**Silambarasan Thiyagarajan**

Lead Java Developer

Email Id: [simbutr163@gmail.com](mailto:simbutr163@gmail.com)

**EXPERTISE SUMMARY:**

* **Around 12+ years** of experience in Software Development Lifecycle (SDLC) core areas such as Analysis, Design, Implementation, Testing and Deployment of Object Oriented, Client-Server, Web-Based, Distributed and Enterprise Applications with **Java/J2EE** technologies.
* Strong experience in Software Development Methodology such as **Agile** (Scrum) Methodology, Iterative and Waterfall.
* Strong hands - on experience in MicroServices with Spring IOC, **Spring Boot**.
* [**AWS (Amazon Web Services)**](https://intellipaat.com/blog/what-is-amazon-web-services-aws/) platform to **secure cloud services, database storage, compute power, content delivery**, and other services to help business level and develop.
* AWS services are used to develop and deploy application in cloud such as **EC2, ELB, S3, RDS, EBS, VPC, Cloud Front, SNS, SES, Beanstalk, Lambda, Auto Scaling and Elastic Cache.**
* **Terraform** tool for automate and manage infrastructure platform and deploying applications.
* **Azure** host entire application stack from **web applications** and **APIs** to databases to storage services using a variety of hosting models from fully managed services to containers to **virtual machines** for **scalability**, **high-availability**, and **security built** into Azure.
* Strong knowledge in **CI/CD** pipeline setup using **Azure Devops** and **Jenkins**.
* Extensive experience in developing Microservices using Spring Boot, Netflix OSS (Zuul, Eureka, Ribbon, Hystrix) and followed domain driven design.
* Experience with container-based deployments using **Docker**, working with **Docker images**, **Docker Hub** and Docker-registries and **Kubernetes**.
* Strong understanding of Java based messaging systems like **IBM MQ** and event streaming platform like **Kafka**.
* Hands on experience in creating UI using components, Forms, Events, keys, Routers, etc.,
* Extensive experience with both Core and Advanced Java concepts which include **Java,Java8, Servlets, JSP, Spring, Spring boot, JDBC, XML,EJB, Struts, Spring MVC, Hibernate, JPA and Web Services**.
* Extensive knowledge in Front End technologies like **JavaScript, HTML, XML and CSS**.
* Understanding of Security Entitlements and implications and implementations(**OAuth2**) and **JWT**
* Have good experience to implement Object Oriented design diagrams for Application system in UML using IBM Rational Rose.
* Expertise in using and implementing various design patterns such as **Singleton, Bridge, Abstract Factory, Decorator** and **Iterator** etc.
* Experience in **OOPS** concepts like **Inheritance, Encapsulation, Abstraction, Polymorphism**.
* Worked in **Multithreading** environment by **Thread Synchronization**, **Thread Pool**, **Class level** and **object level threading**.
* Expertise in designing and developing J2EE compliant systems using IDE tools like **Eclipse**, **RAD** (Rational application Development), **Net Beans** and **JDeveloper**.
* Expertise in Developing and Deploying applications using **Webservers** like Apache and **Application Servers** like Apache Tomcat 5.x/6.x/7.x/8.x, **IBM WebSphere**.
* Experience in building web applications using various frameworks such as Spring Framework (2.0-4.0), **SpringMVC** (Model View Controller), **Spring Boot, IOC** (Inversion of Control) and **DAO** (Data Access Object).
* Worked on **Struts** open source java framework utilizing Strut Tiles, Strut Controllers (Action and Forms) and Validation framework for client, server validation.
* Extensive experience in Core Java concepts like **Exception handling, Multi-threading, Synchronization, Serialization, IO, Collections framework, Java Beans**.
* Involved in development of SOAP based web services using **WSDL, SOAP, JAXB, CXF, AXIS, JAX-WS** and **Restful web services** using **JAX-RS, CXF** and **JERSEY APIs**.
* Experience using **ORM** framework like **Hibernate/JPA** for mapping Java classes with database and using Hibernate Query Language (HQL).
* The Enterprise Service Bus (**ESB**) software architecture which connects all the services together over a bus like infrastructure and communication center in the SOA by allowing linking multiple systems, applications and data and connects multiple systems with no disruption
* Knowledge on service oriented architecture (SOA), workflows and web services using XML Test Driven Development (TDD) Environment, **SOAP** and **Restful** web services.
* **Cassandra , MongoDB** for data modeling to manage huge amount of data with correct methodology
* Expertise in unit testing the applications using **Junit Framework, Mockito and developed logging** and standard mechanism based on **Log4J**, **SLF4j**.
* Strong experience in monitoring tools like **Splunk, SSAM, Sloarwinds, Dynatrace** etc.,
* Experience in implementation of version controls tools like **CVS, Tortoise SVN** and **GIT**.
* Experience in **Oracle 8i/9i/10g/11g, SQL Server, MySQL** database programming using **PL/SQL** and standard built-in Oracle Packages.
* In-depth understanding of **Data Structures and algorithms**.
* Proficient with software build process and writing Maven and **ANT** Build Scripts for automated build and deployment process.
* Experience with Build tools like **ANT, MAVEN, GRADLE**.

