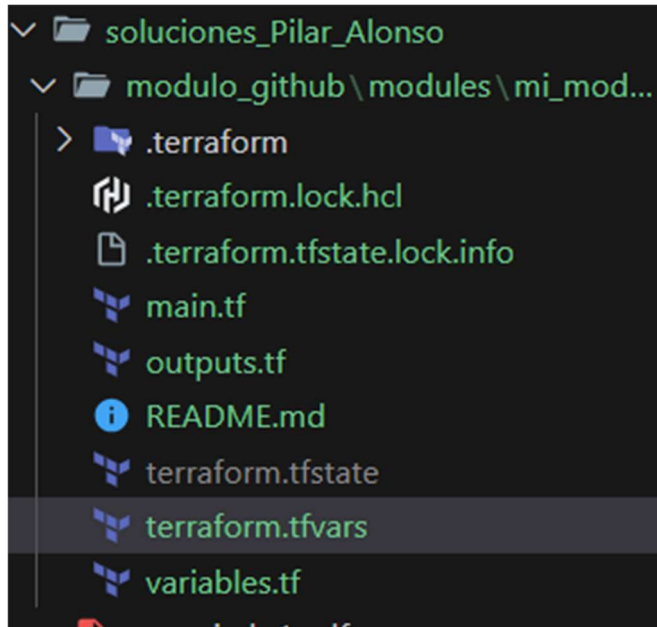


EJERCICIO 6

Subo el modulo a github :



Lo he añadido a mis soluciones y he probado que funcione:

```
you can apply this plan to save these new output values to the Terraform
state.

Do you want to perform these actions?
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes

Apply complete! Resources: 0 added, 0 changed, 0 destroyed.

Outputs:

environment_tag = "dev"
existent_resource_group_nombre = "myTFResourceGroup"
owner_tag = "pilarAlonsoSTEMDO"
net_address_space = tolist([
  "10.0.0.0/16",
])
```

Mi modulo está definido de tal manera que el nombre del recurso se pase desde el modulo que se importe en main.tf para que sea más modular.

```
#MODULO REMOTO
module "mi_modulo_remoto" {
  source = "git::https://github.com/stemdo-labs/terraform-exercises-palonso.git/soluciones/soluciones_Pilar_Alonso/modulo_github/mod
  existent_resource_group_nombre = var.existent_resource_group_name
  vnet_name                      = var.vnet_name
  vnet_address_space            = var.vnet_address_space
  owner_tag                     = var.owner_tag
  environment_tag               = var.environment_tag
}
```

Así se importa desde mi main.tf en local, le tengo que definir la ruta y pasarle las variables obligatorias, además del nombre del recurso.

terraform get --update para actualizar terraform con los cambios del repo.

He añadido el file LICENSE dentro de la especificación del módulo como indica HaShiCorp.