

**OBJECTIVE:**

- Around 8 years of strong software experience in design, development, analysis and deployment of web-based and Client-Server business applications using Object Oriented Analysis and Design (**OOAD, OOPS**) and **Java/JEE/J2EE** technologies in **Financial, Insurance, and Health Sector** Domain with Software Development Life Cycle (**SDLC**).

**PROFESSIONAL SUMMARY:**

- Actively involved in each phase of **Software Development Life cycle** (SDLC).
- Expertise in client scripting language and server scripting languages like **HTML5, CSS3, JavaScript, JQuery, Ajax, AngularJS, NodeJS, JSON, Bootstrap**.
- Utilized **Java 1.8** features like **Lambda** expressions and Stream API for Bulk data operations on **Collections** which would increase the performance of the Application
- Hands on experience with **Amazon** web services (**AWS**) and **Amazon** cloud technologies such as **Amazon EC2** (virtual servers) and **Amazon Cloud Watch** (monitoring).
- Managed Amazon Web Services like **EC2, S3 bucket, ELB, Auto-Scaling, SNS, SQS, AMI, IAM, Dynamo DB, Elastic search, Virtual Private Cloud (VPC)** through AWS Console and API Integration.
- Extensive experience working in **Spring framework, Struts framework, ORM Mapping Hibernate framework and web services**.
- Experience using **Spring MVC, Spring Boot, Spring Cloud, Spring DAO, Spring Data, Spring IOC, Spring Annotations, Spring AOP, Spring Transactions and Spring Security**.
- Experience in developing applications using **Micro Services** architecture.
- Expertise in developing reliable and scalable enterprise applications using **Servlets, JSPs, Struts, JSTL, JDBC, AJAX, EJB and Web Services**.
- Strong work experience in application integration and communicating using **SOA, Web Services** such as **Simple Object Access Protocol(SOAP), Representational State Transfer (REST/ Restful), JAX-RPC, JAX-RS, JAX-WS, WSDL, UDDI, Apache CXF, JAXB, XSD, Axis2**.
- Excellent knowledge and experience in writing SQL queries, Stored Procedures, Triggers and views using various databases such as **MS SQL Server, MySQL, IBM DB2, Oracle** databases and **NoSQL** database systems like **Mongo DB**.
- Expertise in **XML** technologies such as **XSL, XSLT, XML schemas, XPath, XSL-FO** and parsers like **SAX, DOM**.
- Experienced in **JMS** over messaging to exchange the information in more reliable and asynchronous way in enterprise Applications. Used **Apache Active MQ** and **Apache Camel** as **JMS** provider.
- Experience in Creating and configuring the continuous delivery pipelines for deploying **Micro services** and **lambda** functions using **Jenkins CI server**.
- Experience in developing logging and standard mechanism based on **Log4j** and **SLF4j**.
- Worked on implementing full life cycle of software development process (SDLC) in using version control tools like **Git, Rational Clear Case, Tortoise SVN, CVS (Concurrent Version System)**.
- Hands-on experience working with Continuous Integration (**CI**) build-automation tools such as **Jenkins** along with Build Plugins like **Maven, Gradle** and **Apache Ant**.
- Experienced in the formation and mentoring of project teams working on **Agile** methodologies to deliver business critical software applications using tools like **Jira**.
- Experience in web development tools like **Eclipse, RAD, Spring Tool Suite, and IntelliJ**.

- Hands on experience with **Windows, UNIX** and **Linux** Environments.
- Experience in designing, developing, and deploying J2EE applications on **IBM Web sphere, Web logic, Apache Tomcat, Glassfish** and **JBOSS** Servers.

