**Final Project: End-to-End DevOps Deployment**

1. **Configuración del repositorio y entorno de Desarrollo**

**Una captura de pantalla de una computadora

El contenido generado por IA puede ser incorrecto.**

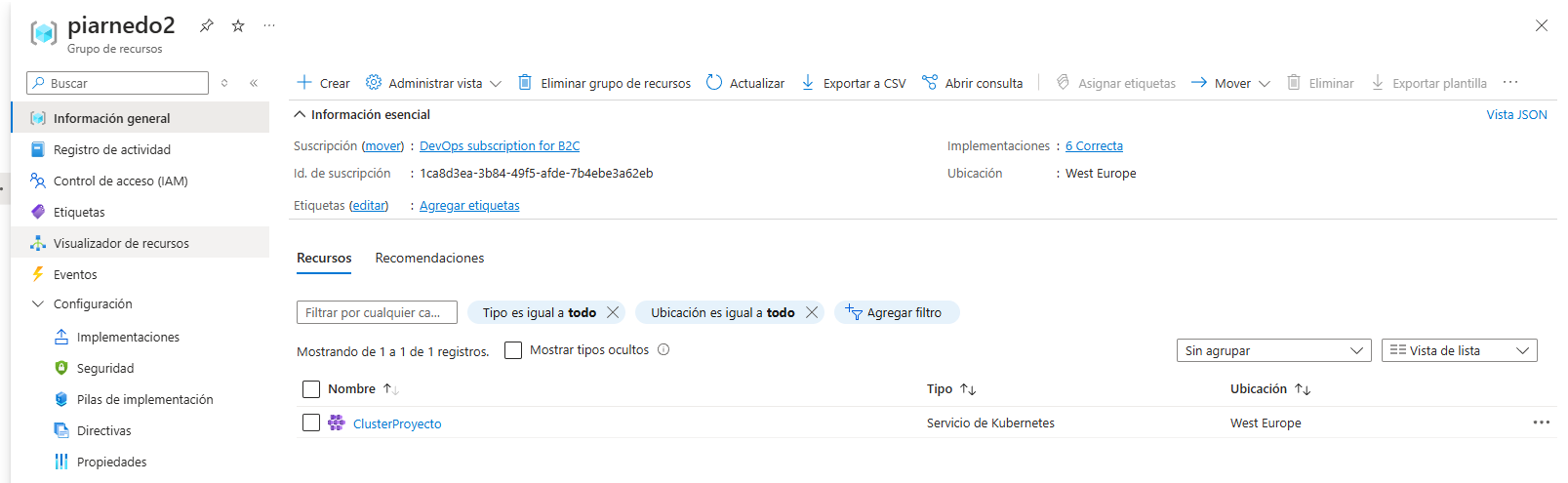
