**Course:** Dev Ops

**Instructor:** Khwaja Abdul Hafeez

**Class:** 98711

**Name:** Muhammad Maaz Arsalan Batla

**ERP:** 22794

**Project: AWS Infrastructure Deployment with Terraform, Docker, and BI Integration**

Table of Contents

[Introduction 1](#_Toc200153140)

[Project Architecture Overview 2](#_Toc200153141)

[Terraform Code Structure 2](#_Toc200153142)

[EC2 Auto Scaling Setup 3](#_Toc200153143)

[RDS Instances 3](#_Toc200153144)

[Application Load Balancer (ALB) 3](#_Toc200153145)

[Dockerized Application Deployment 3](#_Toc200153146)

[Domain and SSL Setup 3](#_Toc200153147)

[Database Access and Dummy Data 3](#_Toc200153148)

[BI Tool Deployment (Redash/Metabase) 3](#_Toc200153149)

[Loom Demonstration Video 3](#_Toc200153150)

[GitHub Repository Link 3](#_Toc200153151)

[Conclusion 3](#_Toc200153152)

# Introduction

The goal of this project was to design and deploy a **scalable**, **secure**, and **containerized** infrastructure on **AWS** using **Terraform** as Infrastructure as Code (IaC). The architecture consists of Auto Scaling EC2 instances running Dockerized Node.js applications behind a secure Load Balancer, private RDS databases, and a containerized Business Intelligence (BI) tool (Metabase) for live database visualization. This project simulates a production-grade deployment and demonstrates key DevOps principles including automation, modularity, high availability, and monitoring.

# Project Architecture Overview

# Terraform Code Structure

# EC2 Auto Scaling Setup

# RDS Instances

# Application Load Balancer (ALB)

# Dockerized Application Deployment

# Domain and SSL Setup

# Database Access and Dummy Data

# BI Tool Deployment (Metabase)

# Loom Demonstration Video

# GitHub Repository Link

# Conclusion

VPC:

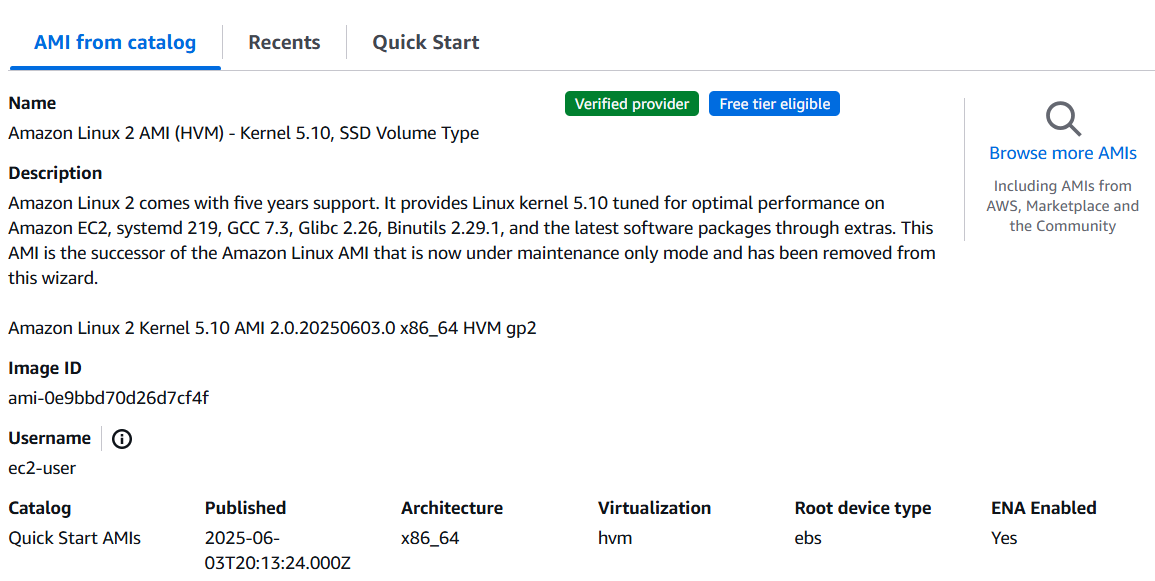
A screenshot of a computer

AI-generated content may be incorrect.

