

Anilkumar Yalagala
Senior Java Full Stack Developer
anil1417usa@gmail.com
484.995.9251

Professional Summary:

- Over 11 years of experience in all phases of Software Development Life Cycle (SDLC) which includes Design, Development, Integration, Documentation, and Testing of all software applications.
- Strong application development experience in using **Agile** Software Development methodologies like **Scrum, Sprint** and Traditional software models like **Waterfall and TDD**.
- Developed web-based enterprise applications using **core Java, J2EE, Servlets, JSP, EJB, JMS, JDBC, Hibernate, Web Services (SOAP and REST), MVC framework (Struts, Spring MVC), SQL and Object-Oriented Programming techniques**.
- Strong **AWS** expertise (EC2, EKS, EMR, Lambda, DynamoDB, S3, CloudFormation) coupled with container orchestration (Docker, Kubernetes, OpenShift) and automated CI/CD (GitHub Actions, Jenkins).
- Experienced in transferring data from different data sources into **HDFS** systems using **Kafka** producers, consumers and **Kafka** brokers
- Domain expertise in **end-to-end payment systems**—authorization, clearing, settlement, fraud detection, reconciliation, chargebacks, and regulatory reporting—across top-tier financial institutions.
- Using **Kotlin** for implementing new modules in the application.
- Have knowledge on **Kotlin** Android Extensions framework.
- Overall 6 years of experience in designing and developing of Enterprise based, EAI and Web applications using Apigee and Java/J2EE Technologies.
- Experience designing high-throughput, low-latency microservices in **Java 17, Kotlin, and Spring Boot** that process 10K+ TPS with sub-second SLAs, meeting PCI-DSS, SOX, and GDPR compliance.
- Have experience in API Management Technologies for more than 2 years in Apigee
- Used **Apache POI** to convert XML files to Excel files and to read Excel files.
- Experience on different IDEs such as **RAD, Eclipse, Spring Tool Suite, and IntelliJ**.
- Solid Experience in creating a **Kubernetes** Clusters in **AWS** and deploying a **Docker** Image on top of **Kubernetes** Cluster.
- Adept at observability and **SRE** practices—instrumenting code with **OpenTelemetry** and exposing **metrics/logs** to **Datadog, Prometheus, and ELK** for proactive monitoring and rapid incident response.
- Good experience in working with Cloud technologies like RedHat OpenShift, Kubernetes, Netflix Eureka
- Proficient in core java concepts like **Collection Framework, Multi-threading, Generics, Annotations, Serialization, Thread pools, JavaBeans, Externalization**.
- Proficient in implementation of frameworks like **Struts, Spring (Spring MVC, Spring Web Flow, Spring IOC, Spring AOP, Spring Security, Spring Boot, Spring Data, Spring Batch), JSF, AJAX** frameworks (**Rich Faces, My Faces**) and **ORM** frameworks like **Hibernate, IBATIS**.
- Expertise in developing **web pages** using **HTML, CSS, JavaScript, JQuery, Ajax, React JS, Angular JS, Angular 2, 4,6,8,9 Bootstrap** with webpage cross browser compatibility and **responsive web design** and Server scripting language like **Node JS, BackboneJS**.
- Proficient in developing **Web Services**, related technologies & frame works: **WSDL, SOAP, REST, JAX-WS, JAXB, JAX-RPC, AXIS and Jersey, SOAP UI** and generating client's client using Eclipse for Web Services consumption.
- 2 years' experience supporting and developing **Perl applications** used in online promotions and **loyalty system** programs
- Experience with and solid understanding of **Microservices**
- Expertise at **JDBC API** for connection and access to databases.
- Skilled in using **TypeScript for strongly-typed, modular front-end development**, ensuring type safety and improved debugging.
- Expertise in Creating **Hibernate configuration** file, **Hibernate Mapping** files and defined persistence classes to persist the data into **Oracle Database**.
- Expertise in Web Front End Technologies **Struts, Tag Libraries** and **Struts custom validation rules, Tiles, JSTL, Java Script, DHTML** and **XML/XSL**.
- Experience in integrating **TypeScript with React, Node.js, and RESTful APIs** for full-stack solutions.
- Leveraged cutting edge technology like Kotlin, Android JetPack, Retrofit, Navigation, ViewModel, Room, Actions.
- Expertise **XML** technologies like **DTD, XSD, XSLT** and various parsers like **DOM, SAX and JAXB**.
- Extensive experience working with RDBMS such as **MySQL, MS SQL Server, Teradata, DB2, Oracle**.
- Experience in working with the **NoSQL** Database like **MongoDB** to manage extremely large data sets.
- Managed **Amazon Web Services** like **EC2, S3 bucket, ELB, Auto-Scaling, SNS, SQS, AMI, IAM, Dynamo DB, Elasticsearch, Virtual Private Cloud (VPC)** through AWS Console and API Integration.
- Experienced in developing UML diagrams with **Star UML, Rational Rose**.
- Actively involved in the DevOps streamlining process through **Jenkins CI** and **CD** Release Automation.

