**Sandeep Kumar Bainddla**

**[Sandeepkumar.b41992@gmail.com](mailto:Sandeepkumar.b41992@gmail.com)**

**515-850-2062**

**JAVA FULL STACK DEVELOPER**

**SUMMARY**

* Senior Java Full Stack Developer with over **5+ years** of software development experience in **Java/ J2EE**, **Spring Boot, Spring MVC, JSP, C, C++, JavaScript, SQL/PL/SQL, Struts, Hibernate, iBatis, Web Services**. Involved in complete **SDLC**from inception, planning, design, coding and testing to project closure mainly in the intranet, Internet and standalone application design, development, maintenance and production support in n-tier architecture.
* Designed & developed several multi-tierWeb based, Standalone, Distributed, Client-Server and Multithreaded applications using **OOAD concepts and Service Oriented Architecture** (**SOA**) mostly in **cross-platform** environments.
* Extensive experience in design and development of J2EE applications using **Core Java, Java collection, JDBC, Servlets 2.4/2.5/3.0, JSP 2.0/2.1/2.2, Struts 1.2/2.x, JSF, iBatis, JMS, JSF My Faces, JPA, JNDI, JSTL, JTA, JAX-WS, JAX\_RS, Spring 2.5/3.0/3.1/4.x** and**Hibernate 2.0/3.0/3.x.**
* In-depth understanding of various Design Patterns like **MVC, Builder, Singleton, Service Facade, Service Locator, Data Access Object, Data Transfer Object/Value Object, Business Delegate, Decorator** and **Factory.**
* Experience in Servers **IBM Web Sphere 6.x/7.x**, **Oracle Web logic 9/10/11g, JBoss 6.0/7/7.1, Apache Tomcat server 6.x/7.x/8.x, XAMPP 1.7.x/1.8.x** for development and deployment of web applications.
* Hands-on experience using deployment scripts in **Ant** for deploying J2EE application in **WebLogic, WebSphere, JBoss application servers and Tomcat web server.**
* Designed the application by implementing **JSF** Framework based on **MVC** Architecture, simple Java Beans as a Model, **JSP** and **JSF** UI Components as View and Faces Servlet as a **Controller**.
* Experience in implementing **Spring Core, AOP, JDBC, MVC** and **ORM** modules.
* Expertise with employment of **Spring Kafka** and **Zookeeper** apart from **JMS** as messaging service.
* Strong Application Integration Experience using **Mule ESB** with **Connectors**, **Transformations**, **Routing**, **ActiveMQ**, **WebSphere**, **JMS** and **RabbitMQ.**
* Diverse experience utilizing tools in N-tier and Microservices architecture applications using **Spring Boot**, **Spring Cloud config**, **Netflix OSS components, PCF, AWS, Rabbit MQ, Kafka, Zookeeper, Cassandra, My SQL, Restful Web Services**.
* Worked in **Jenkins** for continuous integration of the application and System.
* Development of application server part and writing Groovy scripts for Jenkins.
* Extensive work experience on **RDBMS** concepts and worked with **MySQL, Sybase, Oracle 9i/10g/11g** and **DB2.**
* Proficient in **SQL, PL/SQL** and wrote **Stored Procedures, Triggers**besides writing **DDL**, **DML and** transaction queries with development tools like **Toad.**
* Micro Java benchmark
* Strong experience with **MongoDB** development (reports, schema design, map reduce functions) and migrations from **SQL** relational databases to **NoSQL** databases like **MongoDB**, **Cassandra** and **CouchDB**.
* Adept in building highly scalable, pluggable and interactive web applications with open source web frameworks like **Struts, Spring MVC** integrating it with Spring and Hibernate.
* Experience in developing persistence layer applications based on **ORM**between Java and Database using **Hibernate**.
* Extensive experience in **AWS** infrastructure architecture and build out.
* Worked on (AWS) infrastructure with automation and configuration management tools such as Chef and Puppet.
* Good knowledge of AWS, Amazon Cloud Services like Elastic Compute Cloud (EC2).
* Expertise in Client Scripting language and server-side scripting languages like **HTML, DHTML, CSS, JavaScript, JSTL, AJAX, jQuery, ExtJS, JSON, Dojo, bootstrap, AngularJs and node.js.**
* Good knowledge in **Angular 2.0** directives to create reusable elements and attributes, along with easy to test and maintain client-side interactivity in web applications.
* Experience in JUnit framework with **Mockito** and **Easy mock, Spring Junit Runner.**
* Experience in **Agile** software development process, **Test Driven Development** and **Scrum**.

