**Anil Babu Eagala**

**Professional Summary:**

* More than 11+ years of focused expertise developing Backend and Node.js applications.
* Expert in building scalable, decoupled systems with Spring Cloud, Spring Data JPA, and RESTful APIs, enabling seamless deployment and independent service evolution.
* Proficient in AWS cloud ecosystem including EC2, ECS, Lambda, S3, RDS, CloudFormation, and IAM, with hands-on experience automating infrastructure and CI/CD pipelines for high-availability systems.
* Proven ability to build and deploy highly scalable Node.js microservices in enterprise environments across finance, healthcare, and telecom domains.
* Experience designing and supporting multi-client server systems with complex and performance-critical business logic.
* Demonstrated ability to develop responsive and modular frontend applications using React.js, React Hooks, Redux, and TypeScript for dynamic, cross-platform user interfaces.
* Proven success in leveraging messaging systems such as Kafka and RabbitMQ for asynchronous communication and enhancing service resilience across distributed platforms.
* Adept at implementing secure authentication/authorization mechanisms using OAuth2, JWT, and Spring Security, ensuring robust access control and compliance.
* Strong advocate of Agile practices, TDD, and DevOps culture, with extensive use of tools such as Jenkins, Git, Docker, Kubernetes, and Prometheus for modern software delivery.
* Hands-on experience in building Rich UI and RWD Applications using JavaScript, jQuery, JSON, Angular 16
* React JS, Vue JS, Node JS, Backbone JS, Vue.js, HTML5, CSS, Bootstrap, TypeScript, AJAX, JSTL, PHP, Struts.
* Integrated JIRA with Confluence, Bitbucket, and Jenkins for better workflow automation.
* Hands - on experience in Azure Cloud Services (PaaS & laaS), Storage, Web Apps, Active Directory, Application Insights, Logic Apps, Data Factory, Cosmos DB, Service Bus, Traffic Manager, Azure Monitoring, OMS, Visual Studio Online (VSO), Cognitive Services (LUIS) and SQL Azure.
* Expert in distributed systems architecture, including microservices and micro frontends, designing scalable, modular solutions that enhance application flexibility and maintainability.
* Extensive database administration, including MongoDB, Cassandra, Oracle, SQL Server, MySQL, and Aurora PostgreSQL.
* Proficient in the implementation of frameworks like Struts, Spring (Spring MVC, Spring Web Flow, Spring IOC, Spring AOP, Spring Security, Spring Boot, Spring Data, and Spring Batch), JSF, AJAX frameworks (Rich Faces, My Faces), and ORM frameworks like Hibernate.
* Designed a microservices architecture using Spring Cloud Gateway with WebFlux for efficient API routing.
* Built scalable backend services with Node.js and Express.js, handling asynchronous operations and implementing RESTful APIs for efficient data exchange
* Used Bash and python, included Boto3 to supplement automation provided by Ansible and Terraform for tasks such as encrypting EBS volumes backing AMIs and scheduling Lambda functions for routine AWS tasks.
* Practical knowledge of version control platforms such as Bit Bucket, SVN, GitHub, and Git.
* Adept at using extensive testing frameworks such as JUnit, Mockito, and JBehave, guaranteeing robustness and dependability using automated behavior-driven, integration, and unit testing techniques.
* Competent in using AWS Glue for ETL workflow orchestration and in handling AWS components (EC2, S3 bucket, ECS, ELB, Auto-Scaling, Elastic Search, VPC, EMR, Lambda, RDS, DynamoDB, CloudWatch, and IAM, among others).
* Developed RESTful APIs using Node.js and Express.js to facilitate communication between the frontend and backend, ensuring efficient data exchange and user authentication.
* Knowledge of GCP services such as Compute Engine, Cloud Run, Firebase App, Engine, Datastore.
* Ensured adherence to Agile principles and practices by facilitating Agile ceremonies such as sprint planning, daily stand-ups, sprint reviews, and retrospectives.
* Using the new capabilities added in Java 17+ like lambda expressions, streams, and functional interfaces, we created scalable and reliable Java applications.
* Built a rate-limited WebFlux API with resilience patterns like circuit breakers (Resilience4j) to ensure stability.
* Experienced with web servers that host web applications, such as WebSphere, JBoss, and Apache Tomcat.
* Jenkins, AWS CodeBuild, Fargate, Lambda, and other tools are used to orchestrate deployments.

