Poojith Reddy A

Senior Java Full Stack Developer

Phone: 312-536-9779 | Email: poojithreddy.se@gmail.com

LinkedIn

PROFESSIONAL SUMMARY

- Senior Java Full Stack Developer with **11+ years of experience** delivering high-performance enterprise applications using Java/J2EE, Spring MVC, Hibernate, REST APIs, and modern JavaScript frameworks.
- Specialized in **Core Java multithreading, Executor Framework, and concurrent collections**, ensuring scalable and responsive server-side applications with optimized resource utilization.
- Designed and implemented **RESTful APIs** with Spring Boot and JAX-RS, enabling seamless system integrations and supporting **over 10M+ API transactions per day**.
- Developed front-end solutions with **Angular, JSP, HTML5, CSS3, and jQuery**, building responsive and cross-browser compatible applications that improved end-user engagement by **25%**.
- Strong background in Core Java concepts, including Exception handling, Multi-threading, Lambda functions, Synchronization, Serialization, IO, Collections framework, Java Beans, Executor Services, and Thread pools.
- Demonstrated proficiency in Java 17, leveraging its latest features and enhancements for modern Java development.
- Experience in migrating and upgrading existing Java applications to Java 17, ensuring compatibility and optimization.
- Expert in front-end technologies such as HTML, **HTML5**, **CSS**, **JavaScript**, **CSS3**, **SASS, jQuery, Angular (2/4/7/11/14),** AJAX**, JSTL, and JSON,** Bootstrap**, Backbone.js, Node JS, Express.js**.
- Hands on experience with custom tag libraries, **DOM, bootstrap, Tailwind CSS, XML, XSLT, jQuery, Vue.JS and angular JS**.
- Expertise in using **React JS components, Forms, Events, Keys,** and **Router,** plus **Redux,** and **Flux.**
- Extensive experience with Java back-end technologies like Servlets, **JSP,** and **JDBC**. Familiarity with **Quarkus** and **Micronaut** for building lightweight and reactive microservices.
- Developed networking policies for **Docker** containers and worked on open-source development tools like **Docker** Containers, **Mesos** and **Kubernetes**.
- Experienced in various Spring components like **Spring MVC**, **AOP**, **Spring IOC**, **Spring JDBC**, **Spring JPA**, **Spring Securities**, and **Spring Batch**.
- Expertise in Spring Security, **AOP** (Aspect-oriented programming), Spring Core, **Spring Boot**, Spring Integration, Spring Security, and Event Driven architecture.
- Leveraged containerization technologies such as **Docker** and **Kubernetes** to package and deploy microservices, enabling easy scalability and portability across environments.
- Experience in working with **AWS** (Amazon Web Services), **EC2 AMI**, **Glue**, **AWS Config, VPC, IAM, Auto-scaling**, **RDS**, **Dynamo DB**, **S3**, **security groups,** and **Cloud Formation Templates**.
- Working experience with **S3, ELB, EBS, ECS, ECR**. Set up **CI/CD** pipelines for Microservices on AWS using App services.
- Hands-on experience with tools like Hudson, **Sonar**, Jenkins, Apache Ant, **Maven**, Bamboo, and MS Build, OpenAPI v3, OAuth, and Pos.
- Excellence in Apache Kafka, **Apache Flink**, and **Apache Spark** for real-time data processing and leveraged **Avro**, Protocol Buffers, and Apache **Parquet** for efficient data serialization.
- Proficiency in event-driven architectures, fault-tolerant pipelines, and integration with machine learning frameworks. Experience in defining **CAMEL** content-based switches for flow deviation.
- Working experience with middleware technologies like **MQ-Series**, **RabbitMQ**, **Apache Spark**, **Oracle Fusion** (BPEL and OSB), and **Apache CAMEL**.
- Extensive experience using **JSF**, **Servlets**, **JSP**, **JDBC**, and **JMS.**
- Developed **responsive, component-based UIs** with React.js, leveraging hooks, context API, and state management libraries (Redux) for scalable front-end applications.
- Integrated React with **RESTful APIs and GraphQL**, ensuring efficient data flow and optimized rendering for high-performance applications.