- Proven ability in creating page Objects, data driven, keyword driven automation frameworks for J2EE based applications using **JUnit, Selenium Web Driver, Mockito, Groovy Spock, TestNG Testing, Cucumber, Ant, Maven, Bamboo and Jenkins.**
- Worked on deployment, installation, configuration and issues with Application servers like **Apache Tomcat, IBM Web Sphere, BEA WebLogic and JBoss Enterprise Application Server.**
- Experience in implementing Design Patterns like **Singleton, Factory, MVC, Data Access Object** and **Business Delegate.**
- Experience writing apps from scratch in **Kotlin.**
- Has experience from development of **spring boot microservice** application to build and deploy on cloud platform like RedHat OpenShift using automation of Jenkin jobs.
- Worked with Behaviour Driven Development process and **Test Driven Development (TDD)**
- Subversion **SVN, CVS, GIT** and Visual Source Safe as versioning software tools.
- Performed Service Level Testing for Web services using Soap UI, Postman and Apigee
- **Configured and created** application **log files using Log4J** to trace required application messages.
- Having Experience on **UNIX, Linux** commands and Deployment of Applications in Server and ability to work effectively on **Windows** platforms.

Technical Skills:

- **Programming Languages:** Core JAVA & Advanced Java, Python
- **Server-Side Technologies:** Servlets, JSP, JDBC, JSTL, JSF
- **Design Patterns:** MVC, Singleton, Factory, Composite, Session Facade
- **Frameworks:** Spring, Apache Wicket, Hibernate, Ejb3, Struts, Eclipse, Visual Studio
- **Web Technologies:** HTML5, CSS3, Javascript, Ajax, JQuery, Bootstrap, XML, JSON, Typescript, Angular JS, Angular 2/4/6/8/9/11, React JS, Node JS
- **Web Servers:** Jboss, Apache Tomcat Server, Weblogic, Docker Container and WebSphere
- **Web Services:** RESTful services, SOAP.
- **AWS Stack:** EC2, IAM, S3, Lambda, VPC, Cloud Front, Elastic Beanstalk.
- **Databases:** Cassandra, MySQL, Oracle, MongoDB
- **Source Code Control Tools:** SVN, CVS, GIT
- **ORM:** Hibernate, Spring JPA, Eclipse Link
- **Tracking Tools:** Jira, Redmine, Rally
- **Methodologies:** Agile, Waterfall, Test Driven Development (TDD).

Project Experience:

Client: Nationwide Mutual Insurance, Boston, MA

Sep 2024 – Present

Lead Java Full Stack Developer

Responsibilities:

- Involved in full SDLC processes including requirements gathering, analysis, design, development, and testing of multiple enterprise-scale financial applications within an Agile environment, ensuring rapid iterations and high-quality deliverables.
- Engineered high-throughput **payment microservices** with **Java 17, Kotlin, and Spring Boot 3**, sustaining >1,500 TPS at sub-50 ms p99 latency.
- Integrated **Change Data Capture (CDC)** workflows using **Kafka** and **Debezium** to stream data changes from transactional databases into the data lake for real-time analytics.
- Built a **fraud-detection** pipeline using **Kafka Streams + Spark Structured Streaming** on **Amazon EMR**; cut false-positive rates by 23 % while remaining **PCI DSS-compliant**.
- Designed and implemented **REST-based** microservices using **Spring Boot**, leveraging cloud-native principles to enhance scalability by **40%** and streamline integration with existing enterprise banking systems.
- Tokenized and encrypted card data with **AWS KMS & Vault**, passing internal security audits with zero critical findings.
- Designed and implemented **Docker-based** solutions to containerize **Java** microservices, enabling consistent deployment environments and reducing inconsistencies between development, testing, and production stages.
- Integrated **Docker** into the **CI/CD** pipeline, automating the build, test, and deployment processes. This reduced the time required for deployment cycles by **30%**, ensuring faster and more reliable releases.
- Architected a distributed infrastructure on **AWS (EC2, S3, and CloudFormation)**, achieving 95% uptime and improving application reliability, while enforcing security best practices and compliance standards.
- Exposed real-time payment metrics to **React + TypeScript** dashboards via **GraphQL**, giving risk analysts second-by-second visibility.
- Applied **Spring** Annotations and XML-based configurations for Dependency Injection, reducing boilerplate code and increasing developer productivity.
- Optimized **EMR** cluster autoscaling and instrumented services with **Datadog APM & custom Prometheus** exporters, trimming compute costs by 30 %.