**TECHNICAL SKILLS:**

* Java/J2EE, Servlets, JSP, JSTL, JDBC, JMS, Kafka, JPA JNDI, EJB, Java Beans, Applets, Multi-threading, Java Networking, Drop wizard and Jetty Frameworks, Spring data, Spring Boot, Spring Framework, Java mail, Google API, Hibernate
* Programming Languages Java 8/11/17, JDK, SQL, PL/SQL, C, C++, PHP, Python
* Application/Web Servers Oracle/BEA Web Logic, IBM Web Sphere, JBoss, Tomcat, Jetty Server, Jenkins, AEM
* Web Technologies JSP, JavaScript, JQuery, AJAX, XML, XSLT, HTML5, DHTML, CSS3, Bootstrap, Angular14, React JS, Express
* XML Tools JAXB, Apache Axis, XML Spy
* Methodology Agile, Waterfall, Scrum, SDLC and UML
* Modeling Tools CVS, Rational clear case, SVN, GIT, Dimensions Testing Junit, Mockito
* Frame Works Spring 5.0, Spring Batch, Spring Security, Spring AOP, MVC,Spring Core, JSF, IBatis, Spring IOC.
* Database Servers Oracle, DB2, SQL Server, MySQL
* Services SOA architecture, RESTFUL/SOAP, Microservices, Axis 2, Apache CXF
* Design Patterns MVC, DAO Pattern, Singleton Pattern, Factory Pattern
* Build Tools ANT, Maven 3.0, Gradle, NPM, Grunt, Gulp, Jenkins
* IDE/Tools JIRA, Eclipse, Net beans, ANT build script, Maven build
* Cloud Azure Cloud, GCP, Amazon Web services (AWS) Cloud, Pivotal Cloud Foundry (PCF)

