**Yashwanth**

 Sr Java Full Stack Developer

 Email: meyash342@gmail.com

 Phone: +1 (314) 375-6660

Java 17/11/8/1.5 | C | C++ | Golang | Python | Groovy | SQL | PL/SQL | ReactJS | Redux | Vue.js 2 | Angular 14/8/JS | Bootstrap | JavaScript (ES6+) | TypeScript | Node.js | JSP | Servlets | XML | JSON | XSD | XSL/XSLT | SAX/DOM | DOJO | Swift UI | Spring Boot | Spring MVC | Spring Security | Spring Data JPA | Spring Cloud | Hibernate | Micronaut | Struts | Apache Camel | RESTful APIs | SOAP | WSDL | JAX-RS | JAX-WS | Apache CXF | Apache Axis | Kafka | RabbitMQ | ActiveMQ | GraphQL | WebSockets | OAuth2.0 | JWT | SSO | RBAC | AWS (EC2, Lambda, EKS, ECS, S3, RDS, DynamoDB, API Gateway, CloudFront, Route 53, Cognito, SNS, SQS, KMS, Redshift) | Azure (Event Hubs, Data Factory, Cosmos DB, Databricks) | GCP (Compute Engine, App Engine, Cloud Run, Cloud Storage, BigQuery, Cloud SQL) | Terraform | CloudFormation | Kubernetes (EKS, AKS, GKE, OpenShift) | Docker | Jenkins | GitHub Actions | GitLab CI | Maven | Gradle | ANT | Nexus | SonarQube | Veracode | Snyk | Cypress | Selenium | Cucumber | JUnit | Mockito | Jest | PySpark | Apache Airflow | Scala | Snowflake | MongoDB | Couchbase | Cassandra | Redis | PostgreSQL | MySQL | Oracle | SQL Server | IBM DB2 | ELK Stack | Splunk | Prometheus | Grafana | AWS CloudWatch | GCP Stackdriver | Tortoise SVN | CVS | Git | GitHub | GitLab | Spring Tool Suite | VSCode | Eclipse | Visual Studio | JIRA | Rally | Confluence | Agile/Scrum | Waterfall

Senior Full Stack Java Developer with 11+ years of experience in architecting, developing, and deploying enterprise-grade applications and services across cloud and on-premise platforms. Proficient in Microservices architecture, cloud computing (AWS, Azure, GCP), containerization, and modern front-end frameworks like React, Vue.js, and Angular. Deep expertise in API development, event-driven systems, and cloud-native application deployment using Docker and Kubernetes. Skilled in building secure, scalable, and high-performance systems, with a focus on DevSecOps, CI/CD automation, observability, and infrastructure as code.

