Candidate87

(720) 722-0838

candidate871702@gmail.com

Objective:

To work in the most challenging position with an organization that provides ample opportunities to learn and to contribute for the growth of an organization. Possess **8 years** of professional IT experience in designing and developing Web based and Enterprise applications using **JAVA/J2EE** technologies in Telecom, Financial, Insurance and Digital domains.

Professional Summary:

- Experienced in all phases of **Software Development Life Cycle (SDLC)** including Analysis, Design, and Development, Debugging, Support, Integration, Maintenance, and Enhancements of Java/J2EE Applications.
- Experience in working in environments using **Agile (SCRUM)** and **Waterfall** development methodologies.
- Expertise in client and server scripting languages like **HTML5, CSS3, JavaScript, Bootstrap, JQuery, Angular JS, ReactJS, Backbone.js** and **Node JS.**
- Experience with **Java 8** features such as **lambda expressions, Parallel operations** on **collections** and for effective sorting mechanisms and new Date Time API.
- Handled operations and maintenance support for **AWS** cloud resources which includes launching, maintaining, and troubleshooting **EC2 instances**, **S3** buckets, **Auto Scaling**, **Dynamo DB, AWS IAM**, Elastic Load Balancers (**ELB**) and Relational Database Services (**RDS**). Also created snapshots for data to store in **AWS S3.**
- Experience in developing and solid understanding of **Service Oriented Architecture** and **Microservices Architecture.**
- Strong hands on experience in implementing **Design patterns** like **MVC, Data Access Object, Session Façade, Business Delegate, Factory, Proxy,** and **Singleton.**
- Experience in developing web-based enterprise applications using **Core Java, J2EE, Servlets, JSP, JSF, JDBC, Hibernate, Web Services (SOAP and REST), MVC Framework (Struts, Spring MVC)** and **SQL.**
- Experience in both providing and consuming Web Services using **SOAP** and **RESTful** (**JAX-RS** and **JAX-WS**).
- Experience using persistence framework like **Hibernate**, **JPA** for mapping java classes with Database and using **HQL** (Hibernate Query Language).
- Strong experience in **Core Java** concepts like **Collections, Multithreading, Event Handling, Exception Handling,** and **Generics.**
- Experience with different **Spring framework** modules like **Spring Core, Spring AOP**, **Spring IOC, Spring MVC**, **Spring Boot, Spring Batch**, **Spring Security,** and **Spring cloud.**
- Experience in design and development of robust and highly scalable web service platforms and API's based on **REST** and **SOAP** for e-commerce applications using **AXIS, WSDL, UDDI, XML, JAX-WS, JAX-RS, JAXB, JAX-RPC** technologies and **Jersey** frameworks.
- Expertise in **XML** technologies such as **XSL, XSLT, XML schemas, XPath**, **XForms, XSL-FO and parsers like SAX.**
- Experience on CQL (Cassandra query language), for retrieving the data present in **Cassandra** cluster by running queries in CQL.
- Proficient in writing and handling **SQL Queries**, **PL/SQL**, **Stored Procedures,** and **Triggers.**
- Good experience with **Relational** databases like **Oracle, My SQL, MS SQL Server, IBM DB2** and **No SQL databases like Mongo DB** and **Dynamo DB.**
- Experienced in implementation of **JMS** to exchange information over reliable channel in an asynchronous way by using **Active MQ, Rabbit MQ** as a message queue.
- Experience with application servers like **Web Logic, JBOSS, Apache Tomcat** and **IBM Web Sphere.**
- Good experience writing application deployment scripts using **Shell Scripting**. Worked on **Windows**, **UNIX, LINUX**

- platforms to move the project into production environment.
- Experience with **Junit, Jasmine, Karma, Mockito and Log4J** in developing test cases.
- Working knowledge with version control tools like **SVN, CVS, GIT,** and **Visual Source Safe**.
- Hands on experience in building tools like **ANT, Maven, Gradle**.
- Experienced in developing Rich GUI's by using **Eclipse RCP**, HTML, frameworks.
- Highly experience with Java complaint **IDE's** like **Spring Tool Suite**, **Eclipse**, **IntelliJ,** and **RAD.**

