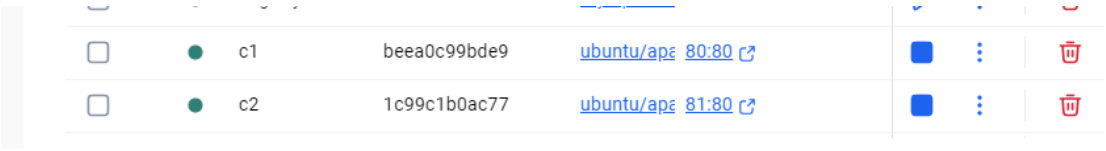
Contenedores con Docker. Proxy inverso con nginx

1. Se trata de emplear la imagen del servicio nginx de la práctica anterior, pero en este caso para habilitarlo como proxy inverso de tal modo que permita redirigir las peticiones que lleguen al puerto del proxy en la maquina anfitrión hacia las direcciones IP y puertos de los servidores web que tenemos en distintos contenedores. Para ello:

    a) Crea en tu sistema Docker dos contenedores de nombres c1 y c2 a partir de la imagen de ubuntu/apache2. Habilita un par de carpetas en la maquina real que permitan el acceso a los directorios /var/www/html de cada contenedor. No es necesario mapear puertos web en este caso ya que la gestión de acceso se realizará desde el proxy



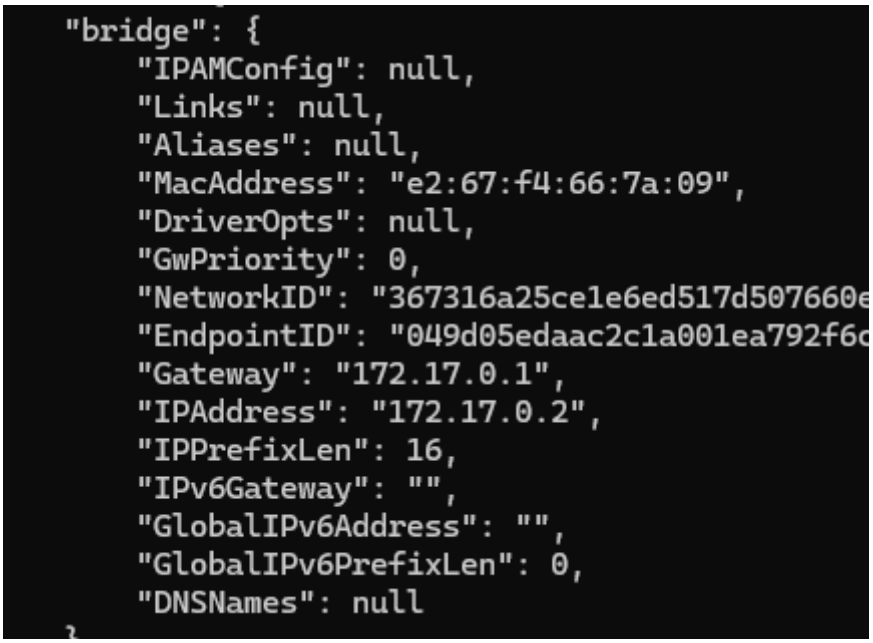
    b) Personaliza cada una de las páginas de inicio index.html empleando las carpetas que se habilitaron anteriormente o bien copia directamente los archivos index.html con el comando:

```
Id CommandLine
-- -----------
 1 docker cp index.html c1:/var/www/html
 2 docker cp index.html c1:/var/www/html
 3 dir
 4 docker cp index.html c1:/var/www/html
 5 cd ../
 6 cd ../../
 7 cd .\actividades\
 8 cd ejercicio3
 9 dir
10 cd du3_act3
11 cd ud3_act3
12 dir
13 cd ejercicio4
14 docker cp index.html c2:/var/www/html
15 cd C:\Users\fabriciog26\
16 docker inspect c1
```

```
docker cp index.html <nombre_contenedor>:/var/www/html
```

    c) Verifica con docker inspect o desde Docker Desktop, la dirección IP privada de cada uno de los contenedores (que estarán en la dirección de red 172.17.0.0/24)

C1



C2

d) Inicia tu contendor nginx y accede a una consola interactiva con este contenedor mediante el comando:

e) 

docker exec -it <nombre_contenedor> bash

o bien desde la consola accesible desde Docker Desktop

f) Una vez dentro debes acceder a la carpeta /etc/nginx/conf.d y crear dentro de ella un par de archivos nuevos de nombres sitio1.conf y sitio2.conf con contenido similar a este:

```
server {
    listen       80;
    server_name  www.sitioX.com;

    location / {
        proxy_pass http://172.17.0.X:80;
    }
}
```

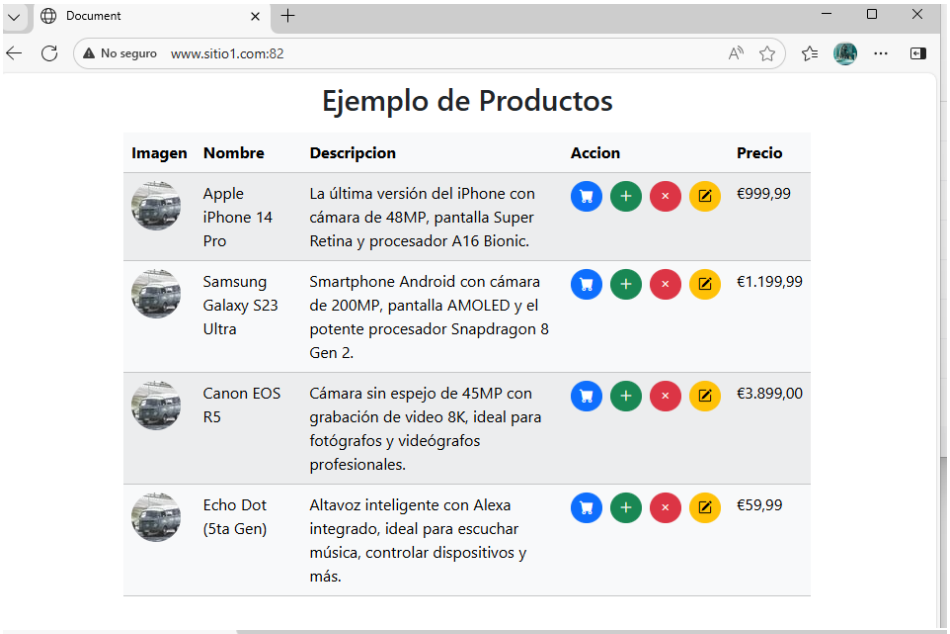
donde www.sitioX.com y 172.17.0.X deben ajustarse por los valores adecuados

No tome capturas de pantalla de pero en el otro se edita a sitio2.com y proxy pass acaba en 3

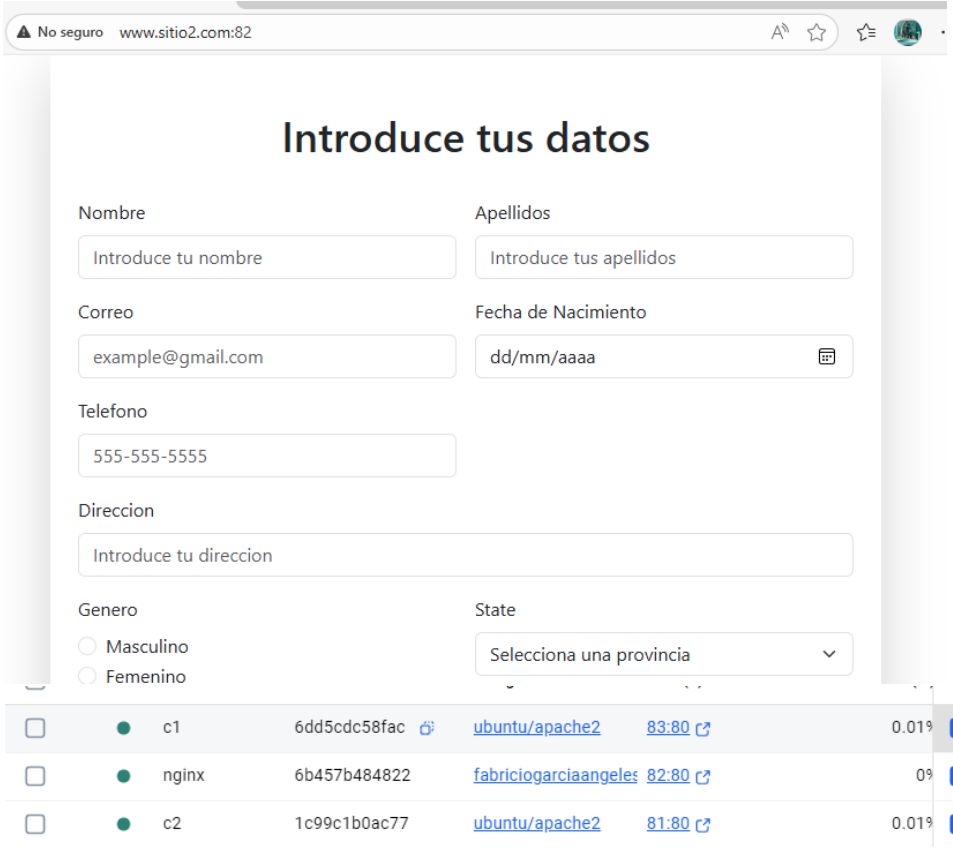
g) Una vez completado debe pararse y arrancar de nuevo el contenedor nginx

h) Finalmente, desde la maquina Windows 10 y en su fichero hosts debe establecerse los dos valores de www.sitioX.com como localhost y verificar desde un navegador cualquiera que al acceder a las direcciones www.sitio1.com y www.sitio2.com se visualizan las páginas index.html de cada uno de los dos contenedores

i)



j)



2. Acceso a contenedores desde VS Code. Se trata de habilitar otra vía de acceso al sistema de ficheros de los contenedores en ejecución. Para ello:

   a) Se debe instalar en VS Code la extensión “Docker for Visual Studio Code”

   b) Se nos creará un acceso a la extensión en la barra de tareas lateral izquierda. Desde ella se deben abrir los directorios /var/www/html de

cada contenedor con el propósito de modificar directamente el fichero index.html

c) Verificar desde el navegador que los cambios son efectivos