**README for Task 3: Docker Container Deployment**

**Project Overview**

This task deploys a Docker container running a simple Flask application on a cloud instance. Terraform is used to provision the server, install Docker, and run the containerized application.

**Prerequisites**

* **Docker** installed locally for building the image
* **Terraform** installed for provisioning infrastructure
* AWS Account with permissions to launch EC2 instances

**Steps**

1. **Build and Tag Docker Image**:

bash

Copy code

docker build -t <your\_docker\_image> .

1. **Push Docker Image to Docker Hub**:

bash

Copy code

docker tag <your\_docker\_image> <your\_docker\_hub\_username>/<your\_docker\_image>

docker push <your\_docker\_hub\_username>/<your\_docker\_image>

1. **Deploy Infrastructure with Terraform**:
   * Modify the main.tf script, replacing <your\_docker\_image> with your Docker image and <your\_key\_name> with your AWS key.
   * Run Terraform commands:

bash

Copy code

terraform init

terraform apply

1. **Access the Application**: Navigate to http://<instance\_public\_ip> in your browser.

**Files in this Directory**

* Dockerfile: Builds the Docker image with a sample Python Flask application.
* app.py: Python Flask application.
* requirements.txt: Lists the application's dependencies.
* main.tf: Terraform script for provisioning AWS infrastructure and running the Docker container.