

**Professional Summary:**

* Around 8 years of professional work experience in Object Oriented Programming, design and development of Multi-Tier distributed, Enterprise application using **Java** and **J2EE** technologies.
* Well versed with different phases of Software Development Life Cycle (**SDLC**) including Design, Implementation, and Testing during the development of **software development**, **Agile** Methodology through **Scrum**, **Test Driven Development**, Waterfall, and many other methods.
* Strong technical knowledge in **Core Java, J2EE development**, with experience in designing functional and technical specifications with understanding of **Lambda Expressions, Stream API, Functional Interfaces, Method References, Garbage Collector, Collections, Multithreading, Event handling, Exception handling, Java Generics, Externalization and collections.**
* Expertise in designing and developing **N-tier Architecture** Applications using **Java/J2EE** using Web services **Servlets, JSPs, JDBC, XML, Java Mail, DAO, JMS, MDB, Apache Struts 1/2, HTML5, CSS3, Angular 2/4/6/7/8, jQuery, AJAX, JavaScript**.
* Strong experience in **MVVM** client-side design and validations using **HTML, DHTML, CSS, Angular , React JS, Java Script, JSP, JSTL, JQuery,** and **MVC** frameworks like **Node JS, Backbone JS, KonckoutJS,** and **HandlebarJS**.
* Extensive experience in using **React JS** components, **Flux concept, Animations, Forms, Keys, Router, Events**, and **plus Redux.**
* Good experience in developing single page application using **MEAN** (**mongo DB, Express, Angular** and **Node**) stack.
* Front-end GUI wireframing/prototyping skills using scripting languages like **JSP, JavaScript, PHP, PERL** and implementing **AJAX** features using **JavaScript** and **jQuery**.
* Working experience on developing a responsive single page application using **AngularJS** framework and **JavaScript**.
* Expert in developing **Server-Side JavaScript** frameworks using **Node JS, Express JS, Hapi JS, Handlebar JS** to build event driven, non-blocking I/O web applications.
* Good experience in **ORM** tools such as Hibernate, IBATIS with **JPA** specifications, **HQL**, **Caching**, **ACID** transaction.
* Extensive programming experience with implementation of **Core Java, Servlets, JSP, JDBC, JNDI, Spring Framework, Spring MVC, Spring AOP, Spring Batch, Hibernate ORM, Spring Web Flow, JPA, JSF, JMS, EJB, Bootstrap, Java Mail, Junit.**
* Hands on experience with **Spring Framework** and associated technologies such as **Spring MVC, Spring Data, Spring Cloud, Spring Boot, Spring Security, Spring Web Flow** etc.
* Designing Web services, **SOA** based applications and writing Web services using **WSDL, SOAP, RESTful** with **Jersey** and UDDIin distributed web application development environment including writing **ANT** and **Maven** for builds and deployments to different environments and Logging tools like **Log4J, SLF4J** and **Splunk**.
* Developed web-based enterprise applications using **core Java, J2EE, Servlets, JSP, JSF, JDBC, Hibernate, Web services** like SOAPand **REST, MVC framework** like **Struts, Spring MVC** and **SQL.**
* Technical experience in J2EE design patterns such **as Singleton, Factory, Abstract Factory, DAO, MVC, Facade &Business Delegate** in the development of enterprise applications.
* Implemented various Java design patterns like **MVC**, **Session Facade, Data Access Object, Factory, Singleton, Data Transfer Object and Front Controller.**
* Hands on experience in deploying new libraries with **Microservice** architecture using **REST API’s** and **Spring Boot**. Used Swaggerfor creating dynamic documentation for **RESTful web services**.
* Expert in implementing Microservices, **Service Oriented Architecture (SOA)** with **XML** based **Web services** like SOAP**/UDDI/WSDL** using **Top-down** Approach and **Bottom-up** Approach.
* Experience in creating RESTful web services using **Jersey**, **spring frame works** and **JAX-RS, JAX-RPC** and **REST** Web services client and exposing existing java components as **Web services using SOAP, WSDL and UDDI.**
* Programming, Developing, Deploying, Configuring **J2EE Application servers** like **Apache Tomcat 6/7/8, JBOSS, Oracle WebLogic, Glassfish** and **WebSphere**.