- Developed batch and streaming ETL pipelines using **Apache Spark** (Spark SQL and DataFrames) to transform raw **CDC** data into optimized, analytics-ready formats.
- Drove **engineering excellence**—ran code reviews, set **Kotlin** style guides, and mentored 2 junior developers, reducing PR cycle time by 40 %.
- Designed and implemented **ReactJS front-end applications using TypeScript**, improving code maintainability by 40%.
- Implemented **Apache Airflow DAGs** to orchestrate and monitor end-to-end data ingestion and transformation workflows.
- Deployed and maintained applications on **Apache Tomcat** servers, executing continuous integration and continuous delivery pipelines for seamless upgrades and zero-downtime deployments.
- Executed build component scheduling and configurations with Maven, managing complex dependencies and improving build consistency across multiple projects.
- Utilized Spring Inversion of Control (IoC) to simplify application complexity, facilitating easier testing and integration of newly developed modules.
- Utilized **AWS S3** extensively for storing raw and processed data, performing **CRUD** operations through Spark jobs running on **EMR**.
- Integrated auto-wired POJO and DAO classes within Spring Controllers, reducing manual wiring efforts and increasing maintainability of the codebase.
- Managed **Kubernetes** clusters on **AWS** to orchestrate the deployment, scaling, and management of containerized applications, improving resource utilization and ensuring high availability in a cloud-native environment.
- Configured **Kubernetes** horizontal pod **autoscaling** and integrated with **AWS Elastic Load Balancing** (ELB) to dynamically scale applications based on demand, ensuring optimal performance and resource allocation during peak loads.
- Implemented custom build packs (WebSphere Liberty and JBoss) in PCF, optimizing containerized workloads and improving deployment efficiency in a cloud platform environment.
- Enhanced system reliability by redesigning **REST**-based microservices with **Spring Boot**, further improving scalability by **40%** and reducing deployment errors by **20%**, contributing to mission-critical trading platforms.
- Developed and maintained Angular 2 and Angular 9/11 applications for customer-facing web portals, utilizing responsive design methodologies and modern **JavaScript** techniques to improve user engagement and reduce page load times.
- Refactored existing JavaScript components to **TypeScript**, reducing runtime errors and enhancing performance.
- Leveraged functional programming techniques with **Lambda** expressions to simplify complex logic, improving code clarity and reducing defects in transaction processing modules.
- Deployed serverless data pipelines using **AWS EMR Serverless** and **Glue Data Catalog** for scalable ETL processing.
- Leveraged GIT for version control, establishing streamlined branching strategies, code reviews, and automated merge checks to ensure code integrity and project consistency.
- Built and optimized a custom **ETL** solution to migrate data from MS SQL Server to Couchbase, designing various document structures for flexible querying and ensuring high performance data retrieval by **RESTful** services.
- Developed enterprise-level web applications using **Spring MVC** frameworks, utilizing annotated controllers and configuration files for comprehensive navigation and request handling.
- Built **Node.js APIs**, integrated Single Sign-On (SSO) and enterprise email solutions, and optimized database operations using **Spring JDBC**—reducing query response times by 25% in large-scale data-intensive workflows.
- Implemented email and SSO integration with Node.js on the backend and Angular 2/4 on the frontend to provide a seamless login experience, increasing user satisfaction and authentication reliability.
- Developed dynamic, object-oriented JavaScript components integrated with AJAX, JSON, HTML5, CSS3, Angular 2/4, and Node.js for enhanced interactivity, reducing client-side rendering time and supporting multiple browsers.
- Integrated **Apache Camel** with ActiveMQ for robust messaging and routing capabilities, improving transaction throughput and enabling asynchronous processing across distributed services
- Developed secure **RESTful** web applications using **Spring Boot** and implemented Single Sign-On with **OAuth2**, enhancing user security and streamlining authentication processes for internal banking tools.
- Built **TypeScript-based APIs in Node.js**, implementing interfaces and generics for robust data handling.
- Established reliable database connectivity via **Spring JDBC**, crafting optimized SQL queries to manipulate large data sets from **DB2** databases, ensuring efficient data retrieval for real-time risk assessments.
- Implemented a Node.js-based configuration loader, enabling runtime modification of application settings, simplifying deployment workflows and reducing operational overhead.
- Created and maintained Hibernate entities that accurately mapped database schemas, leveraging inheritance strategies and caching mechanisms to improve database performance and maintain data integrity.
- Utilized **Spring** DAO concepts to interact with the DB2 database through JDBC templates, improving query performance and fostering consistent coding standards throughout the persistence layer.
- Employed **Log4j** for comprehensive logging and debugging at all application layers, streamlining troubleshooting efforts and proactively identifying performance bottlenecks in high-volume trading environments.

