Mbodou Abakar Mahamat

Jersey City, NJ | (838) 202-2517 | mahamatmbodou33@gmail.com | https://linkedin.com/in/mahamatmbodou | https://github.com/mahamatmbodou33

PROFESSIONAL SUMMARY

AWS Certified Solutions Architect – Associate with hands-on experience designing and automating secure, highly available AWS infrastructure using Terraform, Docker, and GitHub Actions. Built a multi-AZ, three-tier AWS environment from 12 reusable Terraform modules, deployed through OIDC-secured CI/CD pipelines with HTTPS, WAF protection, and Prometheus/Grafana and CloudWatch monitoring. Seeking a junior DevOps or Cloud Engineer role where I can contribute infrastructure automation and cloud security skills while continuing to develop expertise in scalable, production-grade cloud operations.

TECHNICAL SKILLS

- **Cloud:** Amazon Web Services (AWS) — EC2, VPC, Auto Scaling, Application Load Balancer (ALB), RDS, S3, Route 53, ACM, WAF
- **Infrastructure as Code:** Terraform, Terraform Modules, Remote State Management
- **CI/CD & Automation:** GitHub Actions, CI/CD Pipelines, Bash/Shell Scripting
- **Containers & Operating Systems:** Docker, Linux
- **Monitoring & Observability:** Prometheus, Grafana, Alertmanager, Blackbox Exporter, Node Exporter
- **Cloud Security & Identity:** AWS IAM, AWS STS, RBAC, MFA, Least-Privilege Access, OpenID Connect (OIDC)
- **Networking:** VPC Networking, Security Groups, Load Balancing, DNS, Health Checks
- **Databases:** Amazon RDS, MySQL, DynamoDB

CERTIFICATIONS

- **AWS Certified Solutions Architect – Associate | Amazon Web Services (AWS) | 2025**
- **AWS Certified Cloud Practitioner | Amazon Web Services (AWS) | 2024**

EDUCATION

Associate of Science (AS), Computer Information Systems - Borough Of Manhattan Community College **2024**

Relevant Coursework: Web Programming II • Computer Programming II • Linux/Operating Systems • Database Systems II • Mobile Device Programming • Telecommunication Networks

Selected Academic Project: Afro-Style E-Commerce Website

- Built a 7-page e-commerce storefront (Afro-Style) using HTML5, CSS3, JavaScript, and jQuery, showcasing a catalog of 32 clothing products across 4 category pages (Men, Women, Kids, and Couples).
- Built a shopping cart supporting all 32 products, modeling each item as a JavaScript object with 3 properties (name, price, quantity), using the browser's localStorage API to retain cart contents and totals across page reloads.
- Implemented 4 core cart functions (add to cart, quantity increment, quantity decrement, and item removal) with real-time recalculation of cart totals and dynamic DOM re-rendering, all handled client-side without a backend or framework.

TECHNICAL/CLOUD PROJECTS

Production-Ready AWS 3-Tier Infrastructure with Terraform & CI/CD | Jersey City, NJ ***05/2026 - 06/2026***

Independent cloud engineering portfolio focused on designing, automating, and deploying production-ready AWS infrastructure using DevOps best practices.

- Designed a 3-tier AWS network across 3 Availability Zones with a public ALB, private EC2 instances running Dockerized applications, and an isolated database tier; enforced tier-to-tier security groups, SSM-only access, and Auto Scaling with ALB health checks for automatic instance replacement.
- Built 12 reusable Terraform modules for VPC, security groups, IAM/OIDC, ALB, ECR, RDS, ACM, Route 53, WAF, and monitoring, using variables and outputs to eliminate hardcoded values; reused the same module set across isolated dev/prod environments with environment-specific configurations and no duplicated resource logic.
- Secured public traffic with Route 53, ACM-issued HTTPS, and AWS WAF managed rules for common threats and SQL injection, using host-based ALB routing to forward subdomain requests to the correct private application while keeping backend instances inaccessible from the internet.
- Built two GitHub Actions CI/CD pipelines using GitHub OIDC for secure, credential-free AWS authentication: an infrastructure pipeline for Terraform plan/apply with production approval gates, and an application pipeline that builds and pushes Docker images to Amazon ECR and triggers rolling Auto Scaling instance refreshes.
- Configured encrypted Amazon S3 for remote Terraform state and DynamoDB for state locking, separating dev/prod state files by key and preventing concurrent applies from corrupting shared infrastructure state.
- Built a Prometheus, Grafana, Alertmanager, and Blackbox Exporter monitoring stack alongside CloudWatch dashboards and SNS email alerts, providing real-time infrastructure and ALB health visibility validated through simulated CPU/disk stress and instance-failure tests.

- Diagnosed unhealthy EC2 targets behind an AWS Application Load Balancer by verifying the application was listening on port 80 and reviewing target-group health checks and security-group rules; identified that the EC2 security group did not allow inbound port 80 traffic from the ALB security group, corrected the Terraform configuration, and restored healthy targets and application traffic.

Enterprise AWS IAM Role-Based Access Control (RBAC) | Jersey City, NJ ***03/2026 - 04/2026***

Independent cloud engineering portfolio focused on implementing AWS security, identity management, and Infrastructure-as-Code best practices.

- Designed an IAM RBAC model with 9 users across 3 permission-scoped groups: Admin (AdministratorAccess), Developer (EC2, S3, and CloudWatch Logs), and Tester (read-only and monitoring access). Each group's access was matched to its job function, restricting users to only the resources their role required.
- Configured a DevOps role with infrastructure and CI/CD permissions that could assume a separate Admin role through AWS STS (sts: AssumeRole). This granted temporary, time-limited elevated access instead of permanent administrator privileges.
- Enforced MFA and credential-based sign-in for all 9 users, protecting critical AWS resources from unauthorized access and accidental changes while still allowing secure infrastructure operations.
- Validated least-privilege enforcement by testing permitted and denied actions for each group, confirming that Developer and Tester accounts were blocked from IAM changes and actions outside their assigned permissions.