**Professional Experience**

**Client: Bank of America – Pittsburgh, PA. Apr 2024 to Present**

**Role: Sr. Node JS Developer**

**Responsibilities**:

* Designed and implemented enterprise-grade microservices using Spring Boot and Spring Cloud, enabling scalable and independent deployments across business domains.
* Designed and developed high-performance microservices using Node.js and Express.js, orchestrated through REST APIs for scalable financial transaction processing.
* Integrated OAuth2 and JWT-based authentication within Node.js services to ensure secure access and role-based control across banking platforms.
* Architected microservices hosted on AWS (EKS, Lambda, S3, CloudWatch, Route 53), ensuring auto-scalability and real-time monitoring during peak loads.
* Built and maintained multi-client compatible APIs that served both internal dashboards and external partner applications.
* Collaborated with stakeholders to gather business requirements and translate them into scalable Node.js-based solutions, ensuring alignment with enterprise architecture.
* Used Kafka and RabbitMQ to implement asynchronous communication between Node.js services, supporting distributed workflows and event-driven systems.
* Led the development of technical design documents and diagrams for API architecture and backend service interactions.
* Integrated Spring Security with OAuth2 and JWT for secure authentication and authorization in REST APIs.
* Developed device-agnostic, responsive UI modules using React.js, improving customer experience on mobile and desktop platforms.
* Automated CI/CD pipelines using Jenkins and AWS services like EKS, Lambda, and S3, achieving zero-downtime deployments.
* Employed Spring Data JPA for database interactions and optimized backend throughput across PostgreSQL and MongoDB.
* Used AWS CloudWatch, Route 53, and Auto Scaling groups to ensure application availability during peak financial activity.
* Developed dynamic UI interfaces and backend logic using a tech stack including AngularJS, Spring Boot, JavaScript, HTML5, CSS3, jQuery, and Bootstrap to deliver full-stack web solutions.
* Managed and optimized a data lake built on Hadoop, integrating varied data types (structured and unstructured) to support organization-wide analytics and reporting initiatives.
* Proficient in fundamental Java programming principles, including object-oriented programming, multithreading, collections API, exception handling, I/O operations, JDBC, and use of generics.
* Engineered a device-agnostic, single-page application using React.js to deliver an isomorphic interface that adjusts seamlessly across desktops, tablets, and smartphones.
* Followed a test-first approach by applying TDD principles, crafting unit tests prior to produce maintainable and low-defect code.
* Configured GitLab CI pipelines to build Docker images and automate deployments to GKE clusters, supporting seamless updates with zero service interruption.
* Embedded Apache Spark within backend services to deliver customer behavior insights and personalized engagement metrics.
* Implemented secure access control using OAuth2 protocols, JWT tokens, and integrated Spring Security to enforce fine-grained permissions.
* Employed Spring Data JPA for streamlined repository patterns, accelerating interaction with relational databases and enhancing overall data throughput.
* Created modular RESTful services, enabling cleaner interfaces and easing future enhancements for core platform services.
* Used JUnit and Mockito to lead test-driven cycles that validated backend logic, contributing to resilient and predictable deployments.
* Authored and optimized Apache Spark code in Scala for real-time transaction analysis and marketing performance tracking.
* Delivered JSON payloads via Spring Data REST to front-end systems, serving real-time information to user interfaces.
* Leveraged Hibernate for ORM, mapping Java objects to database schemas and ensuring optimal CRUD operations for data-heavy applications.
* Processed vast volumes of retail data using HDFS and Hive queries, feeding forecasting models for demand prediction accuracy.
* Deployed Spring Boot services packaged with embedded Tomcat, simplifying environment setups and accelerating deployment turnaround.
* Integrated TDD scenarios within Jenkins pipelines to ensure early detection of API failures and uphold software quality in agile release environments.
* Delivered end-to-end Generative AI workflows with Kubeflow and AWS SageMaker, covering data transformation, model training, and live inference.
* Engineered database schemas for structured (SQL) and unstructured (NoSQL) environments, enabling efficient scaling and data organization.
* Executed AWS-based deployment strategies for core financial services using EC2, EKS, Lambda, S3, and Route53 for high availability.
* Developed Python tools to automate AWS operations such as backups and recovery using services like Lambda, S3, and DynamoDB.
* Wrote Apache Spark batch processes to handle daily pricing and stock movement data across a distributed retail network.
* Built Python-based ETL flows to process transactional data in near real time, feeding analytical dashboards and reporting tools.
* Built end-to-end CD pipelines with Docker, Kubernetes, Jenkins, and config management tools like Ansible, Puppet, and Chef to streamline delivery.

**Environment**: Java, HTML5, CSS3, JavaScript, Bootstrap, Backbone, JMS, Cassandra, AGILE/Scrum, AngularJS, Bootstrap, JavaScript, SPA, Spring Boot, Spring MVC, AngularJS, RESTful web services, HTTP, Microservices, ReactJS, PostgreSQL, MongoDB, NodeJS, Cucumber, TestNG, NoSQL, AWS IAM, SQS, SNS, MySQL, Cassandra, Docker, Kubernetes, Eclipse, Jira, JSON, Rabbit MQ, Kafka, Junit, Mockito, Bitbucket, Git, Gradle.