* Experience on using **Apache Kafka, Active MQ, RabbitMQ, IBM MQ** as massage-oriented communication which configure and build **asynchronous communication** with **JMS** services with **MQ** Series.
* Expertise in DB-design, normalization, writing **Triggers, Stored Procedures, Functions, Sequences, Cursors, Joins, Index Views, DDL, DML, User defined Data Types** to implement business logic and maintain **data integrity**.
* Proficient in **Relational DBMS** like **Oracle, SQL Server, MySQL, DB2, Toad** for performing **SQL** and **PL**/**SQL** programming and **NoSQL** DB like **MongoDB**, **CouchDB**, **Cassandra**, **Dynamo DB** and **Graph DB**. Can also migrate from SQL relational databases to NoSQL databases.
* Good knowledge on implementing **ODM** framework like **Mongoose** for mapping MongoDB and interacting with NoSQL databases using **Mongoose Connector** and **Cassandra Node driver**.
* Test Driven development like **JUnit, TestNG, Mockito, Cucumber** and performed RD to develop **Selenium cucumber frame** along with developing test cases with JUnit for Unit testing and **Log4J** for extensible logging, debugging and error tracing.
* Experience in Unit testing tools like using **Junit, Karma, Jasmine** and **Protractor** and **Integration** testing tools like **Mockito**.
* Worked on Validating RESTAPI’s using **Postman**.
* Experience on importing and exporting data to HDFS using Scoop and developing applications using Type safe technologies such as **Scala**, Play framework, Akka model.
* Experience in distributed computing or cloud computing, **Java/Scala** performance turning.
* Strong knowledge on using persistence framework like **Hibernate JPA** for mapping java classes with Database and using **HQL** (Hibernate Query Language).
* Pro-found experience on **Unix** and **shell scripting** allowing the user to interact with the OS.
* Expertise in version control tools such as **Git, GitHub, CVS, Jenkins** and **SVN** for and maintaining code repositories.
* Deployment of multiple high-performance **MongoDB** replica sets on **AWS EC2** with robust reliability.
* Experienced in both framework and **CloudFormation** to automate **AWS environment** creation along with the ability to deployment on AWS, using build scripts and automate solutions using **Shell** and **Python Scripts**, created **Terraform modules** to create instances in AWS and automated process of creation of resources is AWS using Terraform.
* Deploying, managing applications and servers on **AWS services** for such as **Elastic Beanstalk, CloudWatch, IAM, S3, RDS, Elastic Cache, SNS, SQS, AMI** and **AWS SDK’s** like **DynamoDB, Lambda.**
* Good experience in deploying **Docker containers** by creating Pipeline processes through **AWS CodePipeline** in AWS ECS using services like **CloudFormation, CodeBuild, CodeDeploy**.
* Managing **Docker Containers** and **Kubernetes Cluster** by **OpenShift** platform.
* Implemented **Microsoft Azure** for moving the on-premises data to **Azure cloud**.
* Hands on experience on Implementing and Architecting **Azure Service Offering**, such as **Azure cloud services, Azure storage, Azure Resource Manager (ARM), Azure Storage, Azure, Blob Storage, Azure VMs, SQL Database, Azure Service Fabric, Azure Monitor, and Azure Service Bus.**
* Designing, configuring, and deploying applications utilizing the **Azure** and **ARM** mainly focusing on auto-scaling, high availability, and fault tolerance.
* Strong Knowledge on implementation of **Pivotal Cloud Foundry (PCF)** services such as **Pivotal Service Container, Pivotal Application Service and Pivotal Functional Service.**
* Good experience with **Web stacks** such as **Apache, Varnish, Lighttpd and Nginx**.
* Experience **with Pivotal Tracker** and **JIRA** tools for ticketing stories in different methodology of Software Development Life Cycle.
* Good documentation skills using **Apache POI** and **I-text** and data visualization using **D3** and **Angular Fusion charts.**
* Development experience in different **IDE’s** such as **Eclipse, Spring Tool Suite, IntelliJ, Android Studio, Visual Studio** and **RAD.**
* Good hands-on experience of Build & Deployment phase and usage of Continuous Integration (**CI/CD**) tools like **Jenkins, Hudson, Puppet, Bamboo**.
* Strong interpersonal and communication skills and self-motivated individual showing ability to learn and use new technologies and frameworks in considerably less time.

