**Adithya C**A blue and white logo with white text

Description automatically generated

**Senior Java Full Stack Developer**

[**adithyajavadev02@gmail.com**](mailto:adithyajavadev02@gmail.com)

**551-295-1264**

**Professional Summary**

**Senior Java Full Stack Developer** with **13+ years** of experience delivering scalable, high-performance enterprise solutions across backend, frontend and cloud platforms and skilled in designing microservices, integrating APIs and driving agile development with a strong focus on security, performance optimization and clean architecture.

* Proficient in managing the **Software Development Life Cycle (SDLC)** using **Agile** and **Kanban** methodologies, ensuring adaptive planning, continuous delivery, and collaborative software development practices.
* Expert in **Java (v6–v19)**, leveraging **polymorphism**, **encapsulation**, functional constructs, and collections (lists, maps, sets, streams, lambdas) for performance and scalability across enterprise applications.
* Proficient in front-end development with **JavaScript**, **TypeScript**, **HTML**, **CSS**, **Bootstrap**, and **AJAX**, building WCAG-compliant interfaces in **Angular**, **React.js**, and **CrownPeak** for accessible, interactive user experiences.
* Expert in the **Spring Framework** ecosystem including **Spring Boot**, **Spring Security**, **Spring MVC**, **Spring JMS**, and **Spring Batch**, along with **JPA**, **Hibernate**, and **Spring JDBC** for robust service-layer logic and secure persistence.
* Skilled in designing and integrating **microservices architectures**, focused on **RESTful** and **SOAP APIs** with **JSON/XML**, enabling seamless front-to-backend communication and application scalability.
* Proficient in backend development using **Node.js**, applying event-driven and asynchronous programming models with **REST APIs** and **Express.js**, tailored for modular, high-performance backend services.
* Experienced in deploying applications to **AWS**, **Azure**, and **GCP**, utilizing **Terraform**, **EC2**, **VPC**, **Lambda**, **IAM**, **S3**, **ECS**, **Azure VMs**, **Active Directory**, and **AKS** for fault-tolerant, cloud-native solutions.
* Skilled in CI/CD automation with **Jenkins**, **Bamboo**, and **GitLab CI/CD**, containerizing with **Docker**, deploying via **Kubernetes**, and ensuring efficient rollouts using blue-green and rolling strategies.
* Proficient in static and dynamic code analysis with **SonarQube**, **JaCoCo**, and **Checkstyle**, integrating with build tools like **Maven**, **Gradle**, **Ant**, and **Terraform** for consistent and automated project builds.
* Experienced with IDEs including **Eclipse**, **NetBeans**, **IntelliJ IDEA**, and **JDeveloper**, utilizing code navigation, refactoring, debugging, and plugin support to accelerate development workflows.
* Skilled in integrating enterprise messaging systems using **Apache Kafka**, **RabbitMQ**, **ActiveMQ**, and **Spring JMS**, supporting distributed systems and asynchronous processing pipelines.
* Proficient in configuring and tuning application servers such as **JBoss**, **Apache Tomcat**, **WebLogic**, and **WebSphere**, optimizing resource allocation, connection pooling, and deployment pipelines.
* Expertise in managing relational databases including **PostgreSQL**, **Oracle Database**, and **MS SQL Server**, as well as NoSQL systems like **MongoDB** and **Cassandra**, with advanced **SQL** tuning and schema design.
* Well-versed in version control systems such as **GitLab**, **Bitbucket**, **GitHub**, **Mercurial**, and **SVN**, ensuring branching strategies, code integrity, and streamlined team collaboration.
* Experienced in issue tracking using **JIRA**, **GitLab Issues**, **Redmine**, and **Bugzilla**, establishing traceability, prioritization, and sprint alignment for feature requests, defects, and epics.
* Mastery in testing strategies with tools like **JUnit**, **Mockito**, **MockMvc**, **Selenium**, **Apache JMeter**, **Karma**, and **Jasmine**, ensuring full-stack coverage for quality and regression control.
* Expertise in logging and diagnostics using **Log4j**, **SLF4J**, **Logback**, and **java.util.logging (JUL)**, supporting centralized logging, filtering, and alert integration for proactive monitoring.
* Proficient in producing and maintaining high-quality documentation using **Confluence**, **GitLab Wiki**, **Bitbucket Wiki**, and **SharePoint**, ensuring consistent cross-team knowledge sharing and traceability.
* Applied **WCAG2.0** accessibility guidelines to build inclusive, standards-compliant UIs in **React.js**, validating ARIA roles, keyboard navigation, contrast ratios, and semantic structure for ADA readiness.
* Successfully delivered full-stack applications adhering to modern architecture principles, combining **Spring Boot**, **React.js**, **Node.js**, and **cloud services**, with reusable patterns supporting maintainability and team scalability.

