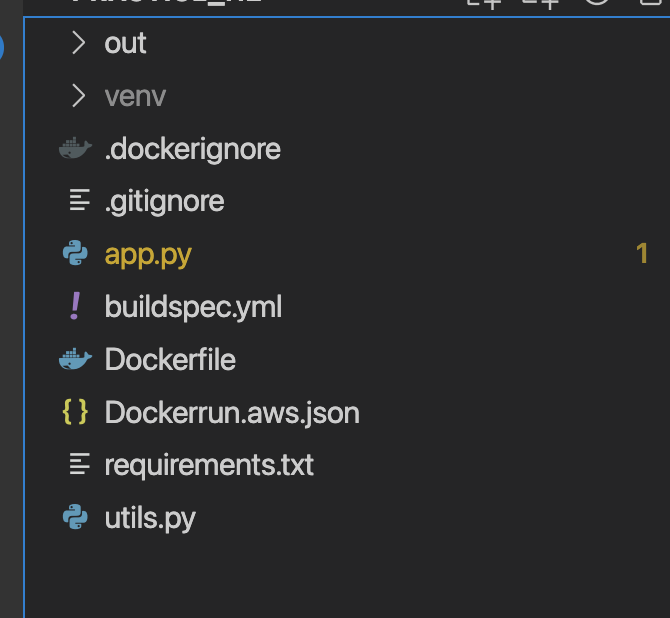
ENTREGABLE NRO 2

DESPLIEGUE CON AWS CODEPIPELINE + AWS BEANSTALK

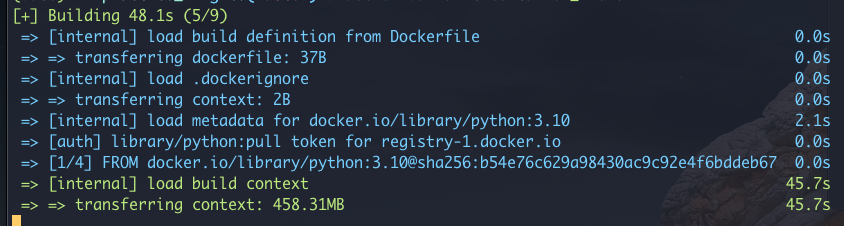
1. Crear la imagen en local

Una vez ya desplegada en local y ambiente desarrollo nuestra App de la practica 2, debemos construir la imagen Docker en nuestro ambiente de desarrollo (local) . Ejecutamos el siguiente comando en consola y la ruta donde se encuentra el proyecto “docker build -t container mle”



Comando para crear imagen docker



Construyendo imagen  


1. Iniciar sesión en AWS ECR

Una vez construida la imagen Docker o en paralelo a la construcción iniciamos sesión en aws con el siguiente comando



Nota: Considerar que debe tener instalado CLI AWS para ejecutar los comandos

1. Crear repositorio en AWS

Una vez que iniciamos sesión y tenemos la imagen Docker construida creamos un repositorio o contenedor en ECR de AWS con el siguiente comando

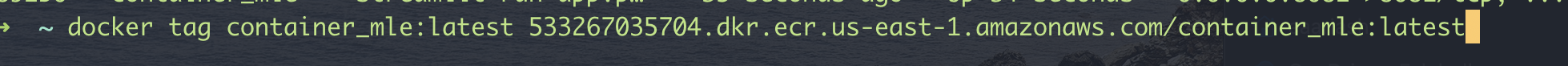


Una vez finalizada la creación del contenedor, aparecerá la metadata del contendor creado