**Technical Skills:**

|  |  |
| --- | --- |
| **Programming Languages** | Java (1.5/1.6/1.7/1.8), C, C++, PL/SQL, Python. |
| **Software Methodologies** | Agile, Scrum, TDD, SDLC, Waterfall, OOAD. |
| **J2EETechnologies&**  **frameworks** | JSP, JSTL, Servlets (3.1,3.0), JDBC, EJB, Java Beans XSD, JAX-RS, JAX-WS, SOAP, WSDL, Axis, Apache CXF, Java Security. |
| **Web Technologies** | HTML, DHTML, JavaScript, AJAX, Bootstrap, jQuery, XML, XSLT, XSL, CSS, JSON, Chase Standards, Node Js, EXTJS, AngularJS, ReactJS |
| **Web service Frameworks** | Spring, Hibernate, Struts (2.5/2.3), SpringMVC, SpringIOC, SpringBoot, SpringAOP, Spring Web Flow, JSF, Log 4J, Junit. |
| **Web services** | SOAP, RESTful, Oracle SOA, UDDI, WSDL. |
| **Design Patterns** | Singleton, Business Delegate, Data Transfer Object, Façade, Data Access Object, MVC, Filter. |
| **IDEs** | Eclipse, Net Beans, IntelliJ, Spring Tool Suite, Android Studio, Visual Studio, RAD. |
| **Application/Web servers** | Web Logic Server 12c, Web Sphere Application Server 8.5, Apache Tomcat (8.5/8.0), Apache HTTP Server, IBM HTTP Server (8.5), JBoss |
| **Databases** | Oracle, SQL Server, DB2, MySQL, PostgreSQL. |
| **NoSQL database** | MongoDB, CouchDB, Cassandra, Dynamo DB and Graph DB |
| **CI/CD tools** | Jenkins, Bamboo, Hudson, Puppet, XL Release |
| **Testing Tools** | JUnit, Log4J, Mock Framework, Cucumber, Selenium |
| **Version Control Tools** | GIT, SVN, Clear Case |
| **Operating Systems** | Windows, Unix, Linux, Mac OS |
| **Cloud Technologies, DevOps Tools & Other** | AWS, EC2, Pivotal Cloud Foundry (PCF), GCP, Azure, Docker, JIRA |
| **O-R Mapping** | Hibernate, JPA, IBatis, ODM. |
| **Testing Tools/Others** | Junit, JMeter, SoapUI, Putty, Mockito, Splunk |
| **Big Data Technologies** | Hadoop, Apache Spark, Scala, Hive, Hbase, Sqoop, Solr, Apache Kafka |
| **Messaging Platforms** | JMS, Apache Kafka, Active MQ, RabbitMQ, IBM MQ |

