

M SAI SREE
Java Developer

Email: smareddy11@gmail.com

Contact number- (281) 603-3796

PROFESSIONAL SUMMARY:

- IT professional with more than 7 years of experience in developing web based, SOA based and **client/server applications in multi-platform environments with Java/J2EE technologies**, Involved in various domains like Finance/Telecom/Commerce/Health.
- Proficient with Software development methodologies like **Agile (Scrum) Methodologies** and **Waterfall methodologies**.
- Experience in developing the User Interface Rich Web Applications and Web Service Applications using User Interface Technologies like **HTML, CSS, jQuery, Ajax** Chrome-Developer Tools and Internet Explorer-Developer Tollbar.
- Experience in **J2EE** framework and design patterns as well as **EJB, web services**.
- Used the fundamentals of **Java 1.8** like **parallel streams** and **filters** through **Lambda expressions** to handle the searching.
- Proficient in core java concepts like **Collection Framework, Multi-threading, Generics, Annotations, Serialization, Thread pools, JavaBeans, Externalization**.
- Good knowledge of **data structures, algorithms, operating systems, and distributed systems fundamentals**.
- Expertise in developing web-based enterprise applications using **core Java, J2EE, Servlets, JSP, EJB, JMS, JDBC, JDWP, Hibernate, Node JS, Web Services (SOAP and REST), MVC framework (Struts, Spring MVC), SQL and Object-Oriented Programming techniques**.
- Experience in design and development of J2EE applications with **Servlets, Struts, Spring Framework, Web Services (SOAP & REST), JDBC, Hibernate and IBatis**.
- Experience in building web applications using spring framework modules like **Spring ORM, Spring MVC, Spring DAO, Spring AOP, Spring Security, Spring Core, Spring IOC, Spring Boot, Spring Batch** and **Web Services** using **Eclipse** and **integration with Hibernate** as well as **Struts**.
- Proficient in developing **Web Services**, related technologies & frame works: **WSDL, SOAP, REST, JAX-WS, JAXB, JAX-RPC, AXIS and Jersey, SOAP UI** and generating client's client using **Eclipse** for **Web Services** consumption.
- Experience in implementing client - server business applications using **Microservices**.
- Implemented **RESTful services** with **JAX-RS** using **Spring Boot** and **Micro Services Architecture**.
- Developed multiple web pages using **JSF** and business logic using **Java** for multiple modules in the application.
- Experience with tools like Experience with **Docker** Images/containers and **Kubernetes**.
- Experience in creating real time data streaming solutions using **Apache Spark and Kafka**.
- Deployed Applications in cloud like **Amazon Webservices** and **Pivotal Cloud Foundry**.
- Experience in designing, developing, and deploying **Java/J2EE** applications on **Web sphere, Web logic, Apache Tomcat** and **JBoss** application Servers.
- Experience in using message brokers **RabbitMQ**.
- Working experience of Relational DBMS like **MySQL, Oracle SQL (PL/SQL), SQL Server (MS SQL), DB2 and NoSQL DB like MongoDB**.
- Experience in implementing Design Patterns like **Singleton, Factory, MVC, Data Access Object** and **Business Delegate**.

- Focused on **Test Driven Development (TDD)** thereby creating detailed **JUnit** tests for every single piece of functionality before writing the functionality.
- Configured and created application **log files using Log4J** to trace required application messages.
- Hands on experience in building tools like **ANT, Maven, Gradle**.
- Implemented the concept of Release branches for the project using **GitHub**.
- Actively involved in the **DevOps** streamlining process through **Jenkins** CI and **CD** Release Automation.
- Having Experience on **UNIX** commands and Deployment of Applications in Server.
- Ability to work effectively in cross-functional team environments and experience of providing training to business users.
- Good understanding of Business workflow, Business logic and Business methods for further implementation of user requirements in a distributed application environment.

SKILLSET

Languages	C, C++, Java, SQL, PL/SQL, Python, Test Driven Development.
J2EE Technologies	Servlets, Spring, Spring Boot 2.x, JSP, JDBC, JPA, JSTL, JNDI.
Web Technologies	HTML, CSS, XSLT, XML, jQuery, JavaScript, Angular JS, Angular, Node JS, React.js.
App/Web Servers	IBM WebSphere Application Server, JBoss, JBoss, WebSphere, Tomcat, Web Logic
Frameworks	Struts, Spring MVC, Microservices – MVC, JSP, JPA, JSF
SOA	REST, SOAP, WSDL, SAAJ, SAXJ, RESTful, AWS, Microservices, JAXB, Apache-CXF
ORM Tool	Hibernate, JPA, IBatis
Tools	LoadRunner, Log4J, Ant, BIRT, Jenkins, Kafka, Rabbit MQ.
IDEs	Eclipse, IntelliJ.
Databases	Oracle, MySQL, MSSQL Server, AWS, Couchbase, Mongodb, Cassandra DB, PostgreSQL
Cloud Technologies	AWS (Elastic Search, EC2, S3, EMI, ELB, DynamoDB), Docker, Azure, PCF
XML Technologies	DOM, SAX, JAXP, JAXB, XML-RPC and XML Beans.