## **TECHNICAL SKILLS**

<b>Languages</b>	C, C++, Java 1.8, J2EE, SQL and PL/SQL
<b>Mark-up/XML Technologies</b>	HTML5, CSS3, Bootstrap, JavaScript, jQuery, AJAX, Angular JS, Node JS, XML, XSD, XSL/XSLT, SAX/DOM, DOJO and JSON.
<b>Tools &amp; Framework</b>	Struts, Spring, Hibernate, Spring MVC, Spring Web Flow, Spring IOC, Spring AOP, Spring Security, Spring Boot, JSF, JUnit, Apache CXF, SOAP UI, GWT, Jersey.
<b>Build Automation</b>	ANT, MAVEN, Jenkins, Gradle.
<b>Web services</b>	SOAP, RESTful, UDDI, WSDL, JAX-RPC, JAX-RS, JAX-WS, JAX-B, Micro services, Apache Axis and Apache CFX.
<b>Cloud Technologies</b>	Amazon Web Services (EC2, SQS, RDS, IAM, S3).
<b>Web/App Servers</b>	Web Sphere, Apache, Apache Tomcat, JBOSS EAP, Web Logic Server and Apache Camel
<b>Databases</b>	Oracle, SQL-Server 2005, MySQL server, MS SQL, IBM DB2, NoSQL DB like MongoDB
<b>IDE tools</b>	Eclipse, Net Beans, Spring Tool Suite (STS) and IntelliJ IDEA, RAD
<b>Modeling/ CASE Tools</b>	Rational Rose and MS Visio Professional.
<b>Testing Tools/ Others</b>	JUnit, TOAD, Soap UI, Putty, Rational Clear Quest, RTC, Load UI and Mockito.
<b>Version Control</b>	Tortoise SVN, CVS, Rational Clear case, Source Tree and Git.
<b>OS &amp; Environment</b>	Windows, Linux, UNIX, Mac
<b>Design Patterns</b>	Singleton, Factory, Prototype, Business Delegate, Session Facade, Service locator, Visitor, Observer, Decorator, Front Controller and DAO pattern.

## **PROFESSIONAL EXPERIENCE**

<b>Client: Capital One</b>	<b>Mar 17 – Till date</b>
<b>Location: Mclean, VA</b>	
<b>Role: Full Stack Java Developer</b>	

**Description:** Capital One is the one of largest banking sector, it has over 480,000 members and is among the top 20 credit unions in the U.S.A. Capital One Home Loans servicing is a diversified bank that offers a variety of checking, savings, mortgage and lending accounts and services to individuals, small businesses and commercial clients.

### **Responsibilities:**

- Involved in the complete **SDLC (software development life cycle)** of the application from requirement analysis to testing.
- Followed **Agile Methodology** in analyze, define, and document the application, which will support functional and business requirements. Coordinate these efforts with Functional Architects.

- Extensively worked on **Responsive Web Design (RWD)** page development using **HTML5, CSS3, JQuery, JavaScript, AngularJS, and JSON**.
- Utilized **Java 1.8** features like **Lambda** expressions and Stream API for Bulk data operations on **Collections** which would increase the performance of the Application.
- Used **Java 1.8** features in developing the code like **Lambda** expressions, creating resource classes, fetching documents from database.
- Deployed and Monitored **Micro Services** Using Pivotal Cloud Foundry, also Managed Domains and Routes with the Cloud Foundry.
- **Designed, Configured and deployed Amazon Web Services (AWS)** for a multitude of applications utilizing the **Amazon cloud formation**.
- Experienced in setting up **Amazon EC2** instances, virtual private cloud (VPCs), and security groups. Setting up databases in **AWS** using **RDS**, storage using **S3** bucket and configuring instance backups to **S3** bucket.
- Implemented the application using **Spring IOC, Spring MVC Framework, Spring Batch, Spring Boot** and handled the security using **Spring Security**.
- Developed **RESTful API** for assessment indicators module and token based authentication for **RESTful** services using **Spring Security**.
- Involved in building database Model, APIs and Views utilizing **Python**, to build an interactive web based solution.
- Used **Jersey** to implement **Restful Web Service** and used **XML** form to transfer the data.
- Developed the application using **J2EE Design Patterns** like Business **Delegate, Factory, Singleton, Session Facade, Service Locator** and **DAO**.
- Worked and involved in deployment of core platform technologies, techniques, and web app frameworks such as **spring, JBOSS, Tomcat, JSON, XML, HTML5, and Web Services**.
- Used **Spring MVC module** to develop **MVC Architecture**.
- Worked on **Apache Camel, RabbitMQ** for sending messages over queue.
- Deployed the Application into **Docker** container and made easily accessible at runtime using Cloud Foundry and other cloud services like **AWS, Netflix Eureka**.
- Used **Spring Security and OAuth2.0** for **Authentication and Authorization** of the application.
- Implemented **Spring AOP** for declarative transaction management.
- Developed a **AWS Lambda** function to send a request for internal service end points through **API Gateway** using apache HTTP Client.
- Managed code versioning with **GitHub** and deployment to staging and production servers
- Used **Rest Controllers** to replace the existing **operations layer**, which acts as bridge between **business layer** and the **UI**.
- Used **Jenkins** as build management tool for continuous integration process.
- Configured **pom.xml** to deploy the project using **Maven**.
- Provided Technical support for production environments resolving the **issues, analyzing the defects**, providing and implementing the solution defects.

