SUMMARY

Highly motivated and results driven certified AWS Cloud Practitioner with over 1 years of professional experience implementing and maintaining IT infrastructure, database management and integrating it into Amazon web Services Cloud solutions in large and medium sized enterprise environment

PERSONAL DETAILS

Date of Birth
29/08/1999

Gender
Male

Marital Status
Single

CONTACT

Address:
64 Bangkok cres Cosmo city, Randburg

Phone:
0608475797/0813808572

Email:
isayamulaudzi@gmail.com

Website:
http://noblemulaudzi.com

LANGUAGE

Tshivenda
English
Sepedi

WORK EXPERIENCE

Tendai ICT
Junior DevOps Engineer
Jan 2022 - Present

- Development: Designing and Developing highly scalable and secure systems within health care industry .
- Operations : Architecting and Supervising the deployment of highly scalable and fault tolerant applications that always ensure highly availability through automation such as auto scaling within AWS environment.

TECHNICAL SKILLS

- **AWS:** Identity and access management, EC2, S3 storage services, Databases, DNS using Route 53, VPCs, ELB, Autoscaling, CloudFront, CloudWatch, CloudTrail, Cloud Formation, Fargate, ECS and high level understanding of application services such as SNS, SQS.
- **Linux :** Managing users, configuring services, Apache-PHP-MYSQL setup, server configuration, Virtual host setup, network troubleshooting, using SSH to connect to remote hosts and general troubleshooting
- **Terraform :** Using Terraform to provision resources such as VPC and more on AWS
- **Docker :** Using Docker and making use of AWS ECS and Fargate service to containerize application and to create infrastructure for deployment with AWS.
- **Kubernetes :** : created clusters on AWS with the help of eksctl, this included setting up worker nodes, region and nodes and making use of kubectl to apply services and developments .
- **SQL:** Database installation, creating and updating tables, writing basic and nested queries and more.
- **Python Scripting and Automation:** Regular expressions, website crawling, document parsing and more.
- **Software development** : ASP.NET, Laravel, PHP, Javascript, React Native, JQUERY,Java, Python :Using several frameworks and programming languages to develop systems that are highly secured

Education

Bsc in Computer science
University of Venda
2018 - 2021

CERTIFICATION

- AWS Certified Cloud Practitioner (AWS)
- Microsoft Azure IoT Developer(Microsoft)
- DevOps on AWS (Udemy)
- AWS Services for Solutions architect(Udemy)
- Studying towards AWS Certified developer (AWS)

SOFT SKILLS

- Time management and communication
- Team work and collaboration
- Flexibility and adaptability

AWS CLOUD SERVICES WORKED WITH

- **Compute services** Amazon EC2, AWS Elastic beanstalk
- **Containers** Amazon EKS, Amazon ECS, Amazon ECR
- **Storage** Amazon S3, Amazon EBS, Amazon EFS
- **Database** Amazon RDS
- **Management and Governance** AWS Autoscaling, AWS CloudFormation, AWS CloudWatch
- **Developer Tools** AWS Codebuild, AWS CodeCommit, AWS CodePipeline
- **Networking & Content Delivery** Amazon VPC, AWS ELB, Amazon Route 53, Amazon CloudFront
- **Billing & Cost management** AWS Cost Explorer, AWS Budgets,

HANDS ON PROJECTS

- AWS PIPELINES : Making use of AWS Code pipeline service to automate steps in my software delivery process. I used code commit to store my code using git, I then used Codebuild service to create a build for my project which was in code commit and went on to configure the buildspec file. Then I used code Deploy service to create the app, specifying EC2 as the platform to deploy to, using Blue green deployment Then went on to link all these phases with the AWS Code pipeline for automation of all these phases. Also making use of Cloud watch to see in case there are some errors during the process.

- Created VPC ,subnets ,internet gateway and configured EC2 instances in public and private subnet,allowing the instances in the public subnet to communicate with the internet by attaching the internet gateway to the VPC and configuring security groups for these instances,then used SSH to login to the EC2 instance to configure the Apache.

- Terraform (IaC) : Used Terraform to provision an infrastructure which was hosting a laravel Application, the infrastructure was composed of a VPC, public subnet, EC2, Routing table, SG and an IGW attached to a VPC.

- Docker : Used Docker to containerize an application stack, the application stack was composed of a laravel Application with a mysql database, using php Yasmin as the DBMS and Redis for handling sessions,
- Kubernetes : Using Kurbenetes for container orchestration, creating clusters with the help of eksctl and managing the cluster with kubectl.

- Elastic Beanstalk : Used Amazon Elasticbeanstalk service to deploy a php application, the php application was also making direct calls to a mysql database which I created using RDS service.

- S3: Making use of AWS simple storage services to store my source code as object, and making the object publicly accessible by other services, then made use of the code build service to build the code which was in the S3 storage, then made use of Code Deploy service to deploy the application to an EC2 instance manually.

- Using AWS Identity and Access Management (IAM) to manage access to AWS services and resources securely, as well as setting up Multi-Factor Authentication (MFA) for extra protection of the environment.

- Used Amazon's Route 53, a highly available and scalable cloud Domain Name System (DNS) web service to route end users to my applications using a custom domain name I bought. I archived this through creating hosted zones within ROUTE 53 which automatically created SN record for me and I created an A record wich pointed my custom domain name to my servers public IP address.

- Using AWS Budgets to set custom budgets for tracking costs and usage of resources in an AWS account. Setting up alerts by email when actual or forecasted cost and usage exceed budgeted threshold.

- Developed several major health care web applications for Bonitas medical fund using Javascript, HTML, CSS, PHP and laravel and hosted these applications using Linux virtual machine through LAMP servers which reduced mistakes and brought consistency within the organization.

REFERENCE

- **Reference available upon request.**