**TECHNICAL PROFICIENCIES:**

* Java & J2EE
* Struts,Spring and SpringBoot
* Oracle 9i/10g/11g, DB2, MySQL and SQL Server
* WSDL, SOAP, AXIS, JAX-WS, REST-Jersey, REST-easy, UDDI, and WS-Security.
* Hibernate.
* JUNIT, Mockito, TestNG, SOAP UI
* AWS,Azure Devops and Jenkins
* XML, DTD, XSD, XSLT, JAXP (DOM & SAX) and JAXB
* HTML, DHTML, CSS, JavaScript, JSP, JQUERY, DOJO.
* UML – Rational Rose, MS Visio, JUDE
* Eclipse, My Eclipse, Net Beans and RAD, Mule ESB
* Docker and Kubernetes
* IBMMQ and Apache Kafka
* Tortoise SVN, CVS and Clear Case
* Jira, Rally, Clear Quest, Clover
* Web sphere, Web Logic, Apache Tomcat & HTTP Server
* WINNT/2000/XP/2003, Unix, DOS and Windows 7
* Netflix OSS (Zuul, Eureka, Ribbon, Hystrix)
* OAuth2
* Log4J, SLF4J, Splunk

**EDUCATION:**

|  |  |  |  |
| --- | --- | --- | --- |
| **MCA** | Master of Computer Applications | Anna University, Coimbatore | 2010 |
| **B.SC** | Computer Science | Bharathidhasan University, Trichy | 2006 |

**PROFESSIONAL EXPERIENCE**

**Client: Western Union,** [**Denver, CO**](https://www.google.com/search?sca_esv=569212662&q=Denver&si=ALGXSlZS0YT-iRe81F2cKC9lM9KWTK4y0m5Atx8g9YliNNw2mdgYZcZzTgz1BrcSouxXByH9ozZ76DQnueeC3R7Ur2stiuzvx6GPIwkpb_A1X-RzVjbzwOPoMAhDpCr6zwRDBj1AqolbCV5yPsYIJscJwBQ2sBvll0cZfH6G3aI9oyHSt-nScfvB0Hz9rBLtQL9sxXcphFbN&sa=X&ved=2ahUKEwjvn9zn9s2BAxWUFVkFHTgaDpMQmxMoAXoECGQQAw) **JULY 2023 – Till Now**

**Title: Lead Java Developer**

**Project: Customer Master Data Management**

Streamline the process of capturing data from customers (collect consistent and accurate data and only ask for it once). Provide a consolidated customer profile and manage profile updates across interactions. Accurately recognize a returning customer, regardless of interaction channel. Provide the business with a view of all data related to a customer’s journey and interactions (360° customer view).Enable customer to manage his engagement with WU through self-service capabilities and the business to act upon the customer data/insight at the point of customer interaction such as Optimized digital experience Faster, improved Care servicing, Personalized marketing/communications, minimized undeliverable and conflicting messaging

Streamlined compliance, risk processes (reciprocity between channels).

**Responsibilities:**

* Develop and maintain Java-based applications for customer data capture and management.
* Implement data validation and consistency checks in the data capture process.
* Ensure efficient and accurate data storage and retrieval.
* Utilize **Spring** and **Spring Boot** for building **customer data management** services.
* Implement **RESTful** APIs for data capture and retrieval.
* Configure Spring-based components for **dependency injection** and **AOP**.
* Design and develop **RESTful** APIs **for data capture, customer profile retrieval, and updates**.
* Ensure secure **authentication** and **authorization** mechanisms.
* Document APIs using OpenAPI (**Swagger**) for internal and external use.
* Deploy and manage the application on **AWS** infrastructure (**EC2** instances).
* Use **S3 buckets** for storing customer data securely.
* Set up Elastic Load Balancers (**ELB**) for scalability and high availability.
* Implement **Elasticsearch** for efficient search capabilities within the customer data.
* Enable rapid and accurate **customer profile retrieval**.
* Implement security measures to protect customer data and ensure compliance with data protection

regulations.

* Monitor and address security vulnerabilities (**Sonar**) and ensure data privacy.
* Collaborate with legal and compliance teams for regulatory adherence.
* Set up **CI/CD pipelines** using **Jenkins** for automated testing and deployment.
* Define infrastructure as code (**Terraform**) for scalable and maintainable cloud resources.
* Containerize the application using **Docker** for easy deployment.
* Use Postman and other testing tools to verify the functionality of **RESTful** APIs.
* Validate **JSON payloads** and **API responses**.

**Client: Western Union,** [**Denver, CO**](https://www.google.com/search?sca_esv=569212662&q=Denver&si=ALGXSlZS0YT-iRe81F2cKC9lM9KWTK4y0m5Atx8g9YliNNw2mdgYZcZzTgz1BrcSouxXByH9ozZ76DQnueeC3R7Ur2stiuzvx6GPIwkpb_A1X-RzVjbzwOPoMAhDpCr6zwRDBj1AqolbCV5yPsYIJscJwBQ2sBvll0cZfH6G3aI9oyHSt-nScfvB0Hz9rBLtQL9sxXcphFbN&sa=X&ved=2ahUKEwjvn9zn9s2BAxWUFVkFHTgaDpMQmxMoAXoECGQQAw) **APRIL 2023 – JUNE 2023**

**Title: Lead Java Developer**

**Project: NextGen Pricing**

NextGen Pricing service acts as a service in between Digital Pricing Framework and Core Pricing Engine (CPE). NextGen Pricing service is integrated with Optimizely, which is an A/B test tool similar to Evergage. The service is getting FICO Engine integration which will allow modifying fees and exchange rates on the fly instead of using a traditional, pricing segment-based approach