**PROJECT SUMMARY:**

**Responsibilities**:

* Based on Business Requirements, Fully Functional responsive modules were developed.
* Participated in Scrum meetings and Followed Agile methodology to track project progress.
* Interacting with business users and creating UI specification documents, design documentation and data modelling for the entire project.
* Designed dynamic and browser compatible pages using **HTML5, SASS/LESS, CSS3, Bootstrap, JavaScript, and React JS**.
* Building User Interface by using **React.js/Redux**, strong knowledge of **state store, middleware, action creator, reducer, and container.**
* **REST service** calls were written by several Axis service calls like **get ()** and **post ().**
* Implementing client-side Interface by using **ReactJS, Redux** for **SPA** development and Used the **ReactJS** to develop user interface.
* Used **React JS** developed a single responsive website that could be served to desktop, Tablets, and mobile users and to create responsive design.
* Developing **Data Grid Items** which can be dragged by **IRD system** users and Building Report **Data Grid, tree grid** by implementing **Redux** Form.
* using **REACT JS** to implement **Virtual Dom** for Client-Side view rendering services.
* Good understanding of the Redux pattern and **Redux/NGRX**
* Generate the server environments by using **browserify, flux** to manage with targeted URL's for Uni-Directional data flows and to monitor the application and writing the Client-Side code using **React JS**.
* Programmatic navigation by **React-Router** and client-side view rendering services by **REACT JS Virtual DOM**, state management by **React-Redux**.
* Created the **Node**.**js** EXPRESS Server combined with **Socket**.**io** to build **MVC** framework to backend **Oracle 11g** Database.
* Developed the application by using JSP, Servlets, Web services, JavaScript, WebLogic.
* Implemented **Multithreading** concepts in Java classes to avoid Deadlocks.
* Worked on java advanced concepts like **Lambda Expressions, Stream API, Functional Interfaces, Method References, Java Generics, Data structures, Exception Handling, Serializations, Java Reflection, Externalization** and **Collections.**
* Building cloud Microservices and implemented back-end using Spring Boot.
* Worked with **Microservice architecture** using **Spring Boot** to modularized code and implemented **REST API’s** using Spring REST and integrated **Swagger API** for documentation.
* Proficient in writing React code with help of **ECMA Script (ES6) and JSX**.
* Deployed our Application on **Pivotal Cloud Foundry** which is used to reduce the development overhead by providing a ready to use platform
* Deployed and Monitored Microservices Using **Spring Cloud Foundry** Managed Domains and Routes with the Spring Cloud Service Registry.
* Implemented **Spring Cloud- Netflix ZUUL** to building Microservices to communicate with each other.
* Used **Amazon Web Services (AWS)** for **Java API** to upload and retrieve user's media data
* Spin up **AWS EC2** Instances and assigned security groups. Used **S3 API** to store and retrieve the image files.
* Designed, configured and deployed **Amazon Web Services (AWS)** for a multitude of applications utilizing the **Amazon cloud formation**.
* Implemented **Netflix OSS** for Feasible changes after the deployment.
* Implementation domain base classes providing basic properties using **Spring Data**.
* Created reusable templates using Angular directives and worked with **Node JS**.
* Used **Kubernetes** to create new cluster and container management tool
* Used Microservices architecture with Spring Boot based services interacting through a combination of REST and **Apache Kafka** message brokers.
* Used Spring Boot for developing **Microservices, REST** to retrieve data from client-side using Microservice architecture and **PCF** for deploying Microservices.
* Used **RabbitMQ** as the Message broker to convert the entire flow as a SOA based architecture.
* Configured Hibernate Cfg.xml to facilitate connection pooling with Data Source and create POJOs to map with backend tables.
* Used Hibernate ORM solution, technique of mapping data representation from **MVC** model to Oracle Relational data model with a SQL-based schema.
* Developed RESTful Web services using **Jersey, JAX**-**RS** to perform CRUD operations on the database server over HTTP and to consume Web services for transferring data between different applications.
* Developed Integration modules for disparate clients using **Apache Camel**.
* Worked on NoSQL database like **Cassandra** to store key values, for data analytics which is used as a Database.
* Developed **Stored procedures** and **triggers** using PL/SQL to calculate and update the tables to implement business logic using Oracle database.
* Designed and implemented a Cassandra NoSQL based database and associated **RESTful Web service** that persists high volume user profile data.
* Worked on Deployment automation of all the Microservices to pull image from the private Docker registry and deploy to **Docker** swarm cluster using Ansible.
* Used **Singleton, Session Facade, and DAO patterns** in implementing the application.
* Deployed web components, presentation components and business components in **WebLogic Application Server**.
* Experience in Microservices development using spring boot and deployment in **Pivotal Cloud Foundry**.
* Created build environment for Java using **Maven**.
* Worked on **SVN** for Version Controlling.
* Used **JIRA** to keep track of the project, bugs, and issues.

