# Aravind Vallamkonda

aravindvallam1@gmail.com | (571) 307-8996| LinkedIn | Seattle, WA.

## SUMMARY

Versatile Software Engineer with 3+ years of extensive hands-on design and development experience building enterprise-grade applications using Java, Spring Boot microservices, React.js,  Go and AWS Cloud technologies. Proven expertise in developing scalable backend services with heavy Java implementation, creating responsive React-based user interfaces, and deploying cloud-native solutions on AWS infrastructure. Strong background in full-stack development with demonstrated ability to architect distributed systems, implement RESTful and GraphQL APIs, optimize database performance, and deliver production-ready applications using Docker, Kubernetes, and CI/CD pipelines. Experienced in Agile methodologies with a track record of reducing system latency by 70%, increasing throughput by 150%, and delivering enterprise solutions for banking and financial services clients.

## TECHNICAL SKILLS

**Programming:**  Java, Go, Python, JavaScript, TypeScript, SQL, C
**Backend Frameworks:** Spring Boot, Spring MVC, Spring Security, Hibernate, Gin (Go), Express.js
**Frontend Technologies:** React.js, Next.js, HTML5, CSS3, Tailwind CSS, Redux, Bootstrap
**API & Web Technologies:** RESTful APIs, GraphQL, JWT, OAuth2, WebSockets, OpenAPI, Swagger
**Cloud & DevOps:** AWS (EC2, S3, RDS, Lambda, ECS, EKS), Docker, Kubernetes, Jenkins, GitHub Actions, Terraform
**Messaging & Event Streaming:** Apache Kafka, RabbitMQ, Amazon SQS
**Databases & Storage:** PostgreSQL, MySQL, Oracle, DynamoDB, MongoDB, Redis, Memcached
**Testing & Monitoring:** JUnit, Mockito, Postman, Cypress, Selenium, Prometheus, Grafana, ELK Stack, Sentry
**CI/CD & Version Control:** Git, GitHub, GitLab, Bitbucket, Jenkins, CircleCI
**Tools & Environments:** VS Code, IntelliJ IDEA, Jira, Confluence, Figma, Postman, Notion
**Development Practices:** Full-stack development, CI/CD, Automation, TDD, Design Patterns, Agile, SDLC, SOLID, and Code Reviews

## PROFESSIONAL EXPERIENCE

### Software Development  Engineer                                                                                            Mar '25 - Jul '25
*Karios* | *Ashburn, VA*

- Encountered deployment instability across FreeBSD-based systems, so developed a secure backend in Go using the Gin framework to orchestrate signed artifact distribution with multi-stage approvals.
- Rollback execution lacked reliability post-update, prompting the integration of ZFS snapshots and bectl automation to guarantee boot-level restoration in critical node environments.
- Public API deployment lacked a scalable and secure hosting environment, so architected and provisioned a comprehensive AWS infrastructure using EC2 instances, Application Load Balancers, and API Gateway to deploy public-facing APIs with SSL termination, enabling secure external client access and supporting high-availability requirements for production workloads.
- Artifact distribution and version management across distributed systems required centralized storage, so established a public repository artifactory using AWS S3 with CloudFront CDN integration, implementing versioned artifact storage and global content distribution that reduced download latency by 40% and provided reliable access to deployment packages for remote datacenter nodes.
- Version tracking across devices was inconsistent, so PostgreSQL-backed metadata storage was implemented with RESTful APIs for real-time status synchronization and update verification.
- Workflow approval required transparency across teams, leading to the creation of a React.js-based admin portal supporting update audit trails and controlled rollout flows.
- Datacenter infrastructure monitoring lacked centralized oversight, so architected and developed a comprehensive React-based Control Center for datacenter management featuring role-based access control, real-time server monitoring dashboards, and resource allocation interfaces enabling operations teams to manage multiple datacenters efficiently.
- Security gaps during remote deployments were mitigated by integrating artifact hash validation and signature verification, ensuring integrity in system-level changes.
- Regression testing delays hindered CI, so integrated Postman test collections within GitLab CI pipelines, reducing QA cycle time and improving release confidence.
- Lack of observability in live environments was resolved by building custom health endpoints and structured logging to track deployment health and rollback triggers.

### Associate Application Developer                                                                                            Nov '20 - Dec '22
*National Payments Corporation Of India* | *Hyderabad, India*

- System bottlenecks in toll processing affected response time, so re-architected the monolithic app into modular Spring Boot microservices, reducing latency by 70%.
- High transaction load caused inconsistencies, so built Kafka-based event pipelines and introduced Redis caching to ensure 150% increase in real-time processing throughput.
- Internal teams lacked efficient tools for business operations management, so developed a comprehensive React.js-based admin portal enabling internal business and operations teams to monitor transaction flows, manage system configurations, and generate operational reports with intuitive user interfaces and responsive design.

- Banks required streamlined dispute resolution processes, so created a dedicated React.js bank portal allowing financial institutions to submit, track, and resolve payment disputes while providing secure access to download transaction records and compliance documentation through role-based authentication.
- Monitoring was reactive and delayed, so implemented Grafana and Prometheus dashboards to visualize service health, lag metrics, and memory utilization.
- Manual bank reconciliation created SLA breaches, so scheduled automated batch jobs that generated compliance-ready financial reports on a daily cycle.
- Issue tracing across logs was tedious, so deployed the ELK Stack for centralized logging, enabling faster debugging and incident response across distributed services.
- Unauthorized access and audit gaps required stricter controls, so implemented role-based access in admin and bank portals with Spring MVC and session tracking.
- Downtime during peak hours impacted reliability, so used heap profiling to detect memory leaks, tuned thread pools, and isolated failures using bulkhead patterns.

### *Software Development Engineer*        **Feb '20 - Oct '20**
*Cognizant Technology Services* | *Bengaluru India*
- Onboarding progress tracking was manual and error-prone, so developed Spring MVC backend modules that automated status updates and streamlined workflow execution.
- Lacked secure provisioning APIs, so created REST endpoints for access management, file handling, and onboarding validations integrated with Salesforce and HRMS.
- High DB latency impacted user experience, so optimized Hibernate queries and tuned connection pool settings, reducing API response times by 20%.
- Agile visibility needed better task clarity, so engaged in daily scrums and retrospectives, using Jira for backlog grooming and sprint planning alignment

## EDUCATION

**Masters in Computer Science - George Mason University** | 3.93/4.0        **Jan '23 - Dec '24**
**Bachelors in Information Technology - Amrita School Of Engineering** | 8.0/10        **Jul '16 - May '20**
**Relevant Coursework:** Distributed Systems, Operating Systems, Networking Fundamentals, Database Management, Virtualisation

## Certifications

- **AWS Certified Developer – Associate**
- **Confluent: Developer Skills for Apache Kafka**