**SOWMYA REDDY BODLAPATI**

**Java Full Stack Developer**

**Phone**: +1 469-885-8469

**Email**: sowmyareddysr234@gmail.com

**Professional Summary:**

* IT Professional with 10 years of experience as Java Full Stack Developer with expertise in **Design**, **Development**, **Analysis** and **Testing** and **Implementation** of Distributed Web Based Applications and Client/Server Applications using **Java**/**J2EE** Technologies.
* Experienced in all stages of the Software Development Life Cycle (**SDLC**), from requirements gathering to **Testing** and deployment, adhering to **Agile** methodologies like **SCRUM** and **Waterfall**.
* Skilled in building responsive and intuitive user interfaces using HTML5, **CSS3**, and JavaScript frameworks/libraries like **Angular** and **React**, with a focus on enhancing user experience and accessibility.
* Extensive experience on Developing Web Applications using **Spring** **Core**, **Spring Boot**, **Spring** **Security**, **Spring** **MVC**, **Spring** **AOP**, **Spring** **Actuator**, **Spring** **ORM**, **Spring** **Data** **JPA** and **Spring** **REST**.
* Strong experience developing Dynamic **Single Page Application** (**SPA**) using **MEAN** (**Mongo, Express Angular,** and **Node**) full-stack development tool kit.
* Extensive experience in **Java** development with proficiency in **Java 8**, **Java 11**, and **Java 17**, leveraging the latest features and enhancements to develop efficient and scalable applications.
* Implementing Core **Java** **SE** **8** concepts like **Streams** **API**, **Lambda** **functions**, **Functional** **programming**, and **Generics**.
* Expertise in developing **SOA** (Service Oriented Architectures) and **Microservice** architectures using **J2EE** web services based on **RESTful** (**JAX**-**RS**), **SOAP** (**JAX**\_**WS**), **JAX**-**RPC**, and **JAXR** using frameworks like **Apache** **CXF**, **Jersey**, and **Apache** **Axis**.
* Used **Java 11 to build modules and utilized Spring Boot framework to construct the microservices.**
* Experience in implementing and deploying **Java** **EE** design patterns such as **Singleton**, **Abstract** **Factory**, **DAO**, **DTO**, and **Business** **Delegate** in the Development of Multi-Tier distributed Enterprise Applications.
* Good knowledge in building real-time dashboard applications using **Splunk**, **open**  **Search**, and **Logstash** (**ELK**) to profile and monitor the streaming data of the application.
* Expertise in front-end technologies including **React**, **Angular 4/5/6/8/9/11/13/15, Node.js**, and proficient in **HTML5, CSS3, Bootstrap, jQuery, AJAX, SASS, and SCSS.**
* Demonstrated ability to architect and develop **Microservices**-based architectures leveraging technologies like **Spring Boot**, **Spring** **Cloud**, and **Kafka**, ensuring scalability, resilience, and performance optimization.
* Proven experience with Application Servers like **IBM** **Websphere**, **Oracle**/**BEA** **Weblogic**, **JBoss**, and **Tomcat**.
* Designed and executed data warehousing solutions with **Azure** Synapse Analytics, enabling real-time data analysis and supporting data-driven decision-making processes.
* Hands-on experience with big data technologies such as **Apache Spark**, skilled in data visualization with **Kibana**, providing actionable insights for strategic decision-making.
* Demonstrated expertise in **Scala** programming and leveraging the **Spark** framework to develop scalable and efficient data processing and analytics solutions.
* Experience in **Spring** **Security** Framework for User Access Management, JWT token-based authentication, and involved **OAuth** **2.0** configuration for authentication and authorization.
* Hands-on experience with **Java 8 features like Lambda expressions, Streams, Optional, Java Time API, Functional interfaces, and BASE64 encryption.**
* Experienced in **MQ** services like **Apache** **Kafka**, **Rabbit MQ**, **Active** **MQ**, and **IBM MQ**.
* Substantial involvement in **UNIX** and **Shell** scripting to automate the **build** **scripts**.
* Good Knowledge in **Amazon Web Services AWS (Amazon EC2, Amazon S3, Amazon Simple DB, Amazon Elastic Load Balancing, Amazon SQS, Amazon EBS, and Amazon CloudFront)**
* Successfully deployed **Spring Boot**-based **Microservices** in **Docker** containers and **Amazon** **EC2** instances using **Jenkins**. .
* Experience in database design and development of **PL/SQL** stored procedures and packages in relational databases like **Oracle**, **SQL Server, DB2**, **MYSQL**, and experience in implementing, designing, and developing NoSQL Databases like **MongoDB**.
* Extensive experience utilizing various **Integrated Development Environments (IDEs)** such as **Eclipse**, **IntelliJ**, **NetBeans**, and **Sublime** **Text**. Proficient in Build Tools including **Maven** and **Gradle**, alongside Logging tools such as **Log4j**.
* Experience in **Testing** with **Junit** Framework using **Mockito** and **Easy** **Mock**, **Spring** **Junit** **Runner**, **JMeter**.
* Used **Kubernetes** **Linux** container for deploying and scaling containerized applications.
* Experienced in utilizing source code variant administration apparatuses like **SVN**, **Git**, **Bit Bucket**, and undertaking sprinters like **Babel**, **Grunt**, **Gulp**, and **Webpack**.