WORK EXPERIENCE

Client: Perficient, TX

June 2019 – Till Date

Role: Java Full Stack Developer

Description: Comcast Corporation is an American Multinational mass media company and is the largest broadcasting and largest cable company in the world by revenue. Xfinity Customer Order Service is a platform to place an orders and report issues on behalf of customers through the comcast portal and project involves the developing online services like number portability from another carrier to Comcast carrier and interacting with database.

Responsibilities:

- Actively involved in Analysis, Design, Development and Testing. Followed **Agile methodology** in the Scrum Cycle model where requirements evolve through collaboration between teams.
- Front-end use **HTML, CSS, jQuery**, to build a single page web app that communicate with back-end via RESTful API.
- Utilize **React JS components, Forms, Events, Keys, Router, Animations**, and **Redux** concept.

- Utilized **Java8** features like **Lambda** expressions and **Stream API** for Bulk data operations on Collections which would increase the performance of the Application.
- Used **J2EE** design patterns like Adapter, Service locator, Session facade, Singleton, Factory and DAO.
- Used a **Microservices architecture**, with Spring Boot-based services interacting through a combination of **REST** and **Spring Boot**.
- Developed Node JS and Spring Framework based **RESTFUL** Web Services for handling and persisting of requests and **Spring MVC** for returning response to presentation tier.
- Successfully designed and developed Java **Multi-Threading** based collector parser and distributor process, when the requirement was to collect, parse and distribute the data coming at a speed of thousand messages per seconds.
- Proficient in Object Relational Mapping (**ORM**) using **Hibernate 4.x** and Java Persistence API Standard (**JPA**) for database mapping and data retrieval facilities.
- Developed different **DAO** Implementation using **Spring JDBC** Template to pre-populate the Run Details, notifications etc. at the start of the application.
- Used **REST-Assured** API framework to publish content and test multiple functionalities. Also, used tools like **Postman**, **SOAP-UI**, **Swagger-UI** for testing web services.
- Implemented Node, **Spring boot microservices** to process the messages into the **Kafka** cluster setup and implemented **Kafka** producer and consumer applications with help of **Zookeeper**.
- For data migration from monolithic to **Microservice** application, we have used **Kafka** as our message streaming service and operated in two modes on-demand and batch migration.
- Secured the API's by implementing **OAuth2** token-based authentication using spring security.
- Developed Data Access Object (**DAO**) interfaces using **Hibernate** to Interact with database.
- Provided connections using **Spring Data JPA** and **Spring Data Cassandra** to the database and developed **SQL** queries to manipulate the data.
- Worked on **DevOps** methodologies and Cloud infrastructure automation tools and engaged in a project to transform all the products into microservices in containers.
- Implemented the **Docker** containers for virtualization, Ship, Run and Deploy the application securely to fasten the Build/Release Engineering, and Virtualized the servers using the **Docker** for the test environments and dev-environment needs.
- Run **Sonar Cube** Analysis on Project and Publish report to Sonar Dashboard and integrate with Jenkins.
- Used **Kubernetes** to orchestrate the deployment scaling and management of **Docker** Containers.
- Used Jenkins pipelines to drive all micro services builds out to the Docker registry and then deployed to Kubernetes, Created Pods and managed using **Kubernetes**.
- Experience in development and deployment on **SaaS** Applications like Amazon Web Services.
- Being a Software Engineer, participated in deploying the services/applications in **AWS EC2** instances and **Kubernetes** as well by creating the pods for load balancing.
- Worked on Big Data Integration & Analytics based on Hadoop, Spark, Kafka and web methods.
- Worked on **spring cloud** for configuration management and maintained the apps on **AWS**.
- Worked with executed unit test cases by using **JUNIT** for accuracy of code and logging with log4j.
- Worked on setting up **Maven** scripts to build, package, and deploy application code to the target **Apache Tomcat** and integrate with **CI/CD** pipeline with the configuration of **GitHub** version control.
- Used **Git** as Version Control System for the application and **JIRA** as defect management.
- Worked with executed unit test cases by using **JUNIT** for accuracy of code and logging with log4j.

