**Sangharsha Dhital**

**Full Stack Java Developer**

sdhital428@gmail.com| **Phone:** 281-940-8061 | www.linkedin.com/in/sdhital1

**PROFESSIONAL SUMMARY:**

* Overall 8+ year experience in the development and deployment of multi-tiered web-based enterprise applications using J2EE technologies like Servlets, JSP, Struts, JMS, Hibernate, Spring, JDBC, React.js, Java Beans and Web services (SOAP, REST).
* Experienced in web development using HTML5, DHTML, XHTML, CSS3, JavaScript, Bootstrap, React, jQuery, and AJAX technologies.
* Experience with Core concepts like **Collections Framework, Exception Handling, I/O Systems, Multi-Threading**, and **Data structures**.
* Experience with **Spring Core, Spring batch, Spring Security, Spring boot, Spring MVC,** and **Spring’s integration** with **Hibernate** and **Struts**technologies.
* Strong experience with J2EE Design Patterns like **MVC, Singleton, Session Facade, Factory, DAO, DTO, Business Delegate, Factory, Abstract Factory, and Service Locator**.
* Expertise in various open-source frameworks like **Spring Framework, Struts, Spring Security, Spring Integration, Java Server Faces (JSF),** and **hibernate**.
* Expertise in all spring framework modules such as **Spring IOC, AOP, Core, Messaging, ORM, and MVC architecture.**
* Expertise in the implementation of Java8 concepts like **Streams API**, **Lambda functions**, and Functional **I**nterfaces.
* Experienced with creating end-to-end web applications using Spring Boot **Microservices**, RESTful APIs, **Spring Security** API, **Spring Data JPA/Hibernate, and ReactJS.**
* Experience in consuming and producing using **Spring REST, and SOAP.**
* Experience in **React JS**and working with **React** **Flux** **architecture**. Experienced in working with **Redux** **architecture**.
* Good experience building **SPA** (Single Page Applications) using reusable and customized React.js Components react routes and working with Life-cycle **hooks** and **React**-**Redux** architecture.
* Experience in developing **Microservices** using **Spring** **Boot**, **Jersey**, & **Swagger** to communicate through **HTTP** protocol for implementing **SOA** approach and web services.
* Experience in working on various spring modules (**Spring** **MVC**, **Data**, **Boot**) along with Hibernate as the back end ORM tool for the implementation of the persistence layer and mapping of **POJOs**.
* Good Knowledge of the Authorization using Json Web Tokens and **OAuth2**.**0**.
* Profound knowledge in configuring and deploying web applications in **Tomcat Server, Web Logic, Web Sphere,** and **JBoss**
* Experience in **AWS EC2, S3, Elastic Beanstalk, Elastic Search, and Lambda.**

**Technical Skills:**

|  |  |
| --- | --- |
| **Programming** | Java, JavaScript, SQL, HTML5, CSS3, XML, ReactJS, JSON,¥r |
| **Core Java** | JDBC, I/O, Collections, Exceptions, Multi-Threading, Lambda Expressions, Streams, Functional Programming |
| **Frameworks/ Tools** | Spring Core, Spring Boot, Spring AOP, Spring MVC, Spring Security, Spring JPA |
| **ORM Frameworks** | Hibernate, JDBC |
| **Web Development Tools** | React, JSP, Servlets, JavaScriptHTML5, CSS, jQuery, XML |
| **Databases** | SQL, Oracle, MySQL |
| **Web Services** | Microservices, APIs |
| **Build Tools** | Maven, Jenkins, Docker |
|  | Tomcat Server, Web Logic, Web Sphere, and JBoss |
| **IDEs and tools** | IntelliJ, VS Code |
| **Version and Source Control** | Git, SVN |
| **Logging/ Debugging Bug Tracking** | Log4j |
| **Cloud Technologies** | AWS |

