**Sriram Alokam**
**Java Developer**
**806-283-6165**
**srirama032000@gmail.com**
**Linkedin**
**Website**
**AWS Developer Associate**
**Github**


## PROFESSIONAL SUMMARY:

- Holds a Master of Science degree in Computer Science with experience in developing practical and time-saving software solutions using Java and web technologies.
- Collaborated on diverse projects involving UNIX-based systems, showcasing proficiency in managing and optimizing system performance.
- Developed responsive and user-friendly front-end interfaces using HTML, JavaScript, jQuery, CSS, and ReactJS, leading to an increase in user interactions.
- Worked on Core Java concepts such as OOP Concepts, Collections Framework, Exception Handling, I/O Systems, Multi-Threading, JDBC, Generics.
- Developed microservices with Spring and tested the application using Spring Boot.
- Used Spring Core annotations for Spring Dependency Injection, Spring MVC for Rest API, and Spring Boot for microservices
- Demonstrated proficiency in managing relational databases, including PostgreSQL and MySQL.
- Applied expertise in database schema design, normalisation, indexing techniques, and other practices to ensure data integrity and maximise query performance.
- Exhibited strong programming skills in designing and implementing multi-tier applications using web-based technologies such as Spring MVC and Spring Boot.
- Spearheaded the integration of Apache Kafka into the data processing pipeline, significantly improving real-time data streaming capabilities.
- Proficiently harnessed core AWS services, including Amazon EC2 for scalable virtual servers, Amazon S3 for efficient object storage, and AWS Lambda for serverless computing.
- Adeptly managed version control using GIT, conducted API testing with Postman.

## TECHNICAL SKILLS:

**languages:**
Java, PL/SQL, Python, C++, C#.

**Java Technologies and Frameworks:**
JSP, JMS, Servlets, Spring Microservices, Spring Boot, Spring MVC, Hibernate, Spring, JUnit, Mockito

**Application Servers:**
Apache Tomcat, WebLogic, WebSphere, JBoss

**Message Brokers:**
Kafka

**Databases:**
Oracle, SQL Server, PL/SQL, and My SQL

**Operating Systems:**
Windows, Linux/UNIX, MacOS

**Web Technologies:**
HTML, CSS, JavaScript, XML, React JS, REST API, JSON, Typescript, jQuery

**Tools and Technologies:**
Postman, Jira

**IDE's:**
Eclipse, NetBeans, Spring

**Methodologies:**
SDLC, Agile (SCRUM)

**Education:**

- Master of Science in Computer Science from Texas Tech University, Lubbock TX in 05/2023.
- Bachelor of Science in Computer Science from Acharya Nagarjuna University, Guntur, India in 04/2021.

**Professional Work Experience:**

**Cognizant, Hyderabad, India**
**Java Intern Trainee**
**01/2021 – 06/2021**

**Contribution:**

- Significantly improved user engagement by developing responsive and user-friendly front-end interfaces using HTML, JavaScript, jQuery, CSS, and ReactJS, leading to an increase in user interactions.
- Enhanced application functionality by designing and developing web interfaces and business logic using Spring, JSP, Servlets, Spring JDBC, JavaScript, HTML, and XML technologies.
- Experience in Core Java concepts such as OOP Concepts, Collections Framework, Exception Handling, I/O Systems, Multi-Threading, JDBC, Generics
- Used Java8 features in developing the code like Lambda expressions, creating resource classes, fetching documents from the database
- Used design patterns like Singleton, Data Access Objects, Factory, and MVC patterns
- Collaborated on diverse projects involving UNIX-based systems, showcasing proficiency in managing and optimizing system performance.
- Developed microservices with Spring and tested the application using Spring Boot
- Used Spring Core annotations for Spring Dependency Injection, Spring MVC for Rest API, and Spring Boot for microservices
- Development of cloud-hosted web applications and REST APIs using Spring Boot with embedded Tomcat
- Wrote build scripts using Maven.
- Successfully established robust communication between frontend and backend components by proficiently creating RESTful APIs using Spring Boot, reducing API response times by 40%.

- Seamlessly integrated third-party APIs into applications using Postman, streamlining data exchange and enhancing functionality, resulting in a 15% reduction in development time.
- Spearheaded the design of serverless architectures using AWS Lambda and maximized the utilization of Amazon S3 for asset storage, leading to a 30% improvement in resource management and cost efficiency.

**Personal Projects:**

---

**Microservices With Spring Boot:**

- Designed and implemented a microservices architecture to modularize the application.
- Leveraged RESTful APIs for communication between microservices.
- Implemented various microservices to handle specific business functionalities.
- Ensured loose coupling and high cohesion among microservices.
- Established communication between microservices using Spring Cloud or other relevant technologies.
- Utilized messaging systems like Apache Kafka for asynchronous communication.
- Employed appropriate databases for microservices, such as relational databases.
- Ensured data consistency and integrity across microservices.
- Implemented security measures, including authentication and authorization mechanisms. Secured communication channels between microservices.

**Spring Boot Security(JWT Token based authentication) :**

- Utilized Spring Boot framework for backend development. Implemented JWT for secure authentication and authorization.
- Designed a robust authentication flow using JWT tokens for user validation.
- Integrated JWT token generation during user authentication. Included necessary user details and roles in the JWT payload.
- Stored JWT tokens securely, ensuring protection against tampering. Validated JWT tokens during subsequent user requests to ensure authentication.
- Implemented RBAC using JWT roles to control user access to different resources. Defined roles for different user types (e.g., admin, regular user).
- Incorporated security measures to protect against common vulnerabilities (e.g., CSRF attacks, session hijacking).
- Ensured password security by implementing encryption techniques (e.g., bcrypt) for storing user passwords.
- Implemented robust exception handling mechanisms to handle authentication errors gracefully.