1. Pushear container Docker a ECR AWS

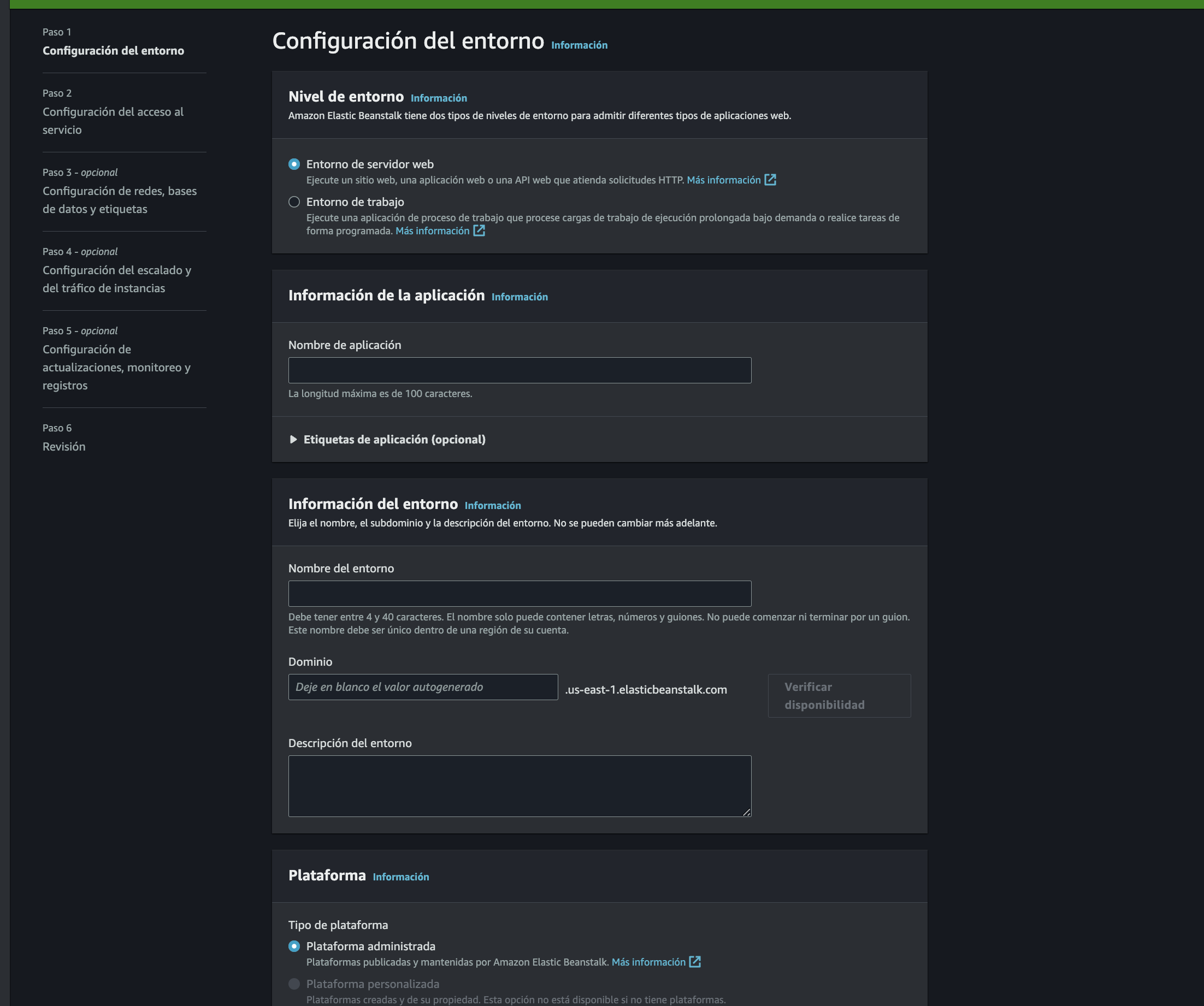
En el contenedor que creamos en AWS vamos a empujar nuestro contenedor local para eso debemos taguear el container con la estructura de ECR

  
luego empujamos (push)

