TALLER DE FASTAPI IN CONTAINERS – DOCKER

JOHANA PAOLA AVILA PEINADO

DOCENTE: SALOMON CADENA

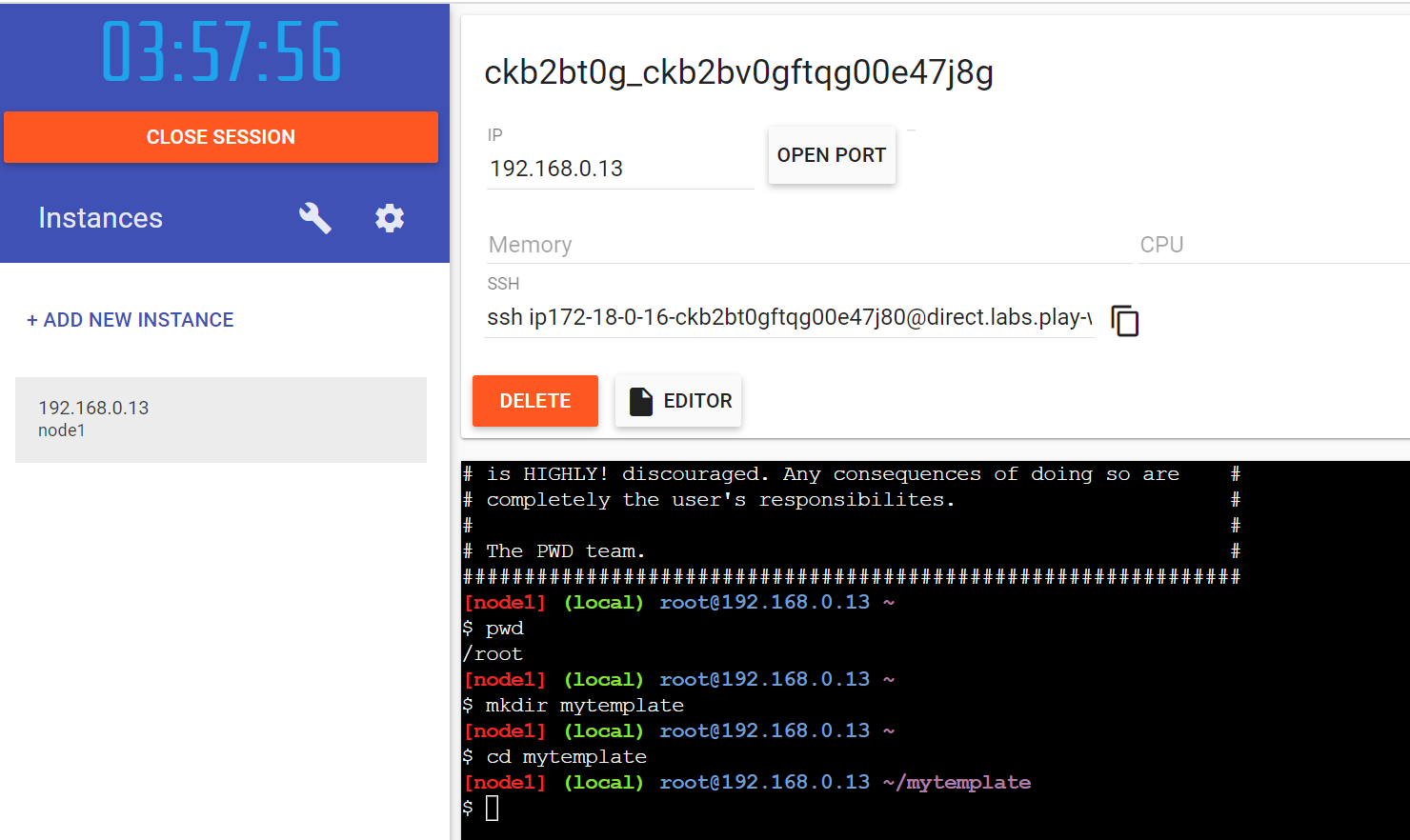
UNIVERSIDAD POPULAR DEL CESAR- FACULTAD DE INGENIERIA Y TECNOLOGIAS

VALLEDUPAR- CESAR

