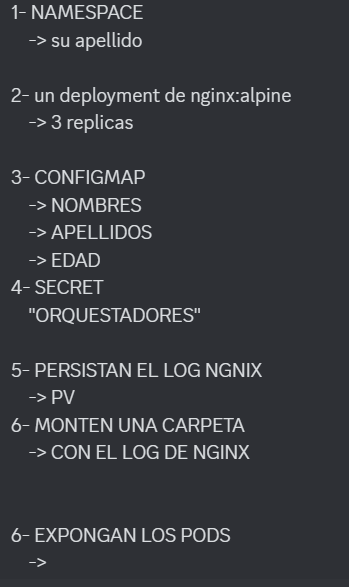
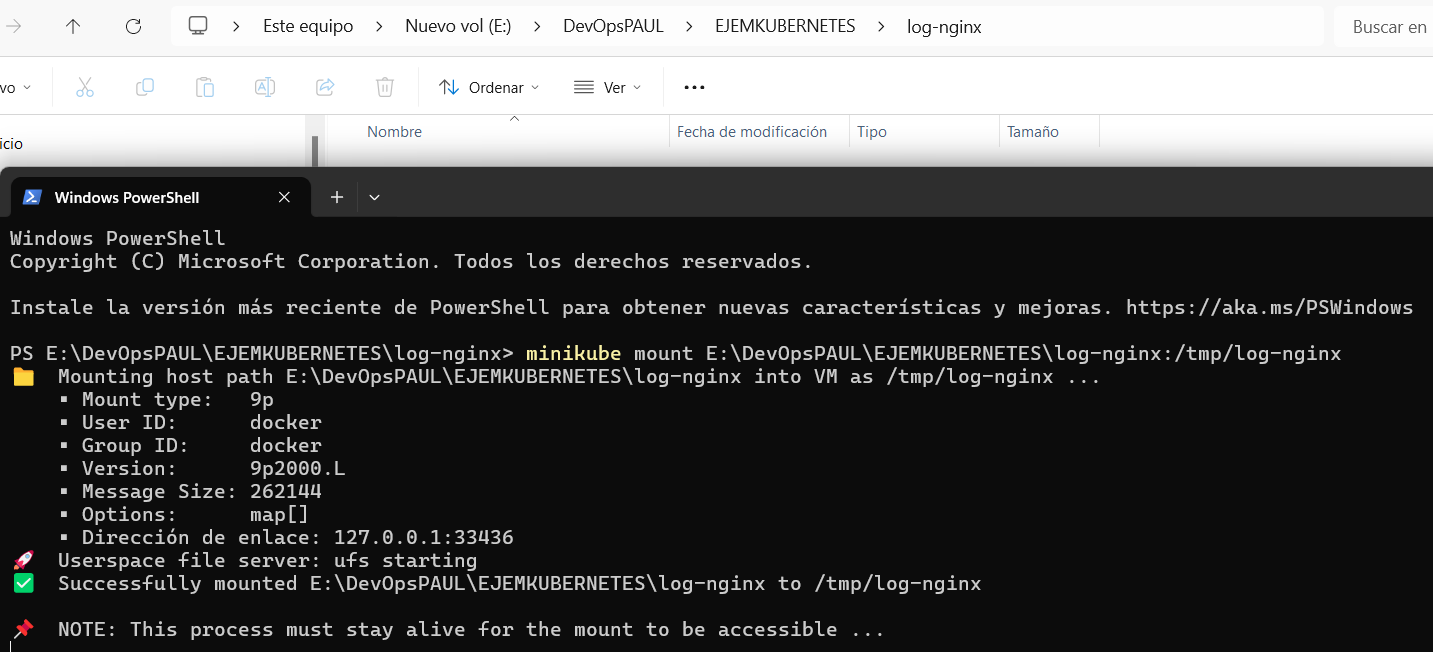
Ejercicio:

* + Namespace: apellido
  + Deployment: nginx:alpine ; 3 replicas
  + Configmap: 03 variables(nombres, apellidos y edad)
  + Secret: “ORQUESTADORES”
  + Persistan el Log de nginx, montar un carpeta con el Log de nginx
  + Expongan los POD: portforward, comando minukube