**Environment: Java, AWS, Swagger, Spring MVC, Spring Core, Spring AOP, Spring ORM, Hibernate, Cassandra, React JS, Type script, Web pack, ECMA6, Karma and jasmine, Spring 4.x, Spring Boot, Spring cloud like Ribbon, Hystrix, Netflix ZUUL, Microservices architecture, Docker, Cassandra DB, OAUTH, Apache Kafka, Splunk, OpenShift, Linux, SQL, PCF.**

**Responsibilities**:

* Analyse requirements for each story card in a **Sprint** through Requirement review meetings and attend daily stand-up meetings (**Scrum**).
* Implementation of **NodeJS** mainly used for authentication, authorization, session maintenance and design of wireframe using **Angular 6** and creating services and exposing it to the backend which runs on **Spring Boot**.
* Made use of customized components and developed UI application using **Angular 4.0.**
* Created services on UI for storing and retrieving data from UI to the backend and Implemented customer side validations using **Angular 6.0.**
* Was under migration of the project converting **Angular 4 to Angular 6**. Implemented the complete set up with configuration file changes of angular components and updating router Modules.
* **LDAP** was used to develop key services for authorization and authentication in Node server and migrated it to Ping Access-**SSO** Application.
* Created **sessions** for Login user and setting session timed out for the entire UI application.
* Strong at attaining the architectural design for **MVC** using **directives, promises** and **pipes** and very good at understanding client-side application.
* Implementation under mobile device access using **cross platform technologies** such as **Ionic** which helps in improving the business and designing the UI.
* Design, development, and deployment of the front-end application using **HTML5, CSS3, AJAX, Bootstrap.**
* Made use of **Reactive Forms** using **Angular 6** where most of the validation is done using component rather than using template-driven which provides in **HTML**.
* Used **Executor framework** for Thread resource pooling.
* **Implemented Multithreading, Concurrency, Exception Handling, Generics** and **Collections** whenever necessary in the application.
* Used **Functional Programming, Parallel Streams** and **Lambda expressions**.
* Moved the project to **Java 8** to extensively use **Lambda expressions** and have better connectivity to Cloud in addition adding **Spring Boot** Libraries and UI Enhancement.
* Developed some test suites for **Automation testing** using **Selenium.**
* Migrating existing application into **REST** based **Microservices** to provide all the **CRUD** capabilities using **Spring Boot**.
* Used Spring Boot which is radically faster in building cloud **Microservices** and develop **Spring** based application with very less configuration.
* Made Services independent from each other by implementing **Microservices** concepts.
* Building cloud **Microservices** and implemented back-end using **Spring Boot**.
* Worked with **Microservices** architecture using **Spring Boot** to modularized code and implemented REST API’s using Spring REST and integrated **Swagger** API for documentation.
* Also created yml files using a swagger API within UI and used **Postman** to test **REST services**.
* Implementation domain base classes providing basic properties using **Spring Data**.
* Designed Target tables as per the requirement from the reporting team and Designed Extraction, Transformation and Loading (**ETL**) using **Talend**.
* Worked on a project which handles searching algorithms using **Elastic Search** which is mainly used for pulling analytics related information using the keywords entered by the end user.