* 1. **Clonar el codigo**

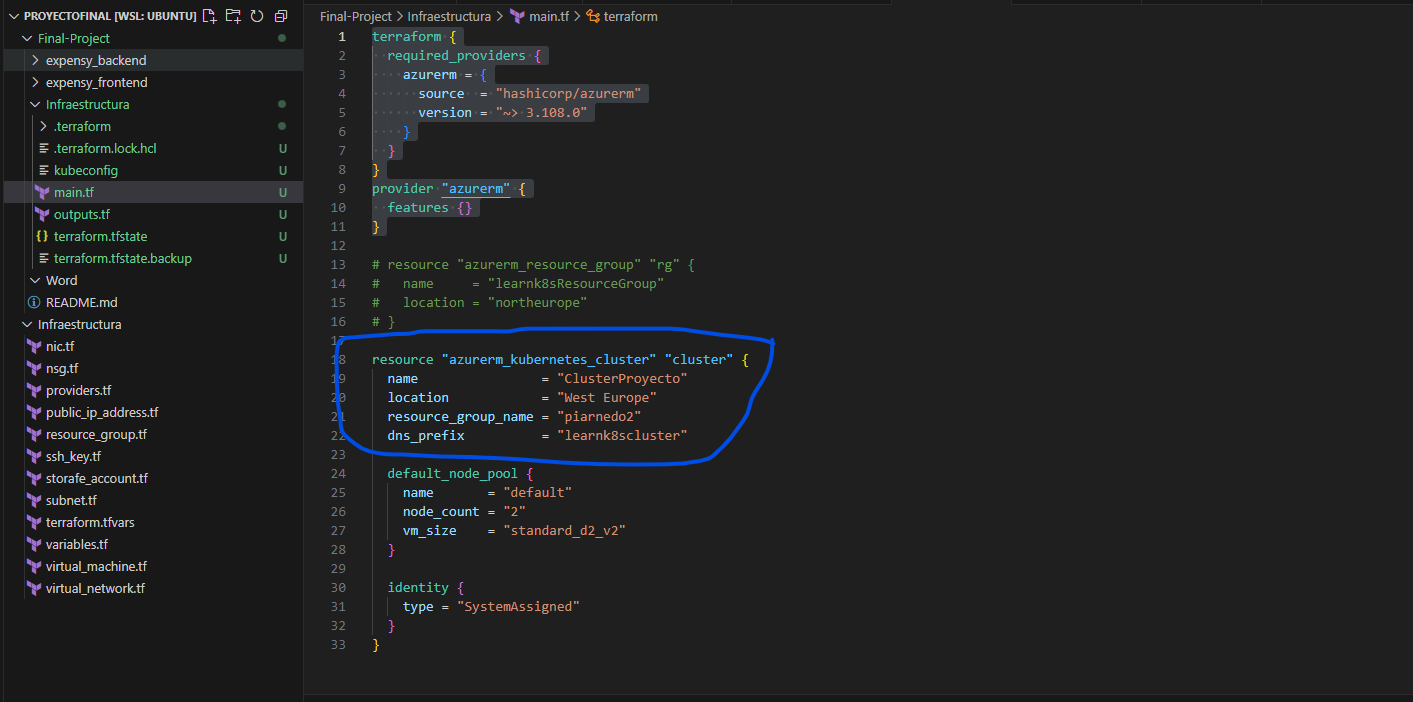
Texto

El contenido generado por IA puede ser incorrecto.

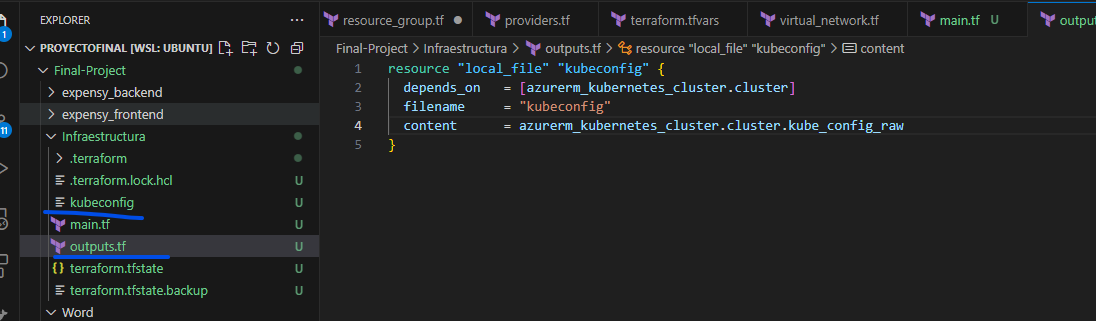
1. **Infrastructure as Code (IaC):**
   1. **Crear Azure AKS:**