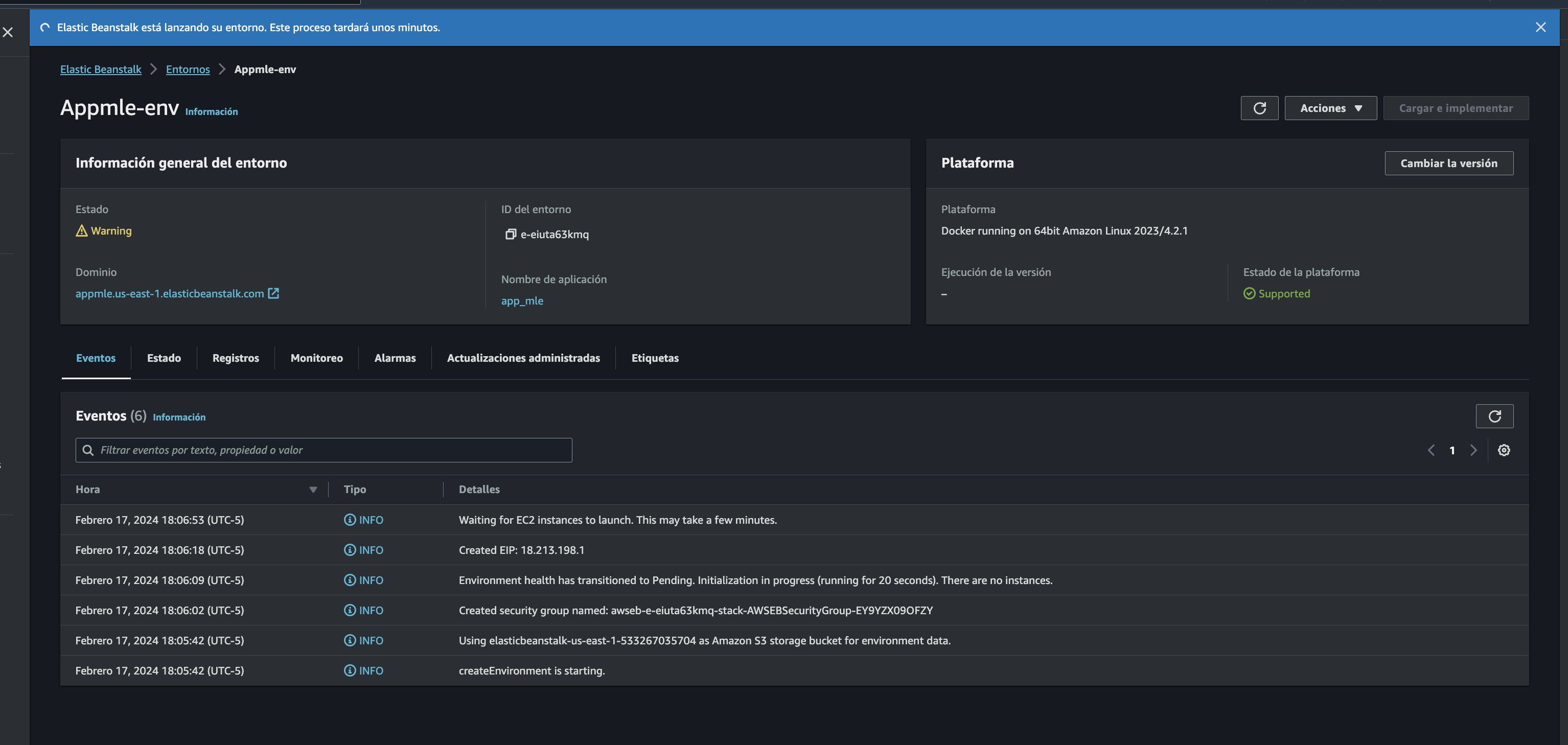


1. Creando entornno Elastic Beanstalk

Ahora debemos preparer los entornos para el despliegue CI/CD

Comenzamos creando el entorno de Elastic Beanstalk, para esto debemos llenar el formulario según lo visto en clases  


Al finalizar es importante crear el ECR Service Rol en el modulo IAM, según lo visto en clases



1. Crear el code Pipeline

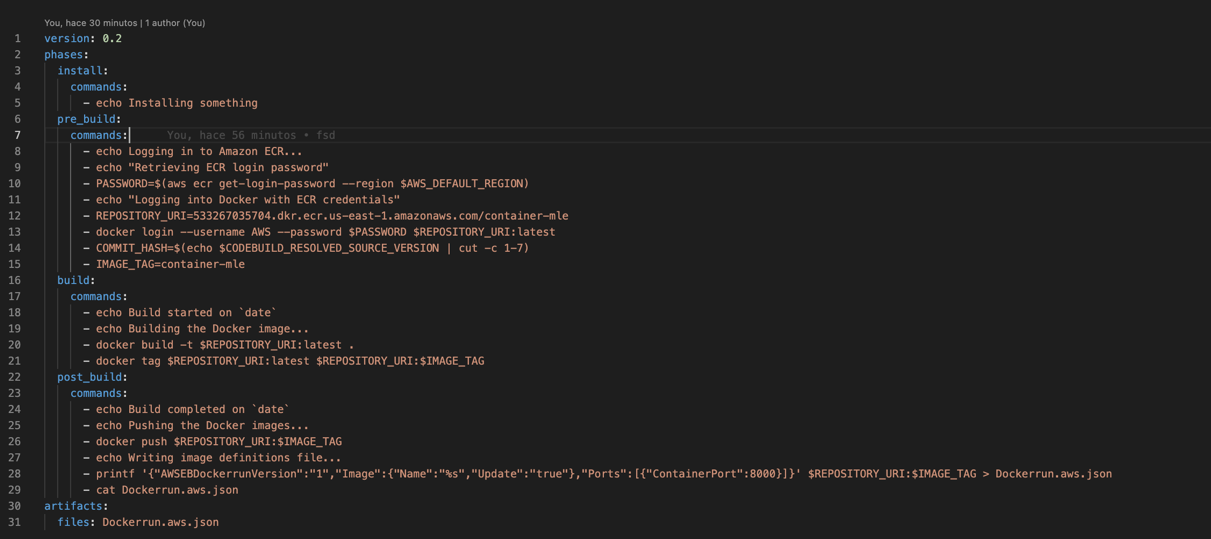
Segun lo visto enc lase llenar el formulario para crear el pipeline.

