**Mahesh V**

**Sr. Full stack Java developer**

**Contact: 9728538231**

**Email: krishna@adven-it.com**

**Summary:**

* Full Stack Java/JEE Developer with **8 years** of solid experience in analysis, design, development, testing, and documenting of web and distributed n-tier applications.
* Expertise in developing and leading software applications end-to-end including Java/JEE middleware, front end, database development, Linux script development, integration with other systems, corporate single sign-on integration, etc.
* Well-versed with the complete Software Life Cycle Development (**SDLC)** process, which includes Analysis, Designing, Developing, Testing, Implementing and documenting the software, working in environments using Agile (SCRUM), Water Fall and Test-Driven-Development methodologies.
* Expertise in Object Oriented Analysis (OOA) and Object-Oriented Design **(OOD**) using Rational Unified Process (RUP) and UML (Unified Modeling Language).
* Experience in application development using **J2EE/JEE technologies** like Servlets, **JSP, JDBC, RMI, JNDI, EJB** and JMS.
* Front-end development technologies such as **NodeJS**, **React JS**,

JavaScript, jQuery, AngularJS, Angular2, Typescript and Bootstrap framework

* Experience in designing, and developing applications using Struts, Spring Core and boot frameworks, Micro services, Web Services using **SOAP, REST / RESTful**.
* Experience with web front-end development using **HTML, HTML 5, XHTML, DHTML, XML, XSLT, JSON, CSS**, **CSS3, JSP, JavaScript, AJAX, jQuery, React JS, Angular2/46/8, Node.js** and Google Web Toolkit (GWT).
* Experience with XML, parsing XML, and converting the XML to XML using XSLT/XPath.
* Expertise in working with code review tools including Fortify and Find bugs.
* Experience in developing applications using Object-relational mapping frameworks including Hibernate and iBatis.
* Experience with Service Oriented Architecture (SOA) environment with a strong focus on Business Process Execution Language (BPEL). Enterprise integrations using MULE Enterprise Service Bus (ESB).
* Developed RESTful APIs for Microservices communication, ensuring scalability and modularity.
* Utilized Docker for containerization of Microservices, facilitating easy deployment and management.
* Experience in creating and consuming data from RESTful services using JSON and Postman for testing the Web Services.
* Experience in providing highly available and fault-tolerant applications utilizing orchestration technologies like Kubernetes and Apache Mesos on the Google Cloud Platform.
* Used Docker and Kubernetes as Containers for deployment and CI process.
* Tested web services using JUnit, Mockito Easy mock, Spring JUnit Runner, Spock and achieved the goals of Test-Driven Development (**TDD**).
* Experience in working and managing Oracle, **My SQL**, and **SQL** Server databases.
* Experience in writing **SQL scripts, PL/SQL** Packages, Procedures, Functions, and Triggers for Oracle and **My SQL** databases.
* Strong understanding of common API technologies, such as **OAuth, SAML, Spring Boot** , and Microservices.
* Expertise with tools including Eclipse, **NetBeans**, **Oracle SQL Developer**, **TOAD**, **Oracle SQL** Developer Data Modeler, and **SOAP UI**.
* Quick Learner, a highly motivated team player with excellent organizational and analytical skills.
* Experience with **J2EE/JEE** web application servers like JBoss Application Server **(4.x/5.x), Apache Tomcat** version (4.x/5.x/6.x), **Web Logic** (10.x/11.x), Web Sphere 6.x and Glassfish.
* Created clusters in Google Cloud and manage the clusters using Kubernetes (k8s). Using **Jenkins** to deploy code to Google Cloud, create new namespaces, create docker images and push them to the container registry of Google Cloud.
* Software development involving cloud computing platforms like Amazon Web Services (**AWS**), Azure , and Google Cloud (GCP).