Technical Skills

Languages	Java 8, J2EE, SQL, and PL/SQL
Enterprise Java	JSP, Servlet, JNDI, JDBC, Java security and JSF
Mark-up/XML Technologies	XML, XSD, XSL/XSLT, SAX/DOM, JSON
Web Technologies	HTML5, CSS3, Java Script, JQuery, Ajax, Bootstrap, AngularJS, ReactJS and NodeJS, Express JS
Tools & Framework	Struts, Spring, Hibernate, Spring MVC, Spring HATEOAS, Spring Web Flow, Spring IOC, Spring AOP, JMS, JSF, Prime faces, Log4J, Groovy, Spring Boot, Spring Cloud, Spring Batch, Spring Security.
Web services	SOAP, Restful, UDDI, WSDL, JAX-RPC, JAX-RS JAX-WS, JAX-B, Axis2, Jersey, Microservices Architecture
Web/App Servers	Web Sphere, Apache Tomcat, JBOSS, WebLogic
Database	Oracle, My SQL, SQL Server, IBM DB2, Mongo DB, Cassandra and Dynamo DB.
Development Tools	Eclipse, Net Beans, IntelliJ, RAD and Spring Tool Suite(STS).
O-R mapping	Hibernate, MyBatis and JPA
Testing Tools/ Others	JUnit, Soap UI, Mockito, Log4j, Jasmine, Karma and JIRA
Version Control	CVS, SVN, GIT, Rational Clear Case
OS & Environment	Windows, UNIX, Linux, and Mac
Cloud Technologies	AWS EC2, S3, Elastic Beanstalk EBS, Elastic Load Balancing ELB, API-Gateway, Cloud Foundry

Certifications

- AWS Certified Developer – Associate (KDNJ3B41CBQQQXSF)　　　　　　　　　　　　Nov '17

Professional Experience:

Client: BCBS, Chicago, IL	**Jan'17 – Till date**
Role: Full Stack Developer	

Description: Well Mark Blue Cross and Blue Shield is a leading insurance company owned by its policyholder. BCBS is the health insurance and health care provider company. "Find a plan" is a web based J2EE application, created to help individuals and families easily shop for, and buy health insurance. Well mark offers a variety of insurance plans, as well as tools to help you choose the plan that's right for you.

Responsibilities:
- Involved in various phases of Software Development Life Cycle (**SDLC**) and worked in a highly dynamic **AGILE** environment and participated in Scrum and Sprint meetings.
- Developed **UI** using **HTML5, CSS3, JavaScript, Bootstrap, JQuery, Angular JS** and **Node JS.**
- Built **AngularJS** modules, controllers, pop up modals, and file uploaders.
- Created reusable directives, factories, and services in **AngularJS.**