Environment: Core Java, Java, RAD 7.5, Web Sphere Application Server 8.5, Hibernate 4.30, Spring Framework 4.0, Angular 2/4, JQuery, Maven 3.3, Bootstrap, AJAX, JSON, JSP, Spring boot, JavaScript,

HTML5, CSS3, DOJO, RESTful, JAX-RS, JDBC, Camel, ActiveMQ (AMQ), Couch Base, DB2, Log4j, Jenkins, Apache Tomcat.DAO.

Client: PNC bank, Charlotte, NC
Senior Java Full Stack Developer

Mar 2022 – Aug2024

Responsibilities:

- Responsible for creating an instance on Amazon EC2 (**AWS**), Pivotal Cloud Foundry and deployed the application on it. Involved in the development of Micro services for all the models using combination of Spring Boot and Spring Security.
- Designed real-time **Spark Streaming** jobs in **PySpark** to process event data from **Apache Kafka** for near real-time processing.
- Designed event-driven reconciliation services on **AWS** (Spring Boot + Kafka + DynamoDB) that balanced 10 M+ ACH & **card transactions** daily.
- Worked on Integration of Micro services deployment patterns with the PAAS platform on top of AWS and Integrating with the Config server and ZUUL proxy.
- Authored **Spark ETL jobs** on **EMR** to hydrate an **S3 data lake**, enabling real-time fraud analytics and historical reporting.
- Worked on POC of high-end APIGEE innovative proxies and On-boarding API's.
- Worked with **jQuery** plugins like **Banner, Slider, Accordion, Tabs, jQuery Dropdown, Image Gallery**.
- Developed RESTful web services using **Jersey, Spring framework** and **JAX-RS**.
- Developed Application to assess JSON and **XML** from Restful web service from consumer side using **JavaScript** and **Angular 11**
- Integrated **React/TypeScript** UI widgets into the member-dispute workflow, boosting agent case-resolution speed by 18 %.
- Developed Spring boot batch applications and deployed on Linux batch servers and cloud platforms like Red hat OpenShift
- Rolled out an observability stack (Prometheus, Grafana, ELK) with **latency/error SLOs**; MTTR dropped from 45 min to 12 min.
- Built modular **ETL** frameworks using **Spark** and deployed them on **AWS EMR** for scalable batch and streaming transformations.
- Migrated legacy JavaScript projects to **TypeScript**, improving static type checking and reducing debugging time by 30%.
- Integrated **Python Lambdas** with **AWS** Step Functions to automate data validation and post-processing flows.
- Developed and maintained **Angular 9/11 applications with TypeScript**, ensuring modular UI components.
- Experience writing apps from scratch in Kotlin with Android studio
- Implemented **Glue Data Catalog** to register metadata for ingested datasets, improving data discovery and schema tracking.
- Embedded **secure-coding** and **PCI DSS** controls into **CI pipelines**, including static analysis, dependency scanning, and secrets detection.
- Produced Spring webservices and deployed on Linux batch servers and cloud platforms like Red hat OpenShift.
- Involved in writing application level code to interact with APIs, Web Services using **AJAX, JSON and XML**.
- Worked **with Angular 11** routers, forms, modules, DOM, expressions, controllers, filters and Http Request.
- Configure and run gulp tasks in **Node.js** runtime and handle injection, **CSS** pre-processing, test-running.
- Developed unit test scripts using Karma and Jasmine for **Angular 11** framework and Mocha for Node JS API end points.
- Productionized **ML fraud-scoring** models by serving **SageMaker** endpoints through **Kafka Streams**, lowering fraud loss by 12 %.
- Worked with **AWS S3** and **EMR** to efficiently manage large-scale data processing with performance-tuned Spark configurations.
- Leveraged **TypeScript with NestJS** for backend microservices, improving maintainability and reducing API errors.
- The mobile application was developed with JAVA mixed with **Kotlin** using Android Studio and Web API with .NET Core 2 using Visual Studio.
- Designed FACADE design patterns for APIGEE using JAVA platform.
- Performed Continuous Code Quality on the using SonarQube.
- Extensively worked in Test Driven Development (TDD) framework using JUnit.
- Worked on transition from legacy products to newer version of APIGEE Edge.
- **Pre-processor's (SASS/LESS), JavaScript, Bootstrap, JDK 1.8, Ajax, Web Services XML, schemas**.
- Designed and developed Micro Services business components using Spring Boot.
- Developed Angular controllers, services, filters, and developed HTML templates with various Angular directives such as ng-app, ng-controller, ng-repeat, ng-if, ng-show, ng-hide, ng-change, ng-blur etc.
- Implemented the Model View Control structure using **Spring** and **Angular 11**.
- Implemented the CI/CD process with adHoc releases for the angular web app including the external libs related to the surveys, source code configurations to be packaged using the maven by including the dependencies versioning