**Responsibilities:**

* Develop and maintain the **NextGen** Pricing Service using Java, ensuring code quality and adherence to coding standards.
* Implement features and functionality required for interacting with the **Digital Pricing Framework** and Core Pricing Engine.
* Collaborate with the team to design, develop, and test Java-based components.
* Write clean, efficient, and maintainable Java code.
* Debug and troubleshoot issues in the Java codebase.
* Utilize **Spring** or **Spring Boot** to create and configure the NextGen Pricing Service.
* Implement **dependency injection** and **AOP** where necessary.
* Work on the integration of the service with other components in the architecture.
* Design and develop **RESTful** APIs for communication between the NextGen Pricing Service and other modules.
* Ensure proper API security mechanisms, including **authentication and authorization**.
* Document APIs using OpenAPI (**Swagger**) or other documentation tools.
* Use Git for version control, ensuring proper branching, merging, and code versioning.
* Collaborate with team members through version control practices.
* Perform unit testing and integration testing of Java components.
* Contribute to the development of test cases and ensure code coverage.
* Collaborate with **QA** teams to resolve issues and ensure high-quality code.
* Identify and address security vulnerabilities within the Java codebase.
* Follow security best practices to protect sensitive pricing data.
* Document code changes, configurations, and architectural decisions.
* Maintain documentation related to Java components and their interactions.
* Work with Postman for API testing and ensure that **JSON payloads** are correctly handled in API requests and responses.
* Set up and configure **CI/CD pipelines** using **Jenkins** or **CloudBees**.
* Automate build, test, and deployment processes for the NextGen Pricing Service.
* Deploy and manage the NextGen Pricing Service on **AWS** infrastructure, such as **EKS** or **EC2**.
* Set up Elastic Load Balancers (**ELB**) for traffic distribution.
* Store and retrieve data securely using **S3 buckets**.
* Implement **Elasticsearch** for efficient search capabilities within the service.
* Ensure that data indexing and querying are optimized.
* Define and manage **AWS** infrastructure as code using **Terraform**.
* Create and maintain the necessary cloud resources for the service.
* Develop and maintain the database schema for storing pricing-related data.
* Write **SQL queries** to interact with the database.

**Client: AT&T, Plano, TX AUG 2022 – MAR 2023**

**Title: Lead Java Developer**

**Project: CSI-Adapter**

CSI Adapter development involves use of adapter framework, spring framework, java and related technologies and much more. It is architectural component that abstract backend systems. Developer is expected to follow the java coding standards and best practices and top of it follow the CSI standards and practices set. There are rules around private schemas, process models, configuration files, adapter code structure, bundle structure. Adapter code review template, SONAR tool and peer reviews help identify any adapter code issues and deliver best quality code.

**Responsibilities:**

* Worked as a Developer and involved in analysis of requirements, design, development, Unit and Integration testing.
* Involved in analyzing the user requirements and specifications for deciding the flex component needed to be developed and their interaction.
* Created MicroServices with **Spring IOC**, **Spring Boot**.
* **Spring Boot** to use Java-based frameworks to create **microservices** and **web apps**
* Simplify the maven configuration using **spring boot** starter pom.xml
* **Spring boot actuator** used for production ready features
* **Shell scripting** is used for Automating the code compiling process, Running a program or creating a program environment, Completing batch, Manipulating files, Linking existing programs together, Executing routine backups, Monitoring a system.
* **CSI-Adapter**  uses **Apache Kafka** for integrating services in an event-driven way and Participant services use the producer-consumer API of **Apache Kafka**, together with **Kafka Streams** and **Kafka Connect**.
* Replicated **Apache Kafka** topic partitions are used to distribute message traffic among event-driven microservice instances
* Unexpected errors are propagated to the exception handler and to the underlying **Apache Kafka** Streaming API to preserve a consistent state on the streaming platform.
* Xml schema design from AID document using **XMLSpy** tool
* Create process model using **Tibco Business studio**
* Validate the generated mappings and process flow on target/cloud runtime
* Build, deploy and execute the services in the cloud environment using GRM tool
* Test the services in Local and cloud environment using **DME/SOAP** UI tools
* Design, develop and implement technical solutions in **Microsoft Azure** environments using Java and other programming languages
* Creating cloud solutions that solve pain points and meet the business’s needs.
* **Executor framework** for creating, executing and managing the thread and improves the performance of the application.
* **ExecutorService** interfaceto manage lifecycle of tasks as well for executor.
* **ScheduledExecutorService** for scheduling tasks at fixed intervals or with initial delay.
* **Executors** class provides factory methods for creating thread pool such as [**newFixedThreadPool**](https://java2blog.com/java-newfixedthreadpool-example/)**,** [**newCachedThreadPool**](https://java2blog.com/java-newcachedthreadpool-example/)**,** [**newSingleThreadedExecutor**](https://java2blog.com/java-newsinglethreadexecutor-example/)**,** [**newScheduledThreadPool**](https://java2blog.com/java-scheduledthreadpoolexecutor-example/)**.**
* Host entire application stack from **web applications** and **APIs** to databases to storage services using a variety of hosting models from fully managed services to containers to **virtual machines** for **scalability**, **high-availability**, and **security built** into Azure.
* **Azure Blob Storage** to store files in the cloud and **Azure Key Vault** to securely store application secrets can be easily added existing apps without changing current application architecture or deployment model.
* **Private registry** for container images, are containerizing an existing app for ease of deployment, deploying microservices based applications, or managing containers at scale.
* **Azure Functions** simplify building solutions to handle event-driven workflows, whether responding to HTTP requests, handling file uploads in **Blob storage**, or processing events in a **queue**
* **Spring Security OAuth2** serves as an open authorization protocol for enabling a third party application to get limited access to an HTTP service on behalf of the resource owner
* Developing and implementing reliable cloud solutions.
* Scaling cloud solutions to match the business’s changing needs.
* Backing up data and ensuring that it can be accessed when needed.
* Securing cloud solutions so that only authorized personnel can access them.
* Monitor and maintain the cloud to ensure that everything is working efficiently.
* **Kubernetes** is being used to orchestrate the deployment, scaling and management of **Docker** Containers.
* Actively worked on Docker and deploying and maintaining Micro services in Dev & QA environment.
* Implemented docker-maven-plugin in maven pom to build docker images for all microservices and later used Docker file to build the docker images from the java jar files.
* Implemented build stage- to build the micro service and push the docker container image to the private docker registry.
* Developed and consuming Restful Web Services using media applications like **JSON, XML**.
* Worked on Selenium web driver with testing team for unit test cases and created test scripts in Selenium and enhanced those scripts using Java API and Test NG.
* Used case analysis, assessing the impacts for deliverables.
* Deployed Elastic Search cluster to serve non-front-end traffic, reducing the load on customer facing search infrastructure by millions of requests day.
* Wrote **SQL, PL/SQL** and **stored procedures** as part of database interaction.
* Wrote unit test cases using **JUnit, Mockito, Easy Mock, Power Mockito framework**.
* Build Micro services for the delivery of Purchase Order module using spring cloud.
* Implemented Multithread /Executor Thread to concurrent for distributed **SOAP** Web Services.
* Implemented navigation using **Spring** **MVC** controllers, configured controllers using Spring MVC annotations and configuration files.
* Customized Restful Web Service using Spring **Restful** API, sending **JSON** format data packets between front-end and middle-tier controller.
* **Kibana elastic search** is a good front-end dashboard that is used for indexing the data / information from the cluster of elastic.
* **Kibana Elastic search** is used for searching, interacting, and viewing the information / data that is stored in the directories of elasticsearch. It will also help the user in performing advanced data analysis. It ensures the users to visualize the data in several varieties of chats, tables and maps.
* It is fully integrated with elasticsearch and Provides reliability, scalability, indexing in order to make the fast search,RESTful,Apache Lucene and helps the users to scale horizontally and vertically.
* Splunk logs, Production dashboard and GRM tools help quickly diagnose a defect
* Used **Log4j** for **logging** errors, **messages** and **performance** logs.