**Certifications**

* AWS Certified Developer Associate.
* Microsoft Certified Azure Developer Associate.

**Educational Background**

* Bachelor's degree in computer science.

**Employment Timeline**

**Client: Truist Bank, Charlotte, NC Jan 2023 - Present**

**Senior Java Full Stack Developer**

**Responsibilities:**

* Directed end-to-end software delivery in **Agile/Scrum** setups for financial systems, facilitating sprint ceremonies, backlog refinement, and incremental rollout of secure, **Java**-driven financial platforms.
* Built high-efficiency backend systems in **Java 19**, using **record patterns**, **virtual threads (Project Loom)**, and **pattern matching for switch**, improving concurrency and scalability in compliance-heavy fintech environments.
* Architected and deployed **cloud-ready microservices** using **Spring Boot**, hosting modular and secure APIs on **AWS EC2** to support trading, payments, and investment service workflows.
* Secured **RESTful financial APIs** using **Spring Security**, implementing **OAuth2** and **JWT** for identity management, with endpoint protection via **AWS API Gateway** and network-level authorization.
* Developed high-throughput batch processing with **Spring Batch**, powering data pipelines for reconciliations, settlements, and compliance-driven reporting in **cloud-native** deployments.
* Optimized persistence using **Spring Data JPA**, **Hibernate**, and native **SQL**, with advanced **Criteria API** queries and **EntityManagerFactory** tuning for minimal latency and stable throughput.
* Built event-driven backend workflows with **Node.js**, designing real-time microservices for banking ecosystems using asynchronous messaging and distributed communication patterns.
* Delivered **accessible**, responsive UIs using **HTML5**, **CSS3**, **Bootstrap**, and **TypeScript**, aligning with **WCAG2.0** standards for inclusive design and consistent multi-device support.
* Engineered modular, enterprise-grade frontends using **Angular 14** and **React.js**, managing application state via **NgRx**, while enhancing performance through **Webpack** optimizations and lazy loading.
* Designed and exposed **secure REST APIs** with **JSON-based token authentication**, integrating observability with **AWS Lambda** and **CloudWatch** to monitor usage and detect anomalies.
* Automated infrastructure provisioning with **Terraform**, deploying **AWS EC2**, **AWS S3**, and **AWS Lambda**, scaling backend services while hosting static assets with built-in fault tolerance.
* Containerized full-stack systems using **Docker**, orchestrated via **Kubernetes**, with production-grade workloads managed on **AWS EKS** for automatic scaling and recovery.
* Established **CI/CD** automation pipelines using **Jenkins**, reducing deployment times through automated builds, tests, and version-controlled rollouts with rollback capability.
* Boosted developer efficiency using **IntelliJ IDEA**, employing smart code completions, deep Git/Bitbucket integration, and powerful debugging across the **Java** and frontend stacks.
* Maintained code quality using **SonarQube**, performing continuous inspection for code smells, vulnerabilities, and duplication across microservices and frontend modules.
* Managed multi-module builds using **Gradle**, standardizing dependency management across **Spring Boot** backends and **Angular/React** UIs for full-stack project consistency.
* Designed document-based schemas in **MongoDB**, indexing for high-volume reads and tuning aggregation pipelines in cloud-hosted applications for financial document processing.
* Implemented **event-streaming architectures** using **Apache Kafka**, deployed microservices on **Apache Tomcat**, and enforced collaboration through **Bitbucket** code reviews.
* Managed logging and issue tracking with **Log4j** and **Redmine**, ensuring full traceability for error detection, team ownership, and release-based triage strategies.
* Conducted automated testing using **JUnit** for backend and **Karma** for frontend coverage, maintaining documentation and component design specs in **Bitbucket Wiki** for knowledge continuity.