**Client: HCA Healthcare – Nashville, TN. Mar 2022 to March 2024**

**Role: Node JS Developer**

**Responsibilities**

* Developed and maintained HIPAA-compliant Spring Boot microservices for health record processing and predictive analytics workflows.
* Developed Node.js microservices with Express.js for processing patient health records, adhering to HIPAA compliance standards.
* Integrated AWS IAM, EC2, S3, SNS, and CloudWatch for secure and observable cloud-native microservices.
* Built and deployed Dockerized Node.js apps in Kubernetes (Azure AKS) using Helm for scalable, production-grade deployments.
* Designed robust auth flows using OAuth2 and token-based auth, enabling secure access to EHR APIs and internal healthcare apps.
* Architected and deployed multi-tenant services using Node.js to serve different care teams and business units with customized logic paths.
* Created performance-tuned services that reduced response time by 30% through Node.js async optimization and Redis-based caching.
* Collaborated with business analysts and product owners to align microservice architecture with predictive analytics and EHR integration requirements
* Integrated Kafka for asynchronous communication across microservices to handle real-time claims processing and EHR integration.
* Created dynamic React UI components for healthcare dashboards, enabling real-time patient monitoring and segmentation.
* Configured AWS IAM, S3, SNS, and EC2 for secure and scalable cloud-native deployments.
* Containerized applications using Docker and managed microservices deployment via Kubernetes Helm charts and Azure DevOps pipelines.
* Participated across all phases of the Agile SDLC lifecycle, contributing to sprint planning, requirement analysis, implementation, and quality assurance using Scrum.
* Employed PySpark within the Apache Spark ecosystem to preprocess claims and EHR datasets for model development and scoring.
* Delivered scalable data-centric microservices using Spring Boot with built-in servlet containers to simplify deployment and lifecycle management.
* Engineered enterprise-grade applications with J2EE stack including JSP, Servlets, and JDBC to maintain persistent data exchange and back-end integration.
* Leveraged Spring framework for IoC and managed ORM with Hibernate to streamline transactional data access layers.
* Developed secure RESTful Python APIs using FastAPI for user profile workflows while maintaining HIPAA compliance via audit logs and role-based access controls.
* Practiced unit-first development using PyTest and unittest to ensure robust testing of domain logic and NLP-driven systems.
* Containerized service workloads with Docker to maintain consistent runtime environments and facilitate cross-platform deployment.
* Utilized Kubernetes Helm templates to standardize deployment configurations across microservices in container orchestration environments.
* Automated ML lifecycle stages using MLflow and Airflow, enabling reproducible medical NLP model training and deployment.
* Reduced data latency using in-memory caches such as Redis and EhCache within Spring Boot ecosystems.
* Designed Spark-based processing workflows to calculate care coordination scores and enable patient segmentation.
* Implemented user interface components using Spring MVC, integrated with AngularJS, JSP, JSTL, and AJAX for a dynamic web experience.
* Scripted lightweight agents in Python to collect logs and metrics, forwarding telemetry data to Prometheus and Grafana dashboards.
* Used Hibernate’s on-demand data retrieval techniques (lazy loading) to optimize backend resource utilization.
* Standardized front-end architecture using reusable React components, focusing on maintainability and performance.
* Managed secure archival of clinical records using Hadoop HDFS and Hive, in compliance with long-term retention mandates.
* Automated REST and SOAP API testing using Postman, SoapUI, and JUnit to reduce QA cycles and improve test coverage.
* Configured AWS IAM to enforce granular access control policies across cloud services and development accounts.
* Integrated SNS and SQS for decoupled communication across distributed service layers, improving system fault tolerance.
* Created reactive listeners and stored procs in Azure Cosmos DB, handling event-based data processing and schema updates.
* Built Kafka-based messaging infrastructure to facilitate real-time data publishing and consumption across services.
* Streamlined event streaming using Spring Cloud Stream abstractions, reducing Kafka boilerplate and improving reliability.
* Managed continuous integration pipelines using GitHub and Jenkins, running JUnit and PyTest test suites to gate deployments.

**Environment**: Java, J2EE, HTML, JSP, JDK, CSS, JavaScript, jQuery, JSP, AJAX, React.js, Spring, Hibernate, JPA, JMS, REST Webservices, JAX-RS, Jackson, ReactJS, NodeJS, Microservices, EC2, S3, ELB, Security Groups, IAM, AWS EC2, Cloud Watch, Oracle, Mockito, JUnit, GIT, Maven, Jenkins, Jira.