**Technical Skills:**

|  |  |
| --- | --- |
| **Programming Languages** | C, Java 8, Java 11, Java 17, J2EE,, PL/SQL, Python, Scala, Kotlin, Drools |
| **Script Frameworks** | AngularJS, Angular 4/ 5/6/ 8/ 9/ 11/ 13/15, jQuery, Node.js, Backbone.js, React.js, Express.js, Ember.js, Handlebars.JS |
| **/J2EE Technologies** | Servlets, Spring, EJB, JPA, JTA, JDBC, JSP, JSTL, Java  |
| **Frameworks** | Struts 2.0/1.0, Hibernate 4.x/3.x, Spring, Spring IOC, Spring Boot, Spring Batch, Spring Security, Spring AOP, JSF, iBATIS |
| **Web Technologies** | HTML, HTML5, CSS/CSS3, JavaScript, AJAX, jQuery, Bootstrap, XML, Grunt, Gulp, JSON. |
| **Design Patterns** | Singleton, DTO Pattern, DAO Pattern, Delegate Pattern, Service Facade, Factory Pattern, Front Controller Pattern, MVC Pattern. |
| **Database** | SQL Server, MYSQL, Oracle, MongoDB, PostgreSQL, Cassandra, DB2 |
| **Cloud Technologies** | Amazon EC2, S3, simple DB, Elastic Load Balancing, SQS, EBS, cloud front, EKS and PCF. |
| **Web Services** |  XSD, WSDL, Apache Axis, SOAP, RESTful, GraphQL, JAX RS, JAX WS |
| **Messaging** | JMS (Java Message Service), IBM MQ, Rabbit MQ, Apache Active MQ, Apache Kafka  |
| **Web Servers** | Oracle Weblogic 10.x/9.x, IBM Websphere 8.x/6.x, Apache Tomcat 8.0/7.x/6.x |
| **Application Servers** | Apache Tomcat, IBM Websphere ,Weblogic, JBoss |
| **Methodologies** | Agile, Waterfall, OOAD, SCRUM, TDD, Kanban |
| **Testing Tools** | Junit, Mockito, Jest, Jasmine, Karma, Mocha, Chai, Selenium  |
| **Version Control** | SVN, CVS, GIT, GITLAB |
| **Build Tools** | Maven, Gradle, ANT, Jenkins |
| **IDEs** | Eclipse IDE, RAD 6.0/7.0, IntelliJ IDEA, NetBeans, Spring Tool Suite (STS) |
| **Operating Systems** | Windows, Linux, and UNIX |
| **Software** | Visual Studio, Rational Rose, and Microsoft Office |

**Educational Details:**

* Bachelor’s in Computer Science Engineering from Jawaharlal Nehru Technological University, Kakinada, India in May 2012.