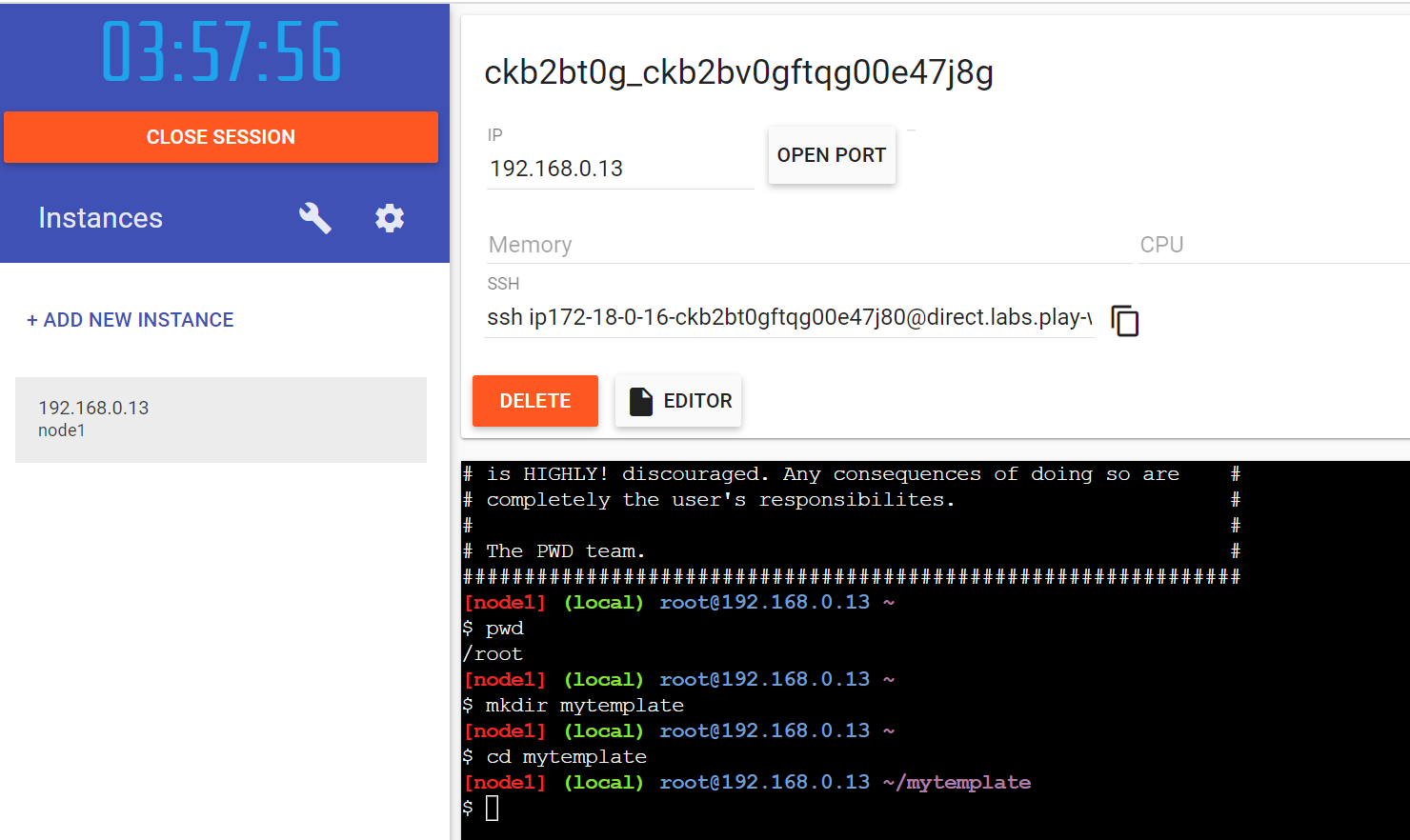
2023

Solución

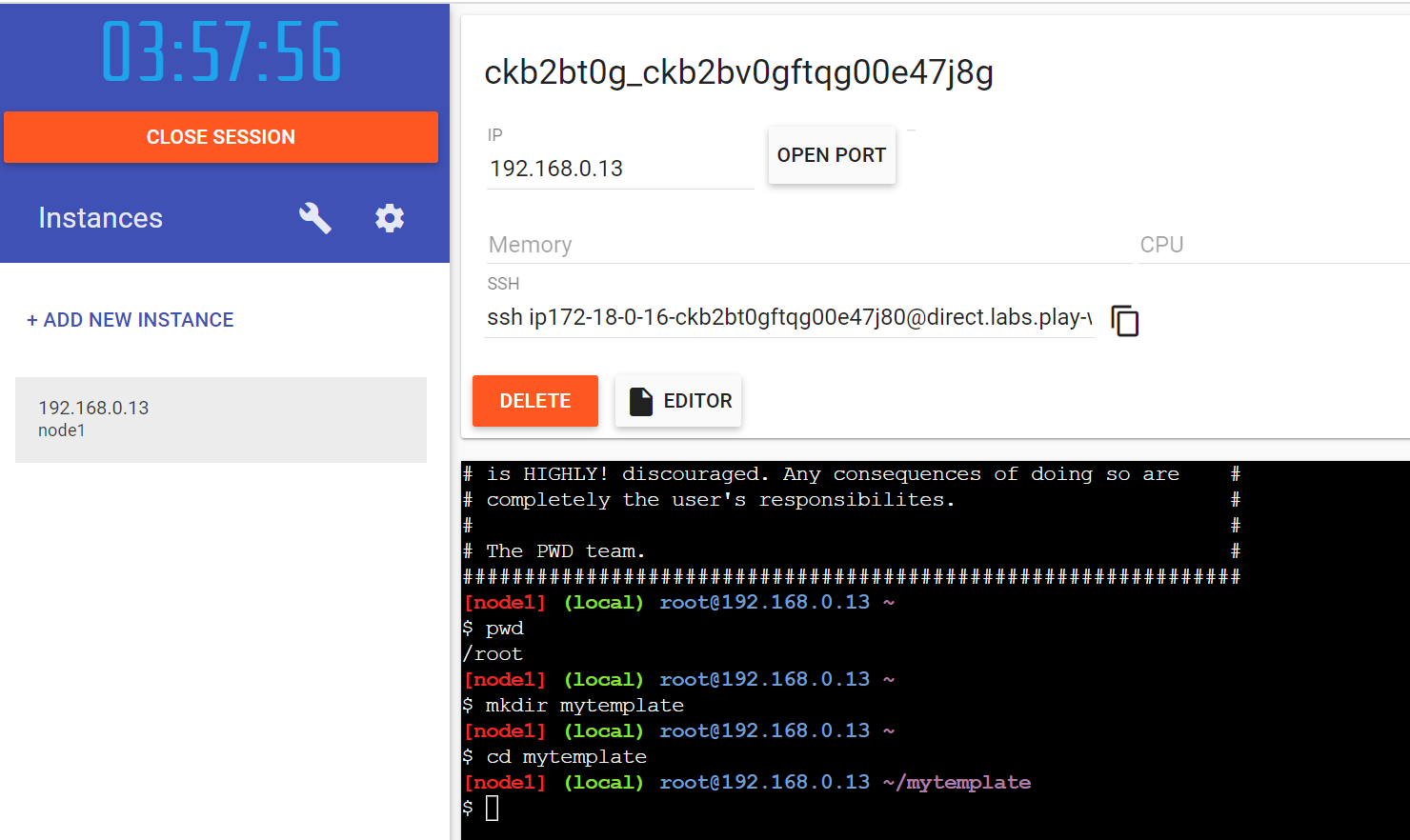
1.Iniciamos ejecutando el comando pwd que me muestra la ruta completa del directorio en el que me encontrada actualmente /root