* + 1. ***Montando la Unidad del Host para Logs***

# minikube mount E:\DevOpsPAUL\EJEMKUBERNETES\log-nginx:/tmp/log-nginx



* + 1. ***Elaborando el archivo YML***

***3. Verificando que existan los objectos***

# kubectl apply -f paulgiraldo.yaml

# kubectl get namespaces

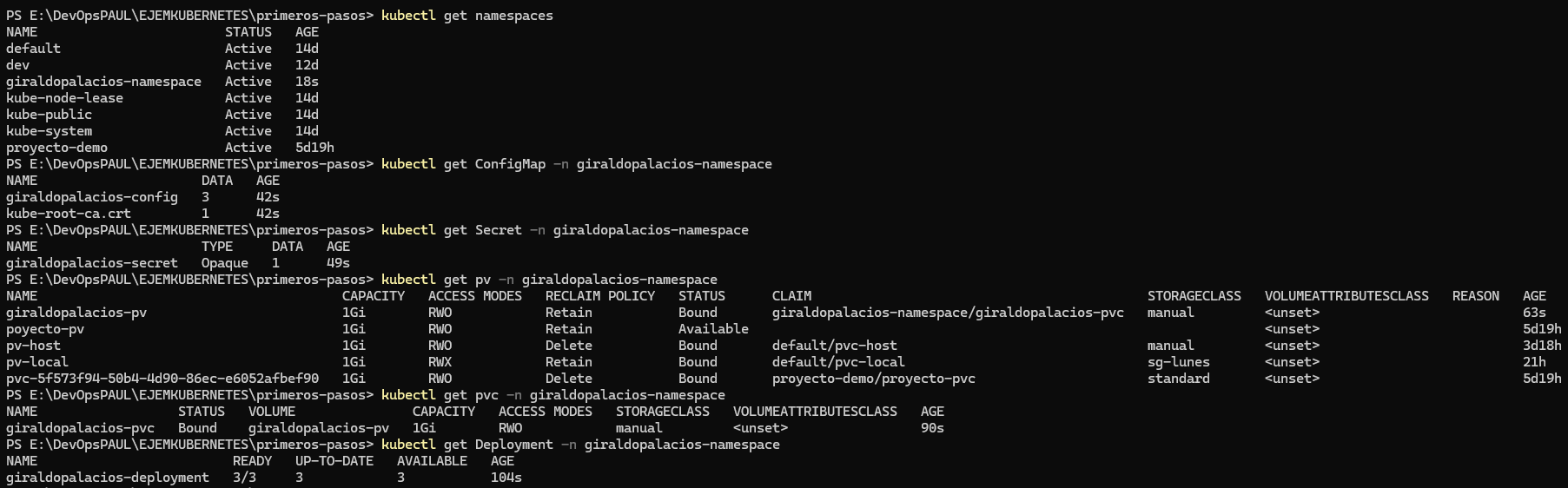
# kubectl get ConfigMap -n giraldopalacios-namespace

# kubectl get Secret -n giraldopalacios-namespace

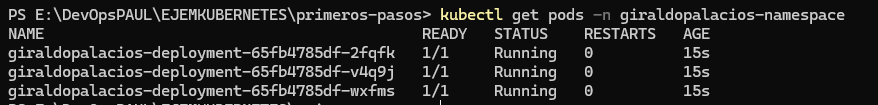
# kubectl get pv -n giraldopalacios-namespace