in the bit bucket, code testing through the Sonar Qube, JFROG repository for storing the artefacts and deploying in to the relevant environment CDN through the bamboo pipeline with Jira build tracking

- Enhanced the Portal UI using **HTML, Java Script, XML, Java, and CSS** as per the requirements and provided the client side **Java script** validations and Server side Bean Validation Framework.
 - Used **J2EE** design patterns like **Model View Controller(MVC), Singleton, Factory, Data Access Object (DAO), Data Transfer Object (DTO)** and **Service Locator** etc.
 - Implemented Micro services architecture using Spring BOOT enabled REST API with the Spring Cloud modules.
 - Expertise in Maven for build, Jenkins and SonarQube continuous integration
 - Used **Angular 9** as framework to create a Single Page Application (**SPA**) which can bind data to specific views and synchronize data with server using **SASS, LESS, Bootstrap**.
 - Utilized **Kubernetes and Docker** for the runtime environment of the CI/CD system to build, test, deploy.
 - Installation of Storm and **Kafka** on 4 node cluster and written **Kafka** producer to collect events from Rest API and push them to broker.
 - Implemented Kotlin with Android content providers for implementing features like sending E-mails, SMS, etc., from the app and designed suitable GUI for these applications
 - Apart from that the application provides various loyalty programs to its customers. This system consumes the JSON data exposed from API services
 - Migrated some modules to KOTLIN from Java.
 - Implemented data-binding using **NodeJS** for front-end development of a current web application.
 - Experience in creating RESTful web services using Jersey, **SPRING** framework and JAX-RS.
 - Used spring **framework** 4.3 to create the beans needed and injected dependency using Dependency Injection
 - Creating and designing data ingest pipelines using technologies such as Apache Storm-**Kafka**.
 - Developed **angular** unit tests for testing **angular** components using Jasmine and Karma.
 - Extensive knowledge on **AWS** EC2, Amazon S3, Amazon Elastic Load balancing and other parts of **AWS** infrastructure.
 - Worked **on AWS** multi factor authentication (MFA) related to security groups including rest security.
 - Changing the AWS infrastructure Elastic Beanstalk to **Docker** with **Kubernetes**.
 - Moving existing monolithic architecture to **micro services** and migrating them to **AWS** cloud.
 - Created some widgets using Kendo UI and successfully integrated Kendo UI components, **node.js** and module.js with polymer.
 - Designed and developed the application using agile methodology and followed Test Driven Development (TDD), Scrum .
 - Configuring windows APIGEE servers using FTP and HTTP protocols.
 - Implemented generating the verification hash code using crypto package on **Node.JS** and Added user validations and user permissions by using the **Node.js**.
 - Coded extensively using **JavaScript, Angular 9** to make rich internet web application for a Single page app.
 - Building application using maven and deploying to local Web Logic or tomcat server
 - Develop Java code (Controller Bean, Delegate, Facade Bean, Manger and Data Access) and testing in local work space.
 - Implemented the Kotlin with Android studio for smooth functioning of the location based services GPS, Bluetooth on Android
 - Worked on building & deploying Java code through Jenkins, used Jenkins Amazon Web Services (**AWS**) Code Deploy plug in to deploy to **AWS** and worked **AWS** Cloud management.
 - Implemented **Spring** boot micro services to process the messages into **Kafka** cluster setup
 - Used **Karma** Test runner and **Jasmine** Testing frameworks.
 - Develop SQL queries based on business rules, find query cost and optimizing the query
 - Writing Splunk Queries, Expertise in searching, monitoring, analyzing and visualizing Splunk logs
- Environment:** HTML, CSS, JavaScript, Web Service XML, AWS, Spring, Angular 9/11, Kafka JQuery, JSP, Spring, Apigee, JSTL, EJB, JMS, XML, XSLT, JDBC, AJAX, HTML, CSS, Bootstrap, Java Script, Typescript, Node.js, Restful, OpenShift, SOAP, Kotlin, jQuery, SonarQube, Hibernate, Docker. Splunk, Apache Kafka, Karma, Jasmine

