

John Ojo

Software Development Engineer II · Amazon

Dallas, TX hello@johnujo.me [linkedin.com/in/john-ojo-573717217](https://www.linkedin.com/in/john-ojo-573717217) github.com/ojo-john
johnujo.me

EXPERIENCE

Amazon · Software Development Engineer II (Promoted to SDE II Oct 2025)

Jul 2023 - Present · Dallas, TX

Platform Migration · Technical Lead · Oct 2025 - Present

Led full-stack migration of a benefits platform (4M+ records, ~200 TPS peak, P99 latency under 55 ms, 100K+ users) from a monolithic shared pipeline into a standalone React application with independent releases.

Reduced the release cycle from 4–5 days to 1–2 days and rollback time from hours to under 30 minutes.

Built an aggregated API consolidating 4 microservices into a single response, eliminating an N+1 call pattern (8+ round-trips to 1) and reducing frontend latency by over 75%.

Designed a dual-auth reverse proxy (SigV4 + token-based) with API Gateway, replacing an intermediary service layer for all GraphQL/REST operations.

Built an automated data validation pipeline (S3, SQS, Lambda) comparing legacy and new endpoints field by field across 4M+ records to guarantee zero data loss during cutover.

Implemented a phased rollout with independent feature flags and wave-based targeting (45 to 100K+ users), with full observability via custom telemetry, CloudWatch dashboards and Athena queries.

Created a Cypress E2E suite (80 automated tests, 134-case QA plan) and authored onboarding documentation adopted as the team standard for new engineers.

Service Modernization · Jul 2023 - Oct 2025

Migrated a high-throughput benefits service (4M+ DynamoDB records) to ECS Fargate (CDK), reducing support tickets 5%+ year on year and eliminating OS patching overhead.

Resolved a production-blocking 403 auth failure by tracing it to API Gateway dropping SigV4 headers, and redesigned the networking (NLB to ALB) as a permanent fix.

Built a serverless CMS (Lambda, Step Functions, DynamoDB) that reduced engineering dependency for content updates by 80% and time-to-market from days to hours, saving 3 developer-months a year.

Developed a geospatial clinic search API (AWS Location Service) with a cache-on-first-visit strategy on Fargate, helping employees across 10+ clinic locations find the nearest care in under 2 seconds.

Built local development stack infrastructure (CDK, Docker) that cut the test cycle from 1.5 hours to 10 minutes for 8 engineers, and led its adoption across the team.

Pioneered chaos engineering with AWS Fault Injection Simulator, organised and led live gameday exercises, and authored adoption documentation now used org-wide.

Migrated containers to ARM64/Graviton for ~20% compute cost reduction and implemented CI/CD with alarm-triggered automated rollback.

Amazon · Software Engineering Intern (Two internships on Supply Chain Optimization, building inventory sourcing tooling used by operations analysts.)

Jun 2021 & May 2022 · Seattle & Bellevue, WA

Second internship · Service migration · May - Aug 2022

Migrated a legacy service onto native AWS, using Lambda with API Gateway as the compute platform.

Created and delegated the service's DNS hosted zone.

Ran end-to-end testing before onboarding internal and external clients.

First internship · FAQ chatbot · Jun - Aug 2021

Built a chatbot with Amazon Lex and QnABot that answered common customer questions automatically, reducing the load on the people answering them.

Built the chat interface with the Lex Web UI and Angular so users could talk to the bot.

TECHNICAL SKILLS

Languages Java, TypeScript, JavaScript, Kotlin

AWS ECS Fargate, Lambda, Step Functions, DynamoDB, API Gateway, NLB / ALB, SQS, S3, CloudWatch, CloudFormation, IAM, Location Service, Fault Injection Simulator, Athena, Glue

Tools & frameworks React, GraphQL, gRPC, Cypress, Docker, CDK

Practices Distributed systems, Microservices, CI/CD, Feature flags, Chaos engineering, Event-driven architecture, Observability, Auth (SigV4, TLS, RBAC)

EDUCATION

University of Texas at Dallas

B.S. Computer Science · 2019 - 2023