**Environment:** Agile Methodology, Java1.8, Lambda, J2EE, HTML5, CSS3, JavaScript, JQuery, AJAX, AngularJS, Bootstrap, JSON, JSP, AWS, Micro Services, Oracle, RabbitMQ, Spring AOP, Hibernate, Spring Cloud, Spring MVC, Spring JDBC, JDBC, Web-Services, SOA (Service-oriented) Architecture, Redis, REST, JAX-RS, Jersey, JUnit, JAX-B, WebSphere, JIRA, Maven, GIT, RAD, Jenkins, MongoDB, Cassandra,.

**Client: State Farm**

**Jan 16 – Feb 17**

**Location: Dallas, TX**

**Role: JAVA Full Stack Developer**

**Description:** Auto Quote Purchase (AQP) application is one of the many applications that are part of SF Auto Systems. Property and Causality being the main forte of State Farm's business the maintenance of all the applications coming under the Auto Systems umbrella is crucial. Auto Quote Purchase (AQP) is one of the many applications that are part of SF Auto Systems. P&C Application support involves production support, defect fixes and minor enhancements based on the SF's requirement. AQP is a web application which is used to gather customer information, generate quote and provides options to purchase the Auto policies.

**Responsibilities:**

- Involved in **SDLC requirements gathering, analysis, design, development and testing** of application developed using **AGILE** methodology.
- Created user-friendly GUI interface using **HTML5, CSS3, JavaScript, JQuery, AngularJS**.
- Implemented **persistence framework using Hibernate** & Handled **Transaction Management** using the provided data source.
- Responsible for designing **Hibernate mapping files** based on business logic and Object relationships.
- Developed restful web service APIs using Spring Framework such as **Spring Core, Spring MVC, Spring AOP and Java Persistence API** with a backend **PostgreSQL and Cassandra** database.
- Involved in developing **Web Services** to send **XML** request and reading the response **SOAP UI** from Java platform.
- Consumed **Web Services** to interact with other external interfaces to exchange the data in the form of **XML** and by using **SOAP**.
- Created **Restful Web services** using **Jersey API (JAX-RS)** while data exchange was carried out between application and services using **JSON**.
- **Spring MVC, Ajax and Dependency Injection** for handling presentation and business logic.
- To maintain **loose coupling between layers** published the business layer as services and injected the necessary dependent components using **Spring IOC** and published cross cutting concerns like Logging, User Interface exceptions, Transactions using **Spring AOP**.
- Integrated **spring** and **hibernate** together and worked on developing backend components and services using Hibernate and spring.
- Extensively used **HQL and SQL** for querying databases.
- Developed **Message Driven Bean** for asynchronous sending Messages using **JMS**.
- Used **Apache Camel** Framework to transform, extract and load the data.
- Used Spring Core Annotations for Dependency Injection and used **Apache Camel** to integrate spring framework.
- Implemented request and response objects using SOAP web services and created routes using **Apache Camel**.
- Used **Rest** methodology for the service layer interface and used **JSON** for the data transformation.
- Developed various **AJAX** and **JSON** calls for the communication between the server and the **UI**.
- Developed **web services** for produce and consumed the services by making **Web Service Calls** to remote services in different applications.
- Used **Spring JMS** module for lookup for the queues and MDBs for the listeners
- Established Database Connectivity using **Hibernate O/R mapping with Spring ORM for MySQL Server**.
- Developed the **Mapping Resource Files** for the database entities.