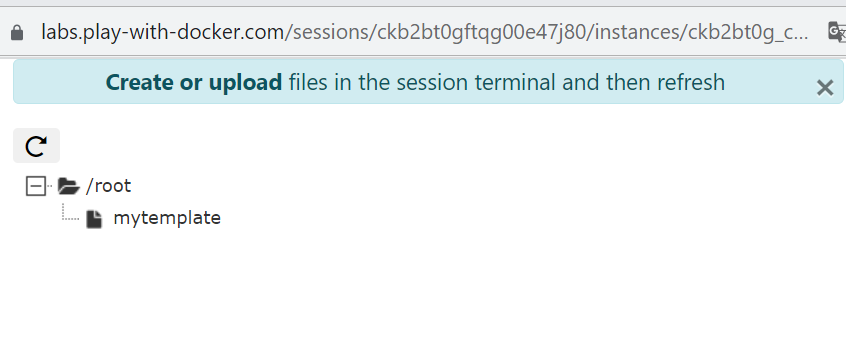


2. Creamos la carpeta mytemplate con el comando mkdir cuando se ejecuta este comando crea una carpeta en el directorio actual

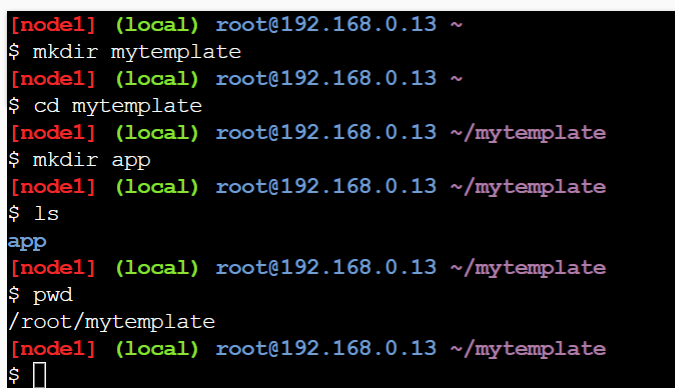


3.Utilizamos el comando cd para cambiar el directorio de trabajo actual por la carpeta previamente creada mytemplate después de ejecutar el comando, estamos dentro de la carpeta donde vamos a realizar diferentes operaciones.

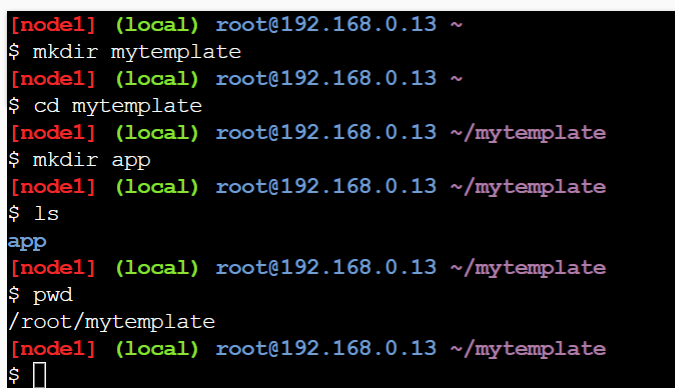




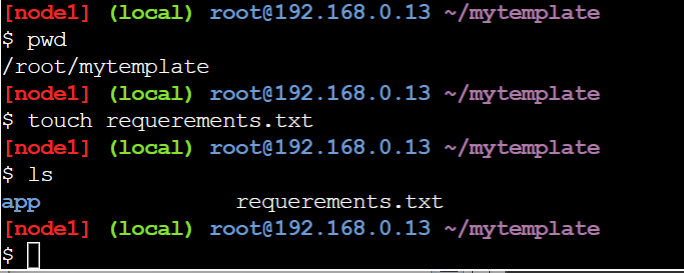
4. igualmente creamos un nuevo directorio con el nombre app

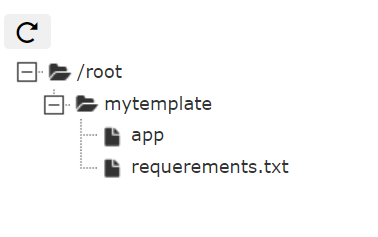


5. enlistamos los directorios con el comando ls que nos muestra el contenido del directorio mytemplate

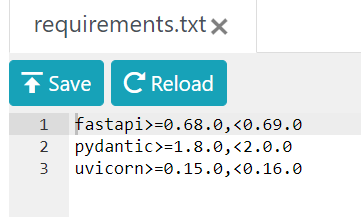


6.Con el comando touch requirements.txt, se crea un archivo vacío con el nombre requirements.txt en el directorio actual y se utiliza el comando ls para visualizar la correcta creación del archivo.

