**Mohammed Abdul Malik**

# Summary

A highly skilled and versatile Full Stack Front-end Developer with over 12+ years of experience delivering innovative web solutions across a range of industries, including retail, healthcare, transportation, and telecommunications. Expertise in building dynamic, responsive, and user-centric web applications utilizing modern technologies such as JavaScript, TypeScript, React, Angular, HTML, and CSS. Strong background in cloud platforms including AWS and Azure, coupled with proficiency in CI/CD pipelines to streamline development workflows. Adept at designing, developing, and integrating robust APIs to enhance system functionality and ensure seamless communication between front-end and back-end systems. Proven track record of working in cross-functional teams, leading projects from concept to deployment, and delivering high-quality solutions that drive business growth and improve user experiences.

**Technical Skills**

* **Web Technologies:** HTML5, CSS3, SASS, JavaScript (ES6+), TypeScript, AJAX, jQuery, JSON, Bootstrap, Tailwind CSS, Express.js, Node.js, Webpack, Babel, Git, GitHub, GraphQL
* **Backend & API Development:** Java, Node.js, Express.js, Nest JS, RESTful APIs, GraphQL, Apollo Server, Web Sockets, gRPC, Microservices Architecture
* **Frontend Frameworks/Libraries:** React.js, Redux, Next.js, Vue.js, Angular, Material-UI, Chakra UI, Ant Design, Styled Components
* **Web Concepts:** Responsive Web Design (RWD), Progressive Web Apps (PWA), Single Page Applications (SPA), Server-Side Rendering (SSR), Cross-Browser Compatibility, Multi-Browser Support, Accessibility (WCAG)
* **Databases:** SQL (MySQL, PostgreSQL, Microsoft SQL Server), NoSQL (MongoDB, Firebase, DynamoDB, Cassandra, Redis)
* **Testing Tools & Frameworks:** Junit 4 & 5, Mockito, Calypso, Power Mock, Selenium, Jest Agile, UML Waterfall, Jasmine, karma
* **Programming Languages:** Java (J2EE, Spring Boot), JavaScript (ES6+), TypeScript, Python, PHP, C++, C#
* **Cloud & DevOps Tool:**  AWS (Lambda, S3, EC2, API Gateway, DynamoDB), Azure, Google Cloud, Docker, Kubernetes, Jenkins, GitHub Actions, Terraform, Chef, Ansible, CI/CD Pipelines
* **Software Development Methodologies:** Agile (Scrum, Kanban), TDD (Test-Driven Development), BDD (Behavior-Driven Development), UML, Waterfall

**EDUCATION:  
BE** in CS RGPV Bhopal in year 2010

**PROFESSIONAL EXPERIENCE**

### ****Lead Full-Stack Developer | Baylor Scott Jan 2024 – Present****

**Dallas, Texas.**

**Overview:** Baylor Scott & White Health is the largest not-for-profit healthcare system in Texas, encompassing 52 hospitals and over 1,300 care sites, including specialized centres, institutes, and regional clinics. The organization is dedicated to promoting the well-being of individuals, families, and communities through a comprehensive range of services, such as inpatient, outpatient, rehabilitation, and emergency medical care