**Professional Experience:**

**Client:** **Country Financial, NY Nov 2022 to Present.**

**Role: Sr. Full Stack Developer**

**Responsibilities:**

* Involved in **SDLC** requirements gathering, analysis, design, development, and Testing of applications by following **AGILE** **methodology** (**SCRUM**).
* Utilized new **Java** **17** features, including Sealed Classes and Pattern Matching for Switch, to write cleaner and more maintainable code.
* Implemented **Bean** classes and configured them in the **Spring** configuration file for **Dependency** **Injection**.
* Developed Controller Classes using **Spring** **MVC**, **Spring** **AOP**, **Spring Boot**, and **Spring** **Batch** modules, and handled security using **Spring** **Security**.
* Developed dynamic, responsive user interfaces with **Angular 15**, TypeScript, Bootstrap, HTML and CSS focusing on high-quality, maintainable code.
* Developed Project Specific **Java APIs** using **Data Structures, Algorithms**, and **Core Java** for new requirements.
* Hands-on experience **Java 8** features like **Lambda, Streams, Optional, Java Time API**, Functional interfaces, and BASE 64 encryption etc.
* Designed and developed **Micro** **Services** business components using **Spring Boot**.
* Implemented resource classes, endpoints, and entity providers following **JAX-RS** specifications.
* Involved with the team in developing UI applications using front-end technologies like **HTML5**, **JavaScript**, **CSS3**, and pure **CSS** layouts, and using **AJAX** frameworks**: Node JS, Express JS, EJS, jQuery, Angular**.
* Developed a **CI/CD** system using Jenkins within a **Kubernetes** container environment, leveraging **Docker** for building, testing, and deploying applications.
* Created multiple **Kubernetes** Clusters in **AWS** and deploying a **Docker** Image on top of Kubernetes Cluster for testing purpose.
* Developed scalable backend services using **TypeScript**, leveraging its strong typing system to enhance code reliability and maintainability.
* Implemented modern utility-first **CSS** styling by integrating **Angular** **12** with **Tailwind** **CSS**.
* Utilized **SASS/LESS** for theming and customization across web applications.
* Integrated third-party libraries and UI frameworks like **Angular** Material and **PrimeNG** to enhance the user interface and user experience.
* Successfully migrated existing **Java** applications from earlier versions to **Java** **17**, ensuring backward compatibility and seamless integration.
* Integrated **Spring Boot** **Microservices** with **Spring** **Cloud** **Sleuth** and **Zipkin** for distributed tracing and monitoring to track and debug requestsacross services.
* Employed **RxJS** observables and operators to handle asynchronous data streams and perform operations like mapping, filtering, and combining.
* Documenting design decisions, implementation details, and usage instructions for **Java 8** components, **APIs**, and frameworks.
* Implemented database interactions with **JDBC**, handling data access operations and connection management.
* Managed **change** **requests** and ensured proper **documentation** and approval using tools like **ServiceNow**, enabling effective communication and coordination of changes.
* Utilized **Angular** **HttpClient** module to make **HTTP** requests to backend APIs and handle responses in a consistent and efficient manner.
* Created web services using **Spring** **REST** Controllers to return **JSON** to the frontend.
* Utilized **JSON** **Web** **Tokens** (**JWT**) for stateless authentication and authorization between **Angular** frontend and backend services.
* worked with **Application** **Lifecycle** **Management** (**ALM**) tools to manage the entire lifecycle of software applications, from conception to retirement.
* Created **Angular 12** projects with strict mode enabled, promoting better coding practices and enhanced type safety.
* Developed and implemented rule-based systems using **Drools and Java 11**.
* **Core Java** with Concurrent **API** is used extensively for parallel processing and chunk processing.
* Updated the legacy application by building new components with new features of **Angular**.
* Implemented centralized configuration management for **Microservices** using **Spring** **Cloud** **Config Server**, enabling dynamic configuration updates without service **RESTarts**.
* Integrated **Kafka** with real-time data processing frameworks such as **Apache Flink** and **Spark** Streaming.
* Extensive experience in developing Java Enterprise Edition (**J2EE**) applications, employing best practices to build robust and high-performance enterprise solutions.
* Utilized **Elasticsearch** for various use cases, including application search, log aggregation, real-time analytics, and business intelligence.
* Designed **relational** **database** schemas, focusing on normalization, data integrity, and optimized query performance.
* Worked on **Spring** **IOC** to inject the services and **dependency** **injection** mechanism and create database objects such as **Triggers**, **Functions**, **PL/SQL packages, Sybase SQL server**, and **MongoDB**.
* Utilized **Kotlin programming language alongside Java to develop microservices, leveraging Kotlin's concise syntax and enhanced null safety features.**
* Created dynamic web pages with **JavaServer** **Pages**, integrating back-end logic with user interfaces.
* Scripting and setup **of CI/CD pipeline to deploy and execute ETL utilities on AWS.**
* Implemented automated scaling policies and managed **OpenSearch** shards for optimal distribution and balancing.
* Created Mongoose schema to perform **CRUD** operations using **MongoDB** database and communicated with a remote server using **REST**-enabled web service on the **Jersey** framework.
* Used **Selenium** to automate testing tasks, ensuring web applications work as expected.
* Integrated **Auth0** for authorization and authentication of users and API calls from different applications.
* Used **Maven** and private **JFrog** **Artifactory** for project builds.
* Ability to write high-performance **Java** **11** code by optimizing algorithms, data structures, and resource utilization.
* Created and executed **JMeter** scripts for performance **Testing** of the application.
* Experience using **Splunk** software for searching logs and reporting.
* Utilized **Spring** **Cloud** **Task** and **Spring** **Batch** for batch processing and scheduling tasks within **Microservices**, enabling efficient data processing and management.
* Performed unit **Testing** with **Junit** and tested using **Swagger**-**UI** and **Postman** for API **Testing**.
* Created **AWS** **Security** **Groups** for deploying and configuring **AWS** **EC2** instances and used various services such as **S3**, **Kinesis**, **VPC**, **Quick** sight.
* Integrated **Apigee** with **CI**/**CD** pipelines, automating API deployment, versioning, and rollback processes for efficient **API** management.
* Configured **Tomcat** server settings for optimal performance and scalability.
* Wrote **Docker** files for creating **Docker** images for **Microservices** and **Docker** containers for environment, used **Kubernetes** to deploy **Docker** containers.
* Involved in **PCF** for developing a module using **Spring Boot** for improving time and efficiency of the process.
* Worked with **Jenkins** for continuous integration, **JIRA** for project management.
* Utilized **Git** **Hub** (**Git** **Bash** & **Git** **Kraken**) for the purpose of version control.