* Used **OAUTH 2.0** for authenticating Users and provided security to **RESTful** Web services.
* Integrated the **ORM Object Relational Mapping** tool **hibernate** to the spring using **Spring ORM** in our app and used spring transaction API for database related transactions.
* Used **Hibernate** as **ORM** tool to configure **JPA**.
* Extensively performed database object mapping using **Hibernate annotations** and configuring Application Context files using **Spring IOC**.
* Spring Validators, persistence layer, and service layer components were implemented using **Spring/Hibernate API** and Spring/Hibernate annotations. Extensively used **Hibernate QL**.
* Created persistence layer using Hibernate for sending and retrieving data from the database.
* Apart from JMS as Messaging service worked with employment of **Apache Kafka** and **Zookeeper**.
* Utilize **Camel** to integrate Microservices with other Microservices.
* Implemented **PL/SQL** queries and used Oracle **stored procedures**.
* **Oracle11g** and **MySQL** were used to write Stored Procedures using PL/SQL and functions and procedure for common utilities.
* Achieved data in a collection to be distributed across multiple systems for horizontal scalability and created dynamic schema design through **NOSQL-CouchDB** by writing rich queries.
* Developed web applications using **Bootstrap, Code Mirror, Brackets and Ember.js**.
* Creation of interface to manage user menu and Bulk update of Attributes using **Node.JS** and **jQuery**.
* Have experience in developing and configuring **JMS** components in **WebLogic Application Server**.
* Maintained a stable code coverage with maximum threshold and wrote test cases using **jasmine-karma, Test Bed.**
* Unit Test Cases using automation Jasmine and **Karma Testing** Frameworks on **Angular**.
* Tested every functionality by Test Driven Development using **Junit** and **Mockito**.
* Took part in solving the bugs during **CI/CD** deployment and familiar with **OSE**. Also made changes in Jenkins Files as per the requirement of the pipeline.
* Responsible for implementation of java integration components for **JBPM** and **DROOLS**.
* Used **Slf4j** for Logging the application.
* Worked on **Bitbucket** for Version Controlling.
* Worked on **Kubernetes cluster** as a part of deployment activity. Familiar with Open Shift commands mainly for deployment.
* **Amazon EC2** instances were used to host the designed and developed frameworks to consume Web services.
* Authored application using **Spring Cloud services** and building cloud Microservices and implemented back-end using Spring Boot. spring version of **Netflix OSS-Eureka** that helps in routing, **Circuit Breaker** and **Ribbon** for client-side load balancing.
* Building Microservices for communication by implementing **Spring Cloud- Netflix ZUUL**.
* Writing server-side request handling and automated functional test cases to validate UI functionality implemented **Spring Circuit breaker pattern**, to monitor spring Microservices.
* During batch loading data integrity was verified on **Apache Cassandra** cluster.
* Used **Docker Container** for better collaboration and innovation and to address every application across the hybrid cloud and Docker.
* Focused on high-availability, fault tolerance, and auto-scaling by designing, configuring, and deploying **AWS** for a multitude of applications utilizing the AWS stack Including **EC2, S3**.
* By using **Route 53**, Configured **AWS EC2** instances, domain mapping.