**Client: Charter Communications – Stamford, CT. Feb 2021 - Dec 2021**

**Role: Java Backend Developer**

**Responsibilities:**

* Built RESTful Spring Boot APIs with Spring Cloud Gateway, enabling consistent routing, load balancing, and service discovery.
* Applied distributed tracing with Spring Sleuth and Zipkin, improving debugging and latency tracking in microservices.
* Developed performant ReactJS and Node.js interfaces with Redux for managing complex state transitions in telecom workflows.
* Developed Node.js services integrated with ReactJS frontends to manage high-volume telecom workflows and customer service interactions.
* Designed backend workflows and microservice orchestration combining Node.js and Java Spring Boot, enhancing inter-service communication.
* Deployed Node.js APIs on AWS EC2 and monitored logs and metrics via AWS CloudWatch, improving response to production issues.
* Created dynamic and responsive UIs for telecom dashboards that interacted with Node.js APIs for real-time event processing.
* Built middleware logic in Node.js to support business rule evaluation and multi-client support with JSON-based payload handling.
* Used AWS services including EC2, CloudWatch, and IAM for cloud infrastructure, and automated deployment pipelines via Jenkins.
* Crafted interactive front-end components using AngularJS, JavaScript, HTML5, CSS3, and jQuery to deliver dynamic and responsive web applications.
* Engineered backend logic and workflow orchestration using Java with Spring Framework and Hibernate ORM to facilitate complex business operations and persistent storage.
* Utilized Spring Boot Actuator for real-time insights into service metrics, endpoints, and health checks, enhancing observability of microservices in distributed systems.
* Integrated distributed tracing tools such as Zipkin and Spring Cloud Sleuth for end-to-end request tracking across services, significantly improving debugging efficiency.
* Developed RESTful web services with JAX-RS and JSON payloads for smooth integration between UI and backend APIs.
* Built scalable microservice-based applications with Spring Cloud technologies, employing tools such as Eureka for service discovery and centralized configuration.
* Designed and executed privacy-preserving ML workflows on Google Cloud (Vertex AI), applying differential privacy to ensure secure, compliant analytics on clinical narratives.
* Implemented MVC architecture using Struts and Spring for modular Java web development, incorporating JSPs and Servlets for dynamic UI rendering.
* Created intuitive admin and user dashboards using JSF, JSP, AJAX, JavaScript, Spring, and Hibernate, offering smooth interaction with enterprise data systems.

**Environment**: JDK 1.8, PostgreSQL, MongoDB, RabbitMQ, IBM MQ, React JS, Spring Boot, Spring JPA, Jenkins, GitHub, AWS Infrastructure, Chef, Angular Framework, Node JS, Drools, Groovy, Logstash, Elastic search, Jira, JUNIT, Maven, and Cucumber.

**Client: Mastek – Mumbai, India. Oct 2017 - Jul 2020**

**Role: Java Developer**

**Responsibilities:**

* Developed a strong framework to manage billing, support, service requests, and client accounts, guaranteeing smooth service delivery.
* Enhanced user experience and engagement by leveraging React to create an interactive and responsive user interface.
* Spring MVC, Spring Cloud, Spring Boot, and Java 8 guaranteed the back-end system's scalability and maintainability.
* Ensured modularity by implementing a microservices architecture to administer web services and RESTful APIs effectively.
* Developed asynchronous communication applications using SOAP Web services, IBM WebSphere, Apache CXF, BEA WebLogic, and JMS.
* Ping Access Proxy Gateway, OAuth, JWT, and Spring Security were used to safeguard client information and guarantee safe transactions.
* JUnit, Mockito, and Spring Integration tests provided thorough testing frameworks that guaranteed system security and dependability.
* Developed Python automated test scripts to verify Verizon's microservices' performance and functionality, guaranteeing high-quality releases and cutting down on manual testing time..
* Connected Go-based services to message brokers like RabbitMQ, Kafka, or others to enable asynchronous communication between dispersed systems.
* Designed and developed GraphQL and REST APIs with Go that are incredibly efficient, guaranteeing quick response times and seamless front-end and other service interaction.
* Developed strong Rabbit MQ message handling and processing methods that guarantee dependable data sharing in the microservices architecture of T-Mobile.
* Associated with AWS Lambda services and several models, including CloudFormation, Serverless Deployment, ECS, API Gateway, AWS CLI, and Docker.
* Elastic load balancer, RDS, EBS, EC2, S3, auto-scaling groups, and AWSSDK were used to connect to Amazon SQS for bulk email processing.
* Developed SQL scripts to combine data from many sources into centralized databases, guaranteeing data availability and consistency for CRM and billing systems.
* Supported collaborative development using GitHub for version control and Maven to manage the build and deployment processes.
* Applied Scrum and Agile approaches to improve teamwork and productivity, guaranteeing on-time delivery of high-caliber software.

