

ASHAYE ADETOMIWA TEMILOUWA

Email: ashayeadetomiwaashaye@gmail.com || GitHub: [Thormie-Harshey](https://github.com/Thormie-Harshey) || LinkedIn: [tomiwa-ashaye](https://www.linkedin.com/in/tomiwa-ashaye)
Website: [Website](#)

PROFESSIONAL SUMMARY

Results-driven DevOps / Cloud Engineer with hands-on experience designing secure, scalable AWS and Azure infrastructure, automating end-to-end CI/CD pipelines, and managing containerised workloads with Kubernetes and Docker. Proven ability to apply Infrastructure as Code (Terraform) and GitOps principles to eliminate manual toil and enforce deployment.

Hands-on experience with AI-assisted engineering workflows including agentic Claude Code frameworks to help teams accelerate infrastructure delivery and reduce operational overhead.

TECHNICAL SKILLS

Cloud Platforms	AWS & Azure
IaC & GitOps	Terraform · GitLab CI/CD · GitHub Actions · Jenkins · ArgoCD
Containers & Orchestration	Docker · Kubernetes · AWS ECS
Observability	CloudWatch (metrics, alarms, logs) · Prometheus · Grafana · Zipkin · Distributed Tracing · Fluentbit · Loki
Security & Compliance	IAM · OIDC · IRSA · External Secrets Operator · Security Group chaining · AWS Secrets Manager · EC2 Instance Connect
Configuration Mgmt	Ansible · Helm
Networking	VPC · CIDR/subnetting · NAT Gateway · ALB · Route tables · Route 53
Scripting	Bash · Groovy (Jenkinsfile)
Version Control	Git
Databases	MySQL · AWS RDS · DynamoDB · HSQLDB
AI-Assisted Engineering	Claude Code (agentic workflows) · Model Context Protocol (MCP) · Multi-model Subagent orchestration · Hooks

PROFESSIONAL EXPERIENCE

The Cloud Advisory

January 2026 – Present

DevOps Engineering Intern

- **Architected** an end-to-end Agentic DevOps framework using Claude Code that automated AWS infrastructure provisioning from spec to deployment, cutting manual configuration effort by ~80% and enabling repeatable, auditable delivery.
- **Engineered** a custom "SAY-DO-LOG" governance layer with Bash safety hooks that intercept and validate all AI-generated infrastructure actions before execution, achieving 100% policy compliance across deployments.
- **Built** a modular Skills library (.claude/skills) that standardised Terraform scaffolding, resource planning, and least-privilege tool permission enforcement (Read/Write/Bash), reducing new-environment setup time significantly.
- **Designed** a multi-model Subagent pipeline — Claude Opus for complex Terraform generation, Sonnet/Haiku for high-speed security auditing and cost optimization, enabling parallel workstream execution and faster review cycles.
- **Integrated** Model Context Protocol (MCP) servers to give agents real-time access to live AWS APIs and Terraform provider docs, closing the knowledge-cutoff gap that typically limits AI-generated IaC accuracy.

- **Enforced** code quality gates in SonarQube (>80% coverage, zero critical vulnerabilities), reducing production deployment risk by ~30% and improving overall release confidence.
- **Redesigned** Docker build configurations using multi-stage builds, reducing image size by ~20% and cutting average CI build times from 4.5 minutes to 3.5 minutes — a 22% improvement in pipeline throughput.
- **Configured** automated SonarQube scanning for Node.js applications within CI pipelines, providing continuous visibility into bugs, security vulnerabilities, and code smells before merge.
- **Deployed** standardised development environments via GitHub Codespaces, enabling developers to onboard in minutes rather than hours and eliminating environment-drift issues across the team.

