Sathwik Reddy Mandadi

Email: [sathwikmfs08@gmail.com](mailto:sathwikmfs08@gmail.com)

Phone: 445-226-2878

**PROFESSIONAL SUMMARY**

Highly skilled **Full Stack Java Developer** with over 6 years of experience designing, developing, and maintaining web applications across diverse domains such as insurance, travel, e-commerce, and healthcare. Expertise in **front-end** and **back-end** technologies, cloud services, microservices architecture, and Agile development practices. Proficient in delivering scalable, high-performance applications with a focus on user experience, secure coding, and system reliability. Passionate about leveraging modern tools and technologies such as **ReactJS**, **Spring Boot**, **Java 17**, **AWS**, and **Azure** to drive business value and technical innovation.

**TECHNICAL SKILLS**

|  |  |
| --- | --- |
| Programming Languages | Java, JavaScript, TypeScript, Python |
| Backend Technologies | Java 8/11/17, Spring Boot, RESTful APIs, Microservices, Event-Driven Architecture, Apache Kafka |
| Frontend Technologies | ReactJS, Angular, JavaScript, HTML5, CSS3, TypeScript, Redux, Bootstrap, Material UI, LESS, NodeJS |
| Database Management | SQL (MySQL, PostgreSQL), NoSQL (MongoDB, DynamoDB), JPA/Hibernate |
| APIs & Web Services | RESTful APIs, GraphQL, Spring REST |
| Cloud Platforms | AWS, Microsoft Azure |
| Container Orchestration | Docker |
| Infrastructure as Code (IaC) | Terraform, CloudFormation |
| Build & Automation Tools | Maven, Gradle, Jenkins, AWS CodePipeline |
| Version Control | Git, GitHub, Bitbucket |
| Testing & Automation | JUnit, Mockito, Karma, Jasmine, SonarQube, Postman, Swagger |
| DevOps & CI/CD Tools | Jenkins, Docker, Kubernetes, AWS CodePipeline, AWS CodeBuild, AWS CodeDeploy |
| Monitoring & Logging Tools | CloudWatch, AWS X-Ray |
| Development Methodologies | Agile, Scrum, Kanban |
| Bug & Issue Tracking | JIRA |

**PROFESSIONAL EXPERIENCE**

**Mutual of Omaha -** - **Present**

**Sr. Full Stack Java Developer**

**Project Title**: **Insurance Policy Management and Claims Processing System**

As a **Full Stack Java Developer** at **Mutual of Omaha**, I contributed to the development and enhancement of a **Policy Management and Claims Processing System**, designed to streamline insurance policy lifecycle management, claims processing, and customer service operations. This system was integral in improving operational efficiency, ensuring timely claim settlements, and enhancing customer satisfaction.

**Responsibilities:**

**Front-end Development:**

* Developed dynamic and responsive user interfaces for the **Policy Management** and **Claims Processing** systems using **ReactJS**, ensuring a seamless user experience for insurance agents and customers.
* Used **Redux** to manage the global application state, enabling consistent data flow across different components, such as policy details, claims status updates, and payment processing screens.
* Created custom React components for managing policy data, submitting claims, displaying payment history, and visualizing policy details in an intuitive and user-friendly format.
* Implemented **React Router** for client-side routing, ensuring smooth navigation between various sections of the application, such as policy details, claim submission forms, and account settings.

**Back-end Development:**

* Developed robust back-end services using **Java 17** features such as **Lambda Expressions**, **Optional**, **Completable Futures**, and the **Java Time API** to write effective, efficient, and reusable code that handled the core business logic for insurance policy management and claims processing.
* Designed and implemented **microservices** with **Spring Boot** to provide a scalable, fault-tolerant, and easily deployable solution. This included services for managing policies, claims, customer profiles, payment integrations, and notifications.
* Utilized **SQL** and **NoSQL** databases to store and retrieve policy details, claims data, and customer records. This involved creating complex SQL queries, stored procedures, and optimizing database performance for large-scale data sets.
* Developed **RESTful APIs** to interact with various internal and external services, including third-party insurance verification systems, payment gateways, and customer support systems.
* Implemented **Spring Scheduler** to automate the retrieval and processing of claims data, policy renewals, and reminders for upcoming policy due dates.