# kubectl get pvc -n giraldopalacios-namespace

# kubectl get Deployment -n giraldopalacios-namespace



# kubectl get pods -n giraldopalacios-namespace

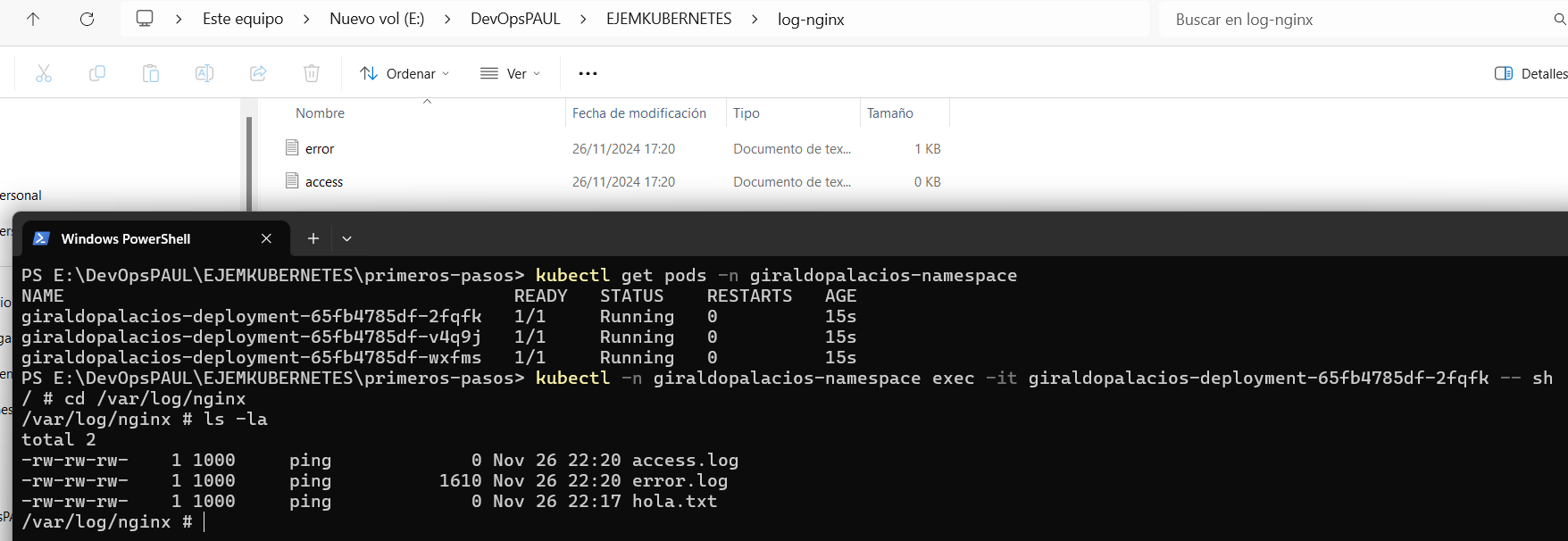


Ingresando a un POD para ver el recurso compartido y las variables

# kubectl -n giraldopalacios-namespace exec -it giraldopalacios-deployment-65fb4785df-2fqfk – sh

# cd /var/log/nginx

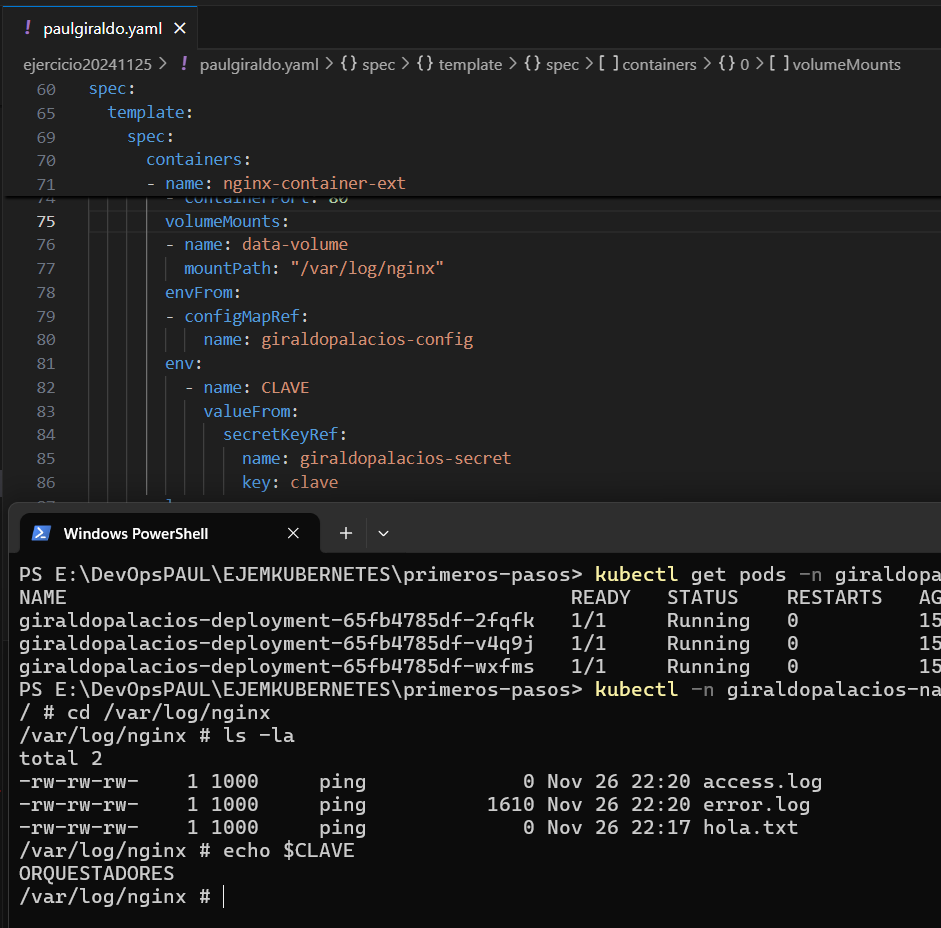