**PROFESSIONAL SUMMARY:**

* Expertise in **UI design and development** using **Angular**, **React JS**, **Bootstrap**, **JavaScript**, **TypeScript**, **JSON**, **XML**, **Ajax**, **jQuery**, **CSS/CSS3** (SASS, LESS), **HTML/HTML5**, and **Swift UI** for responsive web interfaces.
* Proven expertise in building secure, scalable **Microservices** using **Spring Boot**, **Spring Cloud**, **Spring Security**, **Spring Data JPA**, **Spring MVC**, and **RESTful APIs** for distributed cloud-native architectures.
* Deep understanding of **Java 17** features like **Lambda expressions**, **Streams**, **Records**, **Pattern Matching**, and **Sealed Classes**, optimizing code for performance and maintainability.
* Skilled in **API security** using **OAuth 2.0**, **JWT**, **Single Sign-On (SSO)**, and **AWS Cognito** for secure authentication and authorization across distributed systems.
* Proficient in developing **GraphQL APIs** with efficient schema design and resolvers for flexible, real-time data access on web and mobile platforms.
* Experienced in building **event-driven, real-time applications** using **Apache Kafka**, **RabbitMQ**, and **ActiveMQ** to improve system responsiveness and decoupling.
* Strong hands-on expertise in **containerization with Docker** and orchestration using **Kubernetes** (**AWS EKS**, **Azure AKS**, **GCP GKE**, **OpenShift**) for scalable deployments.
* Skilled in **multi-cloud deployments** across **AWS** (**EC2**, **Lambda**, **EKS**, **ECS**, **S3**, **RDS**, **DynamoDB**, **API Gateway**, **CloudFront**, **Route 53**, **Cognito**, **SNS**, **SQS**), **Azure** (**Data Factory**, **Event Hubs**, **Storage**, **Cosmos DB**, **Databricks**), and **GCP** (**Compute Engine**, **Cloud Run**, **App Engine**, **Cloud Storage**, **Cloud SQL**, **IAM**).
* Proficient in **Infrastructure as Code (IaC)** using **Terraform** and **AWS CloudFormation** for automated, consistent, and secure cloud provisioning across **AWS**, **Azure**, and **GCP**.
* Experienced in designing **serverless solutions** using **AWS Lambda**, **GCP Cloud Functions**, and **Azure Functions** for cost-effective, low-maintenance architectures.
* Strong **database expertise** in **Oracle**, **MySQL**, **PostgreSQL**, **SQL Server**, **MongoDB**, **DynamoDB**, **Cassandra**, and **Redis** for structured and unstructured data management.
* Proficient in **backend services development** using **Golang** and **Micronaut** for high-performance, low-latency microservices supporting critical business functions.
* Skilled in **monitoring and observability** using **Prometheus**, **Grafana**, **AWS CloudWatch**, **GCP Operations Suite**, **ELK Stack**, and **Splunk** for real-time system health tracking.
* Hands-on in **CI/CD pipeline automation** using **Jenkins**, **GitHub Actions**, **Maven**, **Gradle**, and **Docker**, integrating **security** and **quality checks** into deployment workflows.
* Extensive experience in **Agile methodologies** (**Scrum**, **Kanban**), using **JIRA**, **Rally**, and **Confluence** to drive iterative development, collaboration, and delivery.

 **TECHNICAL SKILLS:**

|  |  |
| --- | --- |
| **Languages** | C, C++, Java 17.0/11.0/8.0/1.5, groovy, Python, SQL and PL/SQL |
| **Mark-up/XML Technologies** | HTML5, CSS3, Bootstrap, JavaScript, jQuery, AngularJS, React JS, NodeJS, XML, XSD, XSL/XSLT, SAX/DOM, DOJO and JSON. |
| **Tools & Framework** | Struts, Spring, Hibernate, Spring MVC, Spring Web Flow, Spring IOC, AOP, Security, Spring Boot, JUnit, Apache CXF, SOAP UI, Google Web Toolkit, Jersey. |
| **Build Automation** | ANT, MAVEN, Jenkins, Gradle. |
| **Web services** | SOAP, RESTful, WSDL, JAX-RPC, JAX-RS, JAX-WS, JAX-B, Microservices, Apache Axis and Apache CFX. |
| **Cloud Technologies** | Amazon Web Services (EC2, SQS, RDS, IAM, S3), PCF and Spring Cloud, Azure, GCP. |
| **Web/App Servers** | WebSphere, Apache Tomcat, JBOSS EAP, WebLogic Server and Apache Camel |
| **Databases** | Oracle, SQL-Server, IBM DB2, NoSQL DB like MongoDB and Couch base, Cassandra |
| **IDE tools** | Eclipse IDE, Net Beans, Spring Tool Suite (STS) and IntelliJ IDEA, Visual Studio. |
| **Testing Tools/ Others** | Selenium, JUnit, Mockito, Cypress, Cucumber, Jasmine, Karma |
| **Version Control** | Tortoise SVN, CVS and GIT, GitHub, GitLab |
| **OS & Environment** | Windows, Win 2k Server, Sun Solaris, UNIX |
| **Design Patterns** | Singleton, Factory, MVC, Front Controller and DAO pattern. |

 **PROFESSIONAL EXPERIENCE:**

**Client: Charter, Texas July 2023 - Present**

**Role: Sr. Java Full Stack Developer**

**Responsibilities:**

* Collaborated extensively with product owners, business analysts, and development teams to analyze business requirements, decompose epics into detailed user stories, and participate in sprint planning, ensuring alignment with customer service objectives for Charter’s large-scale media and telecom platforms.
* Designed and developed comprehensive Single Page Applications (SPAs) using Angular 14, TypeScript, JavaScript, HTML5, CSS3, and Material UI, focusing on creating reusable components, improving UI consistency, and ensuring a seamless customer experience across Charter’s web and mobile platforms.
* Implemented advanced Angular features such as dependency injection, lazy loading, differential loading, and reactive forms, enhancing application performance, reducing load times, and ensuring maintainability of complex front-end modules.
* Built unit test suites for Angular components, services, and directives using Jasmine and Karma, achieving high code coverage and ensuring code reliability for Charter’s customer self-service portals and internal tools.
* Led the development of scalable, distributed Microservices architecture using Spring Boot, Spring Cloud, and Spring Security, supporting Charter’s high-volume data streaming, billing, and customer management services with fault-tolerant backend processing.
* Utilized advanced Java 17 features such as sealed classes, pattern matching, records, and functional streams to improve backend service logic readability, reduce boilerplate code, and enforce secure programming patterns across Charter’s core enterprise systems.
* Developed RESTful web services with Spring MVC, Hibernate ORM, and Spring Data JPA to expose APIs for real-time billing, subscription management, and network provisioning, ensuring seamless integration between front-end and back-end systems.
* Integrated secure token-based authentication using OAuth 2.0, JWT, and role-based access control to enforce security policies across Charter’s APIs, achieving strict compliance with media and telecommunications security regulations.
* Built high-performance user interfaces with React 17, Redux, and TypeScript to optimize the customer experience for Charter’s billing dashboards and service status portals, reducing page load times and improving application interactivity.
* Enabled real-time communication in React applications using WebSockets and Apache Kafka, allowing live service status updates, event notifications, and streaming data to be displayed dynamically in customer dashboards.
* Migrated legacy Java EE monolithic applications into scalable, containerized Microservices, deploying them onto Kubernetes clusters using AWS EKS and Docker, achieving faster deployment cycles, reduced infrastructure costs, and improved application scalability.
* Automated infrastructure provisioning using Terraform and AWS CloudFormation to deploy and manage EC2, ECS, Lambda, API Gateway, Route 53, and CloudFront services, creating repeatable, version-controlled cloud infrastructure pipelines.
* Integrated centralized logging and monitoring solutions using Splunk and ELK Stack (Elasticsearch, Logstash, Kibana), enabling real-time observability, incident detection, and reducing mean time to resolution (MTTR) for production issues.
* Configured AWS CloudWatch, SNS, and SQS for distributed system monitoring, setting up alerts and event-driven workflows to ensure service availability and streamline incident response for Charter’s streaming platforms.
* Automated CI/CD pipelines using Jenkins and Python scripting, incorporating SonarQube for static code analysis, container image security scanning, and infrastructure validation to ensure secure, consistent deployments.
* Created custom Python-based utilities for API testing, mocking, and automation to validate third-party vendor integrations and ensure reliable communication between internal microservices and external systems.
* Deployed microservices to AWS ECS, EKS, and AWS Lambda environments, balancing cost optimization with performance requirements to handle variable workloads in production environments.
* Utilized Cassandra, DynamoDB, and Redis NoSQL databases to support real-time data processing and caching, ensuring low-latency access to customer profiles, billing data, and service status updates for millions of transactions daily.
* Enhanced system resiliency and service discovery using Netflix OSS tools including Eureka for registration and discovery, Hystrix for circuit breaking, and Spring Cloud Config for centralized configuration management in the microservices ecosystem.
* Developed internal admin portals using Vue.js 2, TypeScript, and reusable component libraries to provide support teams with tools for managing customer accounts, provisioning services, and handling feature toggles with role-based permissions.
* Built Node.js-based backend services to handle asynchronous operations such as notification delivery and data synchronization with NoSQL databases, improving system performance for real-time transaction processing.
* Designed and implemented GraphQL resolvers and schemas to enable flexible data retrieval, reducing over-fetching and improving efficiency for mobile and web client data queries in customer portals.
* Configured RabbitMQ clusters to enable reliable message brokering between microservices, ensuring message durability, high throughput, and transactional integrity for Charter’s real-time operations.
* Managed complex data pipelines using Apache Airflow and AWS Data Services, orchestrating ETL processes to process telemetry data, user interactions, and system metrics for analytics and reporting across Charter’s network infrastructure.

