Voting-App for SwissCom

Terraform IAC using in GCP in GKE cluster :

We could complicate this task as much as we could, I have tried to use the standard structure for a quick workable solution.

Main.tf, Provider.tf, variables.tf and terraform.tfvars

Main.tf: The `main.tf` file contains Terraform code to create a VPC network, a subnet within the VPC, and a GKE cluster. The GKE cluster is created with a specified name, location, and initial node count, and is associated with the VPC network and subnet.
Provider.tf: Access to GCP from terraform.
Terraform.tfvars: Project id in GCP
Variables.tf : Variables used for IAC.

Note : I have used the Terraform v1.6.0
Terraform init
Terraform plan
Terraform apply

Project folder :

1. The project folder contains a simple flask app to choose between your favorite days Saturday ,Sunday and Monday .
2. It contains app.py , index.html , dockerfile and Docker-compose (if you would like to test this locally.
3. I have already built this application and pushed it into my public repo for use .

```
 1823  docker build -t ggvote .
 1824  docker image ls
 1832  docker tag ggvote:latest 5555677/ggvote:latest
 1833  docker push 5555677/ggvote:latest
```

docker.io/5555677/ggvote:latest

Note: No need to do anything here at the moment as the image is accessed from docker.io/5555677/ggvote:latest.

K8’s

The GKE deployment file`voting-app-deployment.yaml` file under the `K8` folder contains the Kubernetes deployment configuration for the voting app. It specifies the Docker image to use, the number of replicas to create, services, ports to expose and ingress for the external access.

Deploy.sh:

> It contains the shell script to deploy voting-app-deployment.yaml in GKE on a brand new namespace called voting-app