**Aslam Email:mkaslam246@gmail.com**

**Sr. Java Full Stack Developer Phone:786584014**

**PROFESSIIONAL SUMMARY:**

* Over 8+ years of IT experience in various phases of software development life cycle with Analysis, Design, Development, Integration and Implementation.
* Strong programming skills in **Java** and **J2EE** with extensive ability to develop scalable applications.
* Expertise in **Java** and **J2EE** technologies including **JSP, JDBC, JPA** and **JMS.**
* Experienced in **HTML, XHTML, XML, CSS, Angular, JavaScript, jQuery, JSP and Servlets**.
* Worked extensively in developing web applications using **Java, Servlets, JSP, JDBC, JNDI** and **AJAX.**
* Experienced in **MVC (**Model View Controller) architecture with **Magento**, **Angular JS** and **SPRING.**
* Strong expertise in using **AngularJS directives** and **filters** to make **SPA’s** better in functionality.
* Extensively worked on custom **AngularJS directives**, **tags** and integrated with **Spring forms**.
* Experienced with UI frameworks such as **AJAX, JavaScript, ReactJS, BackboneJS** and **Angular JS.**
* Experienced in web/Application servers like **Web Logic 10.0,JBoss 7.1.0, WebSphere 8** and **Tomcat 8**.
* Experienced with **SOAP/RESTful** web services and used to scale the applications for better **BOA.**
* Worked on **Service Oriented Architecture (SOA)** such as Apache Axis web services which use **SOAP**, **REST**, and **WSDL**. Exposure to **JAXB, JAX-RPC, JAX-RS & JAX-WS.**
* Expertise in design and development of enterprise level spring based web applications using **Spring REST, Hibernate, EJB, Struts, Servlets 3.x**, **JSP** and **JSF** for **User Interface Applications**.
* Expertise in with **Spring AOP, Security, Event Handling, Spring MVC, Spring IOC Container.**
* Technically sound in understanding and implementation of Spring **Micro services** and their patterns.
* Extensive hands-on on designing **Micro service** applications with **fail safe** methodology using **Circuit Breaker pattern** with **Hystrix/Dashboard** and load balancing using **Spring Ribbon**.
* Expertise with employment of **Spring Kafka** and **Zookeeper** apart from **JMS** as messaging service.
* Implementation of suitable connectors or connecting links between **Hateoas** and **Spring REST** services.
* Experienced with distributed transaction management with isolation/propagation levels with **Spring JTA**.
* Proficiency in developing/maintaining using **Oracle, MySQL** and **SQL Server (2008R2, 2012, 2014)**.
* Proficient in **SQL, PL/SQL, Stored Procedures** and **Triggers** while dealing with automation on **DB.**
* Experienced with **NoSQL** databases like **MongoDB, Couch DB** and **Cassandra** on large data platforms.
* Also worked on creation of **Docker** images, Containers, push/pull and on integration with **Spring Boot**.
* Hands on experience with **Netflix** cloud and fall back technologies like **Eureka, Hystrix** and **Ribbon**.
* Experience using secure authentication mechanisms like **LDAP, SASL** and **Kerberos v4**.
* Extensive working with **Amazon web Services** like using **Elastic Beanstalk**, **EC2** and **AMI** with **Linux**.
* Strongly associated with usage of **Selenium, Jenkins, Junit** and **Curl** for **Unit** and **System Testing**.
* Extensive hands on experience in fine tuning and scaling the applications in Pivotal Cloud Foundry.
* Experience with the Integrated Development Environments like **Eclipse, NetBeans, JBuilder** and **IntelliJ**
* Experienced with **CICD** - **ANT/MAVEN/GRADLE/JENKINS/HUDSON/CVS/GIT/GITHUB.**