- Used **Backbone.js** and **ReactJS** to create Controllers to handle events triggered by clients and send request to server.
- Used plug-in development for **RCP** version for developing the layouts like GridLayout and Fillout for RCP.
- Designed and implemented application using **Spring Boot, Spring MVC, Spring IOC, Spring Annotations, Spring AOP, Spring Transactions, Spring HATEOAS, Spring Cloud** and **Spring JDBC.**
- Implemented the authentication and authorization of the application using **Spring Security** and **Oauth2.**
- Designed and developed **Microservices** business components and **RESTful** service endpoints using **Spring boot**.
- Implemented **RESTful** and **SOAP** based web services using for JAX-RS and JAX-WS respectively.
- Implemented first-hand the best/worst practices for integrating **Microservices** into existing system architecture.
- **Microservices** were being used as a strategy to gradually replace a core monolithic application while still adding business features.
- Worked on integrated **Hystrix** dashboard to monitor **spring micro-services** using **Circuit Breaker**.
- Implemented **Multithreading, Concurrency, Exception Handling, and Collections** whenever necessary.
- Prepared calls through **Hibernate** and **Spring JDBC** for Interacting with the Database.
- Experienced with **AWS** services to smoothly manage application in the cloud and creating or modifying the instances.
- Participated in troubleshooting issues on **AWS**, **Netflix Eureka,** and Cloud environments.
- Authored application using **Spring Cloud** services (spring version of Netflix OSS-Eureka, Circuit Breaker, and Ribbon).
- Involved in writing Java API for Amazon **Lambda** to manage some of the **AWS** services.
- Participated in creation of Spring Configurations using **Java8** features, **Lambda expressions, Streams, filters.**
- Worked **Docker** container management along with volumes, container based DB and services, **Docker** Artifactory configuration and setup.
- Experience in writing complex **PL/SQL** queries using joins, Stored Procedures, Functions, Triggers, Cursors, and indexes in Oracle DB.
- Performed data extraction and loading the JSON data with the server-side content in to the **Mongo DB**.
- Configured **JMS** Server setup and DB Connection pool setup and deployed Returned items in **Web Sphere Server**.
- Used **Active MQ** to communicate with controller and used service layer to implement the business logic.
- Implemented Batch jobs using **Spring Batch** created various item readers, item writers and item processors.
- Involved in support for end to operations support for **OSS/BSS** (Business Support System) identify gaps, and propose solutions and manage issue resolution.
- Validations are done using **SoapUI** and **Ready API** testing tool framework in different environments.
- Worked with **Spring Ribbon and Kafka** broker service to handle heavy traffic.
- Deployed and tested the application on **Web Sphere Application Server**.
- Developed test cases for unit testing in the integration layer using **Mockito** framework.
- Used **UNIX** shell scripts to search/modify files.
- Setup the **(CI)** and **(CD)** process for the application using the **Jenkins.**
- Installed and deployed monitoring solution with **Splunk** services.
- Worked on **Log4J** to validate functionalities and **JUnit** for unit testing.
- Implemented configured and customized **JIRA** issue tracking software with agile plug-in across all company functions.
- Used **GitHub** to maintain the version of the files and took the responsibility to do the code merges from branch to trunk and creating new branch when new feature implementation starts.

Environment: Java1.8, J2EE, HTML5, CSS3, Bootstrap, JQuery, Angular JS, ReactJS, JavaScript, Node JS, Eclipse Rich Client Platform, Spring IOC, Spring MVC, Spring Security, Spring Boot, Spring Batch, Spring HATEOAS, Spring Data, Hibernate, Active MQ, RESTful, Postman, SOAP, Micro Service, AWS, EC2, Docker, Mongo DB, Oracle DB, Maven, STS, Web Sphere Application Server, Mockito, JMS, UNIX, Jenkins, JIRA, log4j and GIT.

Client: Capital One, Wilmington, DE   **Feb'16 – Jan'17**

Description: The move money services help general users and investment users, to carry monetary transactions among friends, family and to manage loans and pay bills. The project consisted of developing and maintaining services of Capital One move money services like bill pay, home loan accounts, mortgage accounts etc. the services were developed maintained and tested for the devices and the web. We worked on the architectures, services and the front-end UI for each of the applications.