**Environment:**

JAVA, J2EE, Eclipse(IDE), Maven (Build Tool), SOAP Web Services,Azure Cloud,Apache Kafka, Altova XML Spy, Jenkins, Business Studio, Contivo Analyst 5.6, Eclipse, AMQ, IBM WMQ, Splunk, Agile Craft, GRM Tool, JMS Tool, Git, windows10.

**Client: Broadridge Financial Solutions, Newark, NJ SEP 2021 – July 2022**

**Title: Lead Java Developer**

**Project: Trading and Allocations**

Broadridge has introduced a centralized utility for the entire market. The trade assignment portal unites originators, broker-dealers and asset managers in a single, secure electronic platform. Zoom is new order management and executions system which captures orders from FAs,CSAs & Traders, route the orders to multiple destinations/exchanges to get the executions, book the trades and send the trades to back office systems for settlement and book keeping. Finally, all parties can rely on a standardized set of digital tools that add transparency and efficiency to every trade assignment.

**Responsibilities:**

* Involved in Agile – Sprint methodologies to do requirements gathering, analysis and planning.
* Worked on design analysis of user stories and documentation for REST API’s
* Rest API implementations using spring boot and oracle database integrating the individual microservices.
* Used Java8 features like stream and lambda expressions
* Spring Security JWT is used to generates the **JWT Token** for Web security
* Worked on getting fixed income orders from Tradeweb, IPERO and Vestmark, validate and route the orders to UBS and maintaining the order status
* Worked on capturing equity and Option orders, Block and Allocations orders from UI, validate and route the orders to different exchange
* Created MicroServices with **Spring IOC**, **Spring Boot**.
* **Spring Boot** to use Java-based frameworks to create **microservices** and **web apps**
* Simplify the maven configuration using **spring boot** starter pom.xml
* **Spring boot actuator** used for production ready features
* ZOOM Api uses **Apache Kafka** as a messaging system, provided and fully maintained by Broad Ridge infrastructure team
* ZOOM accepts and produces FIX messages via Appia servers that are integrated by **Kafka/MQ channels** managed by EISL
* **Apache Kafka HA** Cluster is considered as a messaging backbone of the system, integrated with high cohesion, hence it is not interchangeable with other messaging systems. Several basic architectural concepts of ZOOM are founded on the features of Apache Kafka
* The system state is stored and managed by an **Apache Kafka** cluster and a database cluster. As a result, the overall availability is highly relying on these managed services
* ZOOM forwards messages to external notification hub integrated by **Kafka topic** managed by **EISL**
* [**AWS (Amazon Web Services)**](https://intellipaat.com/blog/what-is-amazon-web-services-aws/) platform to secure cloud services, database storage, compute power, content delivery, and other services to help business level and develop.
* **Identity and Access Management(IAM)** is used to enhanced security and identity management for **AWS** account
* **Elastic Compute Cloud (EC2)** for on-demand computing resources for hosting applications and handle unpredictable workloads.
* **Elastic Block Store (EBS)**for persistent storage volumes that attach to EC2 to allow you to persist data past the lifespan of a single Amazon EC2 instance
* **PaaS,** or platform as a service, is on-demand access to a complete, ready-to-use, cloud-hosted platform for developing, running, maintaining and managing applications.
* **SaaS,** or software as a service, is on-demand access to ready-to-use, cloud-hosted application software.
* Establish a private connection between virtual private cloud (VPC) and the Elastic Load Balancing API by creating an interface **VPC endpoint**. this connection to call the **Elastic Load Balancing** API from VPC without requiring that attach an internet gateway, NAT instance, or VPN connection to VPC. The endpoint provides reliable, scalable connectivity to the Elastic Load Balancing API, which use to create and manage your load balancers.
* **Simple Storage Device (S3)** is a storage device used to store and retrieve the data on AWS service
* **Teraform** tool for automate and manage infrastructure platform and deploying applications.
* **Terraform** scripts are used to define and provision the infrastructure using Private network space,EC2 service, Install Docker and tools, Security
* Created, changed and replicated infrastructure using **Terraform**
* Code Bug analysis and fixes.
* Worked on maven build tool, git.
* Unit and Integration Testing and Deployment using Jenkins in dev and QA environment.
* Used swagger tool to visualize and interact with the API resource without having any of the implementation logic in place
* Ensured deliverables meet the timeline.
* In-depth understanding of **Data Structures and algorithms**.
* Write efficient and optimized computer programs using data structures and algorithms such as a linear data structure can be an **array**, a **stack**, a **linked** **list**, or a **queue**. Non-linear data structures include **trees** and **graphs**.
* Co-ordinate with onsite/clients on daily basis for tasks status.
* Used Git as version control tool for effectively managing the code changes.