**PROFESSIONAL EXPERIENCE:**

**Met Life, Dallas, TX Jun 2021 - Present**

**Full Stack Java Developer**

**Responsibilities:**

* Developed presentation layer using **JSP**, **HTML5**, **CSS3**, **React**, **JavaScript**, and **Bootstrap** to improve responsive designs, manage the layout of the user interface, and build dynamic web visual effects.
* Implemented **J2EE** design patterns such as **MVC** (Model View Controller), **DAO** (Data Access Object), **DTO**, **Front** **Controller**, and **Singleton**.
* Designed and implemented multi-tier applications using web-based technologies like **Spring** **MVC** and Spring Boot.
* Developed the **DAO** layer for the existing applications using Spring, and Hibernate and developed various business logic and reports using **HQL** and Hibernate Criteria
* Developed the application using **J2EE** architecture and was involved in developing **JSP** forms.
* Implemented **Spring MVC, Spring Boot, Transactions, Spring JDBC** template, and JSON features.
* Implemented Spring **Dependency** **Injection** feature of **spring** **framework** to inject beans into User Interface and **AOP** for logging.
* Used features like concurrency, **Stream**, **Filters** **API** and **Lambda** functions, and Functional interface **Java** **8** programming.
* Used **Hibernate** for mapping Java classes with database and created queries with **HQL** (Hibernate Query Language).
* Secured the API by implementing **Oauth2** token-based authentication/authorization scheme using Spring security.
* Implemented **DAO** pattern, performed **CRUD** operations, and O**/**R mapping using Hibernate and **JPA**.
* Created interactive React UI’s using One-way data flow, **Virtual DOM, JSX, and React Native** concepts.
* Worked on Lifecycle Methods, **State**, **Props**, and **Events** to create interactive and **stateful** **React** **Components**, achieved routing using **react**-**router**.
* Developed interfaces with external systems using **REST** web services, **Spring** **Boot,** and **Hibernate** framework.
* Developed **REST** exception handling controller advice to respond to bad requests and server-side exceptions
* Developed REST APIs using **Spring MVC** and **Spring boot**, hosted microservices on **AWS**, and used **elastic Beanstalk** to set up applications and configured environments.
* Utilized the **AWS** platform to upload data into AWS S3 buckets, create EC2 instances and deploy the Dockerized container services.
* Used **GIT** and **SVN** for version control and **Jenkins** to build and deploy the application.

**Bank of America, Irving, TX May 2015 – April 2021**

**Full Stack Java Developer**

**Responsibilities:**

* Responsible for designing the layers and components of the project using **OOAD** methodologies and standard **J2EE** patterns & guidelines.
* Created and maintained the configuration of the **Spring Application Framework** (**IOC**) and implemented business logic using **EJB**.
* Implement business components using spring **dependency injection** and accessed them using Spring-provided **annotations**.
* Developed screens, Controller classes, business services, and **DAO** layers respective to the modules.
* Used various **Core** **Java** concepts such as **Multithreading**, Exception Handling, and Collection APIs to implement various features and enhancements.
* Involved in the integration of spring for implementing Dependency Injection (**DI/IOC**).
* Developed Business Logic using Plain Old Java Objects (**POJOs**).
* Used Spring Core Annotations for Dependency Injection **Spring DI and Spring MVC** for **REST** **API** and **Spring Boot** for microservices.
* Oversaw migration of existing applications into **Microservices** Architecture using Rest APIs, **Spring** **Boot**, Spring Cloud, and **AWS**
* Responsible for implementing Single page applications using **React JS**along with custom directives and components, UI routing, and filters.
* Worked in using React JS components, **Forms, Events, Keys, Routers,**and **Animation** concepts.
* Built the reusable class components using **React** **Redux** based on the designs. Used Fetch for making a server request.
* Developed **Hibernate** with **Spring** **Integration** as the data abstraction to interact with the database and used dependency injection (**DI**) using Spring framework and annotations.
* Experience in configuring and deploying web applications in **Tomcat Server, Web Logic, Web Sphere,** and **JBoss**
* Developed API for using **AWS** **Lambda** to manage the Serverless architecture and run the code in AWS.
* Deployed and developed the project using **AWS** EC2, S3 Buckets, Data pipeline, Elastic Beanstalk, Lambda, Amazon messaging services (**SQS**), and Amazon Kinesis Services
* Used **log4j** to log the error messages to the log files.