**Event-Driven Architecture:**

* Leveraged **Apache Kafka** to implement **event-driven communication** between microservices, enabling real-time updates for claims status, policy renewals, and payment processing, ensuring data consistency across systems.
* Configured **manual offset commits** to guarantee message processing reliability, reprocessing failure messages, and ensuring consistency in the insurance claims lifecycle.

**Cloud & DevOps:**

* Deployed microservices to **AWS ECS** using **Fargate** for container orchestration, ensuring scalability and performance for high-demand periods, such as during the end of the year when policy renewals are most common.
* Set up **Jenkins** pipelines for **Continuous Integration and Deployment (CI/CD)**, automating the process of building, testing, and deploying code. Integrated automated **UI tests** to ensure that the front-end application remained functional after each deployment.
* Used **AWS Managed Streaming for Kafka (MSK)** for real-time event processing and streamlining communication between policy, claims, and payment services.
* Implemented **AWS IAM policies** for data security, ensuring that only authorized users had access to sensitive customer data and financial records. Applied **OAuth 2.0** for secure authentication and authorization of users accessing the system.

**Testing & Code Quality:**

* Wrote **unit tests** and **integration tests** using **JUnit** and **Mockito** to ensure the integrity of both front-end and back-end code. Ensured code coverage and quality by using **SonarQube** to run static code analysis.
* Worked closely with QA teams to ensure comprehensive testing of critical insurance workflows, including claims submission, payment processing, and policy renewals, ensuring the highest levels of reliability and performance.
* Integrated **AWS CloudWatch** for real-time monitoring of application health and performance. Set up **alerts** for potential issues related to application downtime or performance bottlenecks, ensuring proactive resolution of any issues.

**Collaboration & Agile Practices:**

* Worked in an **Agile Scrum** team, attending daily stand-ups, sprint planning, and retrospectives to ensure continuous delivery of features and improvements. Delivered features related to policy updates, claim submissions, and customer communications as per sprint goals.
* Collaborated with **Business Analysts**, **Product Owners**, and the **QA team** to ensure the business requirements for the insurance domain were clearly understood and translated into technical solutions.
* Created technical documentation for developed services and components to ensure easy maintenance and knowledge sharing within the team. Participated in knowledge-sharing sessions and code reviews to maintain high code quality.

**Environment**: Java 17, Spring Boot, Kafka, REST APIs, Microservices, AWS ECS, Fargate, SQL, NoSQL, ReactJS, Redux, React Router, JSX, HTML5, CSS3, JavaScript, Jenkins, Git, GitLab, Docker, AWS Fargate, CI/CD Pipelines, AWS CloudWatch, ELK Stack (ElasticSearch, Logstash, Kibana), SonarQube, AWS IAM, OAuth 2.0, Secure REST API Communication.

**Carlson Wagonlit Travel-Stamford, CT**

**Full Stack Java Developer**

**Project Title**: **Travel Booking and Management System**

As a **Full Stack Java Developer** in the travel industry, I played a critical role in developing and deploying a sophisticated **Travel Booking and Management System**. The project aimed to streamline the travel booking process, enhance customer experiences, and manage travel itineraries efficiently. The system handled everything from flight bookings and hotel reservations to car rentals, ensuring seamless user interaction and quick data processing.

**Responsibilities:**

**Front-end Development:**

* Built dynamic and responsive front-end interfaces using **Angular**, developing various **Angular components**, services, and forms to handle user inputs (e.g., booking forms, search filters, payment details).
* Used **Angular Material** to ensure responsive layouts and created reusable UI components such as buttons, grids, and form controls to enhance the user experience.
* Implemented **property and event binding** for dynamic data binding and designed forms to collect user input, utilizing Angular's **patchValue** for updating data.