**TECHNICAL SKILLS:**

|  |  |
| --- | --- |
| **Languages** | **Java 1.8, J2EE, Scala 2.12.1, Python 3.6.0** |
| **Web Technologies** | **SERVLETS 3.1 , JSP 3.0, JavaBeans 3.0, Java Servlet API 3.0**, Angular2/4 |
| **Frameworks** | **Spring 4.3, Hibernate 4.3, Angular JS 1.0/1.5, Spring Security 3.2,ReactJS** |
| **Application/Web servers** | **Apache Tomcat 8, IBM WebSphere 8.0, Jetty 9.4, Jenkins 2.50, WebSphere MQ 7.5** |
| **Databases** | **Oracle 10g/11g/12C, SQL server 2012/2014/2016, MySQL 5.7, DB2 11.1** |
| **Internet Technologies** | **HTML 5, JavaScript 1.8, XML 2.0, CSS 3 and CSS 4, jQuery 2.11, Angular JS 2.0, BackBone.JS 1.3.1.** |
| **Cloud Environments** | **Azure, AWS, Netflix Eureka, Mesos, Kubernetes.** |
| **IDE** | **Eclipse, NetBeans 8.0.2, IntelliJ 2017.1, Spring Tool Suite (STS) 3.8.3** |
| **Operating system** | **Windows 10, Linux Mint 18.1, Unix** |
| **Bug tracking Tools** | **JUNIT4.12, JIRA 7.0, Bugzilla 4.4.12, Curl 7.36** |
| **Reporting Tools** | **Jasper Reports 6.0, Crystal Reports XI, SSRS 5.0** |
| **Methodologies** | **Agile, waterfall , TDD (Test-Driven-Development), Scrum** |
| **Developer IDE Platforms** | **Eclipse, Edit plus, Notepad ++,TEXT PAD, JBuilder, Net Beans 8.0.2, IntelliJ 2017.1, Spring Tool Suite (STS) 3.8.3, GITHUB 2.12.0** |
| **Build Tools** | **Ant 1.10, Maven 3.3.9, Gradle 3.4.1** |
| **Web Services** | **SOAP 1.2, REST 2.0** |

**PROJECTS UNDERTAKEN:**

|  |  |
| --- | --- |
| **Client** | **Black Knight Financials, Houston, TX** |
| **Role** | **Sr. Java Full Stack Developer** |
| **Period** | **SEP 17– Till Date** |
| **Description** | Black Knight Financials is a leading provider of software, data and analytics solutions to the mortgage and consumer loan, real estate and capital markets verticals.  Developed an application which provides users a web interface application and to view the available fund data in various formats. It also validates various transactions and activities and generate alert messages. Users can log on to the system to research and resolve the alerts. |

