Sai Swetha
Phone: +1 (774) 418-6971 Linkedin Email: devireddysaiswetha@gmail.com

Summary:

Full Stack Developer with 7 years of experience developing RESTful web services using Java Spring Boot in a microservices architecture, with recent experience in Go for backend development. Possesses in-depth knowledge of core Java, Spring Boot, Kafka, data structures, and object-oriented programming (OOP) principles. Industry-standard expertise in cloud technologies, including AWS, Docker, and Terraform.

Technical Skills:

Languages: Core Java, Python, Go.
Frameworks & Scripting languages: Spring Boot, Spring Data JPA, Hibernate, Shell Scripting, Camunda.
Data Transformation: Jolt
Messaging Systems: Kafka, ActiveMQ
DB: MS SQL, Oracle SQL, PostgreSQL 10, Mongo DB, DynamoDB.
Cloud Technologies & Deployments: AWS (VPC, S3, SQS, EventBus, Step Function, Lambda, ECS, EKS), Docker, Terraform, Kubernetes.
Web Design technologies: Server-Sent Events (SSE)
Technologies & Extras: GIT, Gradle, Maven, JIRA, Grafana, Telemetry, Udeploy, Jenkins, Bitbucket, GitLab, GitHub, Agile methodology.
Integrated Development Environments: STS, Eclipse, IntelliJ, Postman, Camunda Modeler, Insomnia.
Business Process Automation: Camunda Modeler
Agile Methodologies: Scrum, Kanban
Test: Junit, Mockito, JMeter, Automation testing, SoapUI, TDD

Professional Experience:

Application Lead Development Analyst | Express Scripts **|** *April 2024 – Present*

- Designed and implemented a serverless architecture involving API Gateway, AWS Lambda, SQS, EventBridge, and Step Functions in Golang to process dynamic payloads and route them based on target specifications, improving system scalability and reducing latency by 30%.
- Automated the retrieval of endpoint details and authentication credentials from DynamoDB, leveraging caching mechanisms to enhance performance and ensure seamless integration with external systems, resulting in a 20% improvement in processing efficiency.
- Developed and cached client instances dynamically based on grant types, streamlining external API calls and reducing execution time.
- Enhanced the insurance change request system by creating a Golang-based workflow that streamlined event processing and reduced delays in Pega, achieving a 30% reduction in processing time.
- Added loggers and integrated Dynatrace telemetry to enhance monitoring, debugging, and performance tracking, resulting in a 40% reduction in downtime.
- Integrated SLAs and automated event forwarding to Pega using AWS EventBridge and Lambda, enhancing data flow consistency and minimizing delays, which improved accuracy and timeliness in handling insurance change requests.
- Built microservices in Golang to support the insurance change request lifecycle, enabling real-time processing and a seamless handoff to Pega. This update improved efficiency, accuracy, and scalability, handling a 20% increase in transaction volume.
- Developed and containerized clinical system microservices on EKS (Elastic Kubernetes Service) in Java and Spring Boot to manage patient prescription assignments:
- Implemented endpoints to create, get, assign, pre-display, and complete tasks related to prescription assignments, ensuring efficient workflow automation for clinical staff.
- Enabled seamless task management for prescription assignments, improving clinical system responsiveness and ease of task handling.
- Automated infrastructure provisioning with Terraform and integrated Jenkins pipelines, leveraging the AWS-D pattern with PLZ configuration, which reduced deployment time by 30% and lowered deployment errors by 50%, enhancing system reliability across both clinical and insurance workflows.
- Automated deployment pipelines by integrating Terraform with Jenkins, optimizing the infrastructure as code (IaC) process, and reducing deployment errors by 50% while increasing system scalability and reliability.
- Enhanced monitoring and alerting mechanisms to improve visibility and minimize downtime, resulting in a 40% reduction in downtime and faster resolution of system issues across both insurance and clinical systems.

Environment: Java 21, Spring Cloud, Go, AWS Lambda, EKS, SQS, Schema Registry, Step Function, Event Bridge, VPC, API Gateway, CloudWatch, MongoDB, DynamoDB, IaC (Terraform), Postman, Jira, Git, CI/CD Pipelines, Jenkins, Telemetry, Dynatrace.