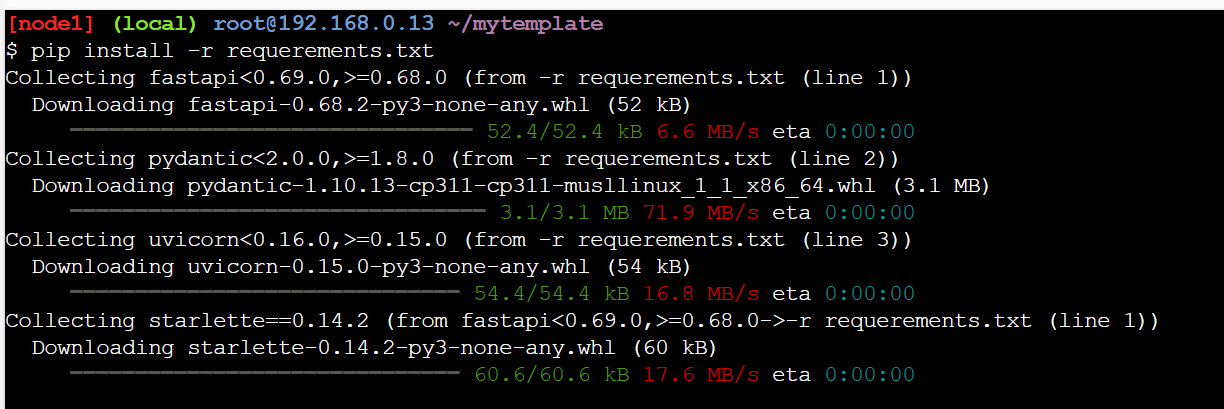




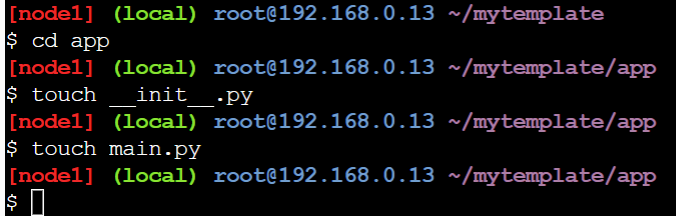
7.Añadimos la información al archivo



8. Utilizamos el comando pip install -r requirements.txt para instalar las dependencias y paquetes específicos en el archivo requerements.txt



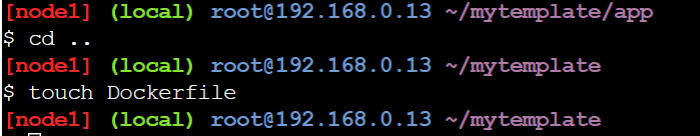
9. Ingresamos al directorio app y creamos dos archivos vacíos \_\_init\_.py y main.py



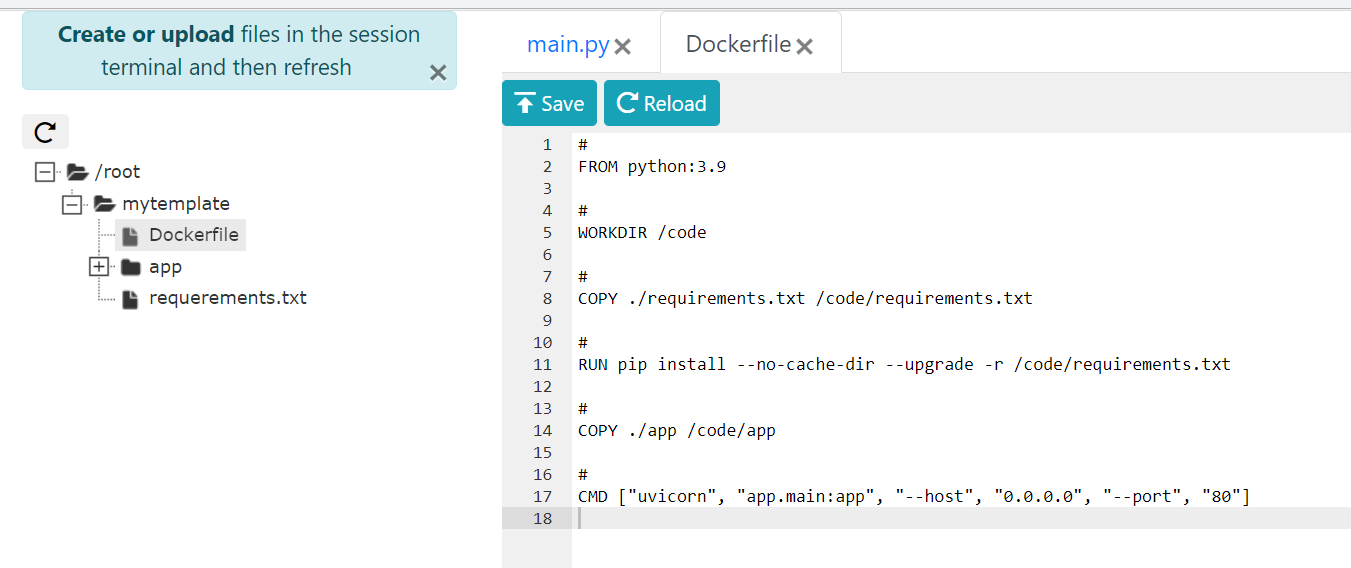
9. Añadimos la información al archivo main.py