- Experience with infrastructure as code tools like **Terraform** and **Ansible**.
- Proficient experience working with relational databases (**RDBMS**) such as **MySQL, DB2, Oracle, PostgreSQL**, **Aurora, and Amazon RDS**.
- Worked with **NoSQL** databases such as **Couch Base, MongoDB, and Cassandra**. Skilled in various aspects of **WEB MVC** (model-view-controller), including Auto Wiring and **JDBC**.
- Proficient in designing, developing, and deploying scalable and reliable Java applications on **Azure** cloud infrastructure.
- Skilled in leveraging **Azure** services to optimize application performance, reliability, and security.
- Used **Azure Resource Manager** (ARM) to deploy, update, or delete all the resources for solution in a single, coordinated operation.
- Built **dynamic, single-page applications (SPAs)** using Angular, with strong expertise in modules, services, RxJS, and dependency injection.
- Implemented **end-to-end solutions with Angular and TypeScript**, integrating REST APIs, forms, and routing to deliver robust and user-friendly interfaces.
- Designed and implemented secure, scalable infrastructure using **Terraform**, **CloudFormation**, and **AWS IAM**, enabling efficient resource provisioning and consistent deployment across environments.
- Integrated **DevSecOps practices** with tools like **SonarQube**, **Checkmarx**, and **Secrets Manager**, ensuring code quality, vulnerability scanning, and secure deployment pipelines.
- Built monitoring and logging solutions with **AWS CloudWatch**, **ELK stack**, and **PagerDuty**, supporting proactive issue resolution and ensuring high system availability in production.
- Hands-on with **SQL performance tuning, indexing, stored procedures, and query optimization**, reducing execution time by **up to 40%** on critical workflows.
- Strong expertise in **Spring, Spring MVC, and Hibernate/MyBatis ORM**, implementing data access layers, transaction management, and caching strategies for high-volume applications.
- Leveraged **GitHub Copilot Agent Mode** to accelerate Java development, improving developer productivity by **20% in code generation and debugging tasks**.
- Proven ability in **scalable application design** on JBoss, Tomcat, and Apache servers, ensuring high availability and secure deployments.
- Built AJAX, XML/XSLT, and JSON-based integrations, enabling interactive user interfaces and real-time communication between client and server systems.
- Experience with Representational State Transfer (**REST**) and Simple Object Access Protocol (**SOAP**) Web Services to expose business logic. Utilized logging frameworks like **log4J**, and **SLF4J**.
- Configured and deployed Elasticsearch clusters for log aggregation, search, and analytics, forming the core of ELK (Elasticsearch, Logstash, Kibana) stack.
- Developed Logstash pipelines to ingest, parse, and enrich log data from various sources into Elasticsearch indices, ensuring **centralized logging and monitoring** capabilities.
- Implemented data visualization and dashboards using **Kibana**, enabling stakeholders to gain insights and monitor system performance through real-time analytics.
- Experience in web services technologies like **REST, JAX-RS, SOAP, WSDL, JMS**, and Service Oriented Architecture (SOA).
- Excellent hands-on experience in configuration management tools like **SVN, CVS** and **GIT**.
- Long-time experience in working with version control systems like GIT and using Source code management client tools like **GitHub**, and **Bitbucket**.
- Experience in using built tools like Maven, Ant, Jenkins and Log4J to automate the process.
- Proficient in **designing, deploying, and managing scalable AWS solutions** using EC2, Lambda, RDS, S3, EKS, and CloudWatch, with automation via CloudFormation and Terraform.
- Strong expertise in **AWS security and compliance** with IAM, VPC, encryption, CloudTrail, and GuardDuty for secure and compliant environments.

Technical Skills:

Languages	Java (8/11/17), SQL, TypeScript, JavaScript, PHP, Python, .NET, C#, C.
Java/J2EE Technologies	Servlets, Spring, JPA, JTA, JDBC, JSP, JSTL, RMI, Applets, XML, DOM, JNDI, Core Java Multithreading (Threads, Synchronization, Locks), Executor Framework (Callable/Future, Fork/Join Pool, Concurrent Collections)
Frameworks	Spring 5.0, Spring MVC, Spring Batch, Spring Security, Spring AOP, Spring Core, Spring IoC, Struts (1.2, 2.1, 2.5), Hibernate 5.x, Django, JSF, GWT, Vaadin, MyBatis.
Scripting Languages	JSP (1.2, 2.1), JSP-EL, JSTL, JavaScript, TypeScript, HTML4, HTML5, CSS2, CSS3, Custom Tags, jQuery (1.8, 1.9), jQuery UI, ECMAScript 5/6, Node.js
Web Technologies	HTML5, CSS3, AJAX, jQuery, Bootstrap, XML, JSON, Kotlin, Scala, Grunt, Gulp.
Databases	SQL Server, MySQL, Oracle, MongoDB, DB2, PostgreSQL, MS-SQL, SQL Performance Tuning (Indexing, Query Optimization, Stored Procedures).
JavaScript Frameworks	AngularJS (2, 4), Angular 8, Angular 11, React.js, Express.js, Node.js, Backbone.js, Hapi.js, Handlebars.js.
Web Services	RESTful, SOAP, GraphQL, JAX-RS, JAX-WS, WSDL, XSD, Apache Axis.
DevOps & Cloud Tooling	GitLab CI/CD, Jenkins, GitHub Actions, Bitbucket Pipelines, Docker, Kubernetes (EKS), Terraform, AWS CloudFormation, HCL, Ansible, AWS CloudWatch, ELK Stack, Prometheus, Grafana, HashiCorp Vault, AWS Secrets Manager.
Web Application Servers	Oracle WebLogic (9.x/10.x), IBM WebSphere (6.x/8.x), Apache Tomcat (6/7/8), GlassFish 4.0,JBoss EAP.
Monitoring & Observability	New Relic, AppDynamics, Datadog, Splunk, CloudTrail, CloudWatch, ELK Stack, Prometheus, Grafana.
Testing Tools	JUnit (3,4), DBUnit, Juniper, HTTPUnit, Load Test, Timed Test, Selenium, FitNesse, EasyMock, Mockito, Jasmine, Karma, Mocha, Chai, JMeter, LoadUI, LoadRunner, HP Quality Center, WinRunner.
Infra / DevOps Tools:	Docker, Kubernetes, Terraform, Ansible, Jenkins, GitHub Actions, Maven, Gradle, Apache Tomcat, Nginx, ELK Stack (Elasticsearch, Logstash, Kibana), Prometheus, Grafana, SonarQube.
Version Control	SVN, CVS, GIT, GitHub, Bit Bucket, and Clear Case.
Build Tools	Maven, Gradle, ANT.
IDEs	Eclipse, IntelliJ IDEA, Spring Tool Suite (STS), Rational Application Developer (RAD 6.0/7.0).
Cloud	AWS (EC2, S3, RDS, Lambda, etc.), Azure (App Service, Functions, SQL DB, Cosmos DB, Storage), IBM Bluemix, Pivotal Cloud Foundry, Spring Cloud.
Security & Secrets Management	Okta, OAuth2, OpenID Connect, Keycloak, AWS IAM, HashiCorp Vault, AWS Secrets Manager, Snyk
Methodologies	Agile, Waterfall, OOAD, SCRUM, Agile (XP, Scrum), Rational Unified Process (RUP), Kanban.
AI & Developer Tools	GitHub Copilot, GitHub Copilot Agent Mode (for Java), ChatGPT, OpenAI Codex, Copilot Chat, AI-assisted Programming, Prompt Engineering

Work Experience:

Bank of America, Jersey City, NJ, USA **Dec 2023 – Current**

Senior Java Full Stack Developer

Responsibilities:

- Engaged in reviewing business requirements and functional designs for applications and participated in creating wireframes.
- Collaborated in Agile process, bi-weekly Sprints, and daily Scrums to discuss the development and achieve TDD (Test-driven development approach).
- Implemented **Executor Framework and Java concurrency utilities** to parallelize vehicle data ingestion, reducing processing latency by 30%.
- Developed **multithreaded microservices** in Spring Boot with thread pools, synchronization, and atomic variables for high-throughput workloads.