**TECHNICAL SKILLS:**

|  |  |
| --- | --- |
| **Programming Languages** | Java, Groovy, C |
| **J2EE Technologies** | JSP, JDBC, JMS, RMI, Servlets, JNDI, JSF, Swings. |
| **Java Frameworks** | Spring 2.5/3.x/4.0, Struts 1.x/2.0, Hibernate 3.0/4.0/5.12 |
| **Web Technologies** | HTML 4/5, CSS2/3, JavaScript, AJAX, JQuery, XML, Angular JS |
| **Web Services** | REST, SOAP, Apache Axis 2,Node JS. |
| **Databases** | MySQL, Oracle 9i/10g/11g/12c, Microsoft SQL Server 2008/2008 R2/2012, MS Access, TOAD. |
| **App/Web Servers** | Apache Tomcat 6.0.x/7.0.x, Oracle Web Logic Server 10g/11g/12c, JBoss 4/5, Web Sphere. |
| **Build Deploy Tools** | Maven, Gradle, Ant, Log4j, JAP7, SDSS. |
| **Operating Systems** | Windows 98/2000/XP/Vista/7/8/10, DOS, UNIX, Linux |
| **Design Patterns** | MVC, Singleton, Factory, Front Controller, DAO |
| **IDE Tools** | Eclipse (Indigo, Europia, Hugo, Luna, Mars, Oxygen), Net Beans, RAD, Visual Studio |
| **Version Controls** | CVS, SVN, GIT |
| **Testing Tools** | JUnit, HP-ALM, Selenium, Eclipse plug-ins |

**Professional Experience:**

**Client: Cigna, Bloomfield, CT Jan 2023 - Present Role: Sr. Full Stack Java Developer**

**Project Description:** Cigna, a global health service company, engaged our team to develop a comprehensive Financial Management System. The project aimed to enhance Cigna's financial operations, ensuring efficient data management, seamless integration, and robust performance to support their extensive range of health services and insurance products.

**Responsibilities:**

* Interacting with the Product manager and users to do requirement gathering. Do brainstorming on requirements, sprint planning, create user stories etc.
* Worked on a green field project namely Market Needs Assessment end to end, including ETLs, Restful APIs using NodeJS, storing data on Postgres, developing front end using **ReactJs** and deploying on AWS Fargate as a Docker image.
* Developed UI using React hooks functional components, React-Redux, React-Router, Redux-Sagas, and Material-UI components.
* Experience in **Azure, C# & VB .Net**, Entity framework, **CSS, LINQ, TFS,VSTS**, Git, Jenkins, **Node JS**.
* Displayed thousands of commercial agriculture products via scrollbar on UI using react-window without compromising performance.
* Used React-Leaflet to show geographical map by pointing using the multi-polygon geometry json structure consisting of latitudes and longitudes.
* Utilized Docker for containerization of Microservices, facilitating easy deployment and management.
* Applied Microservices design patterns such as Saga pattern, CQRS, and Event Sourcing to address complex business requirements.