**Roles and Responsibilities:**

* Worked on various project methodologies like **Agile Software Development**, **Testing** and **Scrum**.
* Used **IBM FileNet** document container to prevent breach of integrity and security of the client data.
* Presently working on enhancements code and supporting the user’s applications to process it.
* Developed UI components using Angular 4 dependency Injection, Models, data binding, controllers
* Implemented **React JS** on server side **validations,** to **handle actions** and **Events** and also **Encapsulation.**
* A few applications were involved in design changes using **HTML5, CSS3** and fraction of **JavaScript.**
* Worked on business logic and calculations, performs analysis, handles data transfers, and generates XSL for presentation layer and **SQL** commands for **database** layer.
* Extensively implemented **Multithreading** concept in order to handle Transaction Management with isolation and propagation levels. Thread handling was used to maintain continuity of execution.
* Hands on experience on **Spring framework** applications like **Spring Core, Spring AOP, Spring MVC, Spring Batch, Spring Security** and **Spring Boot**.
* Experienced first-hand practices for integrating **Micro services** into existing system architecture.
* Experienced in implementing **Micro services**, **Service Oriented Architecture (SOA)** with **XML** based Web Services (**SOAP/UDDI/WSDL**) using Top Down and Bottom Up approach.
* Experience in creating **RESTful**web services using **Jersey, Spring** framework and **JAX-RS**.
* **Java Beans** are used as container of business logic and supporting the services or applications.
* **Micro-services** were deployed in modern containers using **Docker** that increased Isolation.
* Used wide implementation procedures in order to disintegrate whole monolithic application into loosely coupled reusable **Micro services**. This laid foundation for the simplicity in updating versions of existing technologies without hassle of deploying entire modules.
* Employed **fail safe** and **circuit breaker patterns** for the first time in Client’s email applications using **Hystrix** and **Hystrix Dashboard** in **Spring boot** Micro service Applications.
* Exceptionally handled heavy traffic with the help of **Spring Ribbon and Kafka** broker service.
* Good knowledge with cloud services like **Spring Eureka, Cloud Foundry and Aws** etc.
* Creating **REST** based Web Services by establishing links using **Hateoas** from spring based Applications.
* **Docker** containers were used to deploy the applications as part of scaling application using **Micro services**. Image creation, pushing and pulling of images was also part of deploying application.
* Used **DROOLS** engine as java runtime rule engine to validate business rule before executing in runtime.
* Developed the Web Services Client using **REST, RSDL** description to verify the history of the new customer to provide a connection.
* Worked on **XML** parsers (**JAXB**) for data retrieval and **JAXB** parser for marshalling and unmarshaling.
* Used **SQL** statements that were passed through **hibernate 4.3** connection to a Server.
* Writing **SQL Queries** to get the information from **OracleDB**, Saving the data, modifying the data in the DB and removing the data from the DB. Used joins to get the data from more DB Tables. **Views and Unions** were also part of our implementation.
* Used **CURL** to move data from or to a server using available http and https protocols.
* Writing **PL/SQL** with Stored Procedures, functions to get the information from **DB**, Saving the data, Modifying the data in the DB and removing the data from the DB. Using joins to get the data from more DB Tables.
* Used DB components like **Data Access Objects**, **Collections**, and **Value Objects.**
* **GIT** is used for the version control; **Maven** is used for build script and configured **log4j, .log** files.
* Used **AWS** cloud services and deployed couple of applications on **Elastic Bean Stalk** and **EC2** Instances as part of service transition phase, but later on chose to deploy them on Eureka.
* Aware of troubleshooting issues on **AWS**, **Netflix Eureka** and other developer IDE and Cloud environments.
* Using **JUnit** test in Eclipse, developed Unit test cases and test suite**. Curl** was also implemented to test.
* Used **Jenkins** as the integration tool and improved **scalability** of applications on **cross-platforms**.
* Involved in **System Testing, Integration Testing, and Production Testing** on **WebLogic** Server.

**Environment: Java/JDK 1.8,J2EE, Log4j, Angular 4,Web Services, Hystrix, Hystrix Dashboard, Spring Kafka, AWS, Netflix Eureka, Netflix Ribbon, Hibernate 4.3, REST 1.3.0.0, RESTful, Windows SFTP/SCP client (Winscp382), Eclipse, MAVEN build tool, Junit, Jenkins Continuous integration tool, WebLogic Application Server 10.1, Oracle 11g, 12C , Linux OS, Agile Methodology, Curl, GIT, PL/SQL, Docker, Hateoas, Curl.**

|  |  |
| --- | --- |
| **Client** | **Kroger, Cincinnati, OH** |
| **Role** | **SR JAVA/J2EE Developer** |
| **Period** | **OCT 16 – AUG 17** |
| **Description** | This Project is an initiative tasked with reducing complexity, terminating inefficiencies, generating operational efficiencies with enhanced automation and improving speed to market. It also includes developing various web content portals for different departments within the company. These web portals provide employees easier access to the data and other company related information and services. |