**Environment:**

JAVA, J2EE, JAVA 1.8, Spring, Spring Boot, Micro Service, AWS Cloud, IntelliJ(IDE), Apache Maven, Restful Web Services, Jenkins, IBM WMQ, APPIA Messaging tool, Apache Kafka, Splunk, Git, Junit, Log4j, IrisDB, Docker, Terraform, Kubernetes.

**Client: AT&T, Plano, TX April 2018 – Aug 2021**

**Title: Sr Java Developer**

**Project: Expedited Service Systems Delivery**

The Expedited Service Systems Delivery is an infrastructure designed to support application deployment within AT&T environment in an expedited and cost effective manner Its intent is to provide end to end infrastructure such as hardware services and logical connectivity required to serve applications on different platforms such as IPTV STB PC Mobile devices including iPhone blackberry PDAs etc to enable service access on cross platforms and facilitate interactions with IPTV IMS Mobility network as well as other IT middleware services and billing applications.

**Responsibilities:**

* Worked as a Developer and involved in analysis of requirements, design, development, Unit and Integration testing.
* Involved in analyzing the user requirements and specifications for deciding the flex component needed to be developed and their interaction.
* Created POJO layer to facilitate the sharing of data between the front end and the **J2EE** business objects
* Used **Hibernate** for Object Relational Mapping of the database structure created in Oracle,
* Involved in Hibernate configuration properties setup mapping files using generators, associations inheritance etc.
* Implemented Message Driven beans to develop the asynchronous mechanism to invoke the provisioning system when a new service request saved in the database used JSM for this.
* Transformed **XML** documents using **XSL**.
* Implemented application level persistence using Hibernate and **JPA**.
* Developed various components using Data Access Object (**DAO**).
* **Servlets** to generate dynamic web pages and web content.
* Used Websphere for Design, Development and Deployment of applications.
* Wrote scripts in **UNIX** to process the batch jobs on a scheduled basis.
* Have used **Facade**, **Singleton**, **Factory** Design patterns.
* Used Java, J2EE design patterns. Used Business Delegate to maintain decoupling between presentation business layers.
* Used JPA with **Hibernate provider**.
* Involved in configuring web sphere variables, queues, DSs, servers and deploying **EAR** into Servers.
* Involved in developing the business Logic using Plain Old Java Objects POJOs and Session EJBs.
* Developed and debugged the application using Rational Application Developer(RAD) IDE.
* Involved in Hibernate mappings, configuration properties set up, creating sessions, transactions and second level cache set up.
* Interact with Business Analyst and Host Subject Matter Experts (SME) to understand the requirements and for any clarifications required by the team, followed agile methodology and SCRUM meetings to track, optimize and features.
* Developed logging information using **Log4j**.
* Used Log4j for logging errors, messages and performance logs.
* Developed hibernate DAO framework with spring framework.
* Extensively involved in developing core persistence classes using Hibernate framework, writing HQL queries, **DB** schema and **PLSQL** changes.
* Responsible for unit testing the application using JUnit.
* Used tortoise **SVN** as the version control.
* Used **JIRA** for bug tracking, issue tracking and project management.
* Used **Cruise Control** for integrating the applications.
* Used **UNIX** scripting for file transferring.
* Installation, Configuration, Troubleshooting and Maintenance of WebSphere Application Server
* Involved in Deployment code on WebSphere Application Server and configure Source name and **Webspher MQ** in WebSphere Application Server console.
* Deployment of application on **Websphere Application server** in clustered environment, managing the server, monitoring the server logs.

**Environment:**

Java , J2EE, Subversion, XML, JDBC 4.0, RAD, JSF, XML Web Services, SOAP, WSDL, SAX, DOM, JAXB, Log4J, JUnit, Oracle 11g, PL/SQL, Unix, Spring , Hibernate , JPA, WebSphere MQ, Cruise control (Build Tool), WebSphere Application Server, TOAD, Junit, JIRA.

**Client: Apple, Sunnyvale, CA Jan 2017 – March 2018**

**Title: Senior Software Developer**

**Project: i Phone (Uverse**)

CSI-eSSD will deliver the basic level of Robocall blocking to customers using existing i Phone (Uverse) and small business customers with Hiya Aegis API Solution. The blocking capability will filter unwanted scam calls and block them, however, legitimate incoming calls will be allowed. The Apple i Phone (Uverse) customers must be able to use One-Click auto provisioning process in Apple Digital Phone portal to enroll in call blocking service.