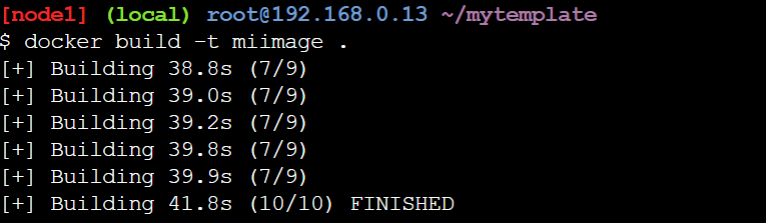
10. Salimos de la directorio app para crear un Archivo con el nombre Dockerfile en el directorio mytemplate



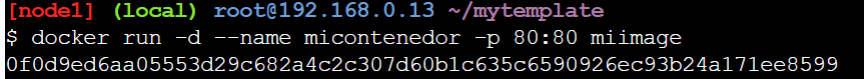
11.Añadimos la informacion al archivo Dockerfile



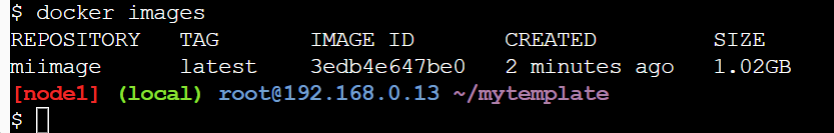
12. Utilizamos el comando Docker build -t para construir una imagen de contenedor con el nombre miimage . como dockerfile se encuentra en el directorio my template se usa (.) para referir el dicrectorio actual



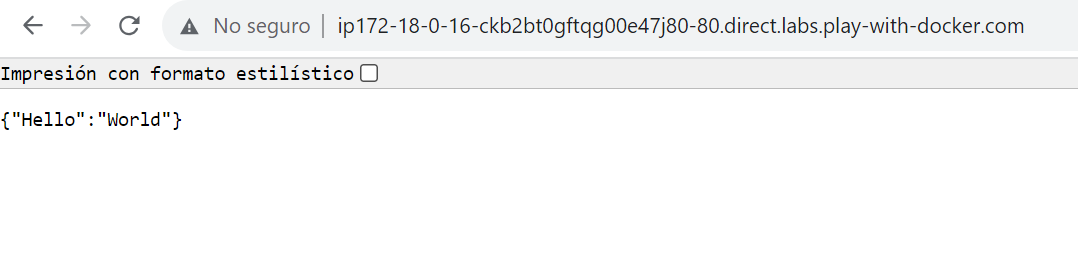
13.Iniciamso la ejecución de un contenedor a partir de la imagen creada anteriormente “miimage” con Docker, utilizamos la asignación de puestos entre el sistema host y el contenedor, el puerto 80 del sistema host se mapeara al puerto 80 del contenedor