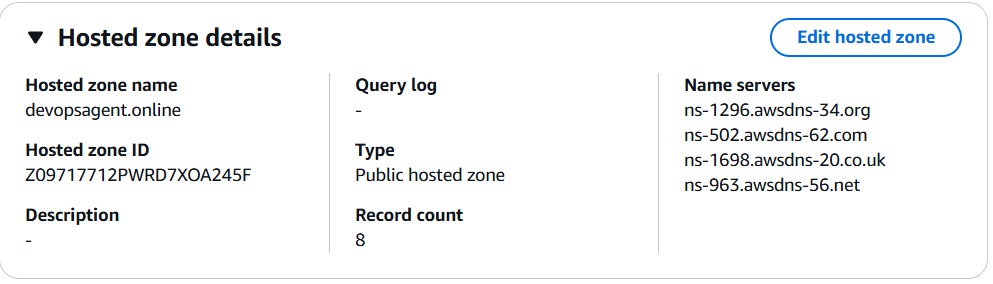
Available Subnets:

A screenshot of a computer

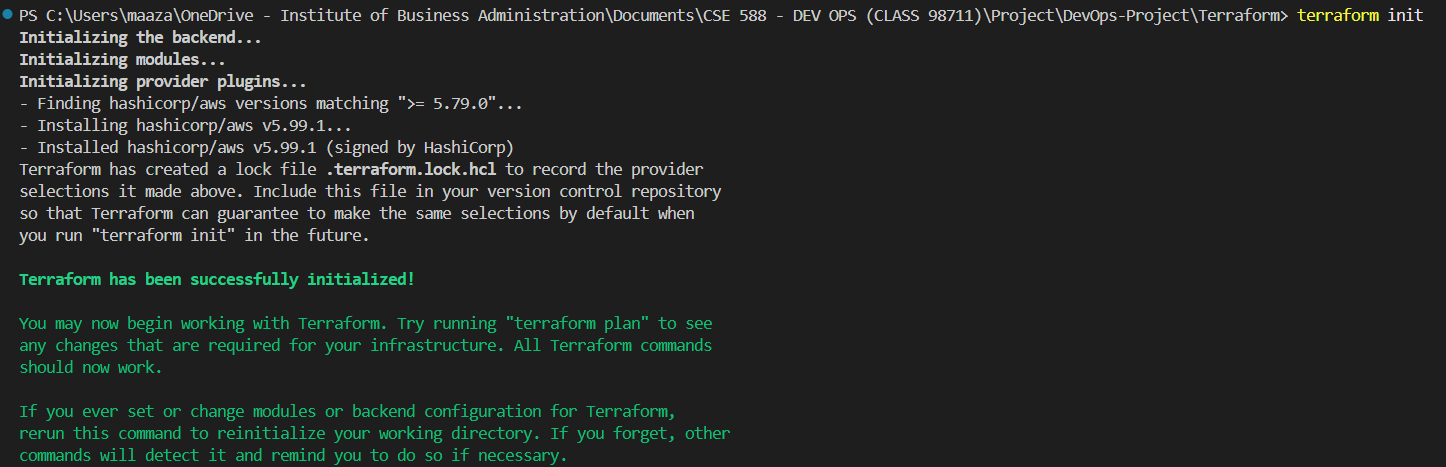
AI-generated content may be incorrect.