* Created routing using React-Router to navigate across different pages of app. In each route, the app performs operations like creating market needs, prioritize the needs and view the created needs.
* Implemented continuous integration and continuous deployment **(CI/CD**) pipelines for Microservices using Jenkins or **GitLab CI/CD**.
* Improved the UI using **Angular 8**, Angular Material, **Bootstrap**, Priming
* Extensively used Angular-cli to generate components, routes, pipes and unit tests.
* Created Res-useable components, Custom Modules, Custom Directives, Pipes, Services in Angular 8.
* Worked on developing applications using Java technologies including Core Java (Java 6 and Java 8), J2EE, Java Server Pages (JSP), Servlets, **Java Script**, **JDBC**, Spring Boot, Hibernate
* Involved in developing UI using JSP, HTML, CSS and **Java Script**
* Implemented different frameworks like Hibernate, Spring Boot
* Extensively used Redux-Sagas to create async actions and perform API calls.
* Extracted data (like sales data, products Info etc.) from various sources via Rest api calls, transformed and loaded into Postgres database.
* Implemented the application using Spring Boot Framework and handled the security using Spring Security.
* Used Micro service architecture with Spring Boot based services interacting through a combination of REST and Apache Kafka message brokers and also worked with Kafka Cluster using Zookeeper.
* Developed **RESTful API’s using NodeJS**, ExpressJs that can be consumed by UI.
* Used GitHub and Git for version controlling.
* Implemented API gateway for Microservices to manage cross-cutting concerns like authentication, rate limiting, and request logging.
* Selecting the appropriate AWS service based on compute, data, or security requirements.
* Integration of **Amazon Web Services (AWS)** with other applications infrastructure.
* Involved in writing Java API for Amazon Lambda to manage some of the AWS services.
* Installed application on   **AWSEC2** instances and configured the storage on S3 buckets.
* Created monitors, alarms, and notifications for **EC2** hosts using Cloud Watch.
* Migrated applications to the AWS cloud.
* Utilized Docker for containerization of Microservices, facilitating easy deployment and management.
* Involved in writing java API for Amazon Lambda to manage some of the AWS services.
* Created several reusable components like drop downs, filters etc.
* Overridden several Material-UI styles using classes provided by material-UI
* Used CSS flexbox for responsive layout.
* Used React-Redux for centralized state management. Created several reducers and composed them to single reducer.
* Written logs using log4Js library.
* Written unit test cases for UI using react-hooks testing-library, chai, and redux-mock-store. Used Chai for API unit testing.