**Environment**: Java, Spring Boot, Swift 5, Hibernate, REST, JUnit, AJAX, REST Web services, Bootstrap, ReactJS, Redis, Maven, GIT, SVN, Docker, Kubernetes, Cloud, and AWS. Java, J2EE, HTML, CSS, Spring, Servlets, JSP, EJB, XML, JavaScript, Ant, SQL, Oracle, Hibernate, Log4j, WebSphere, UNIX, SOAP, XML, SAX, DOM, JMS, Bugzilla, JSTL, Ajax, CSS, jQuery, Spring ORM, GIT, PL/SQL.

**Client: Cyient – Hyderabad, India. Jul 2014 – Sep 2017**

**Role: Java /J2EE Developer**

**Responsibilities:**

* Participated in the Software Development Life Cycle (SDLC) phases, including design, development, analysis, modelling, and requirements collecting.
* Test Driven Development (TDD) was used in the application's design.
* The project made considerable use of Java/J2EE Design Patterns, including Business Delegate, Session Facade, and Singleton, enabling efficient allocation of roles and responsibilities among many processing layers.
* Developed multiple modules utilizing technologies such as HTML5, AJAX, JSON, CSS Media Queries, and JavaScript. Written maps reduce programs that load data into the MongoDB environment.
* Developed server-side applications with Servlets, JSP, JDBC, WSDL, Java, SOAP Web services, ANT, and AJAX.
* Proficiency in frameworks and tools for module development, including Eclipse, JMS, Jersey, and JBOSS.
* Developed asynchronous communication applications using SOAP Web services, IBM WebSphere, Apache CXF, BEA WebLogic, and JMS.
* Agular’s component-based architecture was used to create scalable and reusable components, enhancing the modularity and maintainability of the code.
* Created scalable server-side apps using Node.js that effectively managed middleware, routing, and HTTP requests.
* Effective data management was achieved by integrating Node.js apps with databases like MongoDB, PostgreSQL, or MySQL using ORMs like Mongoose or Sequelize.
* Express.js was used to build RESTful APIs, supporting different HTTP methods, middleware, and routes to provide reliable backend services.
* The Spring-AOP module implemented features like logging and user session validation.
* Developed the front end of an application using HTML, JSP, Ajax/JQuery, Spring Web Flow, XHTML, DHTML, and CSS.
* Spring Web Services (SOAP, WSDL) was used to develop server-side services.
* Web services, WSDL, UDDI, SOAP, SAX parser, and XML were used to transfer data across applications.
* Developed stored procedures in PL/SQL and used HQL considerably.
* Made use of Maven as a build tool and Rational ClearCase as a version control tool.
* Integrated databases (SQL, NoSQL) with Go to ensure excellent performance and effectively handle large-scale data queries.
* RESTful APIs and gRPC services were designed and deployed using Go to provide quick, low-latency communication.
* Developed web applications with SOAP, XML, and WSDL for routing, firewall, callback security, and session services.

**Environment**: Java, JSP, HTML, CSS, SQL, XML, WSDL, UDDI, SAX, EHC, MongoDB, MySQL, Spring Web Flow, SOAP, WSDL, Node,js, AJAX, JDBC, GIT, SVN, Docker, Kubernetes, Cloud, and AWS, Oracle, Spring Web Flow, Ajax/JQuery,

**Education**:

* IWU, master’s in information technology Indianapolis Aug 2020 –Dec 2021
* SRM University, bachelor’s in computer science AP, India June 2010 – May 2014

**Certifications**

* Oracle Certified Professional: Java SE 17 Developer
* AWS Certified Developer – Associate