- Designed **SQL stored procedures and optimized indexing strategies** to improve query performance for reporting dashboards.
- Developed Single Page Applications (SPA) using **HTML5**, **CSS3**, **JavaScript**, **Bootstrap4**, **Angular** 11, Angular CLI, and **Typescript**. Leveraged Kotlin and **Scala's** interoperability with Java, enabling seamless integration.
- Created custom directives in **Angular 11** for reusable components like multi-field form elements and background file uploads. Used Reactive & template-driven forms to associate **HTML** elements.
- Used various predefined components from NPM and redux library and deployed application to **NGINX**.
- Created custom, self-contained, reusable and testable Angular 11 modules, controllers, and directives.
- Worked with **Angular CLI** and Typescript in Angular 11 to create Controllers and Components for web pages.
- Developed responsive and interactive user interfaces using Vue.js and **TypeScript**, leveraging its component-based architecture and reactive data binding capabilities. Integrated databases with **Node.js**.
- Designed and developed REST-based microservices using **Express.js** and Node.js, as well as Spring Boot and Spring Data with JPA. Created **Angular 11 + JSP hybrid interfaces** supporting responsive design and cross-browser compatibility.
- Built **REST APIs with Spring MVC and Hibernate/MyBatis**, ensuring efficient data persistence with caching (EhCache/Redis).
- Integrated **GitHub Copilot in agent mode** to accelerate backend code refactoring and reduce boilerplate.
- Enhanced application **fault tolerance** with error-handling strategies in multithreaded services, avoiding race conditions and deadlocks.
- Conducted **system integration testing (SIT)**, validating API interoperability with Kafka, Elasticsearch, and external vendor services.
- Collaborated with analysts to refine business requirements, producing **low-latency solutions for 10K+ concurrent users**.
- Tuned JVM performance using **JProfiler and VisualVM**, optimizing memory usage under high load.
- Designed the application's architecture, leveraging microservices architecture, Spring Framework, and containerization with Docker and Kubernetes by ensuring scalability, reliability, and security.
- Developed microservices using **Spring MVC**, **Spring Boot**, and **Spring Cloud**. Containerized microservices using Docker and **Amazon EC2** containers using Jenkins. Utilized **Spring Data** and **Spring Security**.
- Produced and consumed **RESTful** web services within the **Spring Boot** framework using **JWT** and **OAuth**.
- Worked with **Splunk** and **ELK** stack for monitoring and analytics solutions.
- Implemented data ingestion pipelines using **Apache Kafka** and **Apache Flink** for real-time data processing.
- Collaborated closely with the **Kafka Admin** team to set up Kafka clusters in QA and production environments.
- Implemented processes to reprocess failure messages in **Kafka** using **offset IDs**.
- Implemented **MQTT**-based solutions for IoT applications requiring low-latency, low-bandwidth communication, and interoperability.
- Developed RESTful APIs and web services using Java and Spring for seamless integration with client applications and external systems.
- Proficient in designing and implementing queue-based architectures using messaging systems such as Kafka, **IBM MQ**, and event streaming platforms.
- Designed scalable and fault-tolerant messaging systems for asynchronous communication, event-driven architectures, and decoupled microservices.
- Leveraged Java Enterprise Edition (J2EE) technologies, including Enterprise JavaBeans (EJB), Servlets, and JSP for building an enterprise-grade application.
- Deployed and managed application on WebSphere Application Server (WAS) for high availability, scalability, and performance.
- Executed SQL queries for relational database management systems (RDBMS) like MySQL, PostgreSQL, and Microsoft SQL Server.
- Gained exposure to NoSQL databases such as Yuga byte, Neo4j, and Cassandra for handling unstructured or semi-structured data, distributed storage, and high availability.
- Developed REST web services using annotations to extract client-related data from databases.
- Built **Restful API** with **Oauth** web services using **MongoDB** and **Apache Cassandra** to store and present assessments. Integrated **Elasticsearch** for real-time search and analysis of unstructured data.
- Written **SQL queries** and PL **SQL-stored procedures**, functions, triggers, cursors, sequences, and indexes.

- Automated CI/CD workflows using **GitLab pipelines** and **Docker**, enabling rapid and consistent deployment of microservices to **AWS ECS**.
- Provisioned infrastructure using **Terraform modules** for VPCs, subnets, and IAM roles across multi-account AWS environments, ensuring scalable and secure deployments.
- Integrated **Checkmarx** and **SonarQube** into Jenkins pipelines to enforce static code analysis and vulnerability scanning during build stages.
- Used **AWS CloudWatch** and custom metrics to create dashboards and automated alerts for vehicle transaction services, ensuring high availability and reliability.
- Utilized Golang's standard library and third-party packages to build robust web applications, command-line tools, and distributed systems.
- Integrated Golang applications with various databases and data stores, including SQL databases, NoSQL databases, and key-value stores, to efficiently manage and retrieve data.
- Implemented RESTful APIs and web services in Golang, adhering to industry best practices and standards for API design and documentation.
- Implemented data replication and disaster recovery strategies using **Snowflake's** multi-cluster, multi-region architecture to ensure high availability and data resilience.
- Deployed Spring Boot-based Microservices **Docker** containers using Amazon **EC2** container services and **AWS** admin console. Configured **AWS Simple Queue Service** (SQS) for interacting with queues.
- Used **GitHub** as a version control tool and **Maven** for building the application.
- Participated in troubleshooting issues on **AWS**, **Netflix Eureka**, and **Cloud** environments.
- Used **POSTMAN** to test API end clients using **REST** web services. Worked in **Agile**, **CMMI** Development Environment. Implemented **GraphQL** for efficient and flexible API queries.
- Participated in the daily **SCRUM** meeting and paired programming to produce quality delivery within time.
- Led the implementation of CMMI Level 3 processes, resulting in a 20% improvement in project delivery timelines and a 15% reduction in defects. Developed documentation compliant with CMMI Level 3.
- Leveraged **TCP/IP**, **SSO** and **OAuth2** knowledge for efficient networking solutions and JPA expertise for seamless database integration. Conducted performance tuning through **Jprofiler** and **VisualVM.**
- Adapted **GIT** as version control, **JIRA**, **Confluence** and **GitHub** for daily routines. Configured settings for development, Test, and Production Environment.