- Packaged and deployed the application in **IBM Web sphere**.
- Used **IntelliJ** as IDE to develop the application and followed the standard features for debugging and running.
- Used **log4J** for application logging and notification tracing mechanisms.
- Analysis and Bug fixing of the production problems and defects along with enhancements
- Used **Rational Clear Case** for the version control
- Used **Maven** as the build tool.
- Used **J-Unit** Testing and **Ant Scripts**.
- Used **JIRA** for tracking the **Project Stories** in AGILE Methodology.

**Environment:** Agile, Java, JDK, J2EE, HTML5, CSS3, Java Script, JQuery, Ajax, AngularJS, Mongo DB, XML, Spring, Apache Camel, MVC design patterns, Spring MVC, Hibernate, Ant, Maven, Netflix, Rational Clear Case, Web sphere Server, IBM Web Sphere, Apache Camel, SOAP UI, JNDI, Log4J, PostgreSQL, MySQL Server, Google cloud platform, JIRA, RMI, IntelliJ, IDE, LINUX.

**Client:** Cigna Healthcare

**Nov 13– Dec 15**

**Location:** Windsor, CT

**Role:** Java Developer

**Description:** One view is a web based application helping Customer Services people to view Benefit Plan information for Customers, health care providers. The application helps to resolve; track customer queries online by reducing overall time. The current project is reengineering the old application by replacing outdated Cordiant service layer with Restful Services and Spring Integration frame work improving performance and scalability.

#### **Responsibilities:**

- Designed and developed the **REST** based **Micro Services** using the **Spring Boot**.
- Writing end-to-end flow i.e. controllers classes, service classes, **DAOs** classes as per the **Spring MVC** design and writing business logics using core java API and data structures
- Design, develop & deliver the **REST APIs** that are necessary to support new feature development and enhancements in an agile environment.
- Used Spring **JMS** related MDB to receive the messages from other team with **IBMMQ** for queuing
- Developed presentation layer code, using **JSP, HTML, AJAX** and **JQuery**
- Developed the Business layer using spring (**IOC, AOP**), **DTO**, and **JTA**
- Developed the Persistence layer **DAO** using **HIBERNATE**
- Worked with Open **symphony** workflows and delivered work flow module to the client.
- Developed the project using industry standard design patterns like Singleton, Business Delegate Factory Pattern for better maintenance of code and re-usability
- Developing application module coded on Spring Batch and **Apache Camel**.
- Written **Jenkins** pipeline template script in **Groovy** programming language for the automation of build and deployment process.
- Created **groovy** scripts to extract transaction volume information and email to management on daily basis.
- Developed unit test cases using **Junit** framework for testing accuracy of code and logging with SLF4j + Log4j
- Worked with defect tracking system Clear Quest
- Worked with IDE as Spring STS and deployed into spring tomcat server, WebSphere 6 & used Maven as build tool

- Responsible for code sanity in integration stream used Clear Case as version control tool.

**Environment:** Java, J2EE, HTML, CSS, AJAX, JQuery, MongoDB, Apache Camel, symphony, JavaScript, JSP, Spring, Spring Batch, Spring web services, XML, UML, JUNIT, IBM WebSphere, Maven, Clear Case, SoapUI, Oracle 11g, Spring JMS, IBM MQ.

**Client:** Liberty Mutual

**Jan 12 - July 13**

**Location:** Seattle, WA

**Role:** Java Developer

**Description:** Liberty Mutual Insurance is an American diversified global insurer, and the second-largest property and casualty insurer in the United States. Ecommerce application provides services such as Auto Insurance and home insurance. The application is used for accessibility into the appropriate Agent Center applications and generating quotes and policies, payments and reports.

