Source Code:

**Terraform:**

terraform {

required\_providers {

aws = {

source = "hashicorp/aws"

version = "5.83.1"

}

}

}

provider "aws" {

# Configuration options

# eu-central-1

region = "us-east-1"

access\_key = "AKIAVRUVVQUSZJMWU2JX"

secret\_key = "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"

}

locals {

servers = ["swarm-leader", "swarm-node-1", "swarm-node-2"]

}

resource "aws\_instance" "docker-swarm-lab" {

ami = "ami-04b4f1a9cf54c11d0"

instance\_type = "t2.micro"

tags = {

Name = each.value

}

for\_each = toset(local.servers)

key\_name = "docker-swarm"

security\_groups = [aws\_security\_group.sg-1.name]

user\_data = file("docker-install.sh")

}

resource "aws\_security\_group" "sg-1" {

egress {

from\_port = 0

to\_port = 0

protocol = "-1"

cidr\_blocks = ["0.0.0.0/0"]

ipv6\_cidr\_blocks = ["::/0"]

}

ingress {

from\_port = 0

to\_port = 0

protocol = "-1"

cidr\_blocks = ["0.0.0.0/0"]

ipv6\_cidr\_blocks = ["::/0"]

}

name = "my-sg"

}

**Docker Setup commands:**

for pkg in docker.io docker-doc docker-compose docker-compose-v2 podman-docker containerd runc; do sudo apt-get remove $pkg; done

# Add Docker's official GPG key:

sudo apt-get update

sudo apt-get install ca-certificates curl

sudo install -m 0755 -d /etc/apt/keyrings

sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc

sudo chmod a+r /etc/apt/keyrings/docker.asc

# Add the repository to Apt sources:

echo \

"deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu \

$(. /etc/os-release && echo "${UBUNTU\_CODENAME:-$VERSION\_CODENAME}") stable" | \

sudo tee /etc/apt/sources.list.d/docker.list > /dev/null

sudo apt-get update

sudo apt-get install -y docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin

**index.php:**

<?php

// Set the response content type to JSON

header('Content-Type: application/json');

header('Access-Control-Allow-Origin: http://172.31.84.12:3000');

header('Access-Control-Allow-Methods: GET, POST, OPTIONS');

require "./app/config.php";

require\_once "./app/todos.php";

$todoModel = new Todo();

$response = $todoModel->getTodos(10);

// Create a response array

// Encode the response array as JSON

$jsonResponse = json\_encode($response);

// Output the JSON response

echo $jsonResponse;

**index.html:**

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>HTML Web Page</title>

</head>

<body>

<h1>Hello, Phani Kesava!</h1>

<div id="todos"></div>

<script>

fetch("http://:172.31.84.12:5000/")

.then((response) => response.json())

.then((data) => {

const todos = document.getElementById("todos");

data?.forEach((item, index) => {

const todo = document.createElement("h2");

todo.textContent = item.\_id + ". " + item.todo;

todos.appendChild(todo);

});

})

.catch((error) => {

console.log("Error: ", error);

});

</script>

</body>

</html>

**Additional Commands used in the projects:**

cd docker-compose-lab-01

vim api/index.php

vim frontend/index.html

docker compose up -d