



ICC0008-UF3-PR01

Ana Cascante

Practica entregable (Enunciado y entregable)

## Guía para el alumno

El alumno debe de entregar la práctica enunciada en este documento antes del cierre programado en el calendario. La nota y corrección de la práctica se publicará en la plataforma en un plazo no superior a 10 días hábiles a contar desde la fecha límite de entrega.

Los entregables son:

- Este mismo documento incluyendo las respuestas solicitadas, los dibujos y las capturas de pantallas indicadas en el mismo:
  - **[CSnn]**: son capturas de pantalla que demuestran el resultado de una práctica. Para realizar la captura se utilizará la tecla "imp pant" o equivalente del teclado, y