# ls -la



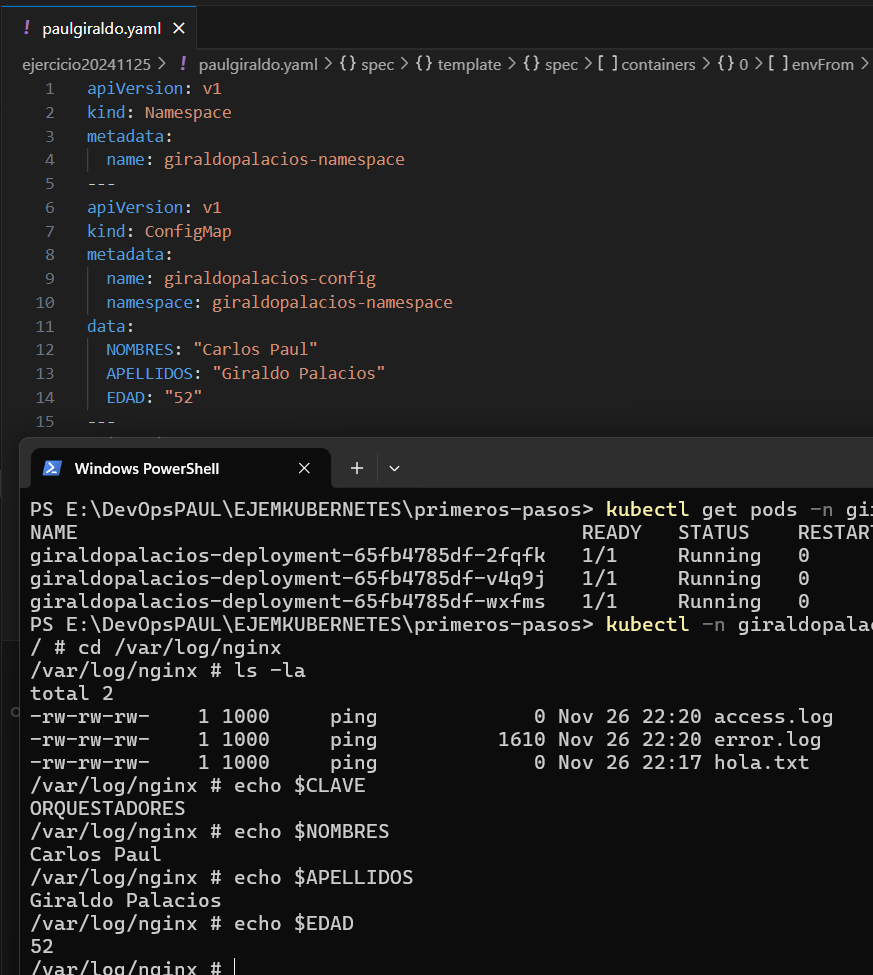
***Verificando la Variable de entorno transferida por el SECRET***

Dentro del POD

# echo $CLAVE



***Verificando la Variable de entorno transferida por el CONFIGMAP***



***Para exponer el servicio, al ser un tipo NodePort podemos usar el siguiente comando:***

# minikube service giraldopalacios-service -n giraldopalacios-namespace –url

Este comando expone el service **temporalmente**

