational Mapping (ORM)**tool to access persistent data.
* Worked with **spring batch**to convert xml files to database records.
* **Quartz scheduler** is used to schedule jobs in the application.
* Configured **JBOSS** Application server and used **Oracle** database.
* Used a highly durable messaging system called **Apache Kafka** for asynchronous exchange of vital information between multiple business applications.
* Behavioral and Integration testing is done using **Cucumber** and **Selenium** respectively.

**Environment:** JavaScript, React JS-Redux, HTML5, CSS3-LESS, AJAX, Bootstrap, jQuery, Spring Security, Rabbit MQ, Microservices, Restful services, Jenkins, Spring Oracle DB, MySQL, Spring Data, Spring AOP, Spring, Spring Batch, RESTful Services, Swagger, Jersey, Docker, Ribbon, JUnit, Mockito, Mocha, Chai, Selenium, Cucumber, Cassandra DB, Kafka,Bootstrap.

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

**Full Stack Developer**

**Responsibilities:**

* Created responsive **UI modules** using **Backbone.JS, Handlebar JS, Ember JS**.
* Updates **data**using**CRUD (Create, read, update, delete)** features of **Backbone.JS**and Paginator for pagination.
* **Grunt** used for Minification of JavaScript files, compilation of **SAAS**files into **CSS** and for monitoring the application.
* Implemented digital case management systems to streamline and digitize the documentation, tracking, and management of child welfare cases.
* Created online portals for stakeholders, including parents, guardians, and legal representatives, to access relevant information and updates on child welfare cases.
* Implemented **React** router for building the **single page application.**
* Built **React** application utilizing **Redux** for **state management**.
* Used **props, states, keys, refs** in **React.js** to build the **UI** components.
* Used **React-router** to route the pages on the client-side and built SPA.
* Implemented **Bootstrap** to build dynamic and responsive user interfaces for digital case management systems, facilitating an intuitive and accessible experience for case workers and stakeholders.
* Leveraged **Bootstrap’s** ready-to-use components and grid system to **rapidly** prototype and iterate on user interface designs, reducing development time and enhancing the speed of delivering new features.
* Implemented **Multithreading** for handling multiple requests and for**high performance**.
* Developed **RESTful services** based on **Spring REST** using **Jersey framework**.
* Orchestrated Dashboard Services at Scene Health utilizing MongoDB Atlas hosted on AWS.
* Conducted data aggregations and facilitated report generation through Tableau integration via MongoDB BI.
* Played a pivotal role in configuring RDS instances at Scene Health using CloudFormation templates and Terraform.
* Successfully transitioned conflicting Terraform modules to CloudFormation templates and maintained AWS components such as EC2, IAM, VPC, ELB, and Security groups.
* Managed DB2 database operations at Scene Health, handling application data loading.
* Developed and optimized PL/SQL procedures and intricate queries for diverse data operations.
* Contributed to efficient data management by writing Hibernate Criteria queries and implementing ORM mappings to connect Java entities with database tables.
* Utilized Spring Inversion of Control (IOC) at Scene Health to maintain the life cycle of beans, seamless integration and session information management.
* Implemented **Hibernate** Template to great extent making use of **HQL, Named Parameters, Named Queries, and interfaces** such as **Expression, Query and Criteria**.
* Developed **Spring Batch** processes for financial reporting applications.
* Message broker services are achieved in the application using **Active MQ**with**JMS framework.**
* Implemented **Spring Dependency Injection (IOC)** feature of spring framework to inject beans into User Interface and **AOP** for **Logging**.
* Used **Spring Security framework** for login authentication, password hashing. Worked on **Java Message Service (JMS) API** for developing message-oriented middleware (MOM) layer for handling various asynchronous request.
* Implemented **Quartz scheduler** to execute batch programming using **Spring Batch**.
* Configured Jenkins to handle application deployment in **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 web hooks plugin.
* **JIRA** tool for project management used in the application to tack bugs, issues, and project management functions.

**Environment:** Angular JS, HTML5, CSS3, AJAX, Bootstrap, JSON, XML, Active MQ, JMS, Hibernate, DB2, SOAP services, JAX-WS, JPA Framework, Spring Rest, Spring MVC, Spring Security, Hibernate.

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

**Full Stack Developer**

**Responsibilities:**

* Designed **User Interface (UI)** web applications using web technologies like **HTML5, HTML DOM, DHTML, XHTML, XML, XSL, XSLT and CSS3**based on**W3C standards**. wrote custom JavaScript utilizing **jQuery** to enhance theme, setup server to handle a greater number of page requests.
* Build data-driven applications for the enterprise using the **Prime Faces** Java server **Faces** UI framework.
* Created views using components of JSF **Prime Faces** in **XHTML files** and configured **JSF framework**.
* Extensively used **Java server Faces (JSF)** Navigation rules and **JSF tags** for enhanced user Interface.
* Implemented data access logic by using **Hibernate Persistence framework** to enable the persistence mapping between java objects and **RDBMS**.
* Developed User Interface view with **Spring MVC** and **AngularJS** as a **Service Oriented Architecture (SOA)** application.
* Performed **Angular JS** end to end testing using **Protractor framework** and **Integration testing methods**and**Behavioral driven** are implemented using **Selenium**and**Cucumber** respectively.
* Created reusable templates using **Angular directives** and worked with **NPM package** manager tools (**Node JS**).
* Good knowledge on using **Angular APIs**
* Implemented HTTP requests using **Rxjs** **Observable library** to handle **multiple values over time.**
* Good knowledge on creating services with **Angular**  **@injectable property**, as to make the service available for **dependency injection** when creating components.
* Used **jQuery UI plugins** like **Draggable, Resizable, Autocomplete, AJAX, JSON, Angular JS**in developing interactive web pages.
* Used **HTML5, CSS, jQuery, Ajax** and performed client-side validations using **JavaScript**.
* Implemented **AngularJS** as the framework to build a **single-page application (SPA).**
* Thorough knowledge of **Micro service architecture** and experience in implementing several design patterns
* Designed and developed  **Micro service** business components using **Java /J2EE, Spring** Used **JaxWS** framework to create **SOAP** web services.
* Wrote Web Services using **SOAP** for sending and getting data from the external interface.
* Used **Spring Batch** to build the **POJO-based development** approach of the Spring Framework.
* Involved in using **JPA (Java Persistence API)** frameworks and APIs such as **JDO** and**hibernate.**
* Worked on **Hibernate** persistence tier. This involves developing mapping files and **Hibernate (POJOs/ DAOs).**
* Developed **UNIX Shell Scripts** for batch jobs to transfer data on different servers.
* Used **Maven** as a build tool and **Jenkins** for monitoring the build status of the project.

**Environment:** HTML, XHTML, CSS, JavaScript, jQuery, Java Server Faces (JSF), Jasmine & karma, Gulp, JSP, Spring AOP, Spring Bean, Spring Security, Hibernate, RDBMS, SOAP, JAX-WS, RDBMS, Drools, Junit, Log4j,

**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.