Client: Krogar, San Antonio, TX

Aug 2020 – Feb 2022

Java Full Stack Developer

Responsibilities:

- Involved in Full stack development based on J2EE using Hibernate and Spring Framework.
- Created web services and desktop applications to access and display data needed by support teams using **Ajax, JavaScript, React JS, Node.JS, CSS and HTML**.
- Worked in using **React JS components, Forms, Events, Keys, Router, Animations and Flux concept**.
- Implemented **React JS** code to handle cross browser compatibility issues in Chrome, IE 7, 8, 9, Safari.
- Developed user interface by using the **React JS**.
- Developed custom **TypeScript hooks** for stateful logic in React applications.
- Setup and built **AWS infrastructure** various resources, **VPC, EC2, S3, Dynamo DB** and **RDS** in Cloud Formation **JSON templates**.

- Built **Spark batch ETL** jobs for cleansing, joining, and transforming large volumes of semi-structured data into curated datasets stored on **S3**.
- Leveraged cutting edge technology like Kotlin, Android JetPack, Retrofit, Navigation, ViewModel, Room, Actions.
- Implemented Micro Services based Cloud Architecture on **Amazon Web Services (AWS)** Platform and Cloud Foundry.
- Leveraged **PySpark DataFrames** and **Spark SQL** for efficient processing of big data workflows.
- Integrated TypeScript with **Webpack and Babel** to ensure cross-browser compatibility.
- Implemented **TypeScript in RESTful APIs** using Express.js for structured, maintainable backend services.
- Used APiGEE Killer to check the performance of each component of Apigee product and fixed lots of issues.
- Using APiGEE configuration and generated proxies service discovery definition, telemetry definition, logging and security.
- Worked on Fixing the bugs and code smells for the using SonarQube
- Scheduled and maintained **Apache Airflow** pipelines to manage dependencies across complex data workflows.
- Followed TDD Test Driven Development and developed test cases by using JUnit for unit testing for each and every module developed.
- Developed data ingestion pipelines from source systems using **Debezium** and **Kafka**, ensuring real-time capture of database changes.
- Applied performance tuning techniques for **Spark** jobs to optimize execution times and resource utilization on **AWS EMR**.
- Involved in developing the Graphical User Interface using **HTML, JavaScript, CSS and JSP**.
- Involved in the development of the User Interfaces using **HTML, JSP, JS, Dojo Tool Kit, CSS** and AJAX.
- Developed **RESTful web services** using **JAX-RS, Apache CXF** and spring frameworks and persistence using **Hibernate** and **JPA ORM** frameworks.
- Jersey implementation of JAX-RS is utilized to develop **RESTful** Web services.
- Expert in HTML5/CSS3 development and have experience **React.js**, Responsive design
- Proficiency in adoption of architecture paradigms such as SOA, Web Services, Middleware technologies.
- Configured and build Asynchronous communication with **JMS** services with MQ services.
- Designed and developed a JDBC driver that intercepts a create view DDL parses it, converts it to a relational model, and stores in the database.
- Experience writing apps from scratch in Kotlin.
- Developed **Spring Batch** application for **batch** processing using Spring4.0 and improved the performance of the backend **batch** processes using Multithreading and concurrent package API.
- Extensively used the backbone.js, Node JS, Angular.js frame works, Amber JS, Require JS, React JS, JSON and JavaScript frameworks jQuery.
- Created hybrid desktop applications for Mac and Windows Redux-saga, nets, styled- components, Gatsby, D3, node.js, cypress, electron, jest, sass,
- Worked in functional programming using **Kotlin** in server-side (Spring Boot) and client - side web development (react-redux, react-router)
- Involved in fetching **JSON** data with in Orchestration Layer with Spring Boot application and displaying it in the UI using **React JS** and followed **ECMAScript** standards for developing.
- Instructed teams on best practices around PCF and Spring Boot.
- Triggered the automation jobs using Jenkins to get the cucumber **JSON** reports.
- Involved in writing test scripts using java and executed it through selenium cucumber.
- Responsible for implementing the business requirements using **Spring Boot**, Spring Data and Spring Security.
- Involved in creating the comparison document for selecting the API Gateway.
- Focused on Test Driven Development; thereby creating detailed JUnit tests for every single piece of functionality before writing the functionality.
- Developed application in Kotlin and Java, implementing new features in Kotlin and migrating older classes into Kotlin
- Worked on relational database design and writing **SQL** queries, stored procedures, views, and functions with **Oracle**.
- Involved in creating various Data Access Objects (DAO) for addition, modification and deletion of records using various specification files.
- Created tables, triggers, stored procedures, **SQL** queries, joins, integrity constraints and views for multiple databases, **Oracle** using Toad tool.
- Developed, debugged, integrated and deployed applications using **Eclipse (IDE)** and used **GIT** as a Version Control.
- Worked on container based deployments using **Docker**, working with **Docker images, Docker Hub** and **Docker-registries** and **Kubernetes**.
- Created **React JS** directives, factories, and services for developing single page web applications.
- Implemented navigation using Spring MVC controllers, configured controllers using Spring MVC annotations and configuration files.
- Development of Java REST API using Spring Boot and PostgreSQL database for data storage. Using Kotlin and Groovy as base coding language for Java programs.