**Environment**: Java 8, Java 11, Java 17, Angular 15, TypeScript Kotlin, Spring, Spring Boot, Core Java, Microservices, Spring Cloud, Spring Batch, Spring Security, HTML5, CSS3, JavaScript, Angular, NodeJS, MongoDB, SQL server, Kafka, Jenkins, PL/SQL, Git, Splunk, AWS (EC2, Kinesis, S3, IAM, VPC, Docker, Kubernetes), Maven, Selenium, APIGEE, Spark, Postman, REST, JSON, Tomcat, Swagger-UI, Jfrog, Junit, JMeter, JIRA.

**Client: Cigna Healthcare, CT April 2021 to Oct 2022**

**Role: Sr. Full Stack Developer**

**Responsibilities:**

* Developed scalable **Java** **8** and **Java** **11** applications focusing on **Lambda** Expressions, **Streams**, and Spring Boot microservices, employing Spring Framework in **MVC** design for backend operations, and adhering to **TDD** principles.
* Developed responsive UI Webpages using **HTML**5, **CSS3**, **JavaScript**, **jQuery**, and **Angular**.
* Implemented HTTP client module in **Angular** to communicate with backend **RESTful** APIs and fetch data asynchronously.
* Worked on **Spring Boot**, **Spring** **MVC**, **Spring** **REST** full template to develop Application-Level **Micro** **Services**.
* Integrated **Angular** with state management libraries like **NgRx** for managing complex application states and data flow.
* Employed **Java** **1.8** features like **Lambda** expressions and **Stream** API for Bulk data operations on Collections to improve Application performance.
* Worked on **Spring** **MVC** Transaction Management, **Spring**/**Hibernate** Batch Transactions and **Hibernate** **cache**, **Spring Boot** components to implement business layer and navigation part of application.
* Utilized Executor Service Interfaces, Thread Pool Executor class for **multi-threading** & **thread resource pooling.**
* Monitored health checks using **Spring Boot** **Actuator** and **Swagger** UI for registered **Microservices**' APIs.
* Implemented service discovery and load balancing using tools like **Netflix** **Eureka** and **Ribbon** to enable dynamic service registration and routing.
* Employed **Spring** **Cloud** **Circuit** **Breaker** (Hystrix) to isolate and control failures in distributed systems, ensuring resilience and fault tolerance.
* Utilized **Eureka's** health checks to monitor the status of microservices and take corrective actions when needed.
* Utilized **Java 11**'s HTTP Client API to build robust and responsive applications.
* Worked with **XML** technologies for schema definition, querying, and automated **XML** generation.
* Implemented asynchronous communication between **Microservices** using messaging systems like **ActiveMQ** for event-driven architectures.
* Developed asynchronous messaging capabilities in **Microservices** using **Spring** **Cloud** **Stream** and **Apache** **Kafka** for event-driven architectures.
* Integrated **Spring Boot** with **Spring** **Cloud** Netflix to simplify the development of **Microservices**, leveraging features like centralized configuration management and client-side load balancing.
* Configured **ActiveMQ** for persistent message storage and message durability to ensure message delivery in case of failures.
* Used **ETL** tools to transform and load data into **relational** databases, maintaining **data** **quality** and consistency.
* Implemented SSH port forwarding (tunneling) with **PuTTY** to securely access remote services and bypass network restrictions.
* Developed microservices using **TypeScript** and **Node**.**js**, focusing on modularity and reusability of components.
* Used **IBM's** Rational Application Developer for building and debugging enterprise applications.
* Developed a large-scale **Angular 12** project, utilizing the latest features and enhancements of the framework.
* Created **REST**-based Web Services with **HATEOAS** links from **Spring**-based Applications.
* Extensive experience in scripting with **Shell, Bash, and** **Python** for automation and task simplification on Unix/Linux platforms.
* Experienced in end-to-end development processes, integrating requirements, development, testing, deployment, and support within **ALM** frameworks.