**Tener el resource group creado o crearlo de antes (resultado final):**

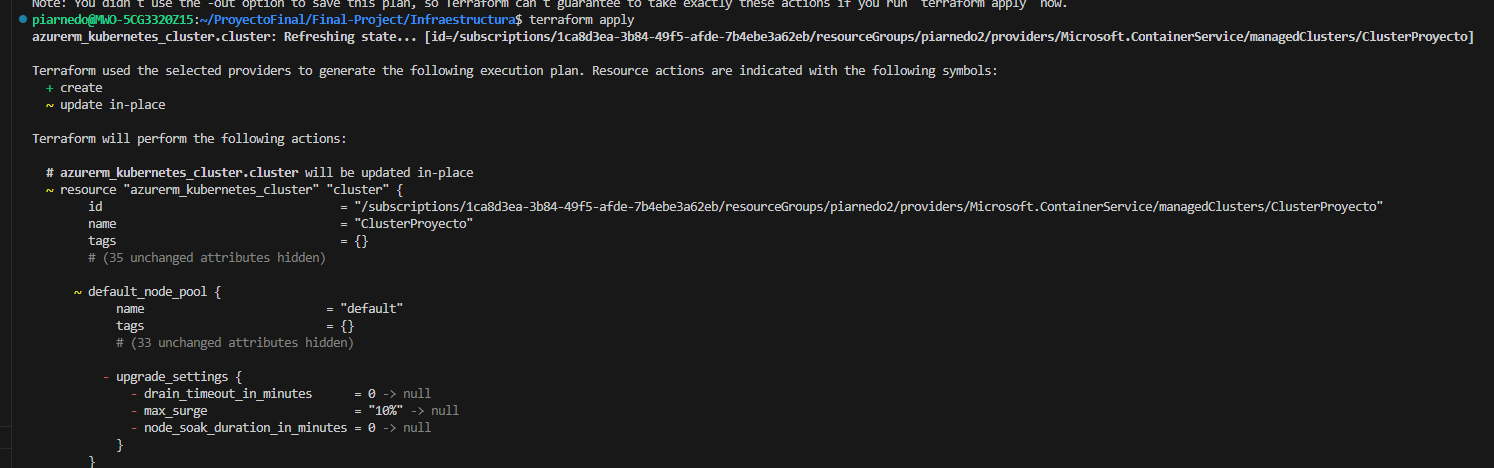


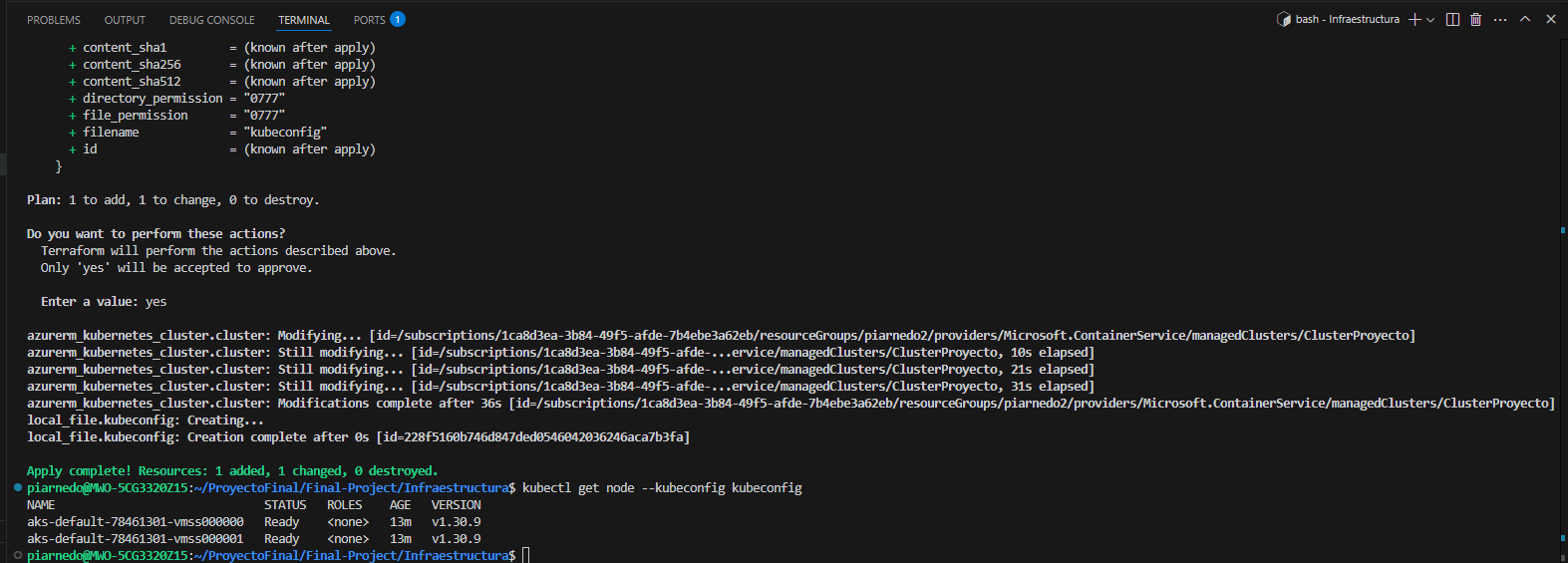
****

**Hacer el Terraform Flow una vez tenemos esto (Se puede crear un kubeconfig para tener el cluster en local mediante el outputs.tf que te lo montará al hacer el Terraform Flow).**

****

**1-Terraform Init, Terraform plan para simular una subida y luego terraform apply para subirlo al azure (Resource Group de antes)**

****

****

1. **Dockerizar imágenes**
   1. **Creamos dockerfiles para front y back**
   2. **Texto

      El contenido generado por IA puede ser incorrecto.**

**Texto

El contenido generado por IA puede ser incorrecto.**

**Creamos Docker-compose para poder generar imágenes y levantar todo a la vez**

**Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.**

* **docker-compose up -d**
* **con Docker images vemos las imangenes creadas**
* **Interfaz de usuario gráfica, Texto, Aplicación

  El contenido generado por IA puede ser incorrecto.**

**Y las tageamos y subimo a Docker hub**

* **docker tag devops-final-project-ih-backend kbaigorri/kbaigorri-proyectofinal:backtag**
* **docker push kbaigorri/kbaigorri-proyectofinal:backtag**
* **docker tag devops-final-project-ih-frontend kbaigorri/kbaigorri-proyectofinal:frontag**
* **docker push kbaigorri/kbaigorri-proyectofinal:frontag**

Interfaz de usuario gráfica, Texto

El contenido generado por IA puede ser incorrecto.