Una vez creado haremos las pruebas haciendo un push al repositorio de github

Que se configuro como origen del pipeline mediante una autorización a github que nos permite obtener un webhook

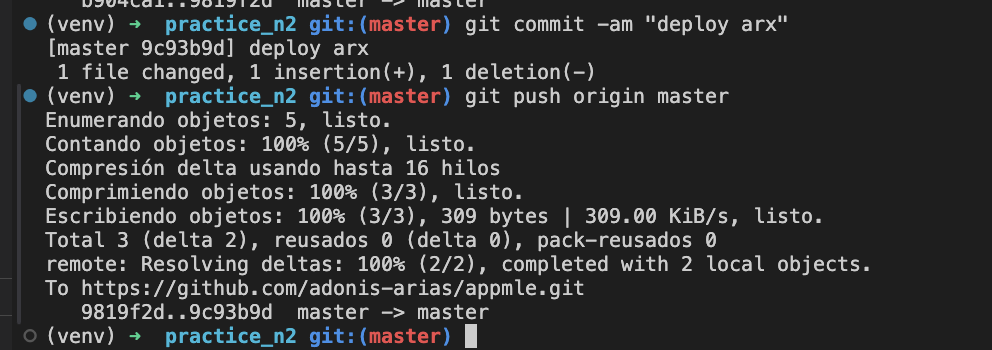
Los cambio que se hagan tambien son procesado por un codebuild que internamente lee un yml llamado builspec.yml donde vamos corriendo los comandos para setear las variables y generar un push hacia el contenedor en ECR que creamos inicialmente

En este script tambien se crea el awsdockerrun.json que es el output del codebuild que servirá para hacer la fase de despliegue en Elastic Beanstalk, este archivo contiene las configuraciones del docker ECR para correr la imagen