**Responsibilities:**

* Worked in agile framework as an individual contributor Responsibilities include - Interaction with Business team in story grooming, reviewing story/acceptance criteria.
* Converted a monolithic app to microservices architecture using Spring Boot using 12-factor app methodology. Deployed, Scaled, Configured, wrote manifest file for various **Microservices**.
* Implemented **REST** Microservices using spring boot. Generated Metrics with method level granularity and Persistence using Spring Actuator.
* Implemented Actuators for distributed Micro Service Monitoring. Integrated Swagger UI and wrote integration test along with REST document.
* Implemented Hysterix a Latency and Fault Tolerance system for Distributed Systems. Wrote Controller, Services and directive code.
* Implementation of token based authentication using **OAuth2**(Spring).
* Responsible for developing Splunk events in applications, **Splunk** queries and dashboards targeted at understanding application performance and capacity analysis.
* Kubernetes is being used to orchestrate the deployment, scaling and management of **Docker** Containers.
* Actively worked on Docker and deploying and maintaining Micro services in Dev & QA environment.
* Implemented docker-maven-plugin in maven pom to build docker images for all microservices and later used Docker file to build the docker images from the java jar files.
* Implemented build stage- to build the micro service and push the docker container image to the private docker registry.
* Developed and consuming Restful Web Services using media applications like **JSON**, **XML**.
* Worked on Selenium web driver with testing team for unit test cases and created test scripts in Selenium and enhanced those scripts using **Java** API and **Test** NG.
* Used case analysis, assessing the impacts for deliverables.
* Creating, maintaining and troubleshooting the Azure pipeline setup for Integration and Deployment.
* Deployed Elastic Search cluster to serve non-front-end traffic, reducing the load on customer facing search infrastructure by millions of requests day.
* Implemented with security vulnerable fixes - Cross Site Scripting (**XSS**), Cross Site Request Forgery (**CSRF**), Force-full Browsing, Request Token, Password Maintenance using **jasypt**.
* Wrote **SQL, PL/SQL** and stored procedures as part of database interaction.
* Wrote unit test cases using JUnit**, Mockito, Easy Mock, Power Mockito framework**.
* Involved in parsing JSON String using JSON parsers like **GSON**, **Jackson**.
* Developed Micro services using Restful services to provide all the **CRUD** capabilities.
* Build Micro services for the delivery of Purchase Order module using spring cloud.
* Implemented Multithread /Executor Thread to concurrent for distributed SOAP Web Services.
* Implemented navigation using **Spring MVC** controllers, configured controllers using Spring MVC annotations and configuration files.
* Customized Restful Web Service using **Spring** Restful API, sending JSON format data packets between front-end and middle-tier controller.
* Update **CTN**, Inquire CTN service using **REST** API with Spring Boot.
* Used **Log4j** for logging errors, messages and performance logs.

**Environment:**

Java, J2EE, Eclipse (IDE), Maven (Build Tool), REST API, Json, Tibco Studio 3.2.2, SVN 1.8, Altova XMLSpy, Eco tool, Jenkins, SpringBoot and Agile Methodology, JIRA, JMS, log4j,NO/SQL,Docker and Kubernetes.

**Client: Hyundai, Los Angeles, CA Nov 2015 – Dec 2016**

**Title: Senior Software Developer**

**Project: Back Office Business Process Management**

Back Office Business Process Management (BOBPM) is an Hyundai business process management system that is developed on M2E (Model to Engine) framework and is part of Common Service Interface application. BOBPM monitors the data usage of both Consumer & Business and informs customers in real-time about their usage and also avoids overcharges by alerting customers about their data and by applying necessary treatments. IoT ConnectedCar, Usage Control, Promotional Notifications, IMEI Updates, Order Management, Metro Cell, VoLTE Firmware Upgrade, Software-defined networking, Alert Customization are some of the key functionalities of BOBPM

**Responsibilities:**

* Designing and developing **SOA** (Service-Oriented Architecture) based Web Services and **REST** applications based on microServices for the communication with adapter, backend and middleware applications.
* The Enterprise Service Bus (**ESB**) software architecture which connects all the services together over a bus like infrastructure and communication center in the SOA by allowing linking multiple systems, applications and data and connects multiple systems with no disruption
* The **ESB** provides authentication and authorization mechanisms is parts of security check to inter connect between the web applications.
* The **SOA** uses **proxy** which interprets the service calls between two different client service protocols
* **ESB** uses the gateway through which it can connect to the different services running in the other networks.
* Creating new **JSON** structures based on business requirements and migrating existing schema to JSON and generating xbo files.
* Using Contivo for data mapping between **BOBPM** and downstream applications.
* Developed and consuming Restful Web Services using media applications like **JSON, XML and Text data**.
* Using Jenkins (**Ant Build tool**) via **Eco pipeline** for building & deploying applications to Quality Center.
* Worked on Selenium web driver with testing team for unit test cases and created test scripts in Selenium and enhanced those scripts using Java API and Test NG.
* Implemented Kafka producer and consumer applications on Kafka cluster setup with help of **Zookeeper** and Used Spring Kafka API calls to process the messages smoothly on Kafka Cluster setup.
* **Cassandra** data modelling play a vital role to manage huge amount of data with correct methodology
* Build database using quick design techniques in **Cassandra**.
* Improve existing model using a query driven methodology in **Cassandra**.
* Optimize Existing model via analysis and validation techniques in **Cassandra**.
* Used case analysis, assessing the impacts for deliverables.
* Experience in working with Docker and creating custom images and exposing through Ngnix and configuring using **Ansible scripts**.
* Deployed Elastic Search cluster to serve non-front-end traffic, reducing the load on customer facing search infrastructure by millions of requests day.
* Worked with NoSQL databases such as **Cassandra**
* Using Python/CQL for data definition and manipulation and to create keyspace and tables for our bundles.
* Migrating applications from M2E framework to microService.
* Wrote **SQL, PL/SQL** and stored procedures as part of database interaction.
* Wrote unit test cases using **JUnit, Mockito, Easy Mock, Power Mockito framework**.
* Involved in parsing JSON String using JSON parsers like GSON, Jackson.
* Designed services to store and retrieve user data using database like Cassandra and communicated with remote servers using **REST** enabled Web Services on Jersey framework.
* Worked with **Gradle** for Dependency Management and as build tool.
* Involved in developing new and enhanced existing **JIRA** plug-ins with Atlassian's plug-in development platform. JIRA is an issue tracking and projects management system that is written in Java.
* Used Apache Tiles as the template framework.
* Worked on SOA Web Services using **SOAP** with **JAX-WS** for transmission of large blocks of XML data over HTTP.
* Using **Postman** for REST applications unit testing.
* Using **Kubernetes** & **Kibana** to check application status & validate logs
* Tier 3 production support to assist clients in identifying the issue & providing support in fixing the issue.