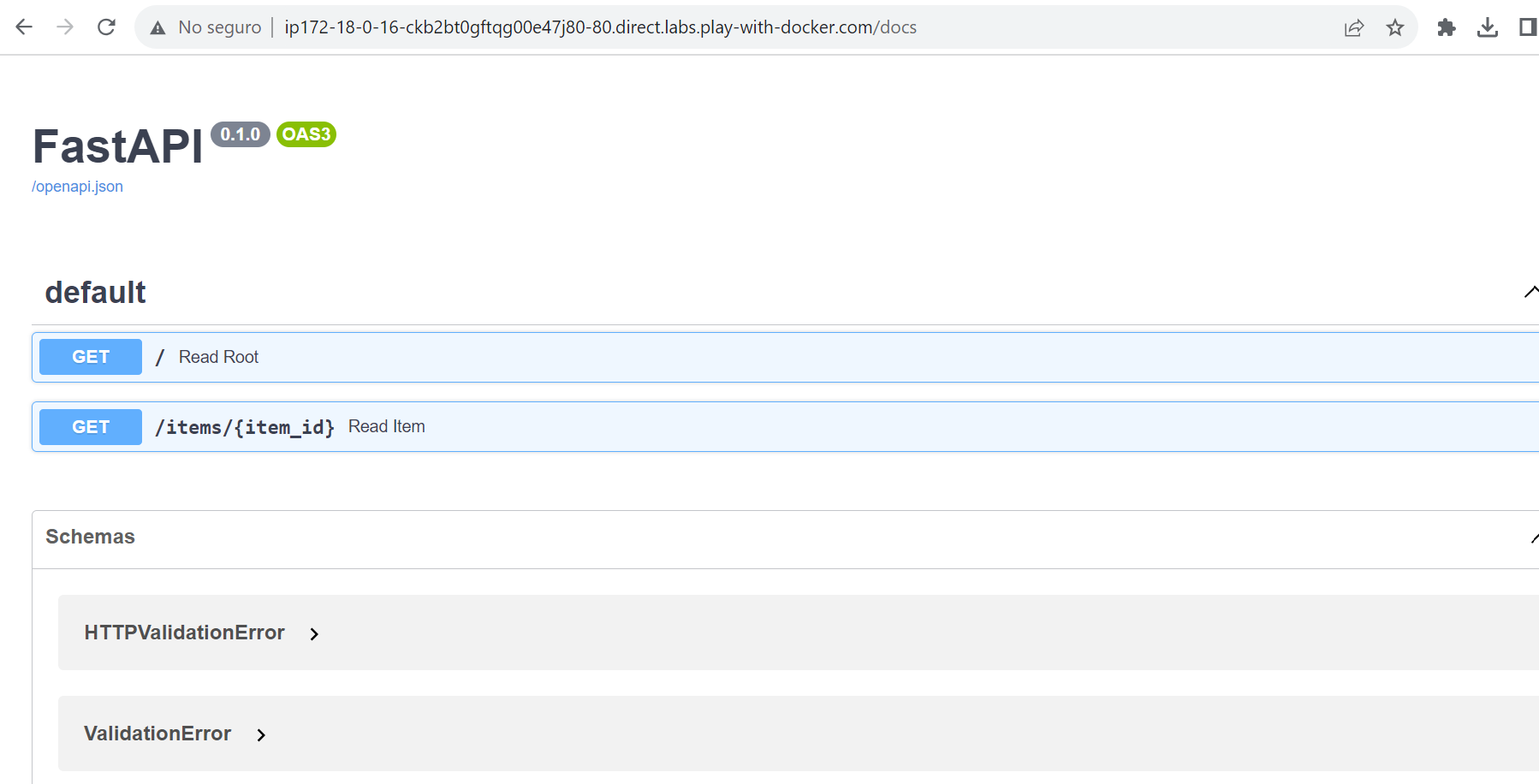


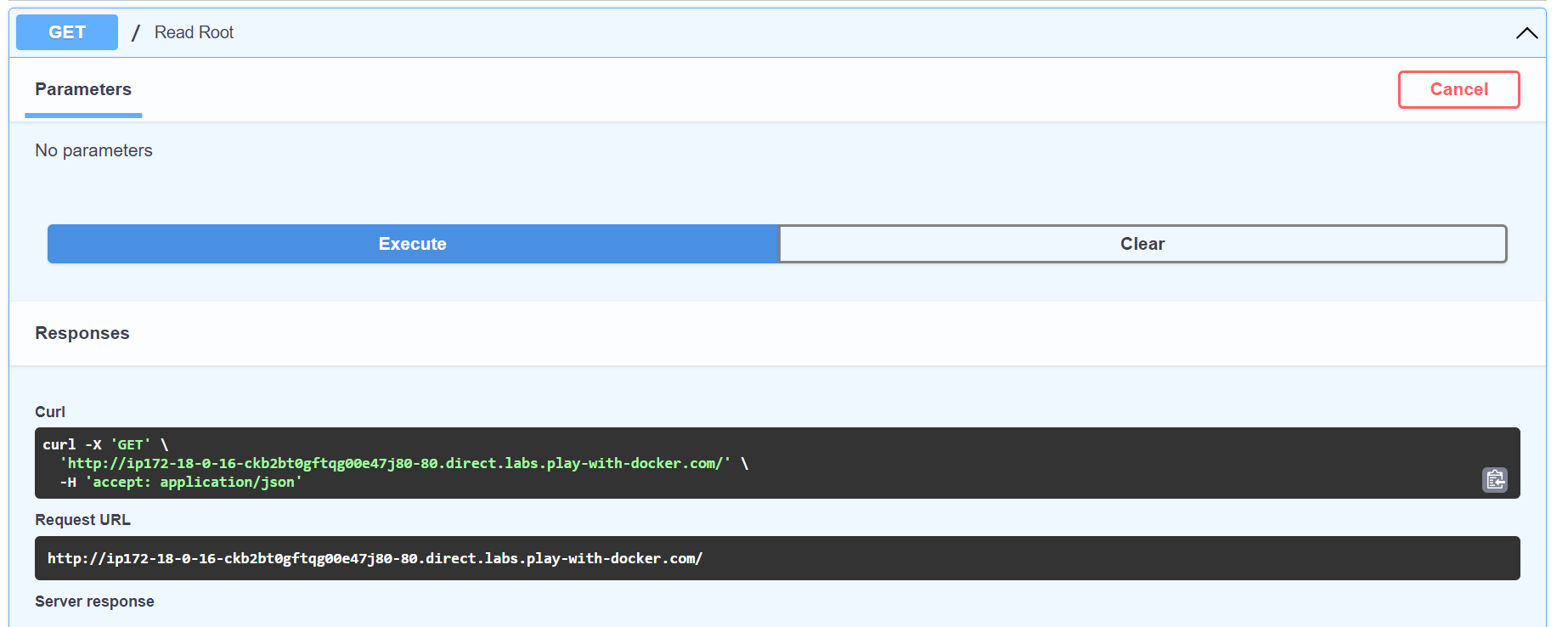
14. verificamos que la imagen de contenedor este almacenada localmente en mi sistema con la información como su nombre etiqueta, id de la imagen, tamaño y fecha de creacion