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.
- Upgraded existing UI with **HTML5, CSS3, jQuery** and **Bootstrap** with **AngularJS** interaction.
- Designed, developed a custom, single-page, responsive web application on front end using **AngularJS**.
- Developed custom directives, Factories and Services in **AngularJS** and created **AngularJS** Controller to isolate the scopes to perform operations.
- Created the **AngularJS** app using **Yeoman, Grunt** and **Bower** and used **Angular UI** for component model.
- Used **JavaScript, jQuery** for user interactions in pages. Creating a responsive web application integrating **Twitter Bootstrap** with **Angular JS** and **media queries.**
- Added user validations and user permissions by using the **NodeJS** with libraries **Express JS**.
- Implemented Grails TagLibs to avoid **groovy** code in GSP pages.
- Used **Grails** command objects for validating non-persistent objects.
- Developed an **API** to write **XML** documents from a database. Utilized **XML** and **XSL** Transformation for dynamic web-content and database connectivity.
- Designed the application using various **Spring** technologies and used **Spring Security** to provide better authentication instances.
- Developed **SOAP** based **JAX-WS Web** Service interfaces for integrating multiple systems, code re-usability.
- Developed **Restful** Web services for other systems to interact with our system and secured the service with Spring-Security Oauth-2.0. Added WS-Security using **XWSS** to secure the **Web services.**
- Used **AWS SDK** for connecting to **Amazon S3 buckets** and **EC2** to deploy on **Docker** Container with **Microservices** architecture.
- Developed **AWS Lambda Functions** in a stateless style and ensure there is no affinity between and compute infrastructure to manage the servers and run the code in the **AWS.**
- Developed services that utilize the **Cloud Foundry** and Azure client libraries (SDK) for Java.
- Deployed application using Pivotal Cloud Foundry (PCF) CLI.
- Used **Micro-services** to ensure the scalability of application on cloud infrastructure**.**
- Refactored existing monolithic application by identifying components that needs to be transformed to **Micro-services** architecture, with **Spring Boot**-based services interacting through a combination of **REST** and **AWS API-Gateway**.
- Installed secured and configured **AWS** cloud servers and virtual servers using **Spring Cloud** for AWS.
- Analysis the logs data and filter required columns by **Log stash** configuration and send it to **Elasticsearch**.
- Developed the persistence layer using **Hibernate** Framework, created the **POJO** objects and mapped using **Hibernate annotations and Transaction Management.**
- Implemented **Web-Services** to integrate between different applications components using **Restful** web services by using **Jersey** for **JAX-RS** implementation.
- Developed **SOAP** based **JAX-WS Web** Service interfaces for integrating multiple systems, code re-usability.
- Worked on **MongoDB** database concepts such as locking, transactions, indexes, Sharding, replication, schema design, etc.
- Extensively used **JSON** to parse the data from server side to satisfy the business requirement.
- Used **WebSphere** server to route our **JMS** queue messages to different business floors and configured routes in

WebSphere.

- 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**.
- Defined several **UML** diagrams; Activity diagrams, Sequence, and Object diagrams.
- Developed unit and integration tests for existing microservices using **JUnit, Mockito** and **Docke**r environments.
- Involved in installing and configuring **Maven** for application builds and deployment.
- Configured and customized logs using **Log4J and** Used **GIT** as a version control tool. Used **JMeter** to test performance of **SOAP** protocol.
- Worked on Creating **Kafka** Adaptors for decoupling the application dependency.
- Used **JIRA** to assign, track, report and audit the issues in the application.
- Setup the **CI** and **CD** process for the application using the **Jenkins.**
- Involved in **System Testing, Integration Testing, and Production Testing** on **WebLogic** Server.

Environment: HTML5, CSS3, JavaScript, Bootstrap, AngularJS, NodeJS, Express JS, jQuery, AJAX, Agile(SCRUM), Groovy, Grails, Hibernate, Spring, Struts, MongoDB, Spring Batch, Spring Boot, Spring Cloud, Spring Security, Servlet, JSON XML, GIT, Spring MVC, Apache Camel, JUnit, Maven, JMS, WebSphere, Restful, JAX-RS, Log4J, AWS, Docker, Microservices**,** Kafka, Elasticsearch, Logstash, JIRA, Jenkins, JMeter.

Client: Sprint, Kansas City, MO	May'14 – Dec'15
Role: Java/J2EEdeveloper	

Description: Sprint Synaptic Storage as a Service is a web services-based storage solution that easily scales up and down to any size you need and allows you to pay only for the storage you use. It provides elastic capacity on demand with no commitments, minimums, or limits and allows access to your stored data from virtually anywhere and anytime.