**Environment:**

JAVA, J2EE, Eclipse(IDE), Maven (Build Tool), SOAP Web Services, Altova XML Spy, Jenkins, Git, Zookeeper 3.4.6, Python, Postman, DBeaver, Business Studio, Contivo Analyst 5.6, Eclipse, AMQ, IBM WMQ, Splunk, Agile Craft, JMS Tool, SVN, windows7.

**Client: AT&T**  **July 2014–Oct 2015**

**Title: Software Developer**

**Project: CSI\_ServiceLayer\_HALO**

CSA is the MDA based framework which is used by clients to connect to their large variety of back ends (e.g. Web services, JMS messages, Databases etc.).The main function of this application is to identify the request to the respective legacy systems, for example: customer’s wants to know about the account summary details, bill dispute details, the front end systems uses CSI to get the account id/ group id to get the details of the account summary or bill dispute details. And it is designed to authorize and authenticate interfaces of Legacy systems for security. Interactions with back end systems will be abstracted via Adapter and Backend implementation.

**Responsibilities:**

* Worked in agile framework as an individual contributor Responsibilities include - Interaction with Business team in story grooming, reviewing story/acceptance criteria.
* Effectively facilitated regular project meetings and follows up with team members as required to focus on business and operations issues or technical issues.
* Participating in the High-level and low-level system design, Production, planning, estimation, Testing and implementation.
* Xml schema design from AID document using **XMLSpy** tool
* Mapping the public and private schemas using **Contivo mapping** tool
* Create process model using **Tibco Business studio**
* Validate the generated mappings and process flow on target/cloud runtime
* Build, deploy and execute the services in the cloud environment using GRM tool
* Test the services in Local and cloud environment using **DME/SOAP** UI tools
* Implemented **REST Microservices** using **spring boot**. Generated Metrics with method level granularity and Persistence using **Spring Actuator**.
* Implemented Actuators for distributed Micro Service Monitoring. Integrated Swagger UI and wrote integration test along with REST document.
* Implemented Hysterix a Latency and Fault Tolerance system for Distributed Systems. Wrote Controller, Services and directive code.
* Implementation of token based authentication using **OAuth2**(Spring).
* Responsible for developing Splunk events in applications, **Splunk** queries and dashboards targeted at understanding application performance and capacity analysis.
* **Kubernetes** is being used to orchestrate the deployment, scaling and management of **Docker** Containers.
* Actively worked on Docker and deploying and maintaining Micro services in Dev & QA environment.
* Implemented docker-maven-plugin in maven pom to build docker images for all microservices and later used Docker file to build the docker images from the java jar files.
* Implemented build stage- to build the micro service and push the docker container image to the private docker registry.
* Developed and consuming Restful Web Services using media applications like **JSON, XML**.
* Worked on Selenium web driver with testing team for unit test cases and created test scripts in Selenium and enhanced those scripts using Java API and Test NG.
* Used case analysis, assessing the impacts for deliverables.
* Deployed Elastic Search cluster to serve non-front-end traffic, reducing the load on customer facing search infrastructure by millions of requests day.
* Wrote **SQL, PL/SQL** and **stored procedures** as part of database interaction.
* Wrote unit test cases using **JUnit, Mockito, Easy Mock, Power Mockito framework**.
* Build Micro services for the delivery of Purchase Order module using spring cloud.
* Implemented Multithread /Executor Thread to concurrent for distributed **SOAP** Web Services.
* Implemented navigation using **Spring** **MVC** controllers, configured controllers using Spring MVC annotations and configuration files.
* Customized Restful Web Service using Spring **Restful** API, sending **JSON** format data packets between front-end and middle-tier controller.
* Used **Log4j** for **logging** errors, **messages** and **performance** logs.

**Environment**

JAVA, J2EE, Eclipse(IDE), Maven (Build Tool), SOAP Web Services, Altova XML Spy, Jenkins, Business Studio, Contivo Analyst 5.6, Eclipse, AMQ, IBM WMQ, Splunk, Agile Craft, JMS Tool, SVN, windows7.

**Client: Black Hawk Community Bank**  **April 2013 – June 2014**

**Company: KSAB Software, INDIA.**

**Title: Software Engineer.**

**Project: Banking Customer Transaction System**

This is a banking application system. The product is used by banking staff for maintaining the business functions like Credit cards, fixed deposits, checking accounts, savings accounts and loan management system. The services are provided by the bank staff to the bank customers. I was mainly working in the Registration & Loan modules. In this application system customer account data and various loan information applied by the customers are stored in Oracle database tables. The various loan categories include Education Loan, Vehicle Loan, Land Loan, Agricultural Loan etc. Screens are provided to create customer accounts, update customer details, applying for new loans, repayments made by customers for the loan taken. Various reports can be generated to determine pending loan balances of customers. The reports can be generated using PDF format.