**Environment:** Java 11, J2EE, JSP, Groovy, Agile, Google Cloud Platform (GCP), Kubernetes, Spring, Cassandra, Web Services (SOAP & REST), Microservices, jQuery, JavaScript,Node JS Angular 2/4, React JS, XML, XSD, JSON, Tomcat 6, Maven, Oracle database, SQL, PL/SQL, Ant, Java Script, SVN, Visio, Spring boot, SQL Developer, SOAP UI, and WebLogic.

**Client: AT & T, Tampa FL Jan 2022 – Dec 2022**

**Role: Full Stack Java Developer**

**Project Description:** AT&T, a global leader in telecommunications, partnered with our team to develop an advanced Financial Management System to optimize their financial operations. The project aimed to create a scalable, efficient, and robust system to handle complex financial transactions, data management, and integration with existing systems.

**Responsibilities:**

* Implemented UI functionality using **HTML5, CSS3, jQuery, Java Script, Angular 2/4**.
* Worked extensively on Spring Integration module, configured channels, gateways & bridges to communicate between different systems/applications in the enterprise in Service Oriented Architecture.
* Hands-on developer to build applications and API’s, creating and consuming both **SOAP** and REST Web Services using spring boot.
* Worked on different modules of spring, including Spring Integrations, spring boot, **MVC**, Webservices, DAO.
* Worked on the Cassandra database for storing the transaction data.
* Worked on the JUnit framework for unit testing.
* Implemented JAX-RS services using Spring Boot.
* Developed Web Services using RESTful to retrieve data for the client side.
* Reviewed the existing code components, and implemented the performance enhancements.
* Created Docker images from scratch, customized, and modified base images from existing environment configurations and maintained the image repo for development teams.
* Developed powerful Single Page Application and dynamic responsive Web Application with Angular 8.
* Upgrading the existing system from Angular 7 to Angular 8.
* Integrated caching mechanisms like Redis or Memcached to optimize the performance of frequently accessed Microservices.
* Designed and implemented data partitioning strategies for Microservices databases to ensure horizontal scalability.
* Developed batch jobs for data processing in multiple applications and scheduled them via crontab. Implemented Java SFTP processes for securely transferring file between servers.
* Worked as a full-stack, Java developer in a global team to develop, support, and maintain Cardinal’s product applications in an agile environment.
* Implement UI of the project using HTML5, CSS3, and jQuery, Java Script, Angular8 React JS and Redux.
* Developed java modules implementing business rules and workflows using Spring Boot
* Involves in Sprint planning for the estimation of efforts for user stories and bugs
* Designed and developed the REST-based Microservices using the Spring Boot
* Used Spring Boot at the back-end which helps to develop applications with ease
* Implemented React components both functional and classical for UI widgets and Used Redux for state management, to store the whole state of the app in an immutable object tree.
* Implemented the various layers of the application using Spring Framework modules such as Spring IOC, Spring AOP, Spring Security, Spring MVC, and Spring Data.
* Developed batch jobs for data processing in multiple applications and scheduled them via crontab. Implemented Java SFTP processes for securely transferring file between servers.
* Responsible for migrating existing modules in IBM MQ to Apache Kafka and worked on creating Kafka adaptors for decoupling the application dependency.
* Worked on AWS provisioning and good knowledge of AWS services like EC2, S3, ELB (Load Balancers) etc.
* Deployed cloud stacks using AWS Op Works and configured Elastic Load Balancers with EC2 Auto Scaling groups.
* Implemented microservices to support highly customizable and scalable content processing system destined in a cloud-based environment.
* Implemented Microservices architecture using spring to make application smaller and independent.
* Used AWS RDS for database configuration.
* Used AWS code pipeline for moving the code across different environments.
* Developed Action classes, Form bean classes and Struts configuration file using Struts framework.
* Wrote complex SQL scripts and tuned them for efficiency and performance.
* Worked on PL/SQL code and developed new procedures and functions.
* Involved in the data model of the application schema and made performance improvements that helped in faster data processing.
* Implemented automated testing strategies, including unit tests, integration tests, and contract tests, to ensure the reliability of Microservices.
* Worked on SoapUI for web service development and written test suites for the web service methods.
* Implemented the Continuous Integration (CI) process within the team using IBM to build and deploy.
* Worked in the Test-Driven Development environment and used JUNIT frameworks like easymock, and jmockit to write and test scenarios in Java.
* Documented and cross-shared the knowledge with other developers in the team.

**Environment:** Java 8, JEE, JSP, Groovy, Agile, Google Cloud Platform (GCP), Kubernetes, Spring, Cassandra, Web Services (SOAP & REST), Microservices, jQuery, JavaScript, Angular 2/4, React JS, XML, XSD, JSON, Tomcat 6, Maven, Oracle database, SQL, PL/SQL, Ant, Java Script, SVN, Visio, Spring boot, SQL Developer, SOAP UI, and WebLogic.

**Client: Ruletronics Systems Pvt Ltd, India Aug 2018 - Aug 2021**

**Role: Full Stack Java Developer**

**Project Description:** RuleTronics Private Ltd, a pioneering technology solutions provider, partnered with our team to develop a sophisticated Financial Management System. This system aimed to streamline RuleTronics' financial operations, ensuring efficient data management, improved performance, and seamless integration with existing services and databases.