Research Engineer | Clark University. | January 2023 – May 2023

- Completed a program-specific project focusing on the development of a single-page web application using Java and Angular, aimed at enhancing the Peer Advising program and SPS student experience.

- Utilized MySQL for efficient storage, retrieval, and management of data within the web application, ensuring seamless user interaction and data integrity.
- Assisted incoming students with course selection and registration, providing personalized guidance on program requirements and elective options through web application.

Senior Software Developer | Elevance Health. **|** *August 2021 – August 2022*
Project: Automation Engine

- Developed and automated the generation of Standard and Modified Plans through API services, significantly reducing manual effort and enhancing efficiency and accuracy.
- Engineered a process automation solution that reduced the time to create plans and contracts to 3 minutes, drastically improving productivity and operational efficiency.
- Led the design and implementation of a microservices architecture using the Spring Boot framework, achieving improved system scalability and flexibility.
- Implemented a Kafka messaging system for real-time data processing and to support an event-driven architecture, thereby enhancing system responsiveness and reliability.
- Utilized the Jolt transformation library for efficient data exchange between microservices, ensuring consistency and compatibility.
- Integrated Eureka for service registration and discovery, facilitating dynamic routing and efficient load balancing across the microservices ecosystem.
- Leveraged PostgreSQL and MongoDB for data storage and retrieval, focusing on ensuring data integrity and system scalability.
- Leveraged Spring Boot's scheduling features, such as @Scheduled annotations, to automate recurring tasks within the application. Configured task scheduling parameters, such as frequency and start time, to meet business requirements.
- Utilized cron expressions to define scheduling patterns for tasks that needed to be executed at specific times or intervals.
- Utilized Postman for extensive testing of RESTful APIs, verifying their reliability and functional performance.
- Developed an Angular single-page web application, focusing on creating a dynamic and responsive user interface. This work significantly enhanced user engagement and experience through interactive design and seamless API integration.
- Managed project workflows and tasks using Jira, streamlining project management processes and enhancing team collaboration.
- Implemented ServiceNow for incident management and ticketing, ensuring the timely resolution of issues and maintaining system reliability.
- Conducted thorough code reviews and provided mentorship to junior team members, fostering a culture of excellence and continuous learning.
- Managed Git repositories and devised effective branching strategies in Bitbucket, ensuring robust version control and code integrity.
- Enhanced development workflows through the integration of Bitbucket with CI/CD pipelines, facilitating automated builds and deployments for efficient continuous integration and delivery.
- Actively participated in troubleshooting and resolving production issues promptly, minimizing downtime and ensuring high availability.

Environment: Java 11, Spring Boot, Microservices Architecture, Kafka, Jolt, Eureka, MongoDB, Angular, Postman, Jira, ServiceNow, Git, Bitbucket, CI/CD Pipelines

Software developer and cloud Engineer | Siemens Technology and Services. **|** *July 2016 – August 2021*
Project: Shared Autonomous Mobility

- Developed a user management service in Java Spring Boot, responsible for handling users, roles, and permissions, ensuring secure and efficient management of user access within the system.
- Led development efforts for SAM, a control center for autonomous fleet operations, enabling efficient management and monitoring of vehicles. Implemented features for importing and exporting GTFS data from agencies, enhancing interoperability and data exchange.
- Utilized ActiveMQ as a Message Broker between all microservices, ensuring reliable communication and message delivery. Integrated with Camunda Modeler to handle Emergency Situations, initiating processes that must be addressed at multiple stages for effective resolution.
- Developed an Event Processor for all assets to store static data and process real-time events, enabling communication to Server-Sent Events (SSE) for real-time updates and notifications.
- Designed and implemented the Notification Service to hold data related to all notifications raised within the system. These notifications indicate issues that dispatchers should address and take corrective actions.
- Integrated assets such as Radars, Lidars, Tccs, Cameras, etc., with ActiveMQ through SSE, providing real-time updates to the dispatcher (controller of vehicles), enhancing situational awareness and decision-making capabilities. Leveraged MongoDB for storing and managing complex data structures, ensuring flexibility and scalability in data storage for SAM's operations.
- Implemented Influx DB for time-series data storage and analysis, enabling efficient handling of telemetry and sensor data from vehicles and infrastructure.