**Responsibilities:**

* Interacted with the Business Analyst to discuss Requirements and acted as first point of contact for Business Analyst Team for any changes or escalations.
* Effectively facilitated regular project meetings and follows up with team members as required to focus on business and operations issues or technical issues.
* **JUnit, JMock Frameworks** were used for performing unit and integration testing by writing test cases.
* Followed **Agile** Methodology in analyze, define, and document the application will support functional and business requirements. Coordinate these efforts with Functional Architects.
* Maven was used as a build tool. Also worked in an agile work environment.
* Developed light weight business component and integrated applications using **Struts** framework Involved in Defect Fixing and after fixing we updated in Jira.
* Daily scrum meeting with client
* Used Struts framework for configuring the application.
* Writing Flow files in **Struts**
* Involved in Writing JSP coding.
* Used **SQL and PL/SQL** Programming extensively to talk to Oracle database.
* Used **Log4j** for **logging** errors, **messages** and **performance** logs.
* Preparing Technical Requirement specifications and **unit** test cases.
* Development of code as per the business requirements.
* Mentored co-developers with new technologies. Participated in Code reviews.

**Environment:**

Java,J2ee,Struts 1.2, Servlets, JSP ,Mysql 5.5,Oracle 9i,HP Quality Center, Putty Jenkins, BMC Remedy, TOAD SVN Agile Methodology and Windows 7.

**Client:** **Missouri Workers Compensation June 2012– March 2013**

**Company: KSAB Software, INDIA.**

**Title: Software Engineer.**

**Project: Workers Insurance Claim Processing**

The Alert Workflow allows the user to set up multiple Business rules such as Successor, SLA definitions, Incomplete Scheduled definitions and Application failure alerts. Escalation Service will generate the alerts based on the rules specified by the Alert workflow setup. The generated alerts will be fed to Tivoli for appropriate action such as opening a PMR and displaying the alert. The ability to define alert rules for Successor Events, incomplete scheduled Events, JCL-SLA jobs and Failed jobs. Escalation Service will generate an alert based on the rules specified by the Alert workflow setup. These alerts will provide the GNS Production Support Team the ability to react faster to failures in the GNS Mainframe space.

**Responsibilities:**

* Interacted with the Business Analyst to discuss Requirements and acted as first point of contact for Business Analyst Team for any changes or escalations.
* Effectively facilitated regular project meetings and follows up with team members as required to focus on business and operations issues or technical issues.
* Involved in writing **JSP** pages.
* Used **Struts 1.2** framework in web tire to handle user request
* Developed the Action classes, Action Form classes and configured the struts-config.xml file.
* Used Struts-validator framework for form-validation and application error Handling.
* Involved in integrating the business layer with **DAO** layer using **ORM** tool Hibernate.
* Used **Struts** framework for configuring the application.
* Used **SQL** and **PL/SQL** Programming extensively to talk to **Oracle** database.
* Used **Log4j** for **logging** errors, **messages** and **performance** logs.
* Preparing Technical Requirement specifications and unit test cases.
* Development of code as per the business requirements.
* Mentored co-developers with new technologies. Participated in Code reviews.
* **JUnit** were used for performing unit and integration testing by writing test cases.

**Environment:**

Core Java, J2EE, Struts, Servlets, JSP, Hibernate, Maven, DB2, SVN and Windows XP Professionals.

**Client:** **Bharathidasan University Colleges Feb 2011 – May 2012**

**Company: KSAB Software**, **INDIA.**

**Title: Software Engineer.**

**Project: E-Library System**

This project is used to organize the book details and facilitate the process of borrowing books. It helps to organize the library and automate all the operations done by the librarian. The user can have search for book details and borrow books. The librarian can add books and view borrower's details and look at the book list. The librarian will inform the user about his overdue. The users who have exceeded their borrowing period will be added into the block list and blocked from borrowing further books.

**Responsibilities:**

* Designed and implemented Java Classes, Interfaces, Model design, and interface layer design with other team members.
* Developed **JSP’s** and **Servlets** to dynamically generate **HTML** and display the data to the client side. Extensively used **JSP** tag libraries.
* Involved in writing services to write core logic for business processes.
* Involved in writing database queries, stored procedures, functions etc.
* Used Java **J2EE** design patterns. Involved in **client** server side validations.
* Involved in writing **JDBC** code to get data base connectivity.
* Implemented server side validations and developed model classes.
* Involved in the Database structuring and Tables creation needed for the project.
* Instrumental in tuning the framework to meet the performance standards.
* Designed and developed web-based software using **STRUTS** MVC Framework.
* Used Java Script to handle asynchronous requests, and **CSS** was used to keep uniformly in look of the application.
* **EJBs** in the application and developed Session beans for the business logic at the middle tier level.
* Created Test plan documents and testing using Junit Framework.
* Used **JAXB** parser for parsing valid .xml files as defined by XSD.
* Created and modified Stored Procedures, Functions and Triggers Complex **SQL** Commands for the application using PL/SQL.
* Monitored the error logs using **Log4j** and fixed the problems.
* Designed **JDBC** Connection for data retrieval and updating purpose, also developed connection pool to avoid waiting time for database connection.
* Used Tomcat server for application development with Eclipse.

**Environment:**

JAVA, J2EE, Struts, Servlets, JSP, Hibernate, Eclipse, Tomcat, Oracle Database, SVN and Windows XP Professionals.