**TECHNICAL SKILLS:**

|  |  |
| --- | --- |
| Java/J2EE Technologies | **Servlets, JSP, JSTL, JDBC, JMS, JNDI, EJB, JPA** |
| Programming Languages | **Java 8.0/7.0/6.0/5.0, PL/SQL** |
| Application/Web Servers | **WebSphere, Jenkins, JBoss, Tomcat 8, IBM WebSphere 8.x/9.x** |
| Frameworks | **Struts 2.x/1.x, Spring 4.3, Hibernate 3.0/4.x, JSF 1.2/2.0, iBatis** |
| IDEs | **Eclipse, IntelliJ 2017.1, NetBeans 8.0.2, Spring Tool Suite (STS) 3.8.3** |
| Web technologies | **JavaScript 1.8, jQuery 2.11, AJAX, XML 2.2, HTML 5, DHTML, CSS 3, CSS 4, Angular JS 1.0/1.5/2.0, Node JS** |
| Web Services | **JAX-WS, JAX-RPC, JAX-RS, SOAP 1.2, WSDL, UDDI, REST 2.2, JAXB, Apache Axis** |
| Methodologies | **Agile, Scrum, TDD(Test-Driven-Development), OOAD, SDLC** |
| Modeling Tools | **UML, Rational Rose** |
| Testing technologies/tools | **JUnit 4.12, JIRA 7.0, Selenium** |
| Database Servers | **Oracle 9i/10g/11g, DB2, SQL Server 2000/2005/2008, MySQL** |
| Version Control | **GIT, CVS, SVN, Rational ClearCase** |
| Build Tools | **ANT, Maven 3.3.9, Gradle 3.4.1** |
| Platforms | **Windows, Mac, LINUX, UNIX** |

**PROFESSIONAL EXPERIENCE**

**Client: Quest diagnostics, Mason, OH April 2018 to Till now**

**Role: Sr java developer**

Quest diagnostics is a web application based on J2EE architecture. It provides web-based access to inpatient and outpatient data from the IHC Clinical Data Repository (CDR). Users can review as well as update or enter new data through CD. Many clinicians use CD daily for accessing patient data, which aids in making important clinical decisions.

