**Documentation for Terraform Deployment and Helm Chart Installations**

**1. Overview**

This document provides a comprehensive guide to setting up and deploying a web application using Terraform and Helm on an Azure Kubernetes Service (AKS) cluster. This documentation covers the steps involved in configuring infrastructure using Terraform and managing deployments using Helm.

**2. Terraform Deployment**

**Terraform Configuration**

Create a Directory: Create a directory for your Terraform configuration files.

Terraform Files: Create the following Terraform files:

main.tf: This is the primary configuration file for defining resources.

variables.tf: This file contains variable definitions used in your Terraform configuration.

outputs.tf: This file defines the outputs you want to retrieve after the infrastructure is provisioned.

provider.tf: This file specifies the Azure provider configuration.

**Main.tf Example:**

provider "azurerm" {

features {}

}

resource "azurerm\_resource\_group" "example" {

name = "myResourceGroup"

location = "East US"

}

resource "azurerm\_kubernetes\_cluster" "example" {

name = "myK8sCluster"

location = azurerm\_resource\_group.example.location

resource\_group\_name = azurerm\_resource\_group.example.name

dns\_prefix = "myk8s"

agent\_pool\_profile {

name = "agentpool"

count = 3

vm\_size = "Standard\_DS2\_v2"

}

identity {

type = "SystemAssigned"

}

role\_based\_access\_control {

enabled = true

}

}

**Variable Definitions (variables.tf):**

variable "location" {

description = "The Azure region where resources will be created."

default = "East US"

}

**Initializing Terraform:**

terraform init

**Plan the Infrastructure:**

terraform plan

Applying the Configuration:

**terraform apply**

Confirm the action by typing yes when prompted.

2.3 Managing Terraform State

**State Files:** Terraform maintains a state file that keeps track of resources. Make sure to manage the state files securely, especially in team environments.

**3. Helm Chart Installation**

Helm Installed: Install Helm on your local machine. Helm Installation Guide.

Kubernetes Cluster: Ensure your AKS cluster is up and running as provisioned by Terraform.

**Creating a Helm Chart**

helm create myapp

**Installing the Helm Chart**

Install the Chart: Once the chart is configured, you can install it to your AKS cluster:

helm install myapp ./myapp

**Verifying the Installation:** After installation, check the status of the pods to ensure they are running:

kubectl get pods

**Accessing the Application:** If using a LoadBalancer service type, retrieve the external IP to access the application:

kubectl get svc myapp --namespace default