Environment: Java 11, Spring Boot, Microservices, HTML5, CSS3, AJAX, jQuery, Node JS, Hibernate, Bootstrap, Servlets, Express.Js, Angular 11, Django, Micronaut, Quarkus, Helidon, RDBMS, Node.JS, Vue.JS, JSF, JavaScript,.NET, XML, XSLT, WCS, JBoss, AWS Glue, Jprofiler, TensorFlow,D3.js, Kotlin, IBM WebSphere Application Server 8, Ratpack, WES, CXF, XPATH, Kubernetes, Jooby, Apache Flink, REST, (JAX-RS), TCP/IP, Docker, Typescript, Axon framework, WES, WMS, IBM WebSphere, JMS, RabbitMQ, Maven, AWS, Cassandra, Azure, Postman, JUNIT, Mockito, GIT, Jira, EMMA, IntelliJ.

United Health Group, Chicago, IL, USA **Aug 2020 – July 2023**

Java Full Stack Developer II

Responsibilities:

- Contributed towards (**SDLC**) Requirement Analysis, Design, Development, and Implementation of the application using **J2EE** technology.
- Revamped the application using the latest Java/J2EE technologies, working with the latest features of **Java 8**.
- Engineered a responsive UI which will scale itself depending on the device, platform and browser using, **HTML5**, **CSS3**, **Bootstrap**, **ReactJS**, **AJAX**, **TypeScript,** and **JavaScript**.
- Worked with Docker and Kubernetes to containerize and orchestrate Golang microservices, enabling seamless deployment and scaling in distributed environments.
- Designed **multithreaded trade-processing modules** using Java Executor Service and Callable/Future, cutting batch execution time by 35%.
- Built **Spring MVC applications** with Hibernate for ORM and MyBatis for advanced query mapping.
- Implemented **RESTful web services with JAX-RS**, integrating with external Treasury APIs and ensuring secure data transfer.

- Developed **Angular + JSP dashboards** for traders, enabling near real-time monitoring of fixed-income securities.
- Conducted code reviews and collaborated with team members to ensure code quality, performance optimization, and adherence to coding standards in Golang projects.
- Contributed to open-source projects in the Golang community, sharing knowledge and collaborating with other developers to enhance the ecosystem.
- Optimized **Hibernate** in Data Access Layer and Implemented Hibernate Template to a great extent making use of SQL, Named Parameters, Named Queries and interfaces such as **Expression**, Query and Criteria.
- Adapted new Java 8 features, such as **Lambda** expressions, **Streams**, and **Functional** Interfaces.
- Developed application using **Spring MVC**, Spring Batch, **JSP**, JSTL, and **AJAX** on the presentation layer, with the business layer built using Spring.
- Enhanced user interfaces using **React** and **JSX** to create reusable and interactive components. Utilized **React** JS components like Forms, Events, Keys, Router, **Redux**, and **Flux** concepts.
- Used **React.Js** to create reusable UI components as well as **Node.Js** and **Express.Js** to improve development effectiveness and maintain uniform design patterns throughout the application.
- Implemented Jest, a popular **JavaScript and Typescript** testing framework, to write unit integration, and snapshot tests for JavaScript and **React** applications.
- Integrated **JAX-RS** with Java/Scala, Java/Kotlin frameworks and technologies to develop comprehensive web applications and microservices architectures.
- Hardened applications against **OWASP vulnerabilities**, securing APIs with OAuth2 and CSRF protection.
- Leveraged **GitHub Copilot AI pair programming**, reducing repetitive coding time by 25%.
- Designed **unit and integration tests** with JUnit5, Mockito, and functional tests using Selenium.
- Designed **GitHub Actions-based CI/CD pipelines** for backend and Python services, automating builds, tests, and cloud deployments to **AWS Lambda** and **ECS**.
- Developed **Terraform modules** to manage AWS resources like **EC2**, **RDS**, and **S3**, with integrated Sentinel policies to enforce infrastructure compliance.
- Collaborated with security teams to manage **Vault secret injections** into containerized applications during pipeline executions, improving secrets rotation and security posture.
- Built centralized logging pipelines using **CloudWatch Logs**, **Kinesis**, and **Elasticsearch**, improving traceability and incident response for Treasury and Trading systems.
- Extensively used JavaScript and React, as Jest is commonly used in conjunction with these technologies for testing purposes.
- Deployed and hosted microservices on the **AWS** cloud platform utilizing **EC2** with **ECS** combination.
- Provisioned Amazon **EC2** instances and configured **AWS** services like **S3**, **IAM**, **RDS, SES**, **Route53**, **CloudWatch**, **CloudFront**, and **Elastic Load Balancers**.
- Integrated messaging systems such as **Kafka Flink** and **RabbitMQ** for **asynchronou**s communication and event-driven architecture.
- Designed and implemented messaging systems using **Apache Kafka**, showcasing expertise in building real-time streaming applications in Java.
- Implemented real-time data streaming and processing pipelines using **Snowflake's** integration with Kafka, Apache Flink and Apache Spark Streaming to ingest and analyze streaming data sources.
- Built event-driven architectures and streaming analytics applications to process, aggregate, and visualize real-time data in **Snowflake** data warehouses.
- Implemented Microservices using the **Pivotal Cloud Foundry** platform built upon Spring Boot Services.
- Developed **DAO** objects and implemented the data access layer using **JPA-Hibernate** to perform **CRUD** operations.
- Augmented Java 17 solutions at Goldman Sachs, leveraging the latest language features to enhance the performance and maintainability of mission-critical applications.
- Collaborated with Goldman Sachs' team to integrate Spring Cloud Netflix components for service discovery, load balancing, and fault tolerance in cloud-native microservices architectures.
- Developed **RESTful** APIs and web services using Java and Spring Framework for seamless integration with client applications and external systems at Goldman Sachs.
- Implemented queue-based architectures using messaging systems such as Kafka and **IBM MQ** for asynchronous communication and decoupled microservices at Goldman Sachs.