**Responsibilities:**

* Responsible for all stages of design, development, and deployment of applications. Active role in **Test Driven Development** termed as **TDD** apart from Agile Software Development.
* Scalability and performance had been the keen concept and we had to modify existing application to **Scala.**
* Complex features like **Tuples, Functions** and **Macros** were implemented using **Scala** for better performance. Also **stability** of the application was achieved with advent of **Scala** programming.
* Worked with the business experts to understand existing business processes and determined a shared approach that would serve their business needs using the new packaged application.
* Documented requirements for necessary customizations and employment of **IBM FileNet**.
* Designed, analyzed and developed the **front-end** and **back-end Servlets, Bootstrap** for designing**, HTML5, CSS3 & Angular JS.**
* Used **JSP** to dynamically generate web pages based on **HTML** and **XML** Document types.
* Use **JSF (Java Server Faces)** to implement component based User Interface Applications.
* **JSON** due to its light-weight nature was used as the main data-interchange format in place of **XML.**
* **CRUD** was used as part of servlets where **DML** operations were required on **RDBMS** like **SQL DB’s**.
* Worked on **Core Java** and its libraries in creating the **Java**web application's business Logic.
* Used JSP to create the front end. Used various third parties to validate the data entered by the User.
* Also used **JavaScrip**t for client side validations of input data. Created **HTML** pages using **CSS** styling.
* Used various **Spring Modules** like **Spring AOP, Spring WEB-MVC** and **Spring ORM** in development.
* Used **RESTful** web services to expose certain functionalities of the application.
* Used **Micro-services** for scalability and reliability of the application in addition to loose coupling.
* Auto scaling functionality of **Micro-services** helped in cloud infra where slowdowns were eliminated.
* **Docker** had been the most important of containers in accelerating the growth of **Micro-services**.
* **Mesos** and **Kubernetes** were used as the best ecosystem to deploy and manage **Micro-services**.
* Used **JAX-RS** plus jersey implementation for **RESTful** web services and containerized it.
* Created hibernate-config.xml and configured the **hibernate 4.0** for the application.
* Configured **hibernat**e with **Spring ORM** as part of **Spring** and **Hibernate** Integration for best results.
* Used **JBOSS** as the chief enterprise level server while dealing the database activities with **hibernates**.
* Used **Dependency Injection** feature for **wiring** and defined dependencies using constructor arguments.
* Implementation of data access layer has been cumbersome for a while and this was greatly resolved using **Spring JPA** through implementation of data access layers there by reducing boiler-plate-code.
* Used **Micro-services** to ensure the scalability of application on cloud infrastructure**.**
* Written **POJO Classes**, .hbm files for Hibernate **Object-To-Relational Mapping**.
* Defined several **UML** diagrams; Activity diagrams, Sequence and Object diagrams.
* Fail-safe/fallback methodology was employed using **Circuit Breaker** Pattern (**Hystrix dashboard**).
* Developed Web services using **Spring RESTful** web Services replacing **SOAP UI**.
* Writing stored procedures in **Oracle 11g/12c** and associated with .hbm to outcast the performance.
* Writing Unit test cases using **JUNIT** testing framework and performed unit and system testing.
* Created spring configuration files to define Handler mappings, Bean definitions, and developed Model classes for View Resolutions and **DAO** implementation classes extended from **Hibernate Dao** Support.
* Worked on **NoSQL** database like **Cassandra** where data storage was scaled rapidly.
* Perfectly utilized the de-centralized nature of **Cassandra** making DB **Always-On** and avoiding regional **Outages.**
* Created **Hibernate** persistent and configuration files for **ORM** implementation.
* **CURL** was used as the core component of testing environment as part of unit and system testing.
* **Selenium’s** playback tool feature for authoring tests has been used in executing more than 100 test cases.
* Worked in **TDD**, **Agile** environment and attended **SCRUM** meetings as part of day to day activity.

**Environment: Java 1.7, Eclipse, Hibernate 4.0, Spring IOC, Web Services, HTML, Java Script, CSS, JQuery, XML, Angular JS, Bootstrap, Log4j, Oracle 11g, PL/ SQL, Junit 4.12, JBOSS 7.1.0, REST, MongoDB, Cassandra, Spring JPA, Curl, Agile and Scrum, Hystrix, Spring MVC, Spring ORM, Spring AOP, POJO, Dependency Injection, Kubernetes, Mesos, Micro-services, Selenium and IBM FileNet.**

|  |  |
| --- | --- |
| **Client** | **CIOX Health, Alpharetta, GA** |
| **Role** | **SR Java/J2EE Developer** |
| **Period** | **APRIL 14 – SEP 15** |
| **Description** | CIOX provides a variety of services in release of information department (ROI), record retrieval and health information management. On the technology side, the company does penetration testing and social engineering testing with 3rd parties and the company itself. The company trains its workforce around HIPAA, security and privacy. |