**Environment:** Java 17, Spring Boot, Spring Cloud, Spring Security, Microservices, REST, OAuth 2.0, JWT, Kubernetes, Docker, Terraform, AWS (EC2, ECS, EKS, Lambda, S3, API Gateway, CloudFront, RDS, DynamoDB, Route 53), Azure (Data Factory, Event Hubs, Cosmos DB, Databricks), GCP (App Engine, Cloud Run), Cassandra, Redis, MySQL, PostgreSQL, Oracle, React, Redux, Vue.js, Angular, Bootstrap, HTML5, CSS3, JavaScript, TypeScript, GraphQL, Node.js, Kafka, RabbitMQ, ELK Stack, Splunk, SonarQube, Jenkins, GitHub Actions, Maven, Git, JIRA, Airflow, Golang, Micronaut, OpenShift, PCF, Cypress, Selenium, JUnit, Mockito, Python

 **Client: Wells Fargo, Florida Jan 2022 - June 2023**

**Role: Sr. Java Full Stack Developer**

**Responsibilities:**

* Developed complex state management solutions in Angular using NGRx for Wells Fargo’s financial dashboards, ensuring predictable data flow and supporting advanced features like undo/redo and session persistence for secure transaction workflows.
* Integrated Angular front-end components with Java-based backend services through NGRx effects, handling side effects, managing RESTful API calls, and orchestrating state updates for transaction and account management flows.
* Built responsive, mobile-first banking interfaces using Bootstrap and Angular 8, providing consistent design and functionality across multiple devices while adhering to strict corporate UI/UX guidelines.
* Constructed front-end modules leveraging Angular 8 dependency injection, RxJS Observables, HTTP Client modules, and Node.js server integration to ensure seamless real-time communication between UI and backend systems.
* Designed and developed secure, high-performance Microservices using Spring Boot, Spring Cloud, and Spring Security, handling Wells Fargo’s online banking services, transaction management, and customer authentication systems.
* Integrated OAuth 2.0 and JWT token-based security into Microservices, implementing granular role-based access control and ensuring regulatory compliance with financial industry standards for data security.
* Built scalable RESTful APIs with Spring MVC and Hibernate ORM, supporting real-time transaction processing, account management, and service request routing across Wells Fargo’s digital banking platforms.
* Developed and deployed cloud-native applications using AWS Lambda, EC2, S3, RDS, and API Gateway, enabling fault-tolerant, horizontally scalable banking workloads with automated disaster recovery.
* Automated validation of microservices deployments using Python scripts in Kubernetes and Docker environments, ensuring consistent configuration management and zero-downtime rollouts across hybrid cloud infrastructure.
* Created Infrastructure as Code templates with Terraform to provision secure AWS resources, maintaining audit trails and version control for regulatory compliance and environment consistency.
* Built high-performance Single Page Applications using React 17, Redux, and TypeScript, enhancing Wells Fargo’s customer experience by providing real-time account updates, transaction monitoring, and fraud alert notifications.
* Designed GraphQL APIs with advanced resolver logic to reduce payload size, eliminate over-fetching, and optimize network communication between web/mobile banking platforms and backend data services.
* Automated service health monitoring using Python scripts integrated with Splunk and ELK Stack, creating custom dashboards and alerts to detect anomalies in banking applications before they impact customers.
* Implemented Kafka-based event-driven architectures to manage transactional events, real-time ledger updates, and inter-service communication, ensuring reliable message delivery across banking services.
* Built comprehensive CI/CD pipelines using Jenkins, Maven, and GitHub Actions to automate build processes, integrate unit tests, security scans, and perform automated deployments to production environments.
* Managed distributed databases with PostgreSQL, MySQL, Oracle, MongoDB, and DynamoDB, supporting high-transaction banking systems, audit logging, and customer profile storage with high availability.
* Developed secure web interfaces using HTML5, CSS3, Vue.js 2, and Bootstrap, ensuring cross-browser compatibility, responsive layouts, and secure interactions for account management portals.
* Integrated AWS IAM policies for fine-grained access control, securing AWS-hosted resources, and aligning with Wells Fargo’s internal security and compliance guidelines.
* Conducted end-to-end testing using JUnit, Mockito, Selenium, Cypress, Jest, and Cucumber to ensure functionality, reliability, and adherence to financial regulatory standards.
* Built reactive microservices using Micronaut for performance-critical backend tasks, improving transaction processing speed and reducing backend system latency.
* Created Node.js backend APIs to support real-time notification systems, fraud detection alerts, and customer communication services, reducing latency for time-sensitive operations.
* Orchestrated large-scale ETL workflows using Azure Data Factory and Databricks to process financial transaction data, customer analytics, and reporting pipelines in compliance with regulatory data governance.
* Optimized cloud infrastructure using AWS CloudWatch for monitoring, CloudFront for CDN delivery, and Auto Scaling groups to maintain system performance during peak online banking hours.
* Managed Kubernetes workloads on Azure AKS and AWS EKS, deploying containerized microservices across hybrid cloud environments and integrating with Azure Event Hubs for real-time banking event processing.

**Environment:** Java 11, Spring Boot, Spring Cloud, Spring Security, Hibernate, Microservices, Python, Node.js, Golang, Angular 8, React, Vue.js, Bootstrap, HTML5, CSS3, TypeScript, GraphQL, Kafka, Kubernetes, Docker, Terraform, AWS, Azure, SQL/NoSQL Databases (PostgreSQL, MySQL, Oracle, DynamoDB, MongoDB, Redis), Jenkins, GitHub, Airflow, Prometheus, Grafana, SonarQube, Log4j, Splunk, ELK Stack, Selenium, Cypress, Jest, JUnit, Mockito, Cucumber, CI/CD.

 **Client: State of Delaware, Delaware Apr 2018 - Dec 2021**

**Role: Java Full Stack Developer**

**Responsibilities:**

* Designed and implemented secure, scalable backend services using Core Java and Spring Boot, architecting modular Microservices to support critical state government workflows including licensing, taxation, document processing, and citizen services.
* Utilized advanced Java multithreading and concurrency features to optimize system performance for high-volume data transactions, reducing processing time for citizen requests across various state-level digital portals.
* Built RESTful APIs in Java, integrating seamlessly with front-end applications while enforcing state-mandated security standards, incorporating comprehensive logging, exception handling, and writing unit tests with JUnit and Mockito to ensure stability and maintainability.
* Developed and maintained dynamic, responsive web interfaces using Angular, integrating with RESTful APIs and backend Microservices to enhance user experience for both public-facing citizen services and internal government portals, with a focus on performance optimization and accessibility compliance.
* Implemented secure authentication and authorization workflows using OAuth 2.0, JWT tokens, Single Sign-On (SSO), and role-based access control (RBAC), enabling unified identity management across multiple state systems handling sensitive citizen data.
* Built highly interactive user interfaces with React, Vue.js 2, TypeScript, and Bootstrap for Delaware’s citizen platforms, ensuring cross-browser compatibility, mobile responsiveness, and compliance with WCAG 2.1 accessibility standards.
* Automated the provisioning and deployment of cloud infrastructure using Terraform, enabling consistent, repeatable deployments for government services across development, staging, and production environments.
* Integrated Python-based monitoring utilities with Grafana and Prometheus, creating real-time dashboards for visualizing system health, performance metrics, and automated alerting on anomalies.
* Streamlined CI/CD pipelines using Jenkins, Maven, and GitHub Actions to automate builds, run security scans, and deploy applications, reducing manual intervention and accelerating delivery cycles.
* Implemented Redis caching layers to reduce data retrieval latency, improving the responsiveness of citizen service applications and ensuring consistent performance during peak usage periods.
* Developed Kafka-based real-time messaging solutions to enable asynchronous communication between distributed state services, supporting emergency alerts, public safety notifications, and transactional event processing.
* Integrated Splunk and ELK Stack (Elasticsearch, Logstash, Kibana) for centralized logging, enabling proactive incident monitoring, forensic analysis, and faster troubleshooting across government systems.
* Built and deployed Kafka real-time event streaming platforms for asynchronous communication between critical state systems, improving event-driven processing for functions like emergency services and document workflow management.
* Engineered scalable NoSQL database solutions using MongoDB, DynamoDB, Cassandra, and Redis to manage semi-structured and unstructured data, including permits, licenses, and citizen profiles with flexible schema designs.
* Orchestrated advanced CI/CD workflows using Jenkins, GitHub Actions, Maven, Docker, and Kubernetes, supporting blue/green deployments, rolling updates, and automated vulnerability scans to maintain continuous compliance.
* Designed robust ETL pipelines with Apache Airflow, automating the processing of large-scale citizen data sets across AWS services including S3, EC2, EKS, Redshift, and Snowflake, supporting statewide analytics, reporting, and decision-making processes.
* Integrated comprehensive security measures including SSL/TLS encryption, AWS IAM policies, OAuth 2.0 authentication, AWS Cognito, and AWS KMS for encrypted data management, ensuring compliance with state and federal security standards.
* Led DevSecOps initiatives by embedding security best practices into CI/CD pipelines, implementing automated code analysis with SonarQube and Snyk, container image hardening, and continuous vulnerability scanning to maintain secure production releases.
* Developed internal administrative portals using Vue.js 2 and TypeScript, providing state employees with efficient tools for managing cases, permissions, and citizen interactions, reducing manual workload and improving government service delivery.
* Built scalable backend APIs using Node.js for real-time transactions, document uploads, and integrations with legacy and modern government databases, improving transaction speeds and API scalability.
* Designed advanced GraphQL schemas with subscriptions and resolvers to support real-time updates in citizen service portals, reducing data over-fetching and improving frontend performance for interactive state applications.
* Automated monitoring, alerting, and performance tuning using Prometheus, Grafana, and AWS CloudWatch, achieving rapid incident detection and maintaining system reliability for Delaware’s public service applications.
* Led Kubernetes-based container deployments on AWS EKS, implementing automated scaling, self-healing, rolling updates, and blue/green deployments to ensure high availability and zero-downtime upgrades for mission-critical services.

**Environment:** Java 8, Spring Boot, Microservices, OAuth2.0, JWT, Kubernetes, Docker, Terraform, AWS (EC2, Lambda, API Gateway, S3, RDS, DynamoDB, SNS, SQS), Python, Redis, Cassandra, MongoDB, Snowflake, PostgreSQL, React, Vue.js, Angular, TypeScript, GraphQL, Node.js, Kafka, Airflow, Prometheus, Grafana, ELK Stack, CI/CD, PySpark, Scala, Test Automation (JUnit, Mockito, Cucumber, Selenium), DevOps Tools (Maven, Jenkins, Git).

 **Client: Ernst & Young, North Carolina July 2016 – March 2018**

**Role: Full Stack Java Developer**

**Responsibilities:**

* Engineered complex enterprise solutions using Java, Spring Boot, and Microservices to develop financial reporting, risk management, and tax automation platforms, following industry best practices and secure coding guidelines.
* Utilized Java Collections, Streams, and Functional Programming paradigms to build efficient data processing pipelines, significantly reducing memory footprint and improving application performance within large-scale distributed systems.
* Developed robust backend modules integrating Java with Oracle databases and Kafka event streaming, ensuring high availability, fault tolerance, and seamless data flow between microservices, supported by automated testing frameworks like JUnit, Mockito, and CI/CD pipelines.
* Designed and implemented OAuth 2.0 and JWT-based authentication with Single Sign-On (SSO) for E&Y’s client and employee portals, enhancing security, regulatory compliance (GDPR, SOC2), and streamlining user access management.
* Created modern, responsive single-page applications using React 17, Redux, React Hooks, TypeScript, and Bootstrap to deliver highly interactive and secure dashboards for financial reporting, tax compliance, and real-time audit insights.
* Architected scalable AWS serverless solutions leveraging Lambda, API Gateway, DynamoDB, S3, CloudFront, and IAM to build secure and cost-effective services supporting data processing, audit automation, and client deliverables.
* Deployed microservices using Docker containers orchestrated with Kubernetes clusters on AWS EKS and Azure Kubernetes Service (AKS), enabling rapid scaling, high availability, and disaster recovery of mission-critical applications.
* Automated infrastructure provisioning and compliance management with Terraform and AWS CloudFormation for multi-cloud environments, ensuring repeatable, secure, and audit-ready infrastructure deployments.
* Integrated enterprise observability solutions including Splunk, ELK Stack (Elasticsearch, Logstash, Kibana), Prometheus, and Grafana for real-time system monitoring, root-cause analysis, and compliance auditing.
* Implemented Kafka as a high-throughput, fault-tolerant event streaming platform to facilitate reliable real-time data processing for audit workflows, financial transactions, and risk management systems.
* Built secure integrations with AWS Connect, Salesforce, and internal E&Y systems to enhance customer support capabilities and streamline cross-platform data synchronization.
* Developed scalable NoSQL database architectures using MongoDB, DynamoDB, Cassandra, and Redis to efficiently manage unstructured data supporting audit automation, tax compliance, and financial reporting systems.
* Automated CI/CD pipelines using Jenkins, GitHub Actions, Maven, Docker, and Kubernetes to support continuous delivery, rapid iteration, and strict adherence to software development lifecycle (SDLC) standards.
* Developed containerized backend services with Node.js for real-time financial data processing, boosting speed and scalability of audit and compliance applications.
* Enforced DevSecOps practices by automating security scanning, vulnerability management, and compliance within CI/CD pipelines using SonarQube, Snyk, and container image hardening techniques.
* Designed GraphQL resolvers, schemas, and subscriptions to enable real-time, efficient data querying across web and mobile platforms for audit, tax, and consulting applications, supported by Vue.js 2, React, and Bootstrap-based responsive administrative dashboards.

**Environment**: Java 8, Spring Boot, Microservices, REST APIs, OAuth2/JWT/SSO, Kubernetes (EKS, AKS), Docker, Terraform, AWS (Lambda, S3, DynamoDB, RDS), Azure (Data Factory, Cosmos DB), GCP, NoSQL (Redis, Cassandra, MongoDB), React, Vue.js, TypeScript, GraphQL, Node.js, Kafka, CI/CD (Jenkins, GitHub), Testing (JUnit, Mockito, Jest), Monitoring (Prometheus, Grafana, Splunk), SonarQube, DevSecOps.

**Client: KeyCorp, Michigan March 2014 - June 2016**

**Role: Java Developer**

**Responsibilities:**

* Participated actively in all phases of the Software Development Life Cycle (SDLC), including requirements gathering, design, development, testing, deployment, and support of enterprise financial applications, following a traditional Waterfall methodology with detailed planning and phased releases.
* Developed dynamic and interactive user interface components using JSP, HTML, CSS, and JavaScript, ensuring cross-browser compatibility and responsive design to enhance customer experience.
* Integrated AngularJS (v1) into existing UI frameworks, leveraging its two-way data binding, directives, and MVC patterns to create a more interactive and seamless user interface for banking and financial services applications.
* Engineered a custom UI tool using Java, XML, and JavaScript to streamline specific business workflows, improving user interaction and overall system usability for back-office users.
* Implemented robust front-end validations using JavaScript to minimize data entry errors, enforce business rules at the client side, and improve form submission integrity.
* Developed complex SQL queries, stored procedures, and PL/SQL scripts to handle backend data transactions using JDBC, facilitating efficient data retrieval and manipulation from Oracle databases.
* Contributed to server-side development using JSP, Servlets, and the Struts framework, focusing on Action Forms, Action Servlets, and validation logic to manage user requests and system responses in a structured MVC architecture.
* Provided production support and issue resolution for existing web applications, conducting root-cause analysis and bug fixes to ensure business continuity.
* Designed and implemented RESTful web services for data exchange between distributed systems, promoting interoperability between internal banking applications.
* Utilized SLF4J as a logging façade to integrate various logging frameworks in distributed system environments, enabling consistent log management and centralized analysis.
* Configured and maintained AWS RDS (Relational Database Service) instances to ensure reliable, scalable, and secure database storage solutions for critical financial data.
* Integrated AWS Simple Notification Service (SNS) and Simple Queue Service (SQS) to enable asynchronous messaging, event-driven communication, and decoupled system architecture for notification and transaction processing.
* Used ANT as the build and deployment tool to automate the compilation of source code, packaging of WAR files, and deployment to application servers, streamlining the release management process.

**Environment:** Java 1.5, HTML, CSS, J2EE, JSP, JavaScript, AWS, AngularJS, Servlets, Hibernate, JDBC, Apache Kafka, Oracle, SQL, ANT, SLF4J, DAO, Struts, PL/SQL, XML, AWS RDS, AWS SNS, AWS SQS, RESTful Web Services, Windows, UNIX.