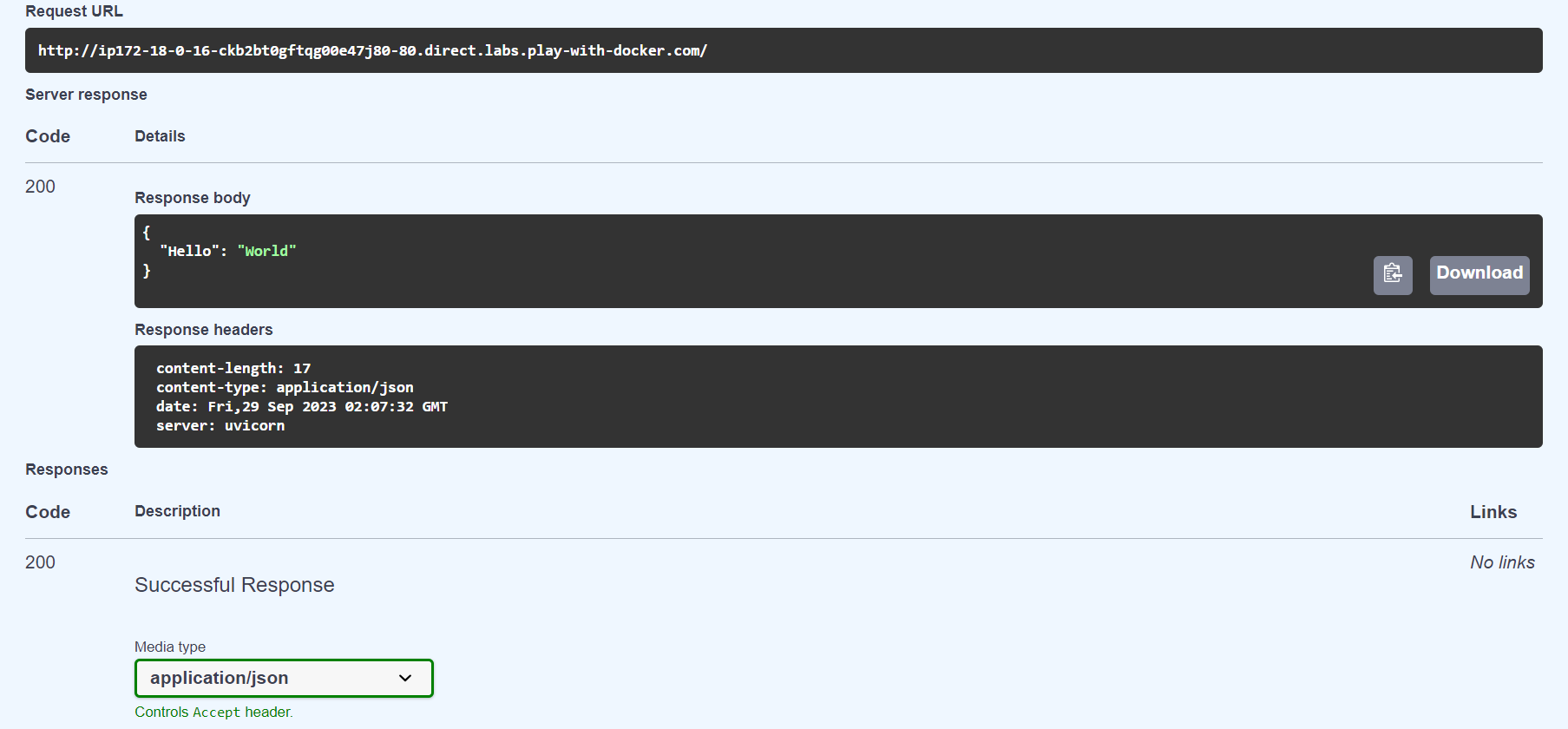


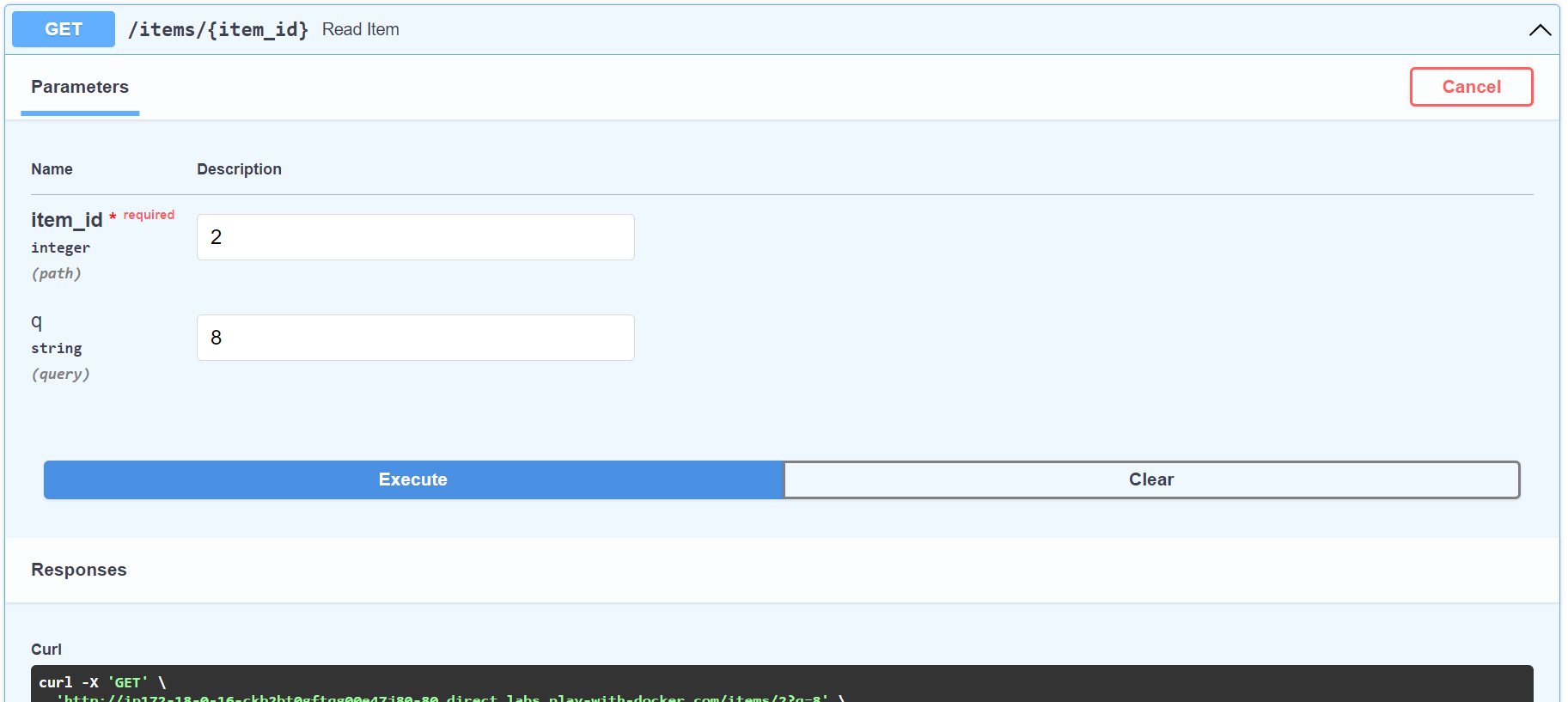
15. abrimos el puerto