**Responsibilities:**

* Involved in System Analysis and Design methodology as well as **Object Oriented Design** and development using **OOAD** methodology to capture and model business requirements.
* Responsible for Object Oriented Design using Enterprise Architect.
* Review and guide the application architects with Object Oriented Design using **Rational Unified Process** (RUP) using Enterprise Architect.
* Involved in Java **Multithreading** to cascade multiple changes on multiple records in only one transaction and developed a batch process framework.
* Responsible for **JAVA, J2EE, JSF, JBPM, EL** and Java **FX** coding / development.
* Implemented the web-based application following the **MVC II** architecture using **Struts** Framework.
* Used **Hibernate** as a persistence layer that is capable of handling high volume transactions as well as managing concurrency issues.
* Implemented **J2EE** patterns like, Data Access Object **(DAO),** Session facade, Business Delegate and Value Object.
* Implemented the service layer using **Spring 2.5** with transaction and logging interceptors.
* Used **Spring** framework for middle tier and **Spring-hibernate** for data access.
* Developed web-based applications using Web Services, both **SOAP/WSDL** and **REST**.
* Used **multithreading** to improve performance in generating the reports in Java.
* Extensively developed User defined Custom **JSP** tags to separate presentation from application logic.
* Designed and developed Web Services **(SOAP, WSDL).** Created and compiled **XML** Schema to generate **JavaBean** classes using Apache Axis.
* Installation, Configuration & administration of **WebLogic** application Server.
* Designed and developed used XSLT transformation components to convert data from **XML** to **HTML**, Used **XML DOM API** for parsing **XML**.
* Involved in writing **SQL**, Stored procedure and **PL/SQL** for back end. Used **Views** and Functions at the Oracle Database end. Developed the **PL/SQL** scripts for rebuilding the application **Oracle** Database.
* Developed **HTML** and **JavaScript**. Developed Scripts for **UNIX** platform deployment.
* Used **ANT** for compilation and building **JAR, WAR** and **EAR** files.
* Used **Log4J**, **Splunk** for logging and tracing the messages.
* Involved in the day to day development activities for the entire application from offshore.
* Involved in **System** and **Unit** testing using **Junit**.
* Coordinate with the onsite and offshore developers with respect to issues and follow up.
* Used **JBPM** to manage workflow, schedule Jobs, Execute procedure, Client Scripts and work on batch processing.
* Project coordination with other Development teams, System managers and web master and developed good working environment.

**Environment: Java, JSP, Servlets, JSF, Spring DI/IOC, Hibernate, XML, JBPM, HTML, JDBC, Web services, Rational Software Architect, Web Services (SOAP/WSDL and REST), Multithreading, Web Logic Application Server, Junit, Maven, Log4J, SVN, Oracle10g, UNIX.**

|  |  |
| --- | --- |
| **Client** | **American Family Insurance Madison, WI** |
| **Role** | **SR Java/J2EE Developer** |
| **Period** | **Aug 12 – MAR 14** |
| **Description** | Sales Tool Project is an Online Auto Insurance Sales tool used to estimate the quote, calculate the premium as per state rules and sell the quote to customers. Customers can access this tool from www.amfam.com and buy insurance online, Sales Tool Rewrite project is an enhancement and Integration project with the following major business requirements. Implement a new Online Auto Insurance Sales process flow for quoting to reduce the number of questions/responses necessary to provide a quote premium. |