Environment: Java (1.8), J2EE, React Js, Microservices, Spring Core, Spring Boot, Spring Data, Spring cloud, Spring Batch, Spring Security, Oracle, Cassandra, RESTful, SOAP, Kafka, Swagger, Postman, JUnit, Mockito, TDD, Log4j, Splunk, AWS, STS, Apache Tomcat, Agile/Scrum Methodology, Maven, JIRA, GIT, Bitbucket, Jenkins.

Client: Anthem, Atlanta, Georgia

Jan 2018 - Jun 2019

Role: JAVA Developer

Responsibilities:

- Designed and implemented the User Interface using JavaScript, HTML, XHTML, XML, CSS, JSP, and AJAX.
- Used Spring JDBC to execute database queries. Created row mappers and query classes for DB operations.
- Used spring framework for Dependency Injection and JDBC connectivity.
- Developed RESTful Web services client to consume JSON messages using Spring JMS configuration. Developed the message listener code
- Design HTML prototypes, visual interfaces, and interaction of Web-based design.
- Developed the business components using EJB Session Beans.
- Involved in developing business components using EJB Session Beans and persistence using EJB Entity beans.
- Involved in the development of the services layer using design patterns such as Session Façade, DAO, and Value Object etc.
- Worked on JBOSS locally and WebSphere in higher environments.
- Involved in development of custom Servlet handling HTTP request and response objects.
- Developed Web service modules based on SOA architecture to consume services.
- Involved in developing the DAO layer to interact with MySQL database.
- Configured Data Sources on the server for database connectivity.
- Developed SQL queries including joins, unions that perform data retrieval for the application, according to the business rules.
- Actively involved in code review, performance tuning and bug fixing phases.
- Used multithreading in programming to improve overall performance.
- Implemented Python based engine to classify emails as spam or not spam using Naïve Bayesian classifier. Used machine learning to train the engine and applied the functionality to classify emails.
- Implemented modules using Core Java APIs, Java collection, Threads, XML, and integrating the modules.
- Involved sending JMS message queues to different layers in the module.
- Good team player with the ability to solve complex issues. Initiated team discussions to improve efficiency of the system.
- Hands-on experience working with version control tools such as SVN and JIRA as issue tracking software.
- Used Maven as the build tool and TortoiseSVN as the Source version controller
- Involved in the deployment of the application in the Integration and Implementation environment.
- Used Subversion (SVN) as the configuration management tool to manage the code repository

Environment: Java, PL/SQL, SQL, HTML, CSS 3.0, Java Script, hibernate, Middleware Technologies, Ajax, Servlets, JSP 2.0, Web logic 10.4, J Boss, WebSphere, XML, XHTML, Eclipse, JMS, Oracle11g, EJB 3.0

Client: Cavista, Dallas, Tx

Aug 2016 – Dec 2017

Role: Java Developer

Description: TCF Bank is the wholly owned banking subsidiary of TCF Financial Corporation, a bank holding company. AIS (Account Information Systems) is Designed and developed for daily operations such as checking and savings account modules that keeps track of record for every customer according to their account information related to branch that includes the financial transaction, service charges and other.

Responsibilities:

- Involved in **Scrum** calls, **Sprint** meetings and updated work status in **Jira** tool as part of **Agile** methodology.
- Utilized Java 8 features like **Lambda** expressions and **Stream** API for bulk data operations.
- Implemented **Microservices** Architecture and made **REST API** calls and parsed the data to client and handled the security using **Spring Security** and **OAuth 2.0** authentication protocol.
- Using **HTML, CSS, JavaScript, Node JS, Angular JS** Involved in the development of user interface applications.
- Refactored existing **monolithic** application by identifying components needs to be transformed to **Micro-services** architecture, with **Spring Boot** based services interacting through a combination of **REST, RabbitMQ**.
- Designed and developed business components using **Spring Boot**, Spring Dependency Injection (Core), Spring AOP and **Spring Annotations**.
- Authored application using **Spring Cloud** services (spring cloud version of Netflix OSS-Eureka).
- Implemented Netflix-Eureka from Spring Cloud for dynamic **Microservices** registry and discovery.
- Exposed the REST API through the **Swagger** tool and Swagger annotations.
- Worked on **REST API** and **Elastic Search** to efficiently handle and searching JSON data.
- Created and managed the messages asynchronously by creating queues on **Rabbit MQ** server and involved in implementing the client layer using **Spring AMQP** framework to consume messages.
- Experience in migrating data from Oracle database using **Spring JDBC** to Apache Cassandra (Datastax) **NoSQL** Database.
- Worked with **NoSQL** database to store JSON like dynamic connections data.
- Scheduled batch jobs using **Spring Batch** framework along with Quartz and Spring Task scheduler.
- Developed customized SQL for retrieving the data from the database.
- Created new tables, **Stored Procedures**, functions, views, **indexes** and constraints, **triggers** and required SQL tuning to reduce the response time in the application.
- Used **SonarQube** code quality tool to maintain the code quality with solid understanding of JVM and monitored its performance.
- **Micro services** development using spring boot and deployment in Azure and **Pivotal Cloud Foundry (PCF)**.
- Working on **spring cloud** for configuration management and maintained the apps on **Pivotal Cloud Foundry (PCF)**.
- Used **JMS** Messaging system to send/receive messages from MQ and differentiated them by using Apache Camel.
- Focused on **Test Driven Development (TDD)** thereby creating detailed **JUnit** tests for every single piece of functionality before writing the functionality and decoupled it by mocking the dependencies using **Mockito** framework.
- Involved in writing **SQL** and Stored Procedures, functions and triggers for handling complex queries with help of TOAD and access them through Java Programs from **Oracle** Database.
- Used **Maven** building tool for creating EAR file to be deployed in application servers. It is also used for dependency management of code in application.
- Used **Git** as Version Control tool to develop the enhancements for every release.
- Triggered the jobs in **Jenkins** for continuous integration and continuous deployments (CI/CD).

- Monitored the error logs using **log4j**.
- Created **Shell Script** to invoke Java programs on UNIX batch platforms.
- Configured **Kafka** to read, write messages from external programs and to handle real time data.
- Involved in production support, resolving the production job failures, interacting with the operations support group for resuming the failed jobs.

Environment: Java (1.8), J2EE, Angular JS, Spring Framework (Spring Security, Spring Boot, Spring Data, Spring Batch), Web services, RESTful, Microservices, OAuth 2.0, XML, JUnit, Mockito, TDD, Log4j, STS, Agile/Scrum Methodology, Oracle, PCF, Tomcat, Maven, JIRA, Git, Jenkins.

LTI - Larsen & Toubro Infotech, India

June 2013 – Dec 2015

Role: Java Developer

Responsibilities:

- Analyzed and extracted the event flows of the UI layer code written in JavaScript, Struts, Java Applets and JSPs to come up with the sequence diagram for various transactions.
- Developed front-end using HTML, CSS, JavaScript, jQuery featuring the Implementation.
- Applied Struts features like Tiles and Validation framework.
- Designed and implemented Struts (MVC Paradigm) components such as Action Mapping, Action class, Dispatch action class, Action Form bean, and a form set for validation.
- Implemented MVC architecture by using Struts/Swings to send and receive the data from front-end to the business layer. Integrated the Struts and Hibernate to achieve Object-relational mapping.
- Used Apache struts to develop the web-based components and implemented DAO. Leveraged struts-validations, struts-beans tag libraries.
- Used Struts Action Servlet to route the request to corresponding Action Classes.
- Developed stored procedures, triggers, and functions for the application.
- Involved in designing data schema and development of SOAP web services with Axis implementation and REST web services with JAX-RS.
- Consumed SOAP and REST web services to get the persisted data from different databases.
- Configured Struts-Config. XML for action mappings.
- Involved in implementing Value Object Pattern, MVC & Factory Pattern.
- Deployed application into WebLogic Application Server.
- Used Subversion for version control management and consistently produced high quality code through disciplined and rigorous unit testing (JUnit).
- Analyzed the existing access system and re-designed it in Oracle by developing data validation and load processes using PL/SQL Stored Procedures, Functions, and Triggers to enhance and improve functionality requested by business for reducing manual processing time.
- Responsible for Code Reviews and PL/SQL Code Tuning.
- Involved in the overall system's support and maintenance services such as Defect Fixing, Feature Enhancements, Testing and Documentation.
- Used JIRA for tracking bugs & Maven script for building and deploying the application.

Environment: J2EE, Web logic Application Server, JSP, Servlets, jQuery, Struts, Hibernate, SOAP, REST, Oracle, Maven, JUnit, and Subversion, PL/SQL, HTML, CSS, JavaScript, JSTL.