**Client: Centene Corporation, St. Louis, MO Jan 2021 - Dec 2022**

**Senior Java Full Stack Developer**

**Responsibilities:**

* Oversaw end-to-end software development for healthcare applications using **Agile** methodologies, coordinating sprint planning, execution, and delivery tracking via **Trac** to foster transparency and ensure timely project outcomes.
* Engineered scalable backend solutions for healthcare systems using **Java 14** and **Java 11**, applying **switch expressions**, **records**, **text blocks**, and improved **NullPointerExceptions** for maintainable, readable codebases.
* Developed secure microservices using **Spring Boot**, exposing **RESTful APIs** and integrating with **Kong API Gateway** for request routing, traffic control, and token-based authentication.
* Enabled reliable service communication through **Spring Cloud**, configuring distributed properties, client-side load balancing, and circuit breaker logic to ensure resilient microservices in production.
* Constructed optimized persistence layers using **Spring JDBC**, **JdbcTemplate**, **NamedParameterJdbcTemplate**, and **Hibernate ORM**, managing **EHR data** in **Amazon DynamoDB** and **Amazon RDS** with efficient query execution.
* Built and consumed standardized **JSON REST APIs**, streamlining integration between microservices, third-party systems, and modern **React.js** and **Angular** frontends for a cohesive client experience.
* Developed responsive, accessible user interfaces using **Angular 7**, **React.js**, **TypeScript**, **HTML5**, **CSS3**, and **Bootstrap**, adhering to **WCAG2.0** guidelines to ensure inclusive, standards-compliant healthcare platforms.
* Deployed healthcare applications using **AWS Elastic Beanstalk** and **Amazon EC2** for monolith workloads, and implemented lightweight, serverless APIs via **AWS Lambda** for scalable service execution.
* Refactored legacy systems into independent microservices, containerized with **Docker** and orchestrated using **Amazon EKS**, improving maintainability, availability, and infrastructure elasticity.
* Automated release pipelines using **GitLab Runner**, achieving consistent, repeatable CI/CD workflows across all environments and reducing manual deployment overhead significantly.
* Orchestrated collaborative version control workflows in **GitLab**, managing branches, merge requests, and PR reviews to enforce development best practices and code consistency across teams.
* Architected asynchronous messaging pipelines with **RabbitMQ**, decoupling services and increasing overall platform resilience in distributed microservice deployments for healthcare platforms.
* Deployed backend components on **WildFly Application Server**, tuning configurations and runtime profiles to meet SLA targets and ensure reliable, high-throughput production operation.
* Maintained clean code practices using **Checkstyle**, standardized builds with **Bazel**, and followed development best practices in **Visual Studio Code** for efficient implementation and debugging.
* Implemented robust testing coverage using **TestNG** for service-layer verification and **Jasmine** for Angular and **React.js** component testing, ensuring reliability and UI consistency through automation.
* Enabled real-time observability with **Logback**, streaming structured logs to Kubernetes-native dashboards and integrating with monitoring stacks to support system health checks and incident tracing.
* Authored technical documentation in **GitLab Wiki**, maintaining architectural diagrams, onboarding guides, CI/CD blueprints, and interface documentation to support internal development knowledge continuity.

**Client: Hallmark Financial Services, Inc., Dallas, TX Feb 2018 - Dec 2020**

**Java Full Stack Developer**

**Responsibilities:**

* Led the end-to-end lifecycle of financial enterprise software development, applying **Agile (Scrum)** and **SDLC** best practices to support iterative releases and accelerate delivery of mission-critical financial systems.
* Developed robust backend modules for financial platforms using **Java 7**, improving throughput with enhancements like String in Switch, multi-catch exception handling and the G1 Garbage Collector to handle high-volume processing.
* Designed and launched scalable microservices using **Spring Boot**, employing clean modular architecture and **RESTful APIs** to streamline functionalities such as payments, account operations and financial reporting.
* Integrated monitoring and diagnostics tools via **Spring Boot Actuator**, enabling real-time system health checks, performance metrics and operational insights to support application availability and uptime.
* Enhanced non-blocking communication between microservices using **Spring WebClient**, enabling real-time transaction processing and dynamic data synchronization across financial systems.
* Developed responsive, cross-platform UI components using **HTML5**, **CSS3**, **JavaScript** and**Bootstrap**, delivering intuitive dashboards tailored for financial advisors, clients and internal operations.
* Built modular frontend applications with **React 16**, applying a component-based architecture within a Service-Oriented Architecture (SOA) environment and leveraging Babel to ensure browser compatibility and efficient transpilation.
* Engineered and maintained secure **RESTful APIs** with **JSON** payloads, utilizing **NGINX API Gateway** for advanced routing, rate limiting and access control across microservices.
* Deployed cloud-based application workloads on **Microsoft Azure**, utilizing services including **Azure VMs**, **Azure Functions**, **App Services** and containerized applications through **Azure Kubernetes Service (AKS)** for scalability and flexibility.
* Established end-to-end **CI/CD pipelines** using **Bamboo** with custom runners, automating build, test and deployment workflows to ensure rapid and error-free software delivery across multiple environments.
* Containerized enterprise applications with **Docker** and orchestrated deployment through **Kubernetes**, ensuring seamless scaling, high availability and uniform runtime environments.
* Enforced static code quality with **SonarQube** to detect bugs, vulnerabilities and maintain clean, reliable code; managed builds and dependencies using **Apache Maven** for consistent, reproducible development and CI/CD workflows.
* Designed and optimized database structures in **Oracle**, including stored procedures and custom data models; implemented asynchronous, event-driven messaging with **Apache Pulsar** to enable real-time streaming and decoupled service communication.
* Oversaw production deployments on **GlassFish Server**, managed source control and release workflows through **Azure DevOps** and tracked defect resolution and change history using **Bugzilla** for full traceability.
* Executed comprehensive testing strategies, utilizing **Mockito** for backend unit tests, Jest for React component validation, maintaining knowledge assets via **Confluence** and incorporating **SLF4J** for consistent, centralized logging.