**Responsibilities:**

* Involved in the analysis, design, and development and testing phases of Software Development Life Cycle (**SDLC**) of building interface for **CRM** System. Hands-on with **Selenium** as part of testing.
* Developed N-tier client-server business application software using Model View Controller **(MVC)** Architecture.
* Developed user interface using **JSP, JSP Tag libraries JSTL, CSS3, JavaScript** to simplify the complexities of the application
* Involved in writing different Design Patterns like **Filter, Action, Business Delegate, Service Locater, Data Transfer Objects (DTOs) and Data Access Objects (DAOs).**
* Used Hibernate in persistence layer and developed **POJO's,Data Access Object** (DAO) to handle all database operations.
* **JSON** being light in nature was used as the chief source of communication between the server and the application.
* Preparation of **Java and JSP** coding standard checklist.
* Involved in the development of **Servlet, Struts Action classes, Business Objects, Web Services.**
* Involved in writing **JUnit** Test Cases and **Ant** Build Script.
* Involved in writing JSP using **Bean, Logic, HTML, and Iterate Tags**.
* Developed client request validations and processing using **JavaScript** and **HTML**.
* Spring boot and REST services are implemented while developing web-based applications.
* Each application was divided into separate or individual **Micro-services** where each service had its own functionality and loosely coupled with the other components of the Application.
* Deployed the Application into **Docker** container and made easily accessible at runtime using **Cloud Foundry** and other cloud services like **AWS, Netflix Eureka, Spring Hystrix** as fallback method, **SpringRibbon** for load balancing and a broker service to maintain status using **Spring Kafka** and **Zookeeper.**
* Developed Struts Action Forms, Action classes and performed action mapping using Struts.
* Used Web Services to publish the credential validation of customer. The Services are published by one bank and all other branches and vendors are using the published services.
* **SOAP** has been used as a protocol to send request and response in the form of XML messages.
* **WSDL** has been used to expose the Web Services. Designed and Developed Oracle Procedure for one-time conversion for Client Demographic and Financial systems.
* Developed **Hibernate** DAO’s and configuration files to access the **ORACLE** database.
* Developed the application by implementing Spring Framework based integration.
* Developed client applications using **JAXR** and **JAXM** to connect to the **UDDI registry.**

**Environment: WebLogic 10.3, Java 1.5/1.6, Apache Axis, JSP 2.0, Hibernate 4.3, Servlets 2.5, XML, SOAP, WSDL, Oracle, Eclipse, JAXR, JAX-WS, JAXM, PL/SQL, Bean, Junit, JBOSS, Struts, DAO, AOP Spring, MVC Spring.**

|  |  |
| --- | --- |
| **Client** | **Carlson Wagonlit, Online Air Ticket Refund Tracking, Denver, CO** |
| **Role** | **Java Developer** |
| **Period** | **Oct 10 – Jul 12** |
| **Description** | This is a travel domain application (on MVC pattern) for Carlson Wagonlit Travel. The objective of this application is to provide facility of refund for air tickets. When a user wants to refund a ticket, user register himself and submit his request for ticket refund, system generates a receipt for user, which contains all the details of ticket and receipt number. User can monitor his refund of tickets using receipt number. Other part of this application is for administrator to link user information to back office support application for refund and generate invoice etc. |