**Projects:**

**Secure Patient Portal Development**

**Healthcare Data Processing**

**E-Commerce Platform for Healthcare Products**

**Responsibilities**

**Secure Patient Portal Development**

* Developed complex healthcare web apps using React 18 and Next.js 13+ with micro frontend architecture, enabling modular deployments of features like patient dashboards, appointment booking, and telemedicine.
* Developed modern, responsive React.js interfaces using React Query and Context API, delivering smooth user experiences in ad performance and targeting dashboards.
* Designed and implemented secure, scalable APIs using Node.js, Next.Js, and Python (FastAPI) to support patient scheduling, billing, and medical records access.
* Developed and optimized GraphQL and RESTful APIs, improving data retrieval speed across critical healthcare modules.
* Collaborated with cross-functional teams and business stakeholders to define technical solutions aligned with healthcare workflows and regulatory needs, documenting user stories and acceptance criteria in JIRA and Confluence.
* Led the development of custom modules and integrations using Java, Node.js, and Python, adhering to enterprise coding standards and healthcare data privacy regulations (HIPAA).
* Utilized AWS services including Lambda, API Gateway, S3, CloudFront, and Cognito to build cloud-native applications ensuring high availability and performance.
* Implemented CI/CD pipelines using Jenkins and GCP Cloud Build, enabling automated testing, containerization, and zero-downtime deployments to Kubernetes clusters.
* Engineered Kafka-based event-driven architecture for real-time communication between appointment services and notification systems.
* Customized backend services to support financial modules such as patient billing, insurance claims, and scheduling workflows, using REST/GraphQL APIs and secure role-based access controls.
* Developed CI/CD pipelines using GitHub Actions and Docker with IaC templates, ensuring consistent deployment across AWS and on-prem Windows/Unix environments.
* Used PostgreSQL, CosmosDB, and MySQL to design normalized schemas and optimize query performance for complex healthcare data sets.
* Consumed and transformed data using Axios and XLSXJS, integrating with external healthcare systems via JSON, XML, and HL7 formats.
* Collaborated with frontend teams on Angular 17+ and React 18 applications, integrating dynamic APIs and improving portal responsiveness.
* Applied HIPAA-compliant security practices including MFA, encrypted tokens, and secure key management using AWS KMS.
* Automated CI/CD pipelines using GitHub Actions and AWS CodePipeline, enabling faster and safer deployments

**Healthcare Data Processing**

* Migrated legacy apps from React 16 to React 18 concurrent mode and optimized UI using React Server Components and Partial Prerendering (Next.js 14).
* Developed a Microservices-based architecture using Spring Boot and GraphQL APIs to process healthcare data, improving system performance.
* Built and consumed GraphQL APIs to enable efficient data querying and improve front-end responsiveness and flexibility
* Architected real-time event processing system using Kafka for medical data streaming
* Built Spring Boot Microservices for patient management, appointments, and billing
* Implemented role-based access control (RBAC) to ensure sensitive patient data was accessible only to authorized users.
* Implemented automated unit tests and regression tests using JUnit, Jest, and Cypress, achieving test coverage across critical modules.
* Designed and deployed cloud-to-site control systems for managing distributed energy resources (DER), including solar panels, BESS, and smart lighting across multiple healthcare campuses.
* Worked with schedulers and optimizers like GAMS and CPLEX to enhance energy utilization and automate energy demand response programs.
* Worked closely with QA and DevOps teams to extend test automation using UiPath for process validation and user journey simulation.
* Mentored junior developers on API security, JavaScript best practices, and test-driven development, fostering a culture of quality and ownership.
* Implemented CI/CD pipelines using Fastlane, GitHub Actions, and App Center, automating app deployment for iOS and Android.
* Developed and maintained web applications using Vue.js 3.x for the front end and Java 17 with Spring Boot 3.x for the back end, ensuring high performance and scalability.
* Implemented telemedicine features using WebRTC, React Native Reanimated, and Firebase, enabling real-time video consultations between doctors and patients.
* Developed and maintained Microservices-based applications using Java (Spring Boot), Python (FastAPI, Flask), Node.js, and Next.Js, ensuring scalability and performance.
* Managed SQL (Azure SQL, PostgreSQL, MySQL) and NoSQL (CosmosDB, MongoDB) databases for efficient data storage and retrieval.
* Integrated Electronic Medical Records (EMR) systems with EMP (Epic) APIs using HL7, RESTful APIs, and GraphQL for enhanced patient data management.
* Created custom Next.Js modules for authentication and logging
* Used AWS ECS (Elastic Container Service) to manage Docker containers and AWS Fargate for serverless container deployment.
* Integrated AWS CloudWatch for real-time monitoring and logging of Microservices.