- Designed GUI and front-end web pages using **JSP** and Bootstrap and developed **JavaScript** solution for user input data validation.
- Configured Staging and Testing and Production environment for APIGEE Edge and APIGEE cloud-based sites.
- Implemented Web based pages and site design using **HTML**, **XHTML**, **DHTML**, **CSS**, Themes, Master Pages, Site Map, XML, TSF, XSLT, JavaScript, jQuery and **Ajax**.
- Utilize and program Restful Web services, EJB/JPA, and Hibernate entity.
- Worked on using **WebSphere** server built in JMS provider for communicating with legacy applications.
- Designed and developed the **REST** based **Micro services** using the **Spring Boot**.
- Worked with java libraries to access **AWS** facilities as Simple Storage service, Simple queue and Simple DB.
- Used **React Js** for creating interactive UI's using one-way data flow, virtual DOM, **JSX**, React Native concepts and maintained states in the stores and dispatched the actions using Redux.
- Used Amazon Cloud Watch to monitor **AWS** services and Amazon Cloud Watch logs to monitor application.
- Responsible for **React UI** and architecture. Including **Tree**, Building components **library**, **Table Grid** and Slide-View.

Environment: HTML, CSS, React Js, Hibernate, Node JS, Micro Services, AWS, EC2, S3 Lambda, Auto Scaling, Apigee, Cloud Watch, SQL, JSP, AJAX, Spring boot, SonarQube, Json, JPA, HTML, CSS, Java Script, Jquery, Spring Boot, Docker, Kotlin, Kubernetes, Jenkins, Apache Kafka, Oracle, Eclipse, Git

Client: Adp , India
Java/J2EE Developer

Jan 2019 – Dec 2019

Responsibilities:

- Developed various J2EE components like **Servlets**, **JSP**, **JSTL**, **AJAX**, **SAX**, **XSLT**, **JAXP**, **JNDI**, **LDAP**, **JMS**, **MQ Series**.
- Used **JDBC** to persist **Java Objects** into the database
- Used **Spring framework MVC** in the development of new modules based on requirements.
- Used **Angular 2/4**, **JavaScript** for validating client-side validations.
- Developing and designing back-end Loyalty System Program for credit card rebate processing
- Utilized various utilities like **JSP**, **JSTL**, **JavaScript**, **HTML**, & **CSS**, **jQuery**, **Angular JS**, **Angular 2/4**.
- Configured and set up Java Workspace which included setting up server and theme installation and configuration. Became familiar with using and navigating through Java Workspace (**Eclipse**).
- Debugged Java classes used to modify web pages such as **Business Objects** and **Data Access Objects**.
- Implemented **JDBC** to perform the database calls from the Java layer.
- Consumed and Exposed webservice and deployed on Linux batch servers and cloud platforms like Red hat OpenShift and Azure.
- Used **Angular 4** and jQuery for the logical implementation part at client side for all the application.
- Refactored existing project to make it more **RESTful** and thread safe.
- For XML marshalling and unmarshalling used **JAXB**. Worked with databases like **Oracle**.
- Tested many components of web application and documented my observations.
- Used **Servlets** as an interface between frontend and backend to handle **HTTP** requests and to send the response back to the front end.
- Involved in the creation of dynamic web pages with the use of **JSP** and **HTML**.
- Developed application using **Spring MVC**, **JSP**, **JSTL** and **AJAX** on the presentation layer, the business layer is built using spring and the persistent layer uses **Hibernate**.
- Used WSDL and **SOAP** protocol for Web Services implementation.
- Worked in **Struts framework** based on **MVC Architecture**.
- Understand existing Java services so that they can be converted to APIGEE based services.
- Developed Restful API's using APIGEE platform.
- Designed and developed Spring and REST based **Micro Services** using the **Spring Boot**, also developed **RESTful web services** with JAX-RS using **Spring Boot** and **Micro Services** Architecture
- Used **SQL** and **PL/SQL** stored procedures, functions to implement business logics in **Oracle**.
- Used micro-framework to implement with REST API and Mongo DB (NO SQL) as back-end database.
- Compiled and built the application using ANT scripts and deployed the application.
- Implemented logging for the whole application using **Log4j**.
- Troubleshooting and monitoring API proxies running on APIGEE using Trace tool.
- Developed **Web Services** using **XML messages** that use **SOAP**. Developed Web Services for Payment Transaction and Payment Release.
- Performed unit testing by using **Junit** testing tool and debugged using Jasmine.
- Developed an automated process for builds and deployments using Jenkins and **Maven**.
- Participated in unit testing, integration testing and installation of the system.