- Conducted in-depth analysis of fixed-income securities, including interest rate derivatives and credit default swaps, to assess risk and pricing dynamics for Goldman Sachs traders.
- Developed complex SQL queries and analytical functions in **Snowflake** to perform ad-hoc analysis, generate reports, and extract actionable insights from data. Leveraged Snowflake's performance optimization.
- Participated in **AGILE** (SCRUM) sprint planning every two weeks and set up daily meetings to monitor the status.
- Responsible for creating and maintaining automation acceptance test suites using **Kubernetes** and **Java 17**.
- Converted automation scripts to a new framework using **Selenium Web Driver, Java**, and **TestNG**.
- Used Entity Framework, ORM and Terraform for data access and database interactions in **.NET** applications.

Environment: Java 8, Spring, Spring Boot, Hibernate, HTML5, Angular 8, CSS3, JavaScript, Micro-services, JSP, JSF, JMS, ReactJS, AJAX, GraphQL, Camunda, AWS DevOps, Apache Cassandra, Kibana, NGNIX, NPM, JSON, XML, DAO, RDBMS, SOAP, Node.js, Java Beans, WES, Kotlin. Django, Terraform, Ansible, Python, Docker, Kubernetes, Postman, Snowflake, Azure, Microservices, Jenkins, Node.js, Maven, EJB, Log4j, JUnit, SAML, OAUTH2.

State of Texas, Austin, TX, USA **Aug 2018 – Nov 2020**
Java Full Stack Developer
Responsibilities:

- Contributed to various phases of the Software Development Life Cycle (**SDLC**), including Requirement gathering, Design, Analysis, and Code development.
- Engaged in production support activities, focusing on performance tuning, and initiating process improvements.
- Worked on the front end, business, and persistence tiers using **JSP**, **JavaScript**, Typescript and **Hibernate** frameworks.
- Developed and customized IBM Curam Social Program Management (SPM) applications using **Curam SPM 7.x**, **Java**, and **Curam model-driven development (MDE)** framework.
- Modified and extended **Curam evidence**, **case management**, and **application workflows** to meet specific client requirements.
- Designed and implemented custom pages, UIMs, and evidence entities using **Curam Development Environment (CDEJ)** and **Eclipse IDE**.
- Integrated third-party services using **Curam Service Layer (SL)**, **RESTful APIs**, and **web services (SOAP/WSDL)** for data exchange between systems.
- Developed web application for the backend system using **Node.JS** along with Express.js for building RESTful APIs, MongoDB for data storage, and JWT for authentication.
- Utilized **Angular 2/4** modules, services, templates, directives, and dependency injection to create a Single Page Application (SPA).
- Utilized a combination of **HTML5**, CSS3, and **JavaScript** frameworks including Node.js, Express.js, and Angular 2/4, alongside jQuery, to craft dynamic layouts within JSP files.
- Contributed to performance tuning activities, monitoring end-of-day loads, and assessing application behavior using tools like JProfiler.
- Developed Microservices and APIs using **Spring Cloud**, Spring Security, Spring Boot, and Spring Integration.
- Implemented UI and interactions with Spring Boot **REST API** Services, creating components, models, and services using Spring Boot.
- Integrated Snowflake with business intelligence (BI) and data visualization tools such as Tableau, Power BI, or Looker to create interactive dashboards, reports, and data visualizations.
- Developed business validation classes and configured connections to services, altering default configurations using Spring Boot.
- Integrated various **Azure** services such as **Azure Storage**, Azure **SQL** Database, **Azure Cosmos DB**, Azure Blob Storage, Azure Functions, and Azure App Service into cloud-based solutions.
- Leveraged **Azure Event Grid**, Azure Service Bus, and **Azure Event Hubs** for event-driven architectures and real-time data processing.
- Deployed Java EE applications on Websphere Application Server (WAS) for high availability and reliability at First Financial, ensuring uninterrupted operation of critical business systems.
- Designed and optimized SQL database schemas for First Financial, leveraging relational and NoSQL technologies such as Yugabyte, Neo4j, and Cassandra.
- Implemented Kafka-based messaging systems for real-time data processing and event-driven architectures at First Financial, enhancing scalability and responsiveness in financial trading environments.

- Developed **RESTful APIs** and web services using Java and Spring Framework for seamless integration with client applications and external systems at First Financial.
- Participated in database design, writing **SQL** Queries, and stored procedures in Oracle.
- Used version control systems like Git and collaboration platforms like GitHub and GitLab for managing codebase, tracking changes, and collaborating with team members.
- Utilized **GGTS** as a Java IDE tool for creating **XML** and **Groovy** (GSP).
- Deployed **WAR**, EAR, and JAR files in the **JBoss** application server.

Environment: Java, Java Script, Spring boot, Cassandra, IBM Curam, Vue.JS, Express JS, JAX-RS, Spring Security, Angular 2/4, Spring Cloud, Rest API, AWS Glue, Kotlin, Node.js, Typescript, WES, Azure App Service, Azure Functions, Azure SQL Database, Azure Storage, Azure Cosmos DB, WMS, WCS, snowflake, Azure, Oracle, OAuth, SAML, OiDC, Mongo DB, Jenkins, Maven, Git, Agile, JIRA.

Costco, Seattle, WA, USA **Jan 2017– July 2018**
Java/ J2EE Developer
Responsibilities:

- Contributed actively in an Agile development environment, spearheading responsiveness to dynamic requirements and clarifications.
- Developed prototypes & proofs of concept for proposed solutions participating in application demos with users.
- Employed front-end technologies such as HTML5, CSS 3, Typescript, **JavaScript**, Node.js, AJAX, JSP and **AngularJS**.
- Worked one-on-one with the client to develop a layout and color scheme and implemented it into a final interface design with HTML5/CSS3, **Bootstrap** and JavaScript.
- Implemented AJAX (Asynchronous JavaScript and XML) techniques to enable seamless communication between the frontend and backend layers of Java applications.
- Utilized AngularJS framework to bind **HTML** template (views) to JavaScript object (models).
- Built AngularJS framework including **MVC** architectures, different modules, specific controllers, templates, custom directives, and custom filters.
- Implemented **RESTful** web services using Spring REST, facilitating communication between different components of the application and enabling interoperability with other systems.
- Designed and optimized relational databases using **MySQL** and **Oracle**, ensuring efficient storage and retrieval of application data.
- Developed and deployed **RESTful** APIs on the Azure cloud platform, utilizing Azure Web Apps, Azure Functions, and Azure API Gateway, to ensure seamless integration and communication across the system.
- Implemented **RESTful** API endpoints following best practices and industry standards, adhering to REST architectural principles such as statelessness, uniform interface, and layered system, to promote scalability and flexibility.
- Utilized ORM frameworks such as **Hibernate** and MyBatis to map Java objects to relational database tables, simplifying data access and persistence operations.
- Integrated Java applications with Azure IoT Hub and Azure IoT Edge for IoT device management and telemetry data processing,
- Developed **Azure** Functions and Azure Logic Apps to automate business processes and workflows.
- Employed Azure Key Vault for encryption and key management, addressing data security and regulatory requirements.
- Implemented **auto-scaling** policies using Azure Autoscale, dynamically adjusting resources based on workload demands, optimizing cost-efficiency and scalability for Java applications on Azure Cloud.
- Used **Node.js**, **AngularJS and Express.JS** frameworks where data from the backend is stored in the appropriate model.
- Designed, Developed, and analyzed the front-end and back-end using **JSP**, Servlets and **Spring**.
- Used Spring framework, **Spring-ORM**, and Spring-JDBC modules.
- Involved in developing business components using **POJO** and **Camunda**.
- Utilized Spring **Inversion of Control (IOC)**, Setter Injection, **Constructor Injection** and Dependency Injection.
- Implemented database transactions and transaction management along with Spring **ORM** and **DAO**.
- Used **Log4J** to print logging, debugging, warning, and info on the server console.