* Integrated **Selenium** with frameworks like JUnit or TestNG to organize and manage tests effectively.
* **Used Kibana for distributed logging mechanism and tracing the system failures.**
* Established architecture and maintained **RESTful** **API** using **Spring Boot**.
* Used **ETL** (Extract, Transform, Load) tools like **Mulesoft**, **StreamSets**, or Databricks to integrate, transform, and process large datasets for analysis.
* Successfully managed data warehousing projects, resulting in improved **dasta accessibility and insights**. Collaborated with cross-functional teams to integrate **Big Query** into data-driven decision-making processes
* Designed and deployed containerized applications on **Fargate** for scalability and cost-efficiency.
* Used **WebSphere Application Server** for deploying various components of application.
* Tested **REST** **API** services using **REST** client tools like **POSTMAN** and **SOAPUI**.
* Wrote **DAO** layer, required **SQL** queries and **Spring** Controller to handle data and request to **Spring** application.
* Developed various **SQL** queries and **PL / SQL** Procedures in **Oracle** **DB** for the application.
* Worked on instantiating multiple **AWS** **EC2** servers by using **YAML** **Script** in Ansible playbook. Experienced DevOps methodologies by implementing **CI/CD** Pipelines using tool sets like **Jenkins**.
* Used **JMeter** for load **Testing** to measure the performance of **REST** services.
* Orchestrated web applications from scratch using **Angular 12** to develop single-page applications.
* Managed **AWS EC2** for scalable web applications and **S3** for secure storage, integrating **AWS Lambda** with API Gateway for serverless **Python** and **Node.js** scripts.
* Built on-premise data pipelines using **Kafka** and **Spark** streaming using the feed from API streaming Gateway **REST** service.
* Developed unit tests and integration tests for **TypeScript** applications using testing frameworks like Jest or Mocha, ensuring code quality and reliability.
* Integrated **AWS** **OpenSearch** with **Amazon** **CloudWatch** for monitoring cluster health, resource utilization, and performance metrics.
* Implemented messaging patterns using **IBM** **MQSeries** for reliable communication in distributed systems.
* Used **REST** client - **POSTMAN** and **SOAPUI** tools to test **REST** API services.
* Developed **MongoDB** Data Access Layer implementing Mongo Repository interface from **Spring** **Data**.
* Implemented **RESTful** web services to retrieve data from client side and made **RESTAPI** calls from **Angular** and parsed the data to project client and handled the security and Authorization using **Spring** **Security** &**OAuth2**.
* Unit Testing **Java** code using **Junit** **4**, **Mockito** and **Power** **Mock**
* **EJB’s** were running under the hood of **WebSphere container/server.**
* **Jenkins** is Built on **Docker** container and the master controller **Kubernetes** controls pods.
* Hands-on experience working with build plugins like **Maven**, and **Apache** **Ant**.
* Wrote **Drool** rules in both rule flow and **drools** for fetching the facts from the database.
* Used Build and Deploy Plan Spec templates for deployments and environment configurations in **EC2** **Clusters**
* Deployed Java web applications on **Apache** **Tomcat** server for production environments.
* Worked on **RESTful** web services using **Jersey**, **Spring** **framework**, and **JAX-RS**.
* Managed code versioning with **GitHub** and deployment to staging and production servers.
* Utilized **AWS** **Lambda** platform to upload data into **AWS** **S3** buckets and to trigger other **Lambda** functions.
* Configured **CloudWatch** monitoring and logging for **AWS** resources.
* Implemented **MongoDB** Data Access Layer with **Spring** Data's Mongo Repository interface.
* Implemented **IAM** policies and roles to manage user access and permissions on cloud platforms.
* Used **Log4j** utility to generate run-time logs and used **Splunk** for applications logging request and response.
* Used **JBoss** for troubleshoot issues.