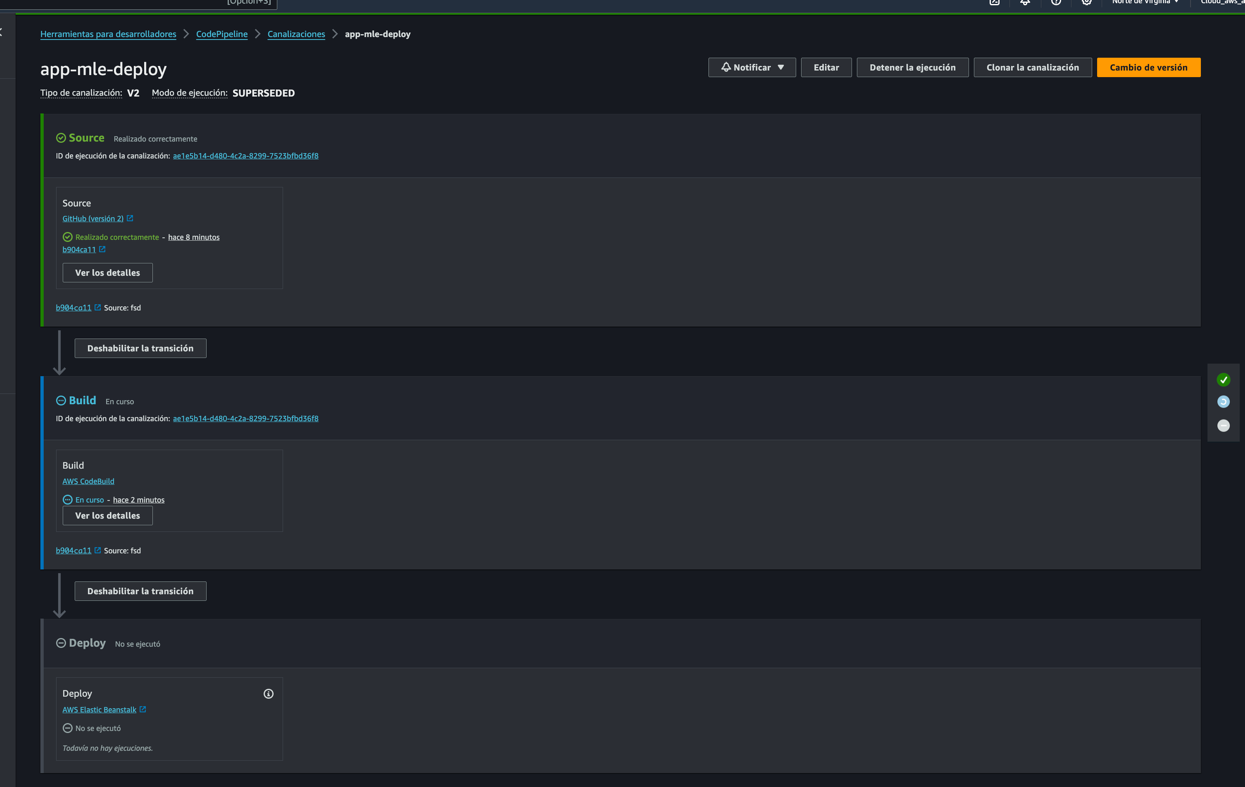


1. Ejecutar pipeline para el despliegue

Mandamos un cambio por github

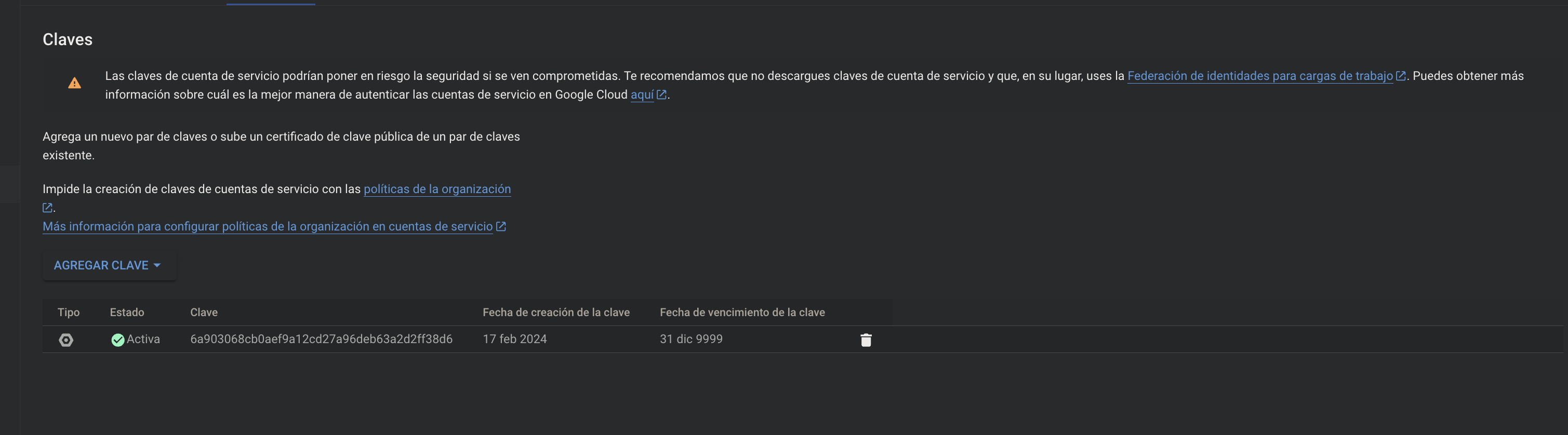


El pipeline se activa con un trigger y genera los pasos configurados anteriormente



DESPLIEGUE PIPELINE EN GCP

1. Creamos una cuenta servicio y una clave tipo json en el modulo de IAM



Validamos en permisos que ya este creado el nuevo service account   