Environment: Core Java, J2EE, HTML, CSS, JSP, JDBC, JSP, JavaScript, Spring Boot, Micro Services, jQuery, Angular JS, Angular 2/4, XML, JSON, Apigee, Servlets, Spring Framework, PL/SQL, Hibernate, OpenShift, Oracle, JAXB, RESTful, Log4j.

Client: Redington – Chennai, India.

Oct 2015 – Dec 2018

Core Java Developer

Responsibilities:

- Involved in the Software Development and Lifecycle for analysis, design, testing and implementation of the project.
- Experience on Collections, Exception Handling, Garbage Collection in **Core Java**.
- Created **HTML, CSS, and JavaScript** and **DHTML** pages for Presentation Layer.
- Used spring framework for implementing **Spring IOC, JDBC, and ORM**
- Designed use case diagrams, class, and sequence diagrams in the design phase of the project and for workflow settings.
- Developed code changes using **core java, J2EE** technologies and used **Eclipse, JDBC and MySQL**.
- Responsible for Writing **SQL Queries** and executing using **JDBC** API
- Good Experience in developing web/enterprise applications using SDLC methodologies like waterfall/agile.
- Used Servlets for maintaining server-side business logic.
- Taken whole responsibility and developed module alone with in time specified.
- Involved in preparation of **System Use Cases (SUC's)** and **Unit Test Speciation (UTS)**
- Experience in writing test cases for Junit testing and thus verifying and validating applications.
- Worked on deployment of virtual machines and troubleshooting on VM management.
- Used **Maven** to build and deploy the application on web logic server.
- Used **Git** for version control across common source code used by developers.
- Experience in writing **SQL Stored** procedures and functions.
- Used various Design patterns like service locator, session facade, singleton, and factory pattern.
- Involved in Daily Scrum meetings, Sprint planning and estimation of the tasks for user stories and participated in retrospective and presenting demo at the end of the **sprint**

Environment: Core Java, Collections, Multi-Threading, AWS, Struts, JSP, JDBC, Junit, SDLC, Eclipse, Tomcat, HTML, CSS, JavaScript, SQL, CVS, Waterfall, Design Patterns, DAO.

Client: IBM, Hyderabad, India

Jun 2014 – Sep 2015

Java Developer

Responsibilities:

- Involved in development, customization and enhancement of the product in Windows, Linux platform.
- Wrote **Form Handlers, Managed Bean Classes, VO's** using **struts MVC framework (Struts)** for the modules
- Developed **Core Java** components to develop the transaction report. Stored the dynamic data on to Collection Objects and used the predefined methods to perform all DB related operations efficiently.
- Written Utility classes for getting Connection using **JDBC**
- Responsible for Developing **Managed Bean and DAO** classes for Business Implementations
- Used **Tag Libraries** for Screen Design using JSP
- Used **JDBC** to interact with the underlying **MySQL** Database.
- Extensively worked on **SQL** for writing complex queries in the business logic layer.
- Implemented **Struts MVC** design pattern and front controller pattern.
- Developed unit test cases using **JUnit**.
- Used **Eclipse IDE** for Development and Version Control as **Clear case**.
- Used **Log4j** for logging purposes during the development of the application.
- Deployed Applications on **Apache Tomcat Server** and used **CVS** as the version control manager.
- Used **Eclipse IDE** to integrate the different frameworks to build the overall application.

Environment: Windows, Linux, Java, Servlets, JSP, ATG Dynamo, JDBC, CVS, CSS, Java Script,