**Environment:**Java 8, Java 11, Python, Angular, TypeScript, NodeJS, Git, Spring, Spring Boot, Spring MVC, Spring Cloud, Microservices, Lambda, Kibana, Websphere, Spark, HTML5, CSS3, JavaScript, jQuery, Junit, JBoss, MongoDB, Oracle, Big query Hibernate, Drools, Dockers, REST, Apache Tomcat, Jersey, Log4j , Splunk, JAX-RS, OAuth2, Eureka server, Postman, SOAP, Kubernetes, Mockito, Maven, Selenium, Jenkins, Apache Kafka, Swagger-UI, Eureka server, AWS(IAM), JMeter.

**Client: State of Colorado, CO Jan 2020 to Mar 2021**

**Role: Full Stack Developer**

**Responsibilities:**

* Designed and developed dynamic, multi-browser compatible, responsive web applications using **HTML**5, **CSS3**, **Bootstrap**, and **ReactJS**.
* Utilized **React** Router for client-side routing to enable navigation between different views.
* Developed **Java** **EE** components using **Spring Boot**, **Spring** **IOC**, **Spring** **Transactions**, **Spring** **Security**, **Spring** **Data**, and **AOP**.
* Integrated **Redux** for state management, enabling efficient data flow and centralized state management.
* Implemented **RESTful** API calls using Axios or Fetch to fetch and update data from backend services.
* Leveraged **React** **Hooks** for managing component state and side effects, improving code readability and maintainability.
* Developed reusable UI components and libraries to promote code reusability and maintainability.
* Utilized the **Spring Boot** for rapid development of cloud-based **Microservices** with minimal configuration.
* Implemented **Hibernate** **ORM** for **Java** class and database table mapping.
* Used **Java** **Persistence** **API** (**JPA**) along with **Hibernate** framework for test class **Implementation**.
* Designed and managed data warehousing solutions using **Oracle** databases for reporting and analytics.
* Serialized **Java** objects to **XML** using **JAXB** for data storage and interchange.
* Integrated **JSON** data into web applications, databases, and APIs for data processing.
* Worked with **Spring** features like **Spring** **DAO**, **Spring Boot**, and **Spring** **Batch**.
* Managed **Kubernetes** clusters, including version upgrades and security patch applications.
* Designed and developed custom message adapter components using **Apache** **Kafka** for message routing through **Microservices**.
* Worked with business intelligence tools like **Tableau** **CRM**, **Cognos**, or **Power** **BI** to create data visualizations and dashboards for business insights.
* Migrated existing applications to **REST**-based **Microservices** using **Spring Boot** for CRUD capabilities.
* Optimizing **MongoDB** CRUD Operations.
* Configured **OAuth** **2.0** Implicit Flow for client-side applications to secure access to protected resources.
* Implemented authentication and authorization features using **JWT** authentication and **React** Context API for managing user sessions.
* Involved in Unit testing of application using JUnit and built the package using Maven and deployed the application on to **WebLogic App Server.**
* Managed **ZooKeeper** configuration files and parameters to optimize performance, reliability, and security of the **ZooKeeper** ensemble.
* Configured **API** Gateway policies for authentication, authorization, and rate limiting.
* Deployed and managed applications using OpenShift.
* Developed data access layer using **Spring** **Data** **JPA** for streamlined database operations.
* Troubleshooted and resolved **Tomcat** server issues to ensure uninterrupted service.
* Developed backend services using **Python** and **Django** to power a real-time data analysis platform.
* Utilized **Pandas** and **NumPy** for real-time data manipulation and statistical analysis.
* Configured **Kafka** clusters for topic management across environments.
* Implemented Selenium Web Drivers and Protractor for **UI** **Testing**.
* Used Continuous Delivery / Continuous Integration (CD/CI) Tools **Docker**, **Jenkins**.
* Used **GIT** for version control and **Eclipse** **IDE** for project development.
* Used **Spring** **Dependency** **Injection**, Annotations, and Components to implement the Business Layer.
* Automated repetitive tasks using **GitHub** Actions workflows.
* Deployed applications on **Amazon Web Services** (**AWS**) for scalable solutions.