EC 2:

Route53:



Running the command “terraform init”:

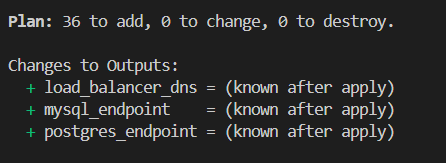


Running the command “terraform validate”:

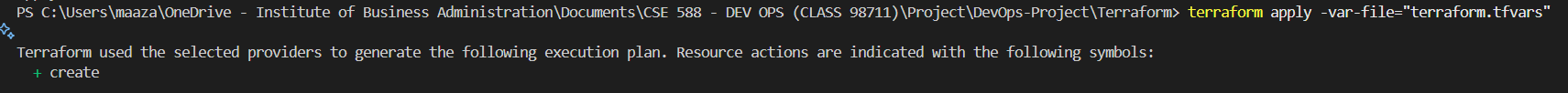


Running the command "terraform plan -var-file=”terraform.tfvars”:



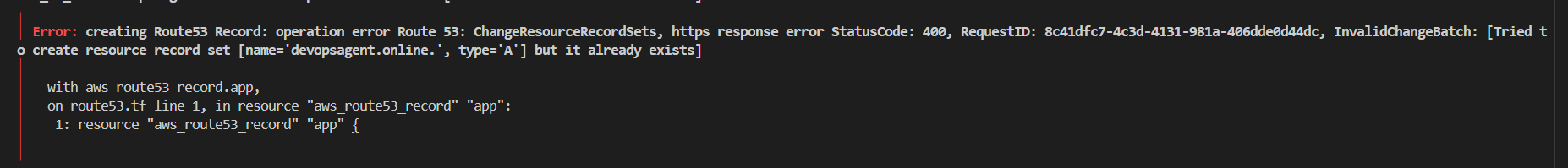


Running the command "terraform apply -var-file= “terraform.tfvars”:



A computer screen shot of a program

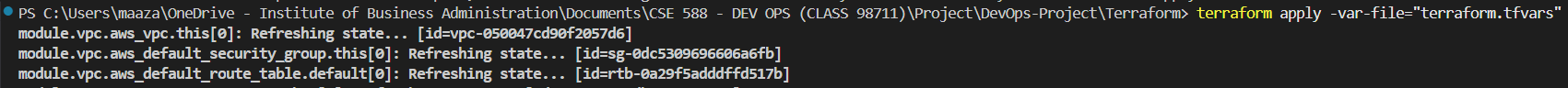
AI-generated content may be incorrect.





A computer screen with green text

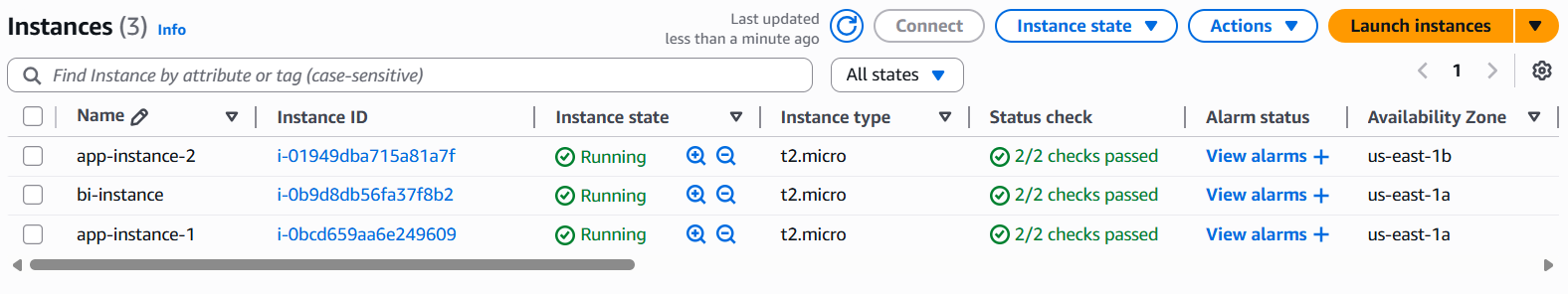
AI-generated content may be incorrect.