#### **Responsibilities:**

- Involved in all phases of the **Software development life cycle (SDLC)**.
- Designed project related documents which includes Use case, Class and Sequence diagrams.
- Involved in design, development, testing and implementation of the process systems, working on iterative life cycles business requirements, and creating Detail Design Document.
- Using **Agile** methodologies to plan work for every iteration and used continuous integration tool to make the build passes before deploying the code to other environments.
- Extensive experience in developing **PL/SQL** Stored Procedures, Functions, Packages, Cursors, Collections (Nested tables and arrays), Records, Object types and Database Triggers.
- Created **HTML** Wireframes based in the Detailed System Specification Document Developed the presentation layer of the application using **MVC** architecture of **spring** framework.
- Created **DAO** classes to interact with **DB2** database.
- Used Spring **ORM** module to integrate with **Hibernate**.
- Used **JMS** API for sending and receiving the messages on the queue.
- Used the **spring validation** and **Tiles Framework** in the presentation layer.
- Developed user interface using **JSP**, **Spring Tag Libraries** to simplify the complexities of the application.
- Developed user interface with **Java Server Pages**, **Ajax** support to speed up the application form filling.
- Used **Apache Axis** as the Web Service framework for creating and consuming **Web Service** clients using **SOAP** and **WSDL**.
- Used **Rally** for tracking purpose.
- Designed various tables required for the project using **DB2** database. Used **JBOSS** server for deploying the application.
- Used **Ant** for build, create JAR, WAR, EAR files and deploy the EAR files automatically.
- Used **JUnit** to test the application classes.
- Used **Log4J** to capture the logging information.

**Environment:** Java, J2EE, JavaScript, JSP, AJAX, XML, Agile, Spring, Eclipse IDE, DB2, Rational Rose, Log4j, Junit, Ant, JBOSS, Rally, Apache Axis, Web Service, SOAP, WSDL.

**Client:** Soft Pro Systems Ltd.  
**Location:** Hyderabad, India  
**Role:** Java Developer

**Sep 10 - Dec 11**

**Description:** This project is mainly an Intranet application meant for employees in the organization. They are Intranet mailing System, Chatting, File Transfer and Remote Login. This project is based on the client server methodology.

**Responsibilities:**

- Involved in the **design** and **development** of an entire Module – Guided Set Up. Developed user interface through **JSP** and **Servlets**.
- Developed Action classes and Action Forms using the **struts framework**.
- Developed Customs tags to display dynamic contents and to avoid large amounts of java code in **JSP** pages
- Connected to database through **JDBC**.
- Extensive experience working with **XML (DOM and SAX)**.
- Used SQL-server to store the items in the database.
- Used **Session beans** for navigation of steps.
- Used **Entity beans** for storing the database in to database. Developed **Session Beans** as the clients of Entity Beans to maintain the Client state.
- Developed **Ant** Scripts to build and deploy **EAR** files on to **Tomcat Server**. Analyzed the **EJB** performance in terms of scalability by various **Loads, Stress tests using Bean- test tool**.
- Extensively used **Eclipse** while writing code as IDE. Written **complex SQL queries, stored procedures, functions and triggers in PL/SQL**.
- Developed test cases and used **Junit** for Unit Testing.
- Used **CVS** for version controlling.

**Environment:** Java, J2EE, Servlets, XML (DOM and SAX), JSP, EJB, JavaScript, CSS, JDBC, Struts, Ant, Web Logic Application Server, WSAD, Eclipse IDE, SQL-Server, Apache Tomcat, Oracle, PL/SQL, CVS, PVCS, Junit, Windows.

**Client:** Login IT Technologies  
**Location:** Hyderabad, India  
**Role:** Java Developer

**May 09 - Aug 10**

**Description:** PRS or Price reporting system is responsible to publish the rates to the customers and as to maintain the users across the FedEx locations and its subsidiaries. It interacts with the customer fusion and existing legacy systems to access the customer information and legacy data.

**Responsibilities:**

- Developed web application using **Struts 2.X, JSP, Servlets, Java beans** that uses MVC design pattern
- Created user-friendly GUI interface and Web pages using **HTML, CSS** and **JSP**
- Wrote JavaScript for client-side validation.
- Used Eclipse as IDE tool for creating **Servlets, JSP, and XML**.
- Wrote SQL for JDBC prepared statements to retrieve the data from database.
- Monitored the logs using **Log4J** and fixed the problem

- The application was deployed on the **Tomcat**.
- Involved in build and deploying the application using ANT
- Used **CVS** as version control system
- Worked on bug fixing and Production Support

**Environment:** Java, JDBC, Struts, JavaScript, HTML, CSS, JSP, Servlets, Tomcat, Oracle, Eclipse IDE, CVS, ANT, Log4J.