**Environment**: Java, J2EE, Python, ReactJS, Spring Boot, Microservices, API, Spring Security, Hibernate, JAX-RS, Weblogic, Apache Tomcat, Zookeeper, Apache Kafka, Scala, Mongo, Oracle, AWS, Selenium, Docker, Jenkins, Kubernetes, GIT, Eclipse, REST, JSON, Maven, Junit, Spring Tool Suite.

**Client:** **AT&T, TX. Oct 2018 to Dec 2019**

**Role: Full Stack Developer**

**Responsibilities:**

* Actively participated in Analysis, Design, Development, System Testing, and User Acceptance Testing, adhering to **Agile** and **SCRUM** Methodologies.
* Engineered **Microservices** with **Spring** and conducted thorough **Testing** leveraging **Spring Boot**.
* Enhanced application performance through **Angular** **Lazy** **Loading** and ahead-of-time Compilation techniques. Leveraged **Angular** extensively for Databinding, crafting Controllers, Directives, and Filters, seamlessly integrating with Backend Services.
* Execute serverless functions with **AWS** **Lambda**, integrating with **S3** for event-driven processing.
* Employed **Java 8** Stream API and Method References for method pointing and harnessed Functional Interfaces.
* Contributed to the **Implementation** of **SOAP** Services using **Axis**-**2** and **Apache** **CXF** Framework, alongside **RESTful** Services with **WSDL**, **Swagger**, and **WAD.**
* Implemented dynamic filtering for **RESTful** Web Services using **Spring Boot** and structured projects based on **Spring** **MVC** Pattern with **Spring Boot**.
* Responsible for migrating existing modules in **Rabbit MQ** to Apache Kafka and worked on creating Kafka adaptors for decoupling the application dependency.
* Crafted and implemented a Single Page User Interface for the Client Service module using **HTML**5, **CSS3**, **Angular**.
* Employed Jackson API for seamless **Java** Object to/from **JSON** conversion.
* Utilized **Spring** **Core** Annotations for **Dependency** **Injection**, **Spring** **MVC** for **REST** **APIs**, and **Spring Boot** for **Microservices** development.
* Developed and executed unit tests and integration tests using Jest and **Selenium**, adhering to TDD methodologies.
* Utilized **Eclipse** **RCP** for creating **POJOs, Servlets**, and **Data Access Objects**.
* Manage configuration of **Web App and Deploy to AWS cloud server through Chef.**
* Configured **AWS** **Oracle** **RDS** Databases for projects and managed data migration utilizing Data Pump.
* Proficient in **RDBMS** **Implementation** and development with **Oracle**, **MYSQL**, **MSSQL**, and **DB2**.
* Expertise in Unit Testing with **Junit**, **Jasmine**, **Mockito**, and **Karma**.
* Leveraged **Spring** **ORM** with **Hibernate**, taking advantage of Annotation Metadata, Auto Wiring, and Collections for DAO layer Implementation.
* Designed, Configured, and deployed **Amazon** **Web** **Services** utilizing **AWS** **API Gateway** and **ELK** **Stack**.
* Utilized **GIT** for Version Control, **JIRA** for Bug Tracking and Project Management, and **Maven** as Build Tools.
* Configured and customized **Maven** plugins (e.g., Surefire, Failsafe) for Testing and reporting.
* Utilized **Spring** Framework with **Hibernate** for **Oracle** mapping leveraging **Hibernate** Annotations.
* Configured **CI**/**CD** processes in **Jenkins** to streamline **Microservices** deployment.
* Managed sensitive data and configurations securely using **OpenShift's** **Secrets** and **ConfigMaps**.
* Spearheaded the **Microservices** Architecture **Implementation** with **Spring Boot** and **Apache** **Kafka** Message Brokers.
* Employed **GitHub** security features for vulnerability scanning and resolution in project dependencies.

**Environment:** HTML5, CSS3, JavaScript, Angular JS, JSON, Java 8, Maven, API, Hibernate, Kafka, Rabbit MQ, Spring, Oracle, SQL, AWS, Spring MVC, MYSQL, Eclipse, Junit/Mockito, Karma, SOAP, REST, Selenium, Jenkins, Openshift, Swagger, GitHub, JIRA.

**Client: Zensar Technologies, Pune, India. July 2012 to August 2017**

**Role: Java Developer**

**Responsibilities:**

* Involved in various phases of Software Development Life Cycle (**SDLC**) of the application like Requirement gathering, Design, Analysis and Code development.
* Developed a prototype of the application and demonstrated to business users to verify the application functionality.
* Created **Hibernate** **XML** files and **Java** class files to map the object relation Mappings.
* Developing front end of application using **HTML**, **CSS**, **backbone.js**, **JavaScript**, **jQuery**.
* Used **Angular** **js** framework where data from backend is stored in model and populated it to UI.
* Extensive experience in producing/consuming **REST** based web services using **JAX-RS, Jersey** providers.
* Developed the administrative UI using **Angular**.**js** and **Node**.**js**.
* Developed and implemented the **MVC** Architectural Pattern using **Struts** Framework including **JSP, Servlets, EJB, Form Bean** and **Action classes.**
* Hands on experience with **MVC** **JavaScript** frameworks such as **Backbone**.**js**, **Angular**.**js** and **Node**.**js**.
* Developed and designed Interfaces using **Swing**.
* Developed the web layer using **Spring** **MVC** framework.
* Worked on a code generation framework that generated **Java** **Swing** code from **XML**.
* Developed web services in **Java**. Experienced with **SOAP**/**WSDL**. Have also designed and developed **REST** interfaces in **Java**.
* Implemented Core Java concepts such as Multithreading, File Handling I/O, Exception Handling, Collection APIs to add various features and enhancements.
* Managed application server environments with **WebLogic**, optimizing for performance and reliability.
* Integrated **Spring** **DAO** for data access using **Hibernate** used **HQL** and **SQL** for querying databases.
* Implemented **EJBs** Session beans for business logic.
* Orchestrated deployment processes on **Azure**, optimizing application performance and scalability.
* Collaborated with cross-functional teams for seamless integration with **Azure** services.
* Utilized **Bitbucket** pipelines for continuous integration and continuous deployment (**CI**/**CD**) processes.
* Wrote stored procedures, triggers, and cursors using **Oracle** **PL/SQL.**
* Wrote **Junit** Test cases for performing unit **Testing**.
* Used **ANT** automated build scripts to compile and package the application.
* Used **Jira** for bug tracking and project management.

**Environment:** Core Java, JSP, MVC, Struts, Hibernate, Angular, NodeJS, Spring, Azure, XML, Jersey, JAX-RX, PL/SQL Developer, Oracle, Weblogic, ANT, Junit, REST, Web services, SOAP, Jira.