**Roles and Responsibilities:**

* Implemented various project development techniques like **Agile Development** and **Testing**, **Test Driven Development** and **Scrum** methodology. A few phases included traditional **Waterfall** methodology.
* Code implemented for enterprise application using combination of **Java, JSP, HTML**, **JavaScript, spring** and **Hibernate** Running on **WebLogic**.
* Used JSP to develop front end UI screens.
* Used **(GWT) Google Web Tool Kit** in developing web applications in AJAX as it is easy to develop for developers and creates high performance applications. Also made the application Compliant.
* Used Java 6 to implement Generics, threading, Annotations.
* Used **JavaScript** for validations and other checking functionality for the UI screens.
* Working on the project for **CSR Dashboard** in the **Prime Brokerage & Financing** Technology organization for on boarding the new Trade Breaks/Settlement Exceptions events message.
* Used XPath to navigate through elements and attributes in an XML document which is an XML message in MQ server.
* Involved in design and development of trading module of the application.
* For Trade Breaks/Settlement Exceptions events message creating the xml message which will useful to test the possible data availability in the database using **String Template API**.
* Designed and wrote design specifications for the Settlement Exception’s Data Capture process receives all incoming external messages from source PBBK, normalizes the messages before forwarding the normalized message to the Event Processing process to persist in the data base.
* Worked on Trade Breaks/Settlement Exceptions events, along with Trade File Upload statuses on the Dashboard. Currently, the CSRs utilize Emulation of their clients’ IDs to view this information. However, placing this information across all clients into the CSRs’ Dashboard view will allow them to be even more proactive in servicing their client base.
* Initialized trading strategies including breakouts, and market-making.
* Transferred tested trading strategies to real client portfolios.
* Used **Argo UML** and **MS Visio tool** to design the Class diagrams, Sequence diagrams and process flow diagrams.
* Worked on the code implementation for Settlement Nate layer in the Oracle data base server.
* Used Hibernate for ORM mapping and persistence layer of the Exceptions Data Capture process receives all incoming external messages to process it, enrich the data and persist the data using the Hibernate Application and Writing POJO’s.
* **JSON** data was used and extensively employed as the common tongue between the application and network layer in-order to make the context of server load easy.
* Used Spring **Web MVC framework** for the applying **MVC Implementation** to the web application.
* Used Spring **Web Module** integrated with **JSP** framework.
* Used Spring **ORM** to integrate **Spring Framework** with **Hibernate**.
* Used Springs **JDBC** and **DAO** layers to offer abstraction for the business from the database related code **(CRUD).**
* Used **SpringAOP module** to handle transaction management services for objects in any Spring-based application.
* Worked in developing Spring related backend services.
* Developed business layer using **spring, Hibernate and DAOs**.
* Generated the Model part of **MVC** with **spring** and **Hibernate** Framework.
* Integration with the **J2EE** Services layer of Unity Platform, using **spring’s** Application Context.
* Create and configure spring to use **Dependency Injection**.
* Using **Spring IOC** developed Java Bean classes, with get and set methods for each property to be configured by spring. These can optionally implement interfaces, created an XML file that describes the Java Beans to be configured. Beans can also be described using a Java property file, but using XML and developed an XML Bean Factory to create configured beans.
* Wrote **SQL** Queries to get the information from DB, Saving the data, modifying the data in the DB and removing the data from the DB. Used joins to get the data from more DB Tables.
* Rational Clear Case is used for the version control, Ant and Maven is used for build script and Created, configured **log4j, .log files**.
* Implemented **team foundation service** from Microsoft in executing project Roles and Responsibilities.
* Using **JUnit** test in Eclipse, developed **Unit test cases and test suite.**
* Involved in **System Testing, Integration Testing, Integrated** and **ProductionTesting** on **WebLogic** server.

**Environment:Java 6, J2EE, JSP, HTML, Java APIs, JavaScript, Spring MVC 3.0, Hibernate, JMS, SQL, XML, XSD/ XML Schema, XPATH, AJAX, CSS, Ant, Maven, Eclipse, String Template, JAXB, Oracle, MQ server, WebLogic 10.3 Application Server, SVN, Quantum, Quartz Enterprise Job Scheduler, DB2, IBM Mainframes, Linux/Unix, Argo UML Tool and Waterfall methodology.**

|  |  |
| --- | --- |
| **Client** | **Value Labs, Hyderabad, India** |
| **Role** | **Java Developer** |
| **Period** | **OCT 09– Sep 10** |
| **Description** | BSHM builds a system, which helps both Doctor and patient in an efficient manner by providing information about the availability of Doctors, maintaining patient records according to their visit and generating reports of patients of their medications, Lab Findings, Problems List and serves the pharmacy maintenance also. The information may be very critical from both doctor's and patient's point of view, BSHM maintains a safe and secure database system for patients. BSHM stores patient's records efficiently and make them easily retrievable. The entire system modules are Appointments, patient chart, Reports, pharmacy, Administration |

**Roles and Responsibilities:**

Involved in various stages of Enhancements in the Application by doing the required analysis, development, and testing.

* Developed logic and code for the registration and validation of the enrolling customer using Spring Boot Application. Used **Hibernate** as the ORM tool to connect application with Database.
* Played active developer role in writing **business rules** and **scripts** during **Service Now** Administration.
* Roles included maintenance of **database** and ensuring round the clock **service availability, Log Shipping, Clustering and Replication**. Involved in change request implementation using **JAVA/J2EE**.
* Active responsibility included automation of various timely activities we used to operate on databases like **backups**, **Snapshots**, **Health check** reports, Experience with **high availability**, **OLAP**&**OLTP** etc.
* Coordinated all the team members during the migration activity of **RBS servers** to **SQL 2012** version.
* Documented and generated reports using **SSRS** and integration was done using **SSIS** services.
* Occasionally responsible for **UNIT** and **System testing** for software’s, designed and fabricated with requested changes from customers.