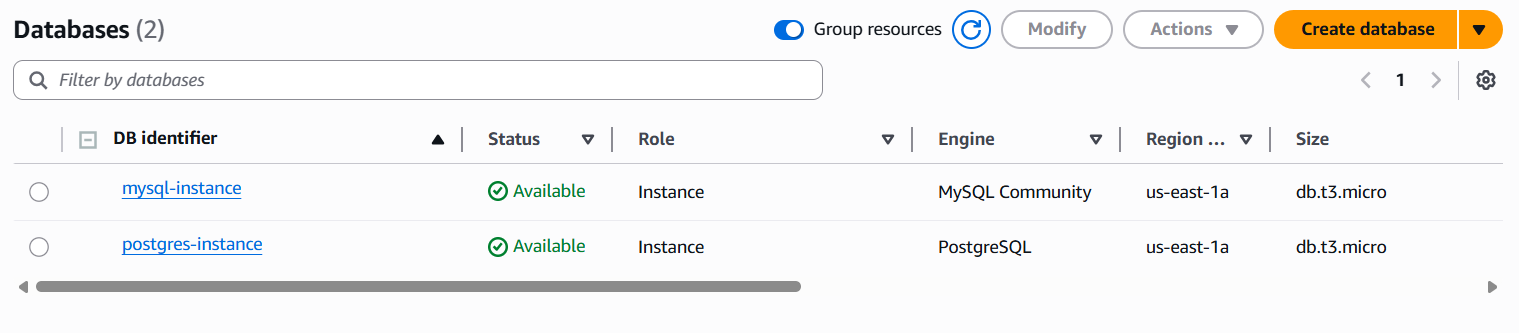
A screen shot of a computer

AI-generated content may be incorrect.

Instances Running Successfully:



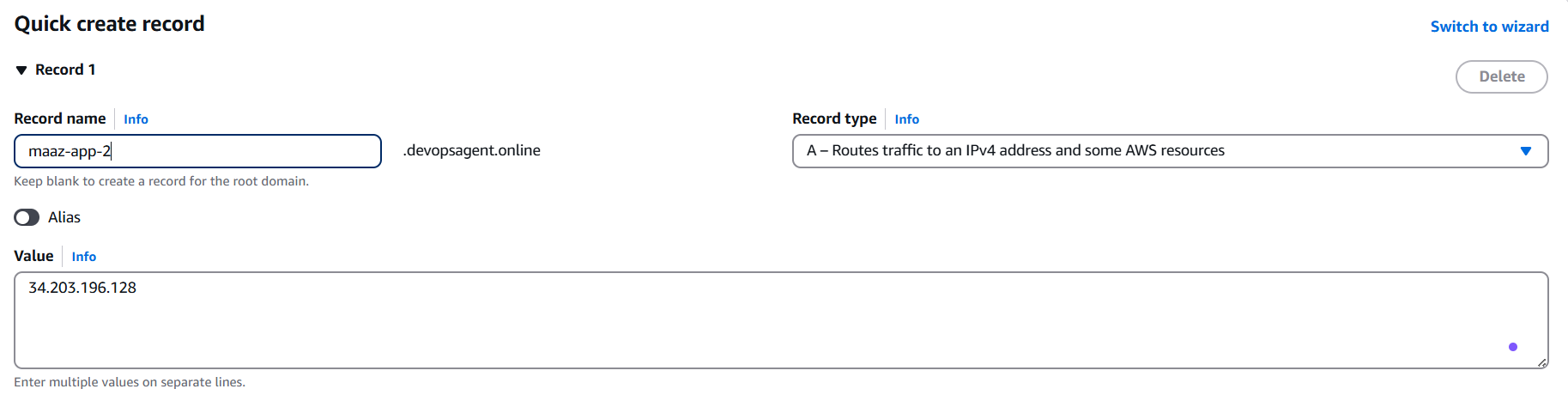
Databases Setup Successfully:



Creating Route53 Records:

A close-up of a computer screen

AI-generated content may be incorrect.



A computer screen shot of a computer

AI-generated content may be incorrect.

Route53 Records Added Successfully:

