

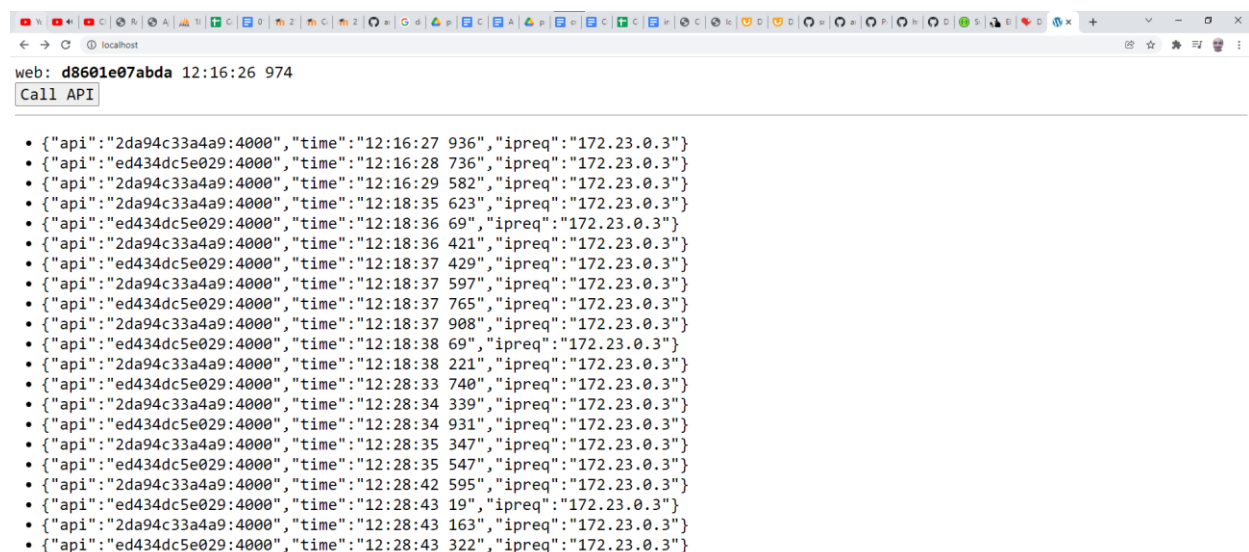
Repositorio: https://github.com/andogoiko/examen_docker_xddddddDDDDdddDddD.git

Contenido del directorio:

Andoitz_Evaluacion_2_Docker				
Nombre	Fecha de modificación	Tipo	Tamaño	
.git	25/01/2022 13:14	Carpeta de archivos		
Andoitz_Repositorio	26/01/2022 12:59	Documento de Mi...	210 KB	
Andoitz_Repositorio	25/01/2022 13:11	Microsoft Edge P...	359 KB	
apiserver	25/01/2022 12:18	JetBrains WebStorm	2 KB	
default	25/01/2022 13:08	Archivo CONF	1 KB	
docker-compose	25/01/2022 13:02	Archivo de origen ...	1 KB	
webserver	25/01/2022 12:18	JetBrains WebStorm	2 KB	

Verificación de funcionalidad:

Desde el webserver, funciona la llamada a las APIs, tal y como se ve en la captura.



¿Por qué he escogido estos puertos?

A el nginx por la 80: que es al que nos conectamos mediante localhost

Webserver por la 3000: el puerto que usa el nginx para mostrar por defecto el webserver (internamente)

Las APIs por la 4000: el puerto que utilizan las 2 APIs para el load balance dentro del nginx (internamente)

Muestra los comandos que:

Arrancan las aplicaciones directamente

```
node apiserver.js  
node webserver.js
```

- Arrancan las aplicaciones con docker run

```
docker run -it -v %CD%:/app -w /app -p 80:3000 node node webserver.js
```

```
docker run -it -v %CD%:/app -w /app -p 80:4000 node node apiserver.js
```

- Arrancan las aplicaciones orquestadas.

Se debe de crear un docker-compose.yml y ejecutar el siguiente comando, docker-compose up.