**Environment: JavaScript, HTML5, CSS3-LESS, AJAX, Bootstrap, jQuery, Spring Security, Rabbit MQ, Node JS, Express JS, HTML5, Bootstrap, Site Minder, Hibernate4.x,OIDC, Eclipse Java EE, Jenkins, Log4j, Redis, Mongo DB, Mongoose Connector, Oracle, JBoss, Kibana, Elastic search, Logstash, EBS, Elastic Load Balancer, Docker, Apache tomcat, Selenium, Spock, Karma, Jasmine, Mockito, Junit, mock MVC, Kubernetes, Dump Analysis, Netflix Zuul, Eureka, AWS.**

**Responsibilities:**

* Using **Agile methodology** where there is a daily **SCRUM** meeting to understand the status and resolve any issues for the development of application.
* Involved in different phases of Software Development Life Cycle (**SDLC**) such as analysis, specification, design, and implementation and testing
* A responsive grid layout was designed for the User Interface **using HTML5, CSS3, jQuery, AngularJS, and XML.  Bootstrap, HTML5** were used to implement a responsive design template to get a new interface.
* Developed Single page application and structured the JavaScript code using **AngularJS**.
* Communication between view components and server was enabled through **AngularJS Ajax module** which is used to handle **RESTful calls**.
* Used **NodeJS** to run **Angular** on server with mock **JSON** data and completed the UI development independent of server.
* **POSTMAN** tool was used to perform the API Web services testing.
* To simplify the complexities of the application a user interface was developed using **JSP**, **JSP Tag libraries**, third party libraries and **Java Script**.
* Communication between different enterprise components was achieved through **Apache Camel** as the Integration Pattern.
* Java based framework was used to test the Automated Application Programming Interface.
* Deployment of the application by **JBoss 6.x Application server**.
* Core Java multi-threading programming and Collection frameworks were used to develop various helper classes.
* Developed Business logic and Entity Beans to manage the Transactions with the database to deliver the content to presentation layer and to store data into the data layer.
* Implemented with **Core Java** with **Multithreading, Concurrency, File Handling I/O, Generics, Data Structures and Collections, Garbage Collection.**
* Developed using Core Java, **Servlets** and **JSP** using Eclipse IDE.
* Developed **Request Handlers** using Servlets as Controlling mechanism.
* Implemented the business requirements using **Spring Boot, Spring Data** and **Spring Security**.
* Using **SOA** Test for developing API automation Scripts.
* Responsible for implementing the application using **Spring Boot** and handling the authentication, authorization, and access- control features by using **Spring Security**.
* Spring boot integration with Apache Camel (**ESB**).
* Development of the End Points, Business Layer, DAO Layer using **Hibernate/JDBC**
* Designed the template using **Spring IOC**.
* **Spring Framework AOP** Module to implement logging in the application to know the application status.
* Database was handled using **Spring DAO Module** along with **Struts Framework**.
* Large number of chunks were handled by batch jobs along with the implementation of **Spring Batch Framework** to execute the similar jobs simultaneously.
* Integration of different applications components through **Web services** using **RESTful** using **Jersey**.
* Adoption of architecture paradigms such as SOA, Web services, Middleware technologies.
* Designed and developed Microservices business components using **Spring Cloud**.
* Performed **CRUD** operations like Update, Insert and Delete data in **MongoDB**.
* In-depth understanding of **MongoDB HA strategies**, including replica sets and/or MongoDB scaling across data centres.
* Creation of the **POJO** objects and mapping using **Hibernate annotations**, Transaction Management and development of the persistence layer using Hibernate Framework.
* Management of people queue for each activity was done by designing and manipulating the **JBPM** process for the client activities.
* Design and generation of the database for the **Flow Control System** of People.
* Involved in technique of mapping data representation from **MVC** model to **Oracle Relational data model** with an SQL based schema using **Hibernate** ORM solution.
* Used **Docker** coupled with load-balancing tool Nginx to achieve Continuous Delivery goal on high scalable environment.
* Created **Azure** services using **ARM** templates and ensured no changes in the present infrastructure while doing incremental deployment.
* Creating, validating, and reviewing solutions and effort estimate of converting existing workloads from classic to ARM based **Azure Cloud Environment**.
* Installed CI/CD automation system. Used **Ant, Maven** as a build tools on java projects for the development of build artifacts on the source code and deployed on Websphere Application Server.
* **Hazelcast** for creation and integration Distributed Caches for clients.
* **JUnit** for unit testing, integration testing and production testing.
* Used **JIRA** to assign, track, report and audit the issues in the application.
* CI setup for the application using the **Jenkins**.
* Installing and configuring **Maven** for application builds and deployment.

**Environment: Azure, Java 1.8, J2EE, Agile(SCRUM), HTML5, CSS3, JavaScript, AngularJS, jQuery, Bootstrap, AJAX, XML, Servlet,  Spring IOC, Spring AOP, Spring DAO, Spring, JBoss 6.2, MVC, Spring Boot, Web services, Hazelcast, REST, SOAPUI, Apache  Camel, SOA, Microservices, MongoDB, Docker, Hibernate, Scala, Websphere, JMeter, JUnit, Log4J, JIRA, Jenkins, STS, Maven  and GIT.**