**Client: Flagstar Bank, Wallington, NJ July 2015 - Jan 2018**

**Java Developer**

**Responsibilities:**

* Oversaw complete software development lifecycles for financial applications, working closely with cross-disciplinary **Agile teams** to ensure incremental delivery and continuous integration of enterprise-grade platforms.
* Engineered backend systems in **Java 6**, leveraging the Scripting API, Pluggable Annotation Processing, JAXB 2.0 and compiler tools for extensibility, XML handling and dynamic scripting support.
* Designed distributed microservices using **Spring Boot**, following modular principles and deploying to **AWS EC2** instances to support scalable transaction flows and financial workloads.
* Built robust data interaction layers using **Spring JDBC**, enabling secure and efficient access to transactional records with long-term storage of financial data in **AWS S3**.
* Crafted responsive and engaging UIs using **HTML5**, **CSS3**, **JavaScript** and**Bootstrap**, optimizing user experience across web-based financial systems and client dashboards.
* Developed dynamic frontend layers with **AngularJS** within an event-driven system, where backend logic was invoked using **AWS Lambda** to support serverless and responsive processing pipelines.
* Connected legacy and modern services using **SOAP APIs** and **XML**, enforcing secure access with **AWS IAM** and isolating service traffic via **VPC** configurations for regulatory and network control.
* Automated infrastructure provisioning with **Terraform**, applying **IaC** principles to standardize cloud resource deployment across multi-environment **AWS** landscapes.
* Implemented automated **CI/CD** workflows using **GitLab CI/CD** and custom runners, accelerating software delivery and reducing manual overhead in deployment pipelines targeting **AWS** infrastructure.
* Containerized microservices with **Docker** and managed lifecycle operations using **Kubernetes** on **AWS**, supporting consistent scaling and resilience across dev, test and prod environments.
* Developed and troubleshooted **Java** codebases in **NetBeans IDE**, utilizing integrated debugging tools and version control capabilities to streamline the development process.
* Enforced static code quality standards via **PMD** and executed repeatable, environment-independent builds using **Gradle** to ensure high code maintainability and test reliability.
* Designed and tuned relational schemas in **PostgreSQL**, while implementing message-based service coordination using **RabbitMQ** for loosely coupled, asynchronous system behavior.
* Handled application deployment and management on **GlassFish Server**, ensured source control using **Mercurial** and maintained transparency in team operations through issue tracking in **Jira**.
* Wrote and maintained test cases using **Mockito** for backend logic, configured structured log output with **Logback** and documented technical solutions and procedures within **SharePoint** for easy reference and knowledge transfer.

**Client: Mercury Insurance, Brea, CA Nov 2012 - June 2015**

**Java Developer**

**Responsibilities:**

* Worked alongside cross-functional teams in an **Agile** environment, contributing to sprint planning, iterative development and the delivery of insurance software through continuous stakeholder collaboration and feedback cycles.
* Built scalable backend modules for insurance applications using **Java**, applying core OOP principles such as Encapsulation and Polymorphism to support business logic for claims processing, policy issuance and underwriting.
* Developed enterprise-level insurance solutions using **Spring MVC** for structured request routing and response management and**Spring Integration** to enable smooth system interactions across modules like billing and policy management.
* Created dynamic user interfaces using **HTML5**, **CSS3**, **JavaScript** and**Bootstrap** and enabled interactive behaviors through **Ajax** and **jQuery** for DOM manipulation, asynchronous data exchange and UI responsiveness.
* Implemented secure web service communication using **SOAP APIs** and **XML**, ensuring reliable data exchange between internal platforms and third-party systems in line with enterprise integration standards.
* Modeled relational databases using **MySQL**, writing optimized schemas, stored procedures andSQL queries; streamlined builds and deployments with **Apache Ant** for consistent packaging and setup.
* Maintained version control using **Apache Subversion (SVN)** for collaborative development and revision tracking, while managing task workflows and bug resolution through **Redmine** to ensure project alignment and timely delivery.
* Created and ran automated acceptance tests using **Cucumber**, enabling **Behavior-Driven Development (BDD)** and improving software quality through readable, testable feature definitions.