Responsibilities:
- Involved in complete development of '**Agile Development Methodology/SCRUM**', developed and tested the application during various iterations.
- Developed screens using **HTML**, **CSS** and **JQuery.**
- Developed the User Interface using **Spring** framework, **JQuery** and **Ajax.**
- Implemented routing logic and navigation from screen to screen and implemented login functionality on the client side in **AngularJS.**
- Developed and configured Jasmine test cases to test the Angular controllers for UI in **Jasmine** framework (JavaScript Testing Framework) in Eclipse IDE.
- Handled Java **multithreading** part in back - end component, one thread will be running for each user, which serves that user.
- Used **spring** frame work Spring **AOP** features and **JDBC** module features to persist the data to the database for few applications.
- Developed the persistence layer using **Hibernate** Framework by configuring the mappings in hibernate mapping files and created **DAO** and **PO**.
- Developed various **Java beans** for performance of business processes and effectively involved in Impact analysis.
- Developed application service components and configured beans using Spring **IOC**, creation of **Hibernate** mapping files and generation of database schema.
- Implemented a highly-scalable vertical and horizontal, performance efficient **Drools architecture.**
- Installed and configured **JBOSS server** with MYSQL persistence. Extracted rules data from the database and decision tables.

- Created **SOAP** web services interface to Java-based runtime engine and accounts.
- Hands on experience on **Cassandra architecture** with better understanding and write processes including mem-table, stable and commit log.
- Actively involved in writing **SQL** using **SQL** query builder.
- Worked on **MYSQL** databases and **Oracle** databases technologies.
- Used **Maven** as build tool and deploying the application.
- Actively used the defect tracking tool **JIRA** to create and track the defects during **QA** phase of the project.
- Used **SVN** to maintain the version of the files and took the responsibility to do the code merges from branch to trunk and creating new branch when new feature implementation starts.
- Developed test cases using **Junit**.

Environment: SCRUM, SDLC, Maven, HTML, CSS, JQuery, Jasmine FrameWork, Spring, Ajax, AngularJS, AOP, JDBC, Spring IOC, Hibernate, DAO, PO, Java beans, SOAP, SQL, Cassandra, JIRA, QA, SVN, Junit, JBOSS Server, Drools.

Client: State Farm, South Houston, TX	**Aug'11 – Dec'13**
Role: Java developer	

Description: State Farm is a group of insurance and financial services companies in the United States. The group's main business is State Farm Mutual Automobile Insurance Company, a mutual insurance firm that also owns the other State Farm companies.

Responsibilities:
- Involved in various phases of Software Development Life Cycle **(SDLC)** such as requirements gathering, modelling, analysis, design, development, testing, and monitoring.
- Involved in the front end using **JSP, HTML, CSS, JavaScript** and **JQuery**.
- Developed GUI using **SWT/RCP** technology.
- Experience in working with different plug-ins of **Eclipse RCP.**
- Developed different **GUI** screens using widgets like Tree and Table form in Eclipse SWT/RCP.
- Developed perspective to show different views together in **RCP.**
- Developed Menus and views using Extension point in RCP.
- Implemented **MVC** architecture using **JSP, Spring, Hibernate** and used **Spring Framework** to initialize managed beans and services.
- Involved in development of Agent Verification System using **Spring MVC** framework.
- Used **Spring Inheritance** to develop beans from already developed parent beans.
- Used **Spring AOP** for **logging**, **auditing,** and **transaction management** to distinguish **business logic** from the cross-cutting concerns.
- Used **Spring Security** for **Authentication and Authorization** of the application.
- Created **data model** and generated **Hibernate mappings** and **domain objects** using Hibernate tools.
- Interfaced with the **My SQL** back-end database by integrating **Spring** with **Hibernate**.
- Extensively used **hibernate** named queries, criteria queries, Hibernate Query Language (**HQL**) and Optimistic Locking and Caching to process the data from the database.
- Developed Unit /Integration test cases using **JUnit**.
- Used **Gradle** tool for building and deploying the Web applications in **Web Logic Server.**
- Used **JIRA** tool for tracking stories progress and follow agile methodology.
- Used logging techniques provided by **Log4J** tool for efficient logging and debugging.
- Developed the application using **Eclipse** as the **IDE** and used its features for editing, debugging, compiling, formatting, build automation and **version control (SVN).**
- Involved in Maintenance and Enhancements for the project.