Environment: Java, Spring, Kotlin, Hibernate, Java Script, Typescript, Node.js, jQuery, Ajax, CSS, Bootstrap, AJAX, HTML5, DB2, CRUD, SQL, JDBC, Apache CXF, REST (Rest API's), Web Services, WES, Python, WCS, WMS, Camunda, Eclipse, Subversion (SVN), SQL Server, JUNIT, Scrum, UML, JSP, Java Server Pages Standard Tag Library (JSTL), SAML, OiDC, Windows, Agile Methodology.

American Express, Phoenix, AZ, USA **Feb 2014 – Dec 2016**

Java Developer

Responsibilities:

- Actively collaborated with the UI/UX team to implement responsive and user-friendly interfaces using HTML, CSS, and JavaScript.
- Designed UI using **HTML**, **DHTML**, Java Script, and **CSS**.
- Developed interactive user interfaces using **AngularJS** (version 1. x) to enhance client-side functionalities, providing seamless navigation and dynamic content rendering.
- Implemented robust server-side solutions leveraging the Spring Framework (version 3. x) with Java EE, Hibernate (version 4. x), and **Apache Tomcat** (version 7. x).
- Used Java script for client-side validations. Developed the middle tier using EJBs, Java Servlets, session, entity beans and message-driven beans.
- Developed SQL queries to store and retrieve data from the database & used **PL/SQL**.
- Used **Struts** framework to maintain MVC and created action forms, action mappings, DAOs, application properties for Internationalization etc.
- Designed and developed applications utilizing Spring (version 3. x), **JSP**, **HTML**, and JavaScript on Azure App Service.
- Developed Microservices using Spring Boot and followed domain-driven design.
- Utilized multi-tiered JEE design patterns, incorporating **Spring MVC** and Hibernate for efficient data processing and management on **Azure SQL Database**.
- Configured Spring framework files to meet application requirements and developed web services for non-Java clients to access policy information on **Azure**.
- Developed build and deployment scripts using Apache ANT to customize WAR and EAR files for deployment on **Azure Kubernetes Service**.
- Designed and optimized **SQL** queries, stored procedures, and database schemas for **MySQL** and PostgreSQL databases to support backend functionalities and enhance data retrieval efficiency.
- Implemented security features such as **OAuth 2.0** and JWT (JSON Web Tokens) for authentication and authorization, ensuring secure communication and access control in backend services.
- Leveraged **Apache Maven** and Gradle as build automation tools to manage project dependencies, streamline the build process, and automate deployment tasks.
- Implemented caching mechanisms using tools like **Redis** and **Ehcache** to improve application performance and reduce database load.
- Integrated logging and monitoring solutions such as **ELK** (Elasticsearch, Logstash, Kibana) stack for centralized log management and real-time application monitoring.
- Used **Struts** Validation framework to do business validation from server side.
- Involved in developing business components using **EJB** Session Beans and persistence using EJB Entity beans.
- Used JMS for Asynchronous messaging and developed the application using **Eclipse IDE**.
- Developed customized reports and Unit Testing using JUnit and Mockito.
- Involved in fixing IST, **QA**, UAT & Production defects. Collaborated with the DevOps team to deploy the application on Apache Tomcat (version 7. x), adhering to industry best practices.
- Utilized **Maven** for management and build and SVN, and **GitHub** as version control tools. Involved in bug fixing and new enhancements.

Environment: Core Java, J2EE, Python, JSP, WES, WCS, WMS, JBoss, Camunda, Snowflake, Azure, Oracle, OAuth, SAML, OiDC, Eclipse, EJB-Session Beans, Entity Beans, JMS, XML, HTML, JavaScript, CSS, AJAX, SVN, Struts, XML, QC.