

Oyame Ogbeche

[✉ Mail](#) [in LinkedIn](#) [GitHub](#)

PROFILE

Hands-on Solutions Architect with a strong understanding of the AWS and Hands-on experience in design of fault tolerant, highly available, highly scalable applications. Recently upskilled in DevOps methodologies and tools, I bring a blend of cloud architecture expertise and proficiency in continuous integration, continuous deployment (CI/CD), infrastructure as code (IaC), container orchestration (Kubernetes, Docker) and automation to streamline deployments and optimize cloud services.

SKILLSET

- Amazon Web Services
- Git/Github
- Kubernetes
- Docker
- Python
- Jenkins
- Linux
- Terraform
- Prometheus
- HTML/CSS

EXPERIENCE

CFO Professionals - June 2022 - Present

AWS Cloud Consultant

- Design and Architecture: Leverage services like Amazon EC2 (for compute power), Amazon RDS (for managed databases), and Amazon S3 (for scalable storage) to design high-availability solutions that meet organizational needs. Use AWS Well-Architected Framework to ensure the architecture is optimized for performance, security, and cost-efficiency.
- Security and Compliance: Employ AWS Identity and Access Management (IAM) to manage access and permissions securely. Use AWS Key Management Service (KMS) for data encryption and AWS Config to enforce compliance across AWS resources. Implement Amazon GuardDuty for continuous threat detection and monitoring.
- Collaboration and Communication: Collaborate with DevOps teams to integrate AWS CodePipeline for continuous integration and deployment (CI/CD) processes. Use Amazon Simple Notification Service (SNS) and Amazon Simple Queue Service (SQS) to enable reliable messaging and notifications across teams and applications.
- Performance Monitoring: Set up Amazon CloudWatch for real-time monitoring of resource metrics, application performance, and custom metrics. Integrate AWS X-Ray for end-to-end debugging and tracing of distributed applications. Use AWS CloudTrail for logging and auditing AWS API calls for compliance and troubleshooting.
- Disaster Recovery: Use Amazon S3 with Cross-Region Replication for secure data backup, along with AWS Backup to automate backup across AWS services. Leverage AWS Elastic Disaster Recovery for fast recovery of applications in the event of failure, ensuring business continuity with minimal downtime.
- Solution Implementation: Implement infrastructure using AWS CloudFormation to automate provisioning and Elastic Load Balancing (ELB) for distributing incoming traffic across instances. Utilize Auto Scaling to dynamically adjust resources based on load, ensuring seamless scaling of applications.
- Documentation: Maintain architecture diagrams using AWS Architecture Icons and store configurations and design decisions in AWS Systems Manager (Document Manager). This central documentation hub facilitates team collaboration and future reference.
- Continuous Learning: Stay updated on AWS service enhancements and emerging solutions, such as Amazon EKS for containerized applications, Amazon SageMaker for machine learning, and AWS IoT Core for IoT deployments, ensuring you can apply the latest innovations to drive organizational value.

Google Ads Specialist - April 2021 - June 2022

- Improved inventory management process using Tableau, resulting in 15% faster order fulfillment
- Created and implemented search Ads for over 100 businesses, increasing exposure by over 40% and resulting in over 25% growth in sales.

CERTIFICATIONS

- [AWS Certified Solutions Architect - Associate](#)
- [AWS Cloud Practitioner](#)
- Google Search Ads Certification

PROJECT

Cloud Resume Challenge

- Technology Used: AWS S3 Bucket, Route 53, Cloudfront, Certificate Manager, API Gateway, Lambda function, Dynamo DB, GitHub Actions.
- Designed the frontend of my resume website with HTML/CSS and JavaScript for the API call to the backend
- Hosted my static resume website with S3 Bucket while using CloudFront for CDN
- Leveraged Route53 for DNS resolution and AWS Certificate Manager to store TLS/SSL certificates for secure HTTP connection.
- I configured DynamoDB for my Database to store data and track my page views
- I configured Lambda Function with Python code as part of the backend serverless architecture to query the DynamoDB
- As a best practice I configured API Gateway, so as not to expose my Database
- I Implemented GitHub Actions for CI/CD
- I provisioned my infrastructure using IaC with Terraform
- [Checkout my project source code](#)
- [My blog post for a more detailed breakdown](#)