**Back-end Development:**

* Developed **RESTful APIs** using **Java 8** and **Spring Boot** to support various functionalities of the travel booking system, including flight booking, hotel management, customer profiles, and payment processing.
* Designed and implemented **microservices** using Spring Boot, enabling modular development and ease of deployment. Each microservice handled different aspects of the travel business, such as user management, payment gateways, and travel itinerary management.
* Worked with relational **SQL databases** to create and manage complex queries involving stored procedures, functions, and joins for data manipulation related to bookings, customer information, and transactions.
* Deployed microservices as **Docker containers** on AWS using **Elastic Beanstalk** and **EC2 instances**. Integrated **CI/CD pipelines** for automated testing and deployment, ensuring smooth rollouts and updates.
* Implemented **Kibana Dashboards** and **CloudWatch Metrics** in AWS to track application performance, set up alerts for potential issues, and ensure the health of the production environment.

**Testing & Quality Assurance:**

* Ensured 100% code coverage by writing **unit tests** for both front-end and back-end code using **JUnit** and **Mockito** for the back-end and **Karma** and **Jasmine** for Angular unit tests.
* Used **SonarQube** to enforce code quality standards, ensuring that all code met team goals for maintainability, readability, and performance.
* Provided support for various testing phases, including functional, regression, and performance testing, to ensure that the application performed well under load and met business requirements.

**Collaboration & Agile Development:**

* Participated in an **Agile** development environment, with daily stand-ups, biweekly sprints, and retrospectives to plan tasks and address any blockers or issues.
* Collaborated closely with **Business Analysts** and **Product Owners** to ensure that user stories were well-defined, understood, and aligned with business goals.
* Provided ongoing support for the production environment by investigating issues, resolving defects, and ensuring the stability of the application. Utilized **Swagger** and **Postman** to test APIs and analyze technical issues.

**Version Control & Project Management:**

* Used **Git** for version control, collaborating with the development team to manage and track changes. The **GitLab** repository served as the central repository for all project code.
* Managed tasks, tracked progress, and logged issues using **JIRA**, ensuring that all work was prioritized and completed on time.

**Support and Monitoring:**

* Provided **technical support** by investigating production alerts and system logs, helping to resolve performance issues and maintain uptime.
* Assisted with the planning and execution of production releases, ensuring that the deployment went smoothly without downtime.
* Used **WireMock** to create mock APIs for testing and integration purposes, allowing developers to simulate external services and test system interactions.

**Environment:** Angular, Angular Material, HTML5, CSS3, JavaScript, TypeScript, Java 8, Spring Boot, REST API, Microservices, Docker, AWS (Elastic Beanstalk, EC2, CloudWatch, Kibana), SQL, Stored Procedures, Functions, Joins, JUnit, Mockito, Karma, Jasmine, Git, GitLab, Jenkins, Docker, AWS CI/CD Pipeline, CloudWatch, Kibana, Swagger, Postman, WireMock, Sonar, Maven, Gradle and Rally.

**Ikea - Jan 2022** - **Nov 2022**

**Application Development Associate**

**Project Title**: **Retail Inventory Management and Customer Experience Enhancement**

As an Application Development Associate at IKEA, I worked on the development of an **Inventory Management System (IMS)** integrated with a **Customer Experience Platform (CEP)**. The goal of this project was to improve the efficiency of inventory tracking, automate restocking processes, and provide personalized shopping experiences for customers both online and in-store. This solution was aimed at reducing inventory errors, improving product availability, and enhancing customer engagement through a seamless omnichannel experience.