**Responsibilities:**

* Planning, analysis, design, and implementation of different modules using **Agile** (Scrum) Methodology.
* Involved in Daily **Scrum** meetings, **Sprint** planning and estimation of the tasks for the user stories, participated in retrospective and presenting Demo at end of the sprint.
* Worked on existing **Angular JS** project for better performance and feasibility of the project.
* Used the **Angular JS** framework and Web pack module loader and successfully created a Single Page Application which binds data to a specific view and synchronizes data with server.
* Deep understanding of **JavaScript** and the **jQuery** Framework.
* Experience on working with **CSS Background, CSS Layouts, CSS Positioning, CSS Text, CSS Border, CSS Margin, CSS Padding, CSS Table, Pseudo Classes, Pseudo Elements and CSS Behaviours in CSS**.
* Implemented design patterns in **Scala** for the application.
* Developed the application using JSP for the presentation layer along with JavaScript for the client-side validations.
* Implemented with **Core Java** with **Multithreading, Concurrency, File Handling I/O, Generics, Data Structures** and **Collections, Garbage Collection**.
* Involved in Java Multithreading to cascade multiple changes on multiple records in only one transaction.
* Using the spring open-source framework for building JSP based web application based on the **MVC** design paradigm.
* The application uses the **STRUTS framework**. The views are programmed using **JSP** pages, Model is a combination of EJB’s**,** and controllers are **Servlet**.
* Used **spring framework** for the server-side logic implementation.
* Configured different layer such as presentation layer, server layer, persistence layer of application using Spring IoC and maintained the **Spring Application Framework's IoC** container.
* Using **Spring IOC’s** Dependency Injection developed Java beans with getter and setter properties to be configured by Spring.
* Used **Spring framework's MVC framework** to better control the presentation, business, and navigational aspects of the applications.
* Configured and implemented Spring Application Context to instantiate all bean classes in Spring MVC application framework.
* Effective use of hibernate ORM tool in the business layer in communicating with the **Spring DAO** and developed mapping files and POJO java classes using ORM tool.
* Modified existing and created new **Hibernate objects-based Queries** for data storage, retrieval, tuning and mapping.
* Used Hibernate 3.0 object relational data mapping framework to persist and retrieve the data from database.
* Designed and Developed Module based persistence layer under the guidance of DAO design pattern, **ORM**/**JPA** principles, using Hibernate.
* Implementation of new web services using **SOAP** and **Apache**-**axis** to provide **HRMS** services for the external systems.
* Developed Database Procedures for updating, Inserting, and deleting table for a complete logical Unit of Work (**LUW**) using PL/SQL under Oracle Database Server.
* Implemented asynchronous multi-thread processing of files using **Java Message Service, Message Driven Beans**, PL/SQL, and Triggers.
* Packaged and deployed the application in **JBOSS Application** server in different environments like Development, testing etc.
* Used **log4J** for application logging and notification tracing mechanisms.
* Experience in running web scale services on **AWS**.
* Responsible for bug fix code check-in and integration of the code with other components.
* Used **J-Unit** Testing for testing components.
* Used **CVS** as version control system.

**Environment: Core Java, J2EE, JSP/Servlet, JDBC, HTML, CSS, JavaScript, JQuery, Ajax, Angular JS, Spring, Hibernate 3.0, Web services, AWS, MVC design patterns, UML, XML, JBOSS Server, JMS, JNDI, Maven, Java beans, RMI, Junit, Toad with Oracle, CVS, My Eclipse IDE, UNIX.**

**Responsibilities:**

* Involved in various phases of **Software Development Life Cycle** (**SDLC**).
* Used **Rational Rose** for the Use Case Diagrams, Object Diagrams, Class Diagrams and Sequence Diagrams to represent the detailed design phase.
* Designed web pages by using **HTML, CSS, JSP, Servlets, JSTL, Ajax and Struts**.
* Developed the front-end functionalities using **Angular** with the use of Components, Directory, Services.
* Involved in writing the exception and validation classes using **Struts** **validation rules**.
* Used **JavaScript** for the web page validation.
* Implemented **Core Java OOPS** technology along with Multithreaded environment to resolve the problem of concurrency.
* Also applied **multi-threading** with Java Collections and Generics.
* Performed application-level logging using **log4j** for debugging purpose
* Created **Servlets**, which route submittals to appropriate Enterprise Java Bean (**EJB**) components and render retrieved information.
* Thoroughly involved in testing phase and implemented test cases using **JUnit**.
* Used **JIRA** ticketing system to keep track of issues and tasks on individuals.
* Written **ANT** scripts for building application artifacts.

**Environment: Java, J2EE, Servlet’s, Struts, JSP, XML, DOM, HTML, JavaScript, JDBC, Web Services, Eclipse Plug-ins**