PROJECTS

PetClinic Platform — AWS EKS Production Deployment

June 2026

- **Led** an 11-person team as Team Lead to design and deploy a production-grade AWS infrastructure for 8-microservice Spring Boot application on AWS EKS using Terraform across dev and prod environments.
- **Architected** the full GitOps pipeline; GitHub Actions building and pushing images to Amazon ECR (commit-SHA tagging), ArgoCD syncing to the cluster on every merge for dev, manual approval for prod, thereby removing all manual kubectl applies from the deployment workflow.
- **Configured** OIDC federation between GitHub Actions and AWS. No long-lived credentials, IAM roles were assumed at runtime via OIDC trust policy. IRSA is used for credential free pod-level AWS authentication and secrets management enforced using External Secrets Operator backed by AWS Secrets Manager.
- **Configured** Kubernetes liveness and readiness probes, init container startup ordering (Config Server → Discovery Server → downstream services), and resource requests/limits across all eight services to ensure stable, self-healing deployments.
- **Deployed** a full observability stack (Prometheus, Grafana, Zipkin, Fluent bit, Loki) exposed via AWS ALB Ingress with subpath routing, enabling live per-service metrics dashboards and end-to-end distributed trace visibility, log aggregation and alerting rules across the microservice.
- **Implemented** Karpenter node autoscaling and HPA Pod autoscaling. The cluster scales from 2 to N nodes automatically based on pending pods, with cost guardrails via resource limits and AWS budgets Alerts.
- **Used AI-assisted engineering** throughout by prompting, reviewing, and validating all Terraform modules, Kubernetes manifests, Helm charts, and CI/CD pipeline with Claude Code, Zero infrastructure hand written.

Three-Tier Book Review Application — AWS & Terraform

March 2026

- **Architected** a production-grade three-tier application on AWS, enforcing strict separation between Presentation (Next.js), Business Logic (Node.js), and Data (MySQL/RDS) layers for security and independent scalability.
- **Implemented** a dual-tier Load Balancer strategy — Internet-Facing ALB for public ingress and Internal ALB for service-to-service communication — enabling secure, scalable traffic routing across tiers.
- **Applied** Security Group Chaining to enforce a least-privilege access model: each tier only accepts traffic from its verified upstream dependency, minimising the blast radius of any potential compromise.
- **Conducted** paper-based architectural modelling before provisioning to eliminate resource sprawl and rework — a discipline that keeps infrastructure lean and cost-predictable.

Epicbook Application Deployment — Microsoft Azure

March 2026

- **Designed** and deployed a secure three-tier architecture on Azure, separating web, application, and database tiers across dedicated subnets to improve security isolation and fault containment.
- **Provisioned** Azure Database for MySQL Flexible Server with private endpoint integration, delegated subnet access, and private DNS resolution, eliminating public database exposure entirely.
- **Resolved** outbound connectivity constraints in Azure Standard Load Balancer by implementing a NAT Gateway egress design, enabling secure package installation and OS updates without opening inbound attack surface.

Scalable WordPress Platform — GitLab CI/CD & Terraform IaC

August 2025

- **Engineered** a full GitLab CI/CD pipeline that automates Terraform-driven AWS infrastructure provisioning, with plan previews and artifacts surfaced to reviewers before any apply is executed.
- **Modularised** infrastructure components (VPC, DB, ECS/EC2, ALB, shared storage) into reusable Terraform modules, enabling independent versioning and ~30% faster environment spin-up.
- **Implemented** OIDC-based authentication for the GitLab runner to assume AWS IAM roles securely, eliminating static credentials from the pipeline entirely.

ContactForm App — Kubernetes on EKS via Jenkins

June 2025

- **Built** a zero-credential Jenkins pipeline by attaching IAM Roles to the EC2 build node via instance profiles — removing all hard-coded AWS secrets and reducing credential-exposure risk by ~99%.
- **Developed** a Jenkins Multibranch Pipeline that automatically detects branch changes and triggers a full workflow: container build → ECR push → Helm-based rollout to EKS.

EDUCATION & CERTIFICATIONS

Bachelor of Technology in Geoinformatics | GPA: 4.65 / 5.0

July 2023

Development & Operations - Landmark Technologies (Bootcamp)

January 2024

DevOps Micro Internship – The Cloud Advisory

January 2026