Environment: Java, J2EE, HTML, CSS, JavaScript, JQuery, Struts, Spring IOC, Spring MVC, Spring AOP, JDBC, Hibernate, My SQL, HQL, SQL, JUnit, Gradle, JIRA, Log4J, Eclipse, SVN, Sybase, RCP and Web Logic Server.

Client: Techno soft Solutions	**Sep'10 – Aug'11**
Role: Java/J2EE Developer	

Description: Techno soft provides a range of professional services delivered by experts with operational experience building and running the largest E-Mail, messaging, and Internet networks. These services range from architecture and design of software solutions, website design & development to deployment services and operational support.

Responsibilities:
- Developed front-end screens using **JSP, HTML, CSS, JavaScript, JSON**.
- Implemented **Struts MVC** design pattern and front controller pattern along with **Action Servlet** as front controller for this application.
- Developed **SCM** by using the **JSP/HTML** like one form for each functionality user interface, standard validations using the **JavaScript**, Servlets used as the controllers for the business logic and business logic using **JDBC**, **XML** parsing techniques etc using **MVC.**
- Developed **SingleSignOn** (SSO) functionality, through which we can run SCM from Oracle Applications.
- Involved in developing the **Routings** and configured **Routing Program** as scheduled the concurrent request.
- Developed **Server-Side components** for the business services for creating Items, BOM, Sourcing Rules, and substitute.
- Involved in raising the Notifications to **Oracle** Users through Mailing Concept for intimating to start the next process using workflow.
- Extensively worked on creating the setups for **Organizations, Templates, Concurrent Requests, Cross Reference Types, User Creations, assigning responsibilities, creating value sets, Descriptive Flex Fields** etc, in Oracle Applications.
- Used **CVS** as version control system.

Environment: Java, JDBC, Servlets, Oracle, JSP, XML, UML, HTML, CSS, JavaScript, UNIX, CVS, DB2 and Ionic Framework.

Client: See Software	**May'09 - Aug'10**
Role: Jr. Java Developer	

Description: see software is a pioneer in providing tax automation products, solutions for both direct and indirect taxes. The application developed offers various computation & filing services for account payable, account receivable and payroll transactions and is a preferred tax solution partner for various ERP vendors and implementers.

Responsibilities:
- Involved in Analysis and Design of the project, and developing prototypes of the product.
- Developed the application using **Struts** Framework that leverages classical Model View Controller **(MVC)** architecture.
- Developed Controller Servlets, Action, and Form objects for process of interacting with **Oracle** database and retrieving dynamic data.
- Wrote custom support modules for upgrade implementation using **PL/SQL, UNIX** Shell scripts.
- Developed backend components, **DB** Scripts for the backend communication.
- Used Apache **Log4j** logging **API** to log errors and messages.
- Involved in Documentation and use case design using **UML** modeling include development of Class diagrams, Sequence diagrams, and use case Transaction diagrams.
- Developed **HTML** and **JSP** pages using **Struts.**

- ➢ Responsible for performance tuning of the product and eliminating memory leakages in the product.
- ➢ Used **Junit** to perform unit testing.
- ➢ Used **Eclipse** tool for application development.
- ➢ Used Configuration Management Tools for version control which is built on **CVS.**

Environment: Java, JSP, HTML, AJAX, XML, DOM, SAX, JNDI, Struts, DB, PL/SQL, UNIX Shell Scripting, Rational Rose, Apache Tomcat, Oracle, JUnit, Log4j, CVS, Eclipse.