**Responsibilities:**

* Involved in various **SDLC** phases like **Requirement** gathering, **Design**, **Analysis** and **Code development**.
* Used **Spring Batch** as spring batch jobs can meet the requirements of reliability, robustness, and traceability, which are essential for automatic processing of large amount of data.
* **Used PL/SQL** stored procedures for applications that needed to execute as part of a scheduling mechanisms.
* Utilized Java Swing framework to develop client-side components.
* Implemented messaging dialog using Camel and **JMS** (Active MQ implementation).
* Extensively worked on **JPA** as persistence layer, involving **EJB3.0 annotations**, synchronizing Data base tables through XML in a secondary project.
* Used **JBoss Fuse ESB** to create mediation or route message between Services.
* Used **IText** to generate **pdf** in the form of a tree.
* Analyzed the **Node.JS** server structure in legacy project, then mimicked the REST service by using Java JAX-WS API and did the corresponding configurations.
* Used **JAX-B parser** for **marshaling** and **unmarshaling**.
* **Multithreading** was used to process data in chunks to improve performance.
* Followed the best practices in coding like removing unnecessary casting, using **generics** in Hash Map, caching, **Collections** like sort, comparator, list, set and Design patterns.
* Worked with **Jboss** as application server.
* Used version control repository **SVN** (Subversion by apache) and **JIRA** for issue tracking.
* Used **Maven** as a build tool and retrieved all the dependencies required for the application.
* Wrote various **select/update/insert SQL queries** and **store procedures** for **Oracle**.
* Code reviews and refactoring do ne during the development and checklist is strictly adhered during development
* **Deployed Docker contained Spring boot micro services into Spring EC2 container service using AWS admin console**.
* Involved in the front end using JSP, JSTL and JQuery, JQuery UI, Servlets, HTML, JavaScript, JSON and AJAX.
* Used **Spring Framework** for **Dependency Injection**.
* Involved in coding for the presentation layer using **Ice faces**, **JQuery** and **JavaScript**.
* Created and deployed web pages using HTML, CSS, JSP, JavaScript and jQuery.
* Implemented the MVC architecture using backboneJS.
* Customized the **CSS** for the ace components used as per the style guide.
* Utilized Bootstrap framework according to internal standards and guidelines.
* Developed the Login, Policy and Claims Screens for customers using **AJAX, JSP, HTML, JavaScript and JQuery.**
* Developed **data tables** using ace components and implemented Sorting, Filtering and Pagination functionalities.
* Develop and implement java Batch files/Unix shell Scripts to run java jar files.
* Implemented the application using **design patterns** such as **Session Façade**, **Bean Factory** and **Singleton**.
* Wrote code to fetch data from web services using jQuery AJAX via JSON response and updating the HTML pages.
* Used Reach JS for rendering views and to update the Document object model dynamically.
* Extensively used CSS and Bootstrap for styling the HTML documents.
* Used **Bootstrap** to create Controllers to handle events triggered by clients and send request to server.
* Development of DAO layer using Hibernate Template, JDBC Template and Mongo Operations.
* Composed and executed **MongoDB** scripts to insert and update NoSQL database; automating data changes and reporting.
* Worked on **SOA Web Services** using **SOAP** with **JAX-WS** for transmission of large blocks of **XML** data over **HTTP**.
* Developed and implemented Swing, spring and J2EE based MVC (Model-View-Controller) framework for the application.
* Used **Spring JDBC** to retrieve the data and represented the data in the form of tree which has seven hierarchies and implemented drag and drop functionality to it.
* Implemented pre-authentication and data base security using **spring security**.
* Implemented crosscutting concerns using **Spring AOP**.

**Environment*:* Core Java, Java Swing, Icefaces 3.0, Spring Dependency Injection, Spring JDBC, IText, Maven, SVN, JIRA, HTML, CSS, Bootstrap, AJAX, JavaScript, BackboneJS, MarionetteJS, MongoDB, XML, JBoss, SOAP, eclipse indigo, Node.JS, React JS, PL/SQL, Oracle 10g, Unix/Linux.**

**Client: Goldman Sachs, Salt Lake City, UT April 2017 to March 2018**

**Role: Site Reliabilty Engineer**

**Description:** E-guidelines IT implements and maintains the client investment guidelines for all asset class (Fixed income, Equities, Money Market, Currencies and Commodities) and all regions (NY, LN, ASIA) using sentinel tool. Sentinel checks orders and portfolio holdings for compliance with client mandates, account regulations and management restrictions. Sentinel takes the accounts, products, credit ratings, benchmarks and market data to evaluate the rules and generate the exceptions are reviewed by Ops and compliance using Exception Management workflow tool to make sure the transactions are compliant with client guidelines

**Responsibilities:**

* Expertise in analyzing and resolving Cross-Brower compatibility issues and provides coding solutions to reusability problems.
* Developed using new features of **Java8 Annotations, Generics, enhanced for loop** and **ENUMS.**
* Used **LINQ** to **Entity Framework** using lambda expressions for communicating between business layer and database and Coded in java 1.8
* Wrote **GWT** code to create presentation layer using **GWT widgets** and **event handlers**.
* **Java8 Stream API** for Bulk Data Operations on Collections using both **streams** and **parallel streams**.
* Developed front end web applications by using web technologies such as **AngularJS**, **Html5, CSS3, JavaScript**.
* Implemented classes, interfaces and methods, constructors and objects definitions using **TypeScript**. Created multiple **TypeScript** modules to manage written codes in **Angular 4**. Registered **Angular 4** components in **Angular4** modules using decorator functions like Component, **NgModule**
* Developed Angular functionality in the application to provide two-way binding to the system
* Developed single page applications using ng-route module in **Angular 2**.
* Used customized services and directives in **AngularJS** to develop dynamic and responsive webpages.
* Built the template, pulled agent information from **JSON** object and rendered it onto the widget.
* Used TypeScript to add optional types, classes, and modules to **JavaScript**.
* Debug the application using Firebug to traverse the documents and manipulated the Nodes using DOM and its Functions.
* Developed **SOAP** and **REST** web services using **Java, Hibernate, JAX-WS, JAX-RS and JAXB.**
* Rest services templates are generated using swagger specifications.
* Following **Agile methodology (SCRUM)** during development of the project and oversee the software development in Sprints by attending daily stand-ups.
* **Java benchmarking** has been used for to measure the computing performance.
* Implemented business logic using **Spring service layer** and **Spring DAO layer using Spring Framework.**
* Configured Spring declarative transaction manager and implemented transaction management in DAO layer.
* Integrated **Spring** and **Hibernate**, configured logging using **log4j** and developed exception handling for the application.
* Wrote **Hibernate** Criteria queries for data retrievals and have performed **the ORM mappings** to map java entities to database tables.
* Used **Spring IOC** as Dependency Injection.
* Used **Microservice architecture** with **Spring Boot-based services** interacting through a combination of **REST and Apache Kafka** message brokers.
* Implementing and debugging code in groovy and rat-pack for micro services for giving flexible callback, polling mechanisms between external micro services.
* Developed Micro services using **Spring MVC, Spring Boot, Spring Cloud**, **Spring AOP, Dependency Injection.**
* Deployed **Spring Boot** based micro services **Docker** container using **Amazon EC2 container services** and **using AWS admin console.**
* Created computing and data resources on **AWS** using **EC2**.
* Utilized **AWS Lambda** platform to upload data into **AWS S3 buckets** and to trigger other Lambda functions.
* Implementation of Hystrix circuit breaker to avoid loss of messages during handling of messages at the consumer level.
* Implementation of **Zuul proxy** as the **API gateway.**
* Experience with basic **WebSphere** Administration- like **Data Source Configurations, Listener port configurations, JVM Property Configuration, Certificates Configuration, deploying EAR’s into various environments (TEST, PILOT) through Admin Console**, etc.
* Implement Persistence layer using **Hibernate** and writing **SQL** queries based on **Hibernate** criteria API.
* Used **Data Mining**for analyzing the data.
* used **Big data** for fraudulent transactions.
* Used **Oracle 11g** as a database in the backend.
* Developed **Pl-Sql** store procedures to convert the data from **Oracle** to **MongoDB.**
* **ETL** is used for data integration
* Developed **UNIX** scripts to run the file validations, automate file manipulation.
* Performed Unit testing using thru URLs, **JUnit & Selenium**and also by making calls into the system.
* Used **JSON** for serializing structured data.
* Configure **Jenkins** build jobs for Continuous Integration and for regression testing.
* Used **GIT** for version control and involved in writing the build scripts using **Maven.**
* Creating unit test strategy, preparing test environments and test cases.
* Prepared documentation for some of the recurring defects and resolutions and business comments for those defects.
* Experienced in writing **Components, Pipes** and **Services** to separate concern and handle different situations by using **Angular2** and Worked on **Angular4** Enhancements.

**Environment: Java 1.8, Java EE, Hibernate4.2, Spring4, Spring Boot, Spring MVC, JPA, TypeScript, Maven, RESTful Web services, JAX RS 1.1, Jersey 1.6, VXML, Web Services1.3, HTML5, CSS3, JavaScript, AJAX, JSON, Node.js, Angular, bootstrap, XML/XSL, Oracle 11g, JUnit, JAXB, WSDL, UDDI, GIT, Groovy, Tomcat, Log4j, SQL,Putty, AWS, Docker, Jenkins, IntelliJ, Firebug, Unix and Windows 7**.

**Client: Master card -O’Fallon, MO June 2016 to March 2017**

**Role: Senior Java/Full stack developer**

**Description:** WellThe E-service Online project is designed and developed for MasterCard Worldwide customers. The Online Reports application to provide MasterCard customers an online view into their mandatory as well as optional reports. MasterCard customer users can request access to the Online Reports tool including the ability to view/export/print specific mandatory and/or options reports. The Online Reports tool receives and loads reports into eService database. The reports are received from provider systems. This tool has billing feed process to provide accuracy & flexibility to handle different kinds of billing models

**Responsibilities:**

* Involved in various phases of **Software Development Life Cycle (SDLC)** of the application like Requirement gathering, Design, Analysis and Code development.
* Generated Use case diagrams, Class diagrams, and Sequence diagrams using **Rational Rose**.
* Developing and editing user Interface Screens using **JSP, HTML5.0, JavaScript and node.js.**
* Developed applications using MEAN**(Mongo-DB + Express.JS + Angular 4 + Node.JS**) stack and created POC's for rest service development using **Node.JS, Express.JS and MongoDB.**
* Improve presentation of webpages **CSS 3.0, Adobe Flash and JQuery plug - INS.**
* **Deployed Docker contained Spring boot micro services into Spring EC2 container service using AWS admin console**
* **Experience with building Docker Images and running them on Docker container**
* Creating customSQ, general use **modules and components,** which extend the elements and modules of core **AngularJS.**
* Used **Twitter Bootstrap** to create responsive web pages, which can displayed properly in different screen sizes.
* Developed the applications using **Java, J2EE, Struts, JDBC**
* Developed **Struts Action classes** using Struts controller component.
* Created **struts form beans, action classes, JSPs** following **Struts framework standards**.
* Used **Jackson** which is a **JSON** processor for both generation and parsing of JSON representations of Java objects
* Used display tag libraries for decoration and used display table for reports and grid designs.
* Worked on core java, using file operations to read system file (downloads) and to present on **JSP**.
* Involved in the development of underwriting process, which involves communications without side systems.
* Used **IBM Operational Decision Manager** to **designing a rule project** and **deploying rules to a runtime environment**.
* Developed enterprise reference architecture, SOA & Cloud strategy in a **SaaS** driven enterprise business model.
* **Used PL/SQL** stored procedures for applications that needed to execute as part of a scheduling mechanisms.
* Developed and debugged the **Servlets** with **JOBSS.**
* Implemented the application using **design patterns** such as **Session Façade**, **Business Delegate**, **Bean Factory, Adaptor** and**Singleton**.
* Used **ReactJS** to create views to hook up models to the DOM and synchronize data with server as a **Single**
* **Page Application (SPA).**
* Developed and implemented **Swing, spring** and J2EE based MVC (Model-View-Controller) framework for the application.
* Developed applications to asses **JSON and XML** from **Restful** web services from consumer side using **ReactJS.**
* Designing and implementing scalable, **RESTful and microservices** based mobile back-end. The backend is being written in Java using **Spring Boot** for simplicity and scalability.
* Used **Spring JDBC** to retrieve the data and represented the data in the form of tree which has seven hierarchies and implemented drag and drop functionality to it.
* Implemented pre-authentication and data base security using **spring security**.
* Used **Apache Camel** for messaging technology glue with routing. It joins together messaging start and end points allowing the transference of messages from different sources to different destinations.
* Used **Spring Batch** as spring batch jobs can meet the requirements of reliability, robustness, and traceability, which are essential for automatic processing of large amount of data.
* Used **SAX XML API** to parse the XML and populate the values for a bean.
* Used **Jasper** to generate rich content reports.
* Implemented persistence layer using **Spring JDBC** that use the **POJOs** to represent the persistence database tuples. These POJOs are serialized Java Classes that would not have the business processes.
* Assisted in planning, installation and execution of new **JRules ILOG server, application server and Team server** and developed designs and strategies for business rules and address problems including configuration of ILOG Rules.
* Involved in building a **scalable email system** using amazon simple email service, s3 for handling heavy loads.
* Worked on **Pivotal Cloud Foundry** to host **Cloud based Applications**.
* Experience in re-architecting monolithic architecture service into **microservices** based architecture using spring boot and **pivotal cloud foundry**.
* Worked on **Mongo DB** concepts such as **locking, transactions, indexes, replications, schema design.**
* Implemented multiple high-performance **Mongo DB replica set.**
* Worked with other **NoSql** and **Big data technologies** such as Cassandra and Hadoop
* **Apache Ant** was used for the entire build process.
* Used **Mockito** for unit tests development of **Test-driven Development (TDD).**
* Deployed the application in **JBOSS** production server running in **Linux platform.**
* Used **log4j** for logging errors in the application.
* Involved in **JAD** sessions to gather requirements from the end clients.
* Used **GIT** as version management, **JIRA** for defect management system
* Involved in **unit testing**, **troubleshooting** and **debugging**. Modifying existing programs with **enhancements**.
* Involved in **fixing bugs** the bugs or defects in the application.
* Successfully executing **offshore modeling** by effective and on time knowledge sharing with offshore project groups through weekly team meetings and daily status update calls.
* Worked as the part of a core team to monitor and enhance the performance of the application.
* Provide guidance and direction to team members in their day-to-day activities.

**Environment: EJB3.0, Spring, JSP1.2, Servlets2.1, Jasper reports, IBM MQ, XML, REST, JDBC, JavaScript, XSLT, XML, UML, HTML, JNDI, Python, SaaS, Apache Camel, Jackson, Spring Boot, JBoss 6.x, ClearCase, Drools, Log4J, ANT, Swagger, Mockito, MongoDB, Eclipse, Pivotal Cloud Foundry.**

**Client: CCNET – HYD Jan 2014 to Nov 2015**

**Role: Java Developer**

**Description**: Host Management is designed with multiple levels of administration including admin, reseller, end user, and email-based interfaces. These multiple levels provide security, ease of use, and flexibility for everyone from the server administrator to the email account user. WHM powers web hosting companies and organizations that have a need to automate and offer competitive hosting services.

**Responsibilities:**

* Involved in the **Requirements collection & Analysis** from the business team.
* Created the design documents with **use case diagram, class diagrams, and sequence diagrams** using **Rational Rose.**
* Implemented the MVC architecture using Apache **Struts1.2** Framework.
* Implemented **Action Classes** and **server side validations** for account activity, payment history and Transactions.
* Implemented views using **Struts tags, JSTL2.0** and **Expression Language**.
* Implemented **Tiles Framework** for the **view’s layout**.
* Implemented **session beans** to handle **business logic** for fund transfer, loan, credit card & fixed deposit modules.
* Worked with **JAXB**, **SAXP** and **XML Schema** for exporting data into XML format and importing data from XML format to data base and **JAXB** in the web service's request response data marshalling as well as marshaling process.
* Implemented**web services** client to consume the third-party service API for validating credit cards. Used **XML Web Services** using **SOAP** to transfer application that is remote and global to different financial institutions.
* Used **JMS** for reliable and asynchronous exchange of important information such as payment status report.
* Developed **Unit test cases** using **JUnit.**
* Developed **Ant scripts** and developed builds using **Apache ANT.**
* Used **Log4J** to capture the log that includes runtime exceptions.
* Used Clear **Case** for **source code maintenance**.

**Environment: J2EE, Java, Struts, Tiles, JSP, JNDI, Clear Case, SOAP, WSDL, UDDI, JAXB, JAXP, Schema (XST), EJB 2.0, ANT, JavaScript, JMS, Rational Rose, WSAD, DB 2.**