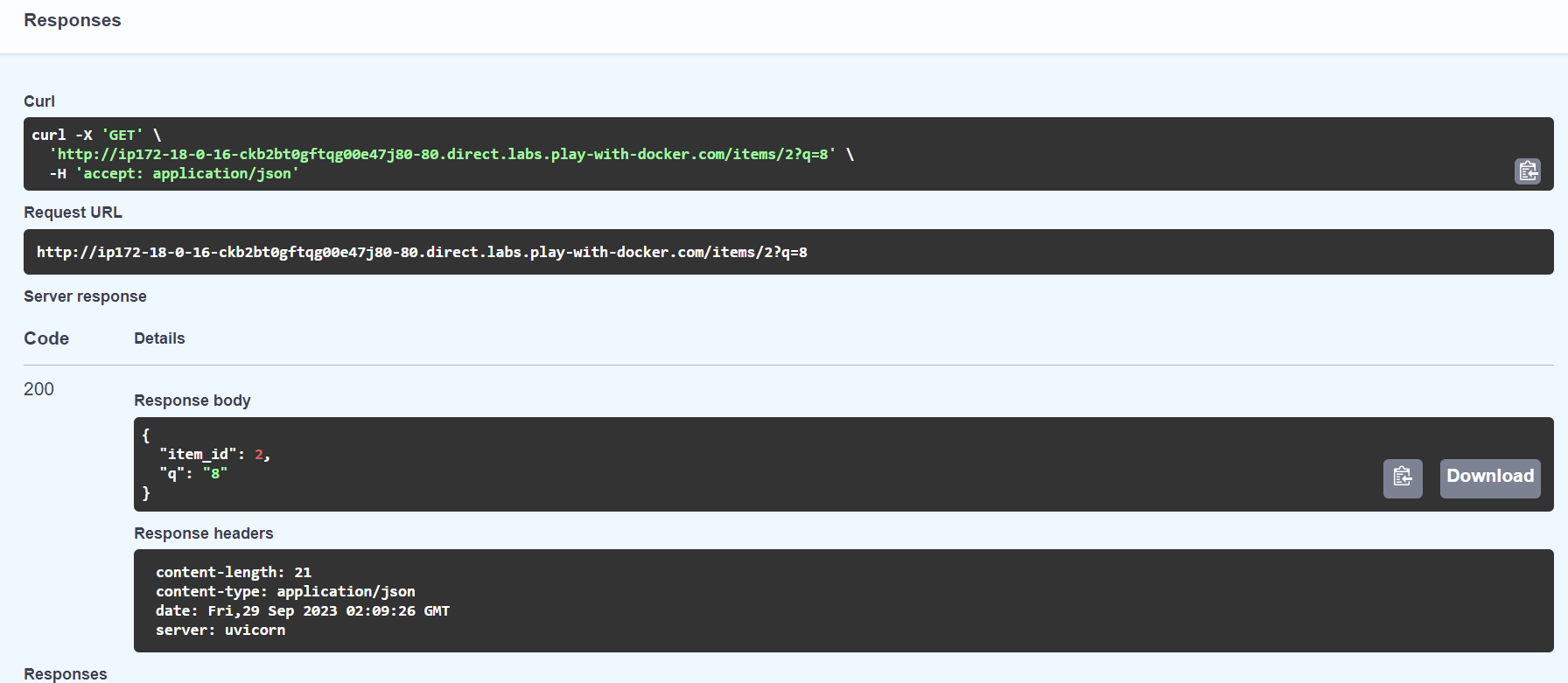












<https://github.com/JohanaAvii/FastApi_docker.git>