**Responsibilities:**

* Developed application components- **JSPs,** Action Form Beans, Action Classes, Value Objects, and model layer logic.
* Used Bootstrap and AngularJS, React.js and Node.js in effective web design.
* Developed views using JSPs and struts tags. Using Tiles framework, improving UI flexibility and providing single point of maintenance.
* Developed the application using Angular framework and Node JS platform using JavaScript and Typescript programming
* Developed the code for asynchronous update to web page using **JavaScript** and Ajax.
* Developed front-end pages using **CSS, HTML**.
* Developed application using JavaScript for Web pages to add functionality, validate forms, communicate with the server.
* Redesigned and re-implemented services using **NGINX, Node JS**, Express JS, and **MySQL**.
* Used Spring IOC, Writing Java Bean classes, with get and set methods for each property to be configured by spring.
* Designed and implemented data partitioning strategies for Microservices databases to ensure horizontal scalability.
* Deployed and hosted the application on **AWS** using OpsWorks, automated single click deployment onto AWS using Chef.
* Used Angular 6 as framework to create Single Page Application (SPA) which can bind data to specific views and synchronize data with server using Sass, Bootstrap and Angular 6.
* Improved the UI using Angular 6, Angular Material, **W3.CSS,** and Priming.
* Created Reactive forms and performed validations using Angular 6.
* Created Res-useable components, Custom Modules, Custom Directives, Pipes, and Services in Angular 6.
* Configured Jetty Server to instead of Tomcat to run for Spring Boot apps.
* Hands on experience with one or more Cloud Infrastructure and Services like **Amazon AWS**.
* Developing a product-level cloud-based architecture in AWS.
* Installing and automation of application using configuration management tools Puppet and Chef.
* Involved in deploying systems on Amazon Web Services Infrastructure services **EC2, S3, RDS, SQS**, and Cloud Formation.
* Worked on **AWS**, High Availability Practices and deploying backup/restore infrastructure.
* **SOAP and REST** based webservices are implemented using Apache CXF framework.
* Modified the configuration of the Spring Application Framework IOC Container.
* Worked on Rest Web services for backend services, used Couchbase DB (NoSQL) for database services.
* Used Hibernate ORM framework as persistence engine, actively engaged in mapping, and hibernate queries.
* Involved in writing Hibernate mapping files (HBM files) and configuration files.
* Used Log4j for logging Errors.
* Using JUnit test, extensively written test cases for this system to test the application.
* Worked on Oracle as the backend database.

**Environment:** J2EE, JDK5, NodeJS, Angular JS, Struts1.2, Spring 2.5.1, Hibernate 3.2, JavaScript, JSP, Ajax, XML, XSLT, HTML, UML, SOAP, WSDL, Eclipse 3.6, RAD 7.0, IBM WebSphere7.0, Oracle11g, TOAD 10.1, MySQL, JUnit

**Client: YPAY – India April 2016 - July 2018**

**Role: Java Developer**

**Project Description:** The YPAY Enterprise Financial Management System was designed to comply with J2EE specifications, ensuring a robust and scalable architecture. The system incorporates various technologies for front-end and back-end development, along with web services integration to support YPAY's financial operations.

**Responsibilities:**

* The system was designed per **J2EE** specifications. Servlets were used as a Front Controller gateway into the system.
* Developed data persistence layer using Hibernate as Object Relational Mapping Tool for the backend data persistence
* Used HQL queries to retrieve the required data using Hibernate
* Implemented Web Services using **SOAP, Restful Web Services**
* Developed Restful Web Services using Spring Boot to migrate the existing application for changing the existing methods to use POST and GET functionalities
* Consumed SOAP web services exposed by the backend system and performed validations and check logic before sending these to the **WEB U**I
* Used multi-threading in programming to improve overall performance.
* Involved in fixing the defects raised.
* Created Reactive forms and performed validations using Angular2.
* Design the frontend Web form-based application using **Angular JS 2.0**, **Ajax**, **CSS** and **HTML 5**.
* Used **Ajax Controls, Web forms, JavaScript and HTML** for Commission, Payment and Inventory reports.
* Working on view page to enhancing the UI interface using HTML5, REST API, JQuery and JavaScript Angular JS for web Application.
* Good experience in writing test cases using Java testing frameworks like JUnit, Mockito for Unit testing
* Used SOAP requests to test the functionality of payments and transfers through SOAP UI to validate all the scenarios and make sure everything is working fine
* Developed Action Classes, which acts as the controller in Struts framework
* Created table and different batch programs to clean up tables in **DB2** database
* Extensively used Collections and Exceptions in the batch program for database cleanup
* Establishing JDBC connection using database connection pool

**Environment**: JAVA, J2EE, HTML, XML, CSS, SASS, JavaScript, windows/Unix, Struts, Servlets, JSP, EJB, Eclipse, JDBC, DB2.