**E-Commerce Platform for Healthcare Products**

* Built the front-end for a secure e-commerce platform for healthcare products, including Prescription medications and medical supplies.
* Secured React and Next.js applications with JWT, OAuth2, and AWS Cognito authentication, ensuring HIPAA-compliant user data handling.
* Used Webpack 5 Module Federation to dynamically compose Next.js micro frontends that interact smoothly with shared React modules.
* Provided a unified data visualization interface for energy managers using React.js, pulling in metrics from various sources with API-driven dashboards.
* Integrated Stripe API for secure payment processing, ensuring PCI-DSS compliance with SSL/TLS encryption.
* Developed backend APIs using Node.js 16.x for order management and payment processing, securing transactions with JWT tokens.
* Used AWS Amplify to deploy and host the front-end application, enabling seamless CI/CD integration.
* Implemented AWS Cognito for user authentication and authorization, enhancing security and scalability.
* seamless integration with other JavaScript libraries and frameworks.
* Used AWS S3 to store and serve product images, optimizing load times with AWS CloudFront CDN.
* Set up AWS DynamoDB for caching frequently accessed product data, reducing database load.
* Participated in SAFe Agile ceremonies, contributing to sprint planning, retrospectives, and release demos while ensuring business goals were met.
* Collaborated with hardware integration teams and utility providers to ensure proper metering, alerting, and compliance with local energy and healthcare facility codes.
* Contributed to system resilience by supporting real-time alerting, failover logic, and live energy usage predictions, significantly improving facility uptime and power continuity.
* Integrated financial data sources using secure XML/JSON APIs, enabling accurate reconciliation and reporting for internal healthcare finance teams.
* Implemented a recommendation engine using machine learning algorithms to suggest related products, increasing cross-selling revenue.
* Integrated Google Analytics to track user behaviour and improve marketing strategies.
* Designed and optimized UI/UX components using Figma, ensuring an intuitive and responsive design compliant with accessibility standards.
* Implemented robust state management using NgRx to handle complex financial data flows and user interactions.

### ****Senior / Lead Software Engineer | Verra Mobility June 2019**** – Dec 2023

**Austin, Texas.**

**Overview:** Verra Mobility is a global leader in intelligent transportation systems, specializing in fleet management, automated tolling solutions, traffic enforcement, and smart mobility services. It provides advanced technology solutions for government agencies, commercial fleets, and rental car companies to improve traffic safety, compliance, and operational efficiency.

**Projects:**

**Intelligent Fleet Management System**

**Toll Pay – Automated Toll Collection System**

**Customer Support Ticket Management System**

**Responsibilities**

**Scalable Fleet Management System**

* Designed and led the development of a scalable fleet management system for vehicle tracking, maintenance, and fleet optimization.
* Integrated Redux for managing application state, handling large-scale fleet data such as vehicle status, location, and real-time updates.
* Led the backend development of a real-time citations and violations platform using Java (Spring Boot) and Kafka, focusing on Microservices architecture for modularity and scalability.
* Upgraded and optimized legacy modules by implementing API versioning, introducing asynchronous messaging using Kafka topics, and improving throughput.
* Integrated communication protocols such as OPC UA, MQTT, and DNP3 to enable reliable, real-time data exchange between field hardware and centralized control systems.
* Supported cloud-to-edge architectures connecting AWS-hosted Microservices with roadside equipment, enabling telemetry, traffic enforcement, and environmental data streaming at scale.
* Designed and maintained CI/CD pipelines using Bitbucket, Jenkins, and Artifactory, automating build validation and deployment across environments.
* Led backend development of Microservices in .NET Core and Java, processing tolls, citations, and vehicle data using Kafka for real-time messaging and SQS/SNS for async workflows.
* Developed CRUD APIs using C#/.NET Core and Node.js, integrated with SQL Server and Cosmos DB, ensuring ACID-compliant data handling.
* Built XML/JSON based integrations with law enforcement agencies, ensuring interoperability with third-party systems.
* Used AWS Lambda, S3, and CloudWatch for building serverless components, data archiving, and monitoring of high-throughput backend systems.
* Implemented GraphQL APIs to efficiently fetch fleet data from multiple backend services, reducing API calls and improving performance.
* Applied security best practices by integrating SonarQube for code quality and vulnerability scanning as part of the DevSecOps pipeline.
* Monitored application health and optimized performance using Stack driver, Prometheus, and Grafana within the GCP ecosystem.
* Developed automated tolling and traffic violation systems using Vue.js 2.x, Java 11 with Spring Boot 2.x, and PostgreSQL 13, improving processing efficiency.
* Set up Kafka Connect for data integration with PostgreSQL
* Deployed the platform using Azure Kubernetes Service (AKS) and Docker for consistent containerized environments.
* Implemented real-time notifications using WebSocket’s to alert fleet managers about vehicle maintenance schedules and emergencies.
* Integrated Google Maps API for real-time vehicle tracking and route optimization.

**Toll Payment Processing System**

* Developed the front-end application using React 17.x and TypeScript 4.x, ensuring an optimized user interface and smooth payment process across devices.
* Developed a reusable component library using React and integrated it into Next.js micro frontend projects with Tailwind CSS and Storybook, maintaining UI consistency across teams
* Utilized Redux for state management, handling user payment data, toll amounts, and processing status efficiently.
* Developed GraphQL resolvers in Node.js and Spring Boot to interface with both SQL and NoSQL data stores.
* Built federated GraphQL gateways to unify APIs from multiple Microservices into a single query layer, improving developer experience and API discoverability.
* Enhanced UI/UX for various mobility solutions using Vuetify (Vue.js 2.x), BootstrapVue, and TailwindCSS, improving user engagement and accessibility.
* Built GraphQL APIs to retrieve toll transaction data, allowing the front-end to query specific information (e.g., toll history, balance) without unnecessary over-fetching of data.
* Implemented .NET Core for creating RESTful APIs that handled payment transaction processing, working with SQL Server for secure Leveraged Kafka Streams for real-time processing of traffic violation events and publishing to analytics dashboards.
* Developed backend logic for license plate recognition systems and citation issuance using Java 11, Spring Boot, and MongoDB.
* Implemented systems for grid-connected assets, such as electric vehicle (EV) charging stations and solar-powered toll monitoring units, supporting real-time performance monitoring and load distribution.
* Collaborated closely with hardware vendors, field engineers, and market interface teams to deploy end-to-end data pipelines for device control, diagnostics, and over-the-air updates.
* Integrated optimization engines like CPLEX to calculate traffic routing decisions and prioritize signal timing under constrained resource conditions.
* Interfaced directly with stakeholders and business analysts to gather requirements and deliver functionality aligned with clearing and compliance workflows, ensuring adherence to SLAs.
* Conducted peer code reviews and architecture walkthroughs to ensure adherence to coding standards, security practices, and modular design.
* Customized application workflows (task stations, data routing) to process incoming law enforcement events and citations, mirroring Calypso workflow and SDI design principles.
* Optimized data pipelines and reporting APIs using Python Scripts for ETL, data validation, and cleanup of ticketing system records.
* Developed Angular-based admin portals for managing toll transactions and user accounts.
* Used Java Spring Boot to build Microservices for payment processing and transaction history.
* Integrated C# .NET Core APIs for secure payment gateway integration and fraud detection.
* Leveraged Azure Redis Cache to cache frequently accessed payment data, for reducing database load.
* Integrated SMS and email notifications to keep users informed about toll payments and balances.
* Conducted load testing using JMeter to ensure the system could handle concurrent users.

**Customer Support Ticket Management System**

* Designed the React-based front-end using TypeScript, ensuring an intuitive user interface for support agents and end users.
* Implemented .NET Core 5.x to create backend services that processed customer tickets, support history, and interactions, while securing sensitive data.
* Architected secure and performant API interactions within React and Next.js apps, consuming backend services built in Spring Boot and .NET Core.
* Integrated Python Microservices to analyze customer feedback and categorize tickets based on historical trends, providing faster ticket routing.
* Created real-time data dashboards using D3.js and backend aggregations from Kafka topics, providing operational insight into tolling systems.
* Conducted SQL performance tuning on stored procedures and large joins to reduce latency in violation processing reports.
* Mentored junior developers and led Agile ceremonies, improving sprint velocity and documentation standards, Payment data storage.
* Managed multi-platform deployments across Windows and Unix, implementing monitoring and logging using Splunk and CloudWatch for proactive issue resolution.
* Extended and maintained APIs to integrate third-party data (vehicle history, license plate readers), leveraging Java Servlets, XML, and RESTful architecture.
* Participated in Calypso-like task management and static/reference data mapping, ensuring synchronization between the citation system and state agency databases.
* Documented application flows and system architecture in Confluence and led KT (knowledge transfer) sessions for global teams.
* Developed Angular components for ticket management and agent dashboards, improving usability and productivity.
* Used Java Spring Boot to build backend services for ticket processing and analytics.
* Integrated C# .NET Core APIs for secure data handling and integration with third-party CRM systems.
* Deployed backend services using Azure Kubernetes Service (AKS) and Docker for scalability and seamless updates.
* Developed a sentiment analysis tool using Python to prioritize high-priority tickets based on customer emotions.
* Automated ticket reporting using Power BI.
* Delivered highly secure, responsive platforms using React, Node.js, and AWS IoT Core, ensuring low-latency data visualization and event-triggered alerts.
* Implemented Microservices architecture using Next.Js, RabbitMQ, and Kafka for distributed systems.
* Collaborated with Python engineers to integrate real-time traffic data processing and vehicle health monitoring.
* Managed data storage using PostgreSQL for fleet information and Redis for caching real-time vehicle data.
* Deployed the platform using Azure Kubernetes Service (AKS) and Docker for consistent containerized environments.
* Built Angular-based dashboards for visualizing real-time traffic and vehicle health data.
* Used Java Spring Boot to build Microservices for traffic data processing and analytics.
* Integrated C# .NET Core APIs for real-time data integration with third-party traffic data providers.
* Leveraged Azure Functions for serverless data processing and analytics.
* Built a predictive analytics module using Python to forecast traffic patterns and optimize fleet routes.

### ****Full-Stack Developer | Ahold Delhaize Jan 2018 – Aug 2019 Quincy, Massachusetts.****

**Overview:** Ahold Delhaize is a global leader in supermarket retail, operating brands like Stop & Shop, Food Lion, Giant. It specializes in grocery e-commerce, digital retail experiences, and supply chain optimization, delivering online and in-store shopping solutions across multiple markets.

**Projects:**

**Grocery E-Commerce Platform**

**Loyalty Rewards Program**

**Inventory Management System**

**Responsibilities**

**Grocery E-Commerce Platform**

* Developed and led the creation of a grocery e-commerce platform for customers to browse products, place orders, and schedule deliveries, contributing to Ahold Delhaize's online presence.
* Designed and implemented RESTful APIs for product catalog, cart management, and order tracking, ensuring seamless integration with the front-end.
* Implemented Redux for managing complex state between various components (cart management, product listings, and user authentication) to improve data consistency and app performance.
* Developed Node.js (10.x) Microservices for handling user authentication, order processing, and inventory management, utilizing Express.js for building RESTful APIs.
* Used RxJS for data fetching and parsing of data from REST endpoints. Utilized Ngrx for state management using store, selectors, actions, and reducers.
* Implemented Authentication and Authorization using JSON web tokens. Used Jasmine for unit testing and Karma for integration testing of Angular components.
* Set up Jenkins pipelines for continuous integration and continuous deployment (CI/CD), ensuring rAPId iteration and deployment cycles.
* Containerized the application using Docker and orchestrated with Kubernetes for seamless deployment and scaling on Azure.
* Utilized Azure App Services for hosting the backend services and SQL Server for transaction data and inventory management.
* Reduced deployment times by 40% through automated CI/CD pipelines and containerization.

**Loyalty Rewards Program**

* Built the front-end with Angular 6.x and TypeScript 3.x, ensuring a dynamic user interface that integrated with the company’s existing e-commerce systems.
* Implemented Spring Boot 2.x for backend services, including the management of customer points, rewards catalog, and redemption process.
* Developed RESTful APIs using Node.js for real-time points calculation and redemption tracking.
* Integrated Swagger for API documentation, improving developer productivity and collaboration.
* Automated CI/CD pipelines using Jenkins to ensure seamless deployment of
* Designed and implemented role-based access control (RBAC) for secure access to loyalty program data.
* Used Azure Blob Storage for storing bulk inventory reports and logs, ensuring scalability and cost-efficiency.
* Improved API response times by 30% through query optimization and caching strategies.

**Inventory Management System**

* Used Java 8-11 and Spring Boot 2.x to build Microservices for stock management, order processing, and demand forecasting.
* Developed a mobile application for a leading e-commerce client using React Native
* Developed and optimized SQL queries to improve database performance.
* Utilized angular services and RESTful web services to connect the user interface to backend databases such as Oracle and MongoDB
* Developed RESTful APIs using Node.js for real-time inventory updates and stock level monitoring.
* Implemented CI/CD pipelines using Jenkins to automate the deployment of inventory management Microservices.
* Integrated Azure API Management to secure and monitor APIs, enabling throttling and analytics.
* Integrated with SQL Server to store inventory data and used Azure Blob Storage for storing bulk inventory reports and logs.
* Implemented Docker for containerizing backend services and set up a Kubernetes cluster on Azure to handle scaling and orchestration.
* Utilized Angular's component-based architecture to create modular, reusable UI components and dynamic templates, reducing code duplication and enhancing the maintainability of large-scale applications.
* Built and managed both template-driven and Reactive forms in Angular, with dynamic form validation to ensure a seamless user experience and enhanced data integrity.
* Developed RESTful APIs using Node.js for form data submission and validation.
* Integrated Swagger for API documentation, improving developer productivity and collaboration.
* Automated CI/CD pipelines using Jenkins to ensure seamless deployment of front-end and back-end services.
* Used Azure Blob Storage for storing form submissions and logs, ensuring scalability and cost-efficiency.

### ****Frontend Developer |Verizon May**** 2016 – Dec 2017 New York, NY

**Overview:** Verizon is a global leader in telecommunications and digital services, offering wireless networks, broadband solutions, connectivity, and enterprise IT services. It specializes in cloud computing, IoT, cybersecurity, and Next-gen communication technologies for consumers and businesses.

**Projects:**

**Customer Self-Service Portal**

**Network Monitoring Dashboard**

**Billing and Payment System**

**Customer Support Ticket Management System**

**Responsibilities**

**Customer Self-Service Portal**

* Designed and developed a customer self-service portal for managing accounts, billing, and service plans, enhancing user experience and reducing call center volume.
* Built the front-end using Angular 4.x and TypeScript 2.x, ensuring a responsive and dynamic user interface.
* Implemented Redux for state management, handling complex data flows between account management, billing, and service plan components.
* Developed RESTful APIs using Node.js (6.x) and Express.js for account management, billing, and service plan updates.
* Integrated Swagger for API documentation, improving developer productivity and collaboration.
* Set up Jenkins pipelines for continuous integration and continuous deployment (CI/CD), ensuring rAPId iteration and deployment cycles.
* Containerized the application using Docker and orchestrated with Kubernetes for seamless deployment and scaling.
* Utilized AWS EC2 for hosting the backend services and RDS for secure storage of customer data.

**Network Monitoring Dashboard**

* Developed a network monitoring dashboard for real-time tracking of network performance and outages.
* Built the front-end using React 15.x and TypeScript 2.x, integrating D3.js for real-time data visualization.
* Implemented Redux for managing complex state between various components (network status, outage alerts, and performance metrics).
* Developed RESTful APIs using Node.js (6.x) and Express.js for real-time data fetching and processing.
* Integrated Swagger for API documentation, improving developer productivity and collaboration.
* Automated CI/CD pipelines using Jenkins to ensure seamless deployment of front-end and back-end services.
* Used AWS S3 for storing network logs and reports, ensuring scalability and cost-efficiency.

**Billing and Payment System**

* Designed and developed a billing and payment system for managing customer invoices and payments.
* Built the front-end using Angular 4.x and TypeScript 2.x, ensuring a responsive and dynamic user interface.
* Implemented Redux for state management, handling complex data flows between invoice management, payment processing, and customer notifications.
* Integrated Swagger for API documentation, improving developer productivity and collaboration.
* Containerized the application using Docker and orchestrated with Kubernetes for seamless deployment and scaling.
* Utilized AWS RDS for secure storage of billing and payment data.

**Customer Support Ticket Management System**

* Designed and developed a customer support ticket management system for tracking and resolving customer issues.
* Implemented Redux for state management, handling complex data flows between ticket management, customer notifications, and support agent workflows.
* Developed RESTful APIs using Node.js (6.x) and Express.js for ticket management and customer notifications.
* Integrated Swagger for API documentation, improving developer productivity and collaboration.

### ****Mean Stack Developer |M&T Bank March**** 2014 – April 2016 Buffalo, NY

**Overview:** M&T Bank is a leading financial institution offering commercial and retail banking services, wealth management, digital banking, and loan solutions. It specializes in secure payment processing, fraud prevention, and regulatory compliance solutions for both personal and business banking customers.

**Projects:**

**Online Banking Platform**

**Fraud Detection System**

**Responsibilities**

* Developed RESTful APIs using Java & Spring Boot for secure banking transactions.
* Built responsive front-end using AngularJS, improving user experience.
* Developed a static data consistency checker that compared reference data across systems, reducing trade failure due to mismatched counterparties or securities.
* Automated clearing settlement file processing using custom Java utilities with batch ingestion logic, cutting down manual intervention
* Integrated a lightweight internal accounting ledger to track daily balances and adjustments at the fund and desk levels.
* Designed and developed RESTful APIs to facilitate communication between the booking system and other internal
* systems, ensuring real-time data synchronization.
* Automated the deployment pipeline with Jenkins and Docker, accelerating the release cycle and minimizing
* deployment-related issues.
* Integrated a lightweight internal accounting ledger to track daily balances and adjustments at the fund and desk levels.
* Contributed to regression testing using custom JUnit test suites and enhanced code coverage from 40% to over 85% in core financial modules.
* Utilized AWS services such as EC2, S3, and RDS to build a highly scalable and reliable infrastructure, supporting peak
* travel seasons with minimal downtime.
* Designed and implemented trade booking interfaces using Java Servlets and JSP, supporting lifecycle processing for fixed income and money market instruments.
* Built modular custom applications to track and reconcile daily settlements across front, middle, and back-office systems, improving audit transparency.
* Developed XML-driven communication interfaces for trade affirmation and broker confirmations, mirroring Calypso message adapter configurations.
* Created dashboard components in JavaScript/HTML to visualize task queues, pending actions, and trade exceptions using real-time status indicators.
* Engineered a multi-step corporate actions processor with approval workflows and role-based routing logic, simulating Calypso’s task station modules.
* Implemented XML schema validation tools to validate inbound FIXML and SWIFT messages, helping catch data discrepancies before integration.
* Configured and monitored scheduled Java tasks for overnight position snapshots, reconciliation jobs, and position aging reports.
* Assisted in Calypso upgrade preparation by reverse-engineering custom workflows, testing backward compatibility, and validating SDI logic.
* Supported pre-trade compliance checks by building rules-based engines using Java and SQL triggers for exposure, eligibility, and risk factor validation.
* Collaborated with business users to capture UAT scenarios and formalize acceptance criteria in Confluence, improving documentation quality.
* Developed helper utilities in Python to transform and ingest large historical datasets into SQL Server, used for risk analytics and backtesting.
* Integrated MongoDB for scalable and flexible user data management.
* Worked closely with iOS and Android teams to seamlessly integrate APIs.
* Developed .NET-based APIs in C# for managing customer interactions.
* Wrote Excel VBA macros for traders to bulk import deal sheets into internal systems, replacing legacy macros and improving reliability.
* Used Microsoft Office (Excel, Word) to produce interactive reporting templates that pulled live data via embedded REST API calls
* Created a monitoring dashboard to track microservice uptime and task failure rates using shell Scripts and scheduled Windows services.
* Participated in organization-wide compliance audits, mapping data access logs, code history, and exception flows to security best practices.
* Conducted thorough testing using JUnit and Selenium, ensuring the stability and reliability of the booking system before each release.
* Optimized Oracle DB queries for faster data retrieval and improved performance.
* Created Node.js & Express.js APIs to support new mobile banking features.
* Designed and maintained SQL Server databases for efficient data storage.
* Built Python/Django APIs to process real-time transaction data for fraud detection.
* Integrated machine learning algorithms to identify suspicious activities.
* Managed PostgreSQL databases, ensuring data integrity and quick access.

### ****Software Developer |Deloitte Oct**** 2012 – Mar 2014 Atlanta, Georgia

**Responsibilities**

* Developed a financial risk assessment tool for evaluating corporate loan applications, ensuring compliance with Basel III regulations.
* Built Spring MVC-based REST APIs for risk calculation models, processing large financial datasets efficiently.
* Developed dynamic front-end dashboards using AngularJS and JavaScript, allowing analysts to visualize risk scores and financial trends.
* Implemented automated unit testing with JUnit and Mockito for code coverage.
* Managed CI/CD pipelines using Jenkins, ensuring smooth deployments across dev, staging, and production environments.
* Built Spring Boot Microservices to fetch real-time inventory data and predict demand using statistical models.
* Created interactive web interfaces using JSP, Servlets, JavaScript ES5, and jQuery, improving data accessibility for supply chain managers.
* Optimized Oracle SQL queries for high-volume transactions, ensuring fast data retrieval for reports and dashboards.
* Designed Spring MVC-based workflows for case filing, document approvals, and automated reminders.
* Implemented Oracle PL/SQL stored procedures for efficient case data retrieval and audit tracking.
* Developed custom UI components in JSP and JavaScript, enabling real-time updates for case statuses.