**Responsibilities:**

**Front-end Development:**

* Developed intuitive and responsive front-end solutions using **ReactJS** and **NodeJS**. The SPA design improved user engagement by enabling fast load times and dynamic content updates without full-page reloads.
* Ensured that the platform was fully responsive across all devices (mobile, tablet, desktop) using **Bootstrap** and **CSS3**. This provided a seamless shopping experience for customers, both in-store and online.
* Used **LESS CSS pre-processors** to create reusable, dynamic styles with variables and functions, improving the maintainability of the codebase.

**Back-end Development:**

* Developed **RESTful APIs** using **Spring Boot** to handle communication between the front-end and back-end systems. These APIs served data related to product details, inventory status, pricing, and promotions.
* Designed and implemented a **microservices-based** backend architecture to enhance scalability and modularity. Each microservice handled different business logic, including order processing, inventory management, and user personalization.
* Built backend services that interfaced with the **real-time inventory tracking system**, updating stock levels and providing alerts when product quantities reached a threshold for restocking.
* Utilized **Java 8** functional programming features, such as **Streams** and **Lambda Expressions**, to process large volumes of transactional data efficiently, which helped in tracking product movements and sales patterns.

**Automation and Scheduling:**

* Developed automation features that used **Spring Scheduler** to trigger automatic stock replenishment requests based on predefined thresholds, reducing manual intervention and ensuring that inventory was always up to date.
* Built services that provided personalized recommendations and promotions for customers based on their browsing history and past purchases. These services were integrated with the **Customer Experience Platform (CEP)** to enhance customer engagement across various touchpoints (website, mobile app, in-store kiosks).

**Reporting and Analytics:**

* Developed real-time dashboards using **Grafana** and **Prometheus** to monitor key metrics such as product availability, sales trends, and customer behavior. This helped the business team make data-driven decisions for inventory optimization and targeted marketing campaigns.
* Integrated **Apache POI** to generate downloadable reports in **Excel** and **CSV** formats, enabling the business to track inventory movements and generate analytics reports for further analysis.

**System Integration and Performance Monitoring:**

* Integrated the system with third-party APIs for pricing updates, stock availability from suppliers, and payment gateways for seamless transactions.
* Used **multithreading** and **parallel processing** techniques to optimize heavy tasks like inventory updates and daily reports, significantly improving the system's response time.
* Deployed the entire application in a **Red Hat OpenShift** containerized environment, ensuring that the application could scale efficiently during high-demand periods, such as holiday sales or promotions.

**Security & Compliance:**

* Ensured the protection of sensitive customer data by implementing **encryption** techniques for user credentials and transaction data. The application also adhered to data privacy regulations such as **GDPR**.
* Implemented **role-based access control (RBAC)** to restrict sensitive data and features, ensuring that only authorized personnel could access certain parts of the system.

**Collaboration & DevOps:**

* Collaborated closely with business analysts, UI/UX designers, and other developers in an **Agile Scrum** environment. Participated in sprint planning, daily stand-ups, and retrospective meetings to ensure that project deadlines were met.
* Managed code using **Bitbucket** and tracked tasks through **JIRA** to streamline collaboration and ensure clear communication between team members.

**Environment:** ReactJS, NodeJS, HTML5, CSS3, Bootstrap, LESS, Java, Spring Boot, REST APIs, Microservices, MongoDB, PostgreSQL, Grafana, Prometheus, Apache POI (Excel & CSV Generation), Red Hat OpenShift, Spring Scheduler, SSL/TLS Encryption, Role-Based Access Control (RBAC), Bitbucket, JIRA.

**Conquerors Software Technologies Pvt. Ltd Feb 2021** – **Dec 2021**

**Java Developer**

**Responsibilities:**

**Project Overview:** The project at Conquerors Software Technologies Pvt. Ltd. was focused on developing a robust healthcare management system aimed at improving patient management, healthcare service delivery, and regulatory compliance. The system integrates various healthcare services, such as patient records, appointment scheduling, and real-time data processing, while ensuring high levels of security, scalability, and performance. The solution was developed using a microservices architecture, deployed on Azure Cloud, and implemented best practices for secure coding and data privacy standards.

**Responsibilities:**

**Application Design and Development:**

* + Designed and developed a **secure, scalable, and high-performing healthcare management system** using **Java**, **JavaScript**, and **ReactJS**.
  + Built **Single Page Applications (SPA)** utilizing **ReactJS** with modern JavaScript (ES6), implementing key features like states, props, and dynamic forms.
  + Developed modular **React components** to manage complex user interactions such as patient data entry, appointment scheduling, and medical history tracking.

**Backend and Microservices Development:**

* + Designed and implemented **Java-based backend services** for **entitlement filtering**, processing user data based on predefined business rules for healthcare services and benefits eligibility.
  + Created **RESTful APIs** to integrate internal and third-party healthcare systems for streamlined data exchange, including appointment booking systems, electronic health records (EHR), and insurance providers.
  + Ensured the backend system could handle high-volume transactions typical in the healthcare domain, including **patient information updates** and **claims processing**.

**Cloud Deployment (Azure):**

* + Leveraged **Azure Cloud Platform** for the deployment of microservices-based healthcare applications, ensuring optimal scalability, availability, and security.
  + Designed and deployed applications on **Azure App Services** and **Azure Kubernetes Service (AKS)** to manage microservices and scale according to system demand.
  + Integrated **Azure Blob Storage** for secure patient data storage and implemented **Azure Active Directory (AAD)** for user authentication and access control.

**System Performance and Optimization:**

* + Used monitoring and performance optimization tools such as **Dynatrace** and **New Relic** to track application performance, troubleshoot issues, and ensure high system reliability.
  + Optimized system performance to handle high transaction volumes, reduce latency, and ensure the application meets the response time requirements set for healthcare services.

**CI/CD Implementation:**

* + Implemented **Continuous Integration and Continuous Deployment (CI/CD)** pipelines using **Jenkins** and **Git**, automating the process of building, testing, and deploying the application on the Azure cloud platform.
  + Integrated automated unit tests and integration tests into the CI pipeline to ensure the reliability of critical features like patient data entry and healthcare service updates.

**Collaborative and Agile Environment:**

* + Collaborated with cross-functional teams including **business analysts**, **product owners**, **testers**, and **DevOps engineers** to align technical solutions with business goals and requirements in the healthcare sector.
  + Participated in **Agile Scrum ceremonies** such as sprint planning, daily stand-ups, sprint retrospectives, and sprint demos to ensure timely delivery of features and improvements.

**Secure Coding and Compliance:**

* + Followed industry best practices for secure coding, ensuring that sensitive healthcare and patient data was protected through data encryption, access control mechanisms, and secure API designs.
  + Worked in alignment with **SOX (Sarbanes-Oxley)** and other relevant healthcare compliance regulations to ensure that the system adhered to data privacy and regulatory standards.

**Debugging, Support, and Maintenance:**

* + Actively involved in debugging and resolving production issues, ensuring the **minimal downtime** for critical healthcare services, such as patient portals and appointment scheduling.
  + Provided on-call support for the healthcare system, promptly addressing any issues related to system outages or performance degradation.

**Documentation and Knowledge Sharing:**

* + Maintained detailed technical documentation for system design, API specifications, and workflows to support future onboarding, troubleshooting, and upgrades.
  + Contributed to knowledge sharing sessions with junior developers, providing mentorship and guidance on best practices and coding standards.

**Environment:** ReactJS, JavaScript (ES6), HTML5, CSS3, REST APIs, Java 8, Spring Boot, Microservices, RESTful Web Services, Azure Cloud (App Services, AKS, Blob Storage, Active Directory), Jenkins, Git, Azure DevOps, Dynatrace, New Relic, OAuth 2.0, SSL/TLS encryption, Azure AD authentication, SQL Server, Azure Cosmos DB, Git