- Demonstrated expertise in agile life-cycle management tools, software testing standards, methods, and conventions, ensuring the delivery of high-quality software solutions that meet customer requirements and industry best practices.
- Integrated Swagger UI into the development process, allowing for real-time visualization and interaction with APIs during development and testing phases.
- Utilized Swagger UI's capabilities to generate interactive API documentation, improving communication and collaboration between development teams and stakeholders.
- Managed Kubernetes deployments using Helm charts, streamlining the installation, configuration, and management of Kubernetes applications.

Project: CRSPNG

- Spearheaded the development of Remote Systems utilizing Java Spring Boot Microservices, ensuring easy scalability and robustness of the backend infrastructure.
- Designed, developed, and owned 5 microservices in Java and Spring Boot, covering functionalities such as file transfer, browsing, Distributed File System (DFS), and Scheduled file transfer.
- Explored various cloud computing concepts in AWS including ECS, EC2, SQS, Lambda, Route53, and RDS to achieve a low-cost highly scalable environment for deployments.
- Deployed Spring Microservices and created GitLab CI/CD pipelines for continuous integration and deployment into AWS ECS instances, ensuring efficient deployment processes and system maintenance.
- Designed and optimized Oracle SQL databases for efficient data storage and retrieval, ensuring adherence to best practices and performance standards.
- Wrote comprehensive JUnit test cases using testing frameworks like Mockito and Power Mockito to ensure the reliability and robustness of the developed software.
- Contributed to the development of the initial integration between the Mind Sphere and CRSP NG platforms, enabling seamless communication and data exchange between systems.
- Developed automated solutions to dynamically scale Java web services deployed in AWS, optimizing resource utilization and ensuring optimal performance during peak loads.
- Engineered automated solutions to provide scalability to the CRSP Platform, supporting over 100k connected devices simultaneously, enhancing system reliability and performance.
- Designed and implemented RESTful APIs in Java Spring Boot for backend services, facilitating data exchange between the UI and server-side components.

Project: Mindsphere

- Collaborated closely with senior developers to reconcile conflicting requirements and resolve complex design issues, ensuring the delivery of high-quality software solutions.
- Investigated and developed skills in new technologies, staying updated with industry trends and best practices to enhance project development.
- Played a key role in developing web APIs using Spring Boot for deploying third-party applications in Cloud Foundry, enabling the utilization of IoT data and notification of customers about important events.
- Integrated JWT (JSON Web Token) authentication mechanism for stateless and secure token-based authentication, enhancing the security and scalability of microservices architecture.
- Deployed applications to Cloud Foundry platform, managing application lifecycle including start, stop, and restart operations via Cloud Foundry CLI or APIs.
- Utilized AWS Redis Cache for caching frequently accessed data, improving application performance and reducing database load.
- Exposed APIs for starting, stopping, and restarting applications, providing seamless management and control over application instances.
- Utilized PuTTY for SSH connections to remote servers, facilitating secure access and administration of Linux-based systems.
- Implemented infrastructure as code using Terraform, automating the provisioning and management of cloud resources across multiple environments.
- Demonstrated interest in infrastructure-as-code (IaC) principles by creating and modifying AWS resources using Terraform scripts, optimizing resource provisioning and management.
- Wrote comprehensive JUnit test cases for unit testing, validating the correctness of code implementations and ensuring robustness and reliability of software components.
- Actively participated in Agile methodologies, including sprint planning, grooming, and retrospective meetings, ensuring alignment with project goals, prioritization of tasks, and continuous improvement of development processes.

Environment: Java 11, Spring Boot, Cloud Foundry, AWS, Terraform, JWT, Redis, AWS Lambda, SQS, SNS, Postman, JUnit, PuTTY, Agile Methodologies, AWS (ECS, EC2, SQS, Lambda, Route53, RDS), GitLab CI/CD, ActiveMQ, Camunda Modeler, Server-Sent Events (SSE), MongoDB, Influx DB, Agile Methodologies, Swagger UI, Kubernetes, Helm Charts.

Education:

Master of Science in Computer Science

Clark University, Worcester, MA. | GPA: 3.6 | Aug 2022 – Dec 2023