Full Stack Java Developer Full Stack Java Developer Full Stack Java Developer - LexisNexis Risk Solutions Java/J2EE Programmer with 5years of IT experience in Java Application Development, Distributed Application Development, and Object Oriented Programming, developing, testing and implementation. Expertise in JEE Technologies-Spring, Servlets, JSP, JDBC, Hibernate. Used Multithreading to improve the performance/scalability of the applications. Experience in J2EE Design Patterns such as MVC, Business Delegate, Service Locator, Singleton, Transfer Object, Singleton, Session Fa ade, Data Access Object. Expertise in UML, XML (SAX, DOM, JAXB) and JavaScript. Strong Experience in SOA implementations - SOAP (JAX-WS), REST (JAX-RS). Familiar with all aspects of technology projects including Business Requirements, Design Specification, Design Patterns and Deployment. Experience in developing, deploying enterprise applications on IBM WebSphere, BEA WebLogic, Oracle Application Server, JBoss, and Tomcat. Extensive experience working on databases: Sybase, Oracle, DB2, SQL Server, MySQL. Excellent written and verbal communication skills. Excellent Client interaction skills and proven experience in working independently as well as in a team. Good at Documentation on User requirements and System Specifications. Familiar with container based technologies like Docker. Used container orchestration tools like Docker Swarm, Kubernetes. Work Experience Full Stack Java Developer LexisNexis Risk Solutions - Alpharetta, GA August 2017 to Present Claims-Report Portal (CLRP) is the interface by which customers will order incident reports from LexisNexis. Overall purpose of the application is to consolidate all the police reports, which also allow access to high volume claims related reports ordered by claims adjusters. CLRP deliver secured access to the reports to end users quickly, easily and reliably. Responsibilities: Worked in multiple projects/applications. Used Agile methodology for the application development. Developed JUnit test cases for Test Driven Development approach. UI layer written in JavaScript and Typescript, using the Single Page Application architecture. React Component framework utilized so that on-screen component written once and be reused. Functional components such as handlers, Connectors and Routers each allow a clean separation of concerns in the application source. The Middle Tier (MT) layer is written using spring web Application architecture. Developed System architecture based on different design patterns like SOA, DAO, Singleton, Factory, etc. Developed independent & re-useable service components using Web services, Spring Dependency Injection and Design Patterns. Used Hibernate as persistence (DAO) layer to interact with database. Used Mockito ha the mocking Framework. Involved in Database Schema/XSD/WSDL Design for the various components. Implemented JAXB Framework to convert XML files into Java Objects and vice versa (marshalling and unmarshalling). Used Core Spring for Dependency Injection of various component beans. Containerized the application using Dockers. Implemented Kubernetes as the best ecosystem to deploy and manage containers. Used Jacoco to check the unit test coverage. Environment: Java 1.8, Spring 4.2, Spring IOC, Design Patterns, UML, Mockito Hibernate 5, Tomcat, Eclipse, MySQL, Maven, XML XSD, React JS, MobX, Typescript, Java Script, Log4j, Web Services, JIRA, TOAD, SOAP UI, UNIX, Docker, Kubernetes. Java/J2EE Developer JPMorgan Chase - Columbus, OH July 2016 to November 2016 Image Document Access Layer (IDAL) application that provides comprehensive document search and retrieval capabilities without preference to the client's architecture or development environment. It can deliver documents in a variety of output formants including PDF, TIFF and PNG. IDAL enables to build applications that deliver secured documents and images to end users quickly, easily and reliably. Responsibilities: Used Agile methodology for the application development. Worked with creation of reports in PDF and excel/csv formats for admin users. Wrote and maintained Technical Documents and Release Documents. Developed System architecture based on different design patterns like SOA, DAO, Singleton, Factory, etc. Developed independent & re-useable service components using Web services, Spring Dependency Injection and Design Patterns. Responsible for the performance improvement of batch processes using Core Java: Collections API & Multithreading. Used Hibernate as persistence (DAO) layer to interact with database. Developed Complex SQL queries, Views for DAO Layer. Involved in Database Schema/XSD/WSDL Design for the various components. Implemented batch jobs using Spring batch to load the data from files to database. Implemented JAXB Framework to convert XML files into Java Objects and vice versa (marshalling and unmarshalling). Used Core Spring for Dependency Injection of various component beans. Developed UI to monitor the process. Involved in Unit & Integration testing, bug fixing, Design Reviews, Code Walk through, Client Interaction and Prod Support. Used PMD Check style to measure the quality of the code. Environment: Java 1.7, Spring 3.2, Spring MVC, Spring IOC, Design Patterns, UML, Hibernate 3.4, Tomcat, Eclipse, Oracle 11g, Ant, Maven, XML XSD, HTML/DHTML, Java Script, JSP, Log4j, Web Services, Apache Axis, JIRA, TOAD, SOAP UI, UNIX. Java/J2EE Developer SPRINT - Overland Park, KS February 2016 to November 2016 The Device Services Infrastructure (DSI) application is an application developed which interface with Sprint back-office systems. The DSI application will manage communication to/from the customer's device for, Push notifications to applications on devices, Reporting / logging service for applications on devices, Back office data access for applications on devices, Authentication, Authorization, and Identity services for applications on devices. Responsibilities: Worked in multiple projects/applications. Used Agile methodology for the application development. Created Test plan documents for applications for executing the scripts for major enhancements. Developed JUnit test cases for Test Driven Development approach. Worked with Android and IOS developers Developed System architecture based on different design patterns like SOA, DAO, Singleton, Factory, etc. Developed independent & re-useable service components using Web services, Spring Dependency Injection and Design Patterns. Implemented Message-layer and Transport-Layer Security Responsible for the performance improvement of batch processes using Core Java: Collections API & Multithreading. Used Hibernate as persistence (DAO) layer to interact with database. Developed Complex SQL queries, Views and Stored Procedures for DAO Layer. Involved in Database Schema/XSD/WSDL Design for the various components. Implemented JAXB Framework to convert XML files into Java Objects and vice versa (marshalling and unmarshalling). Used Core Spring for Dependency Injection of various component beans. Designed customized global exception handling which will take care of the SOAP fault codes and HTTP error codes. Designed Mule flows for polling and crone jobs to preload the data automatically based the configuration time. Used Log4J for debugging and error logging purposes. Used Jacoco to check the unit test coverage. Environment: Java 1.7, Spring 3.2, Spring MVC, Spring IOC, Design Patterns, UML, Hibernate 3.4, Tomcat, Mule Soft 3, Git, Eclipse, Oracle 11g, Ant, Maven, XML XSD, HTML/DHTML, Java Script, JSP, AngularJS, Log4j, Web Services, REST, CXF, JIRA, TOAD, SOAP UI, UNIX. Java Developer Guardian Insurance Company November 2012 to August 2014 A Health Insurance Portability and Accountability Act (HIPAA) disclosure is the legally authorized sharing of individual physical or health information by Health Care facilities. HIPAA disclosure application provides the ability to upload new disclosure in bulk. It facilitates provision to view pending, denied, approved disclosures, and individual client reports based on the criteria and download a CSV file. Responsibilities: Collaborated with multiple teams to implement the requirements and deploy releases. Used Agile/Scrum methodology for the application development. Created Test plan documents for applications for executing the scripts for major enhancements. Developed JUnit test cases for Test Driven Development approach. Developed RESTful web services API using Apache Jersey for client applications. Implemented web-tier using Spring MVC. Worked with creation of reports in PDF and excel/csv formats for admin users Developed System architecture based on different design patterns like MVC, SOA, DAO, Singleton, Factory, etc. Used AJAX for Asynchronous data communication between UI and server side. Developed independent & re-useable service components using Web services, Spring Dependency Injection and Design Patterns. Implemented Message-layer and Transport-Layer Security Responsible for the performance improvement of batch processes using Core Java: Collections API & Multithreading. Used Ibatis as persistence (DAO) layer to interact with database Implemented JAXB Framework to convert XML files into Java Objects and vice versa (marshalling and unmarshalling). Used Core Spring for Dependency Injection of various component beans. Used Log4J for debugging and error logging purposes. Involved in Unit & Integration testing, bug fixing, Design Reviews, Code Walk through, Client Interaction and Prod Support. Environment: Java 1.7, Spring 3.2, Spring MVC, Spring IOC, Design Patterns, UML, Hibernate 3.4, Web sphere 8.5, Git, RAD 8.0, Oracle 11g, Ant, Maven, XML XSD, HTML/DHTML, Java Script CSS3, JSP, Log4j, JUnit4.0, Web Services, REST, Apache Jersey 2.2, JIRA, TOAD, SOAP UI, UNIX. Education Master of Science University of Missouri Kansas City - Kansas City, MO December 2015 Bachelor of

Technology in Technology JNTU April 2014 Skills JAVA (4 years), JAXB (4 years), SOAP (4 years), UNIX (4 years), WEB SERVICES (4 years) Additional Information TECHNICAL SKILLS: Languages & Tools Java 1.6/1.7/1.8, J2EE, SQL, Subversion, SVN, Git, Mule Soft. Java Technologies Multi-Threading, Servlets 2.3, JSP 1.2, Spring, Spring MVC, Hibernate, JDBC, JAXB, Web Services, WSDL, SOAP, REST. Web Technologies HTML, HTML5, DHTML, CSS, CSS3, Bootstrap, XML, XSLT, DTD, XML Schema (XSD), JSON, React, JavaScript. Databases: Oracle 11g/10g/9i/8i, Sybase, DB2 10.1, MS Access Application Servers IBM WebSphere 4.x/5.x/6.0/8.1, WebLogic 6.x/7.x/8.x/9.x, JBoss AS 4.x/5.1, Tomcat 5.x/6.0/7.0, Apache Web Server. Frameworks Spring, Spring MVC, Spring boot, Spring AOP, Spring Security, Hibernate, Mobx. Operating Systems UNIX (AIX), Red Hat Linux, Windows. IDE RAD 6.0/6.1/7.0, Eclipse, WebLogic Workshop. Container Technologies Docker, Docker Swarm, Kubernetes.

Name: Dr. Emily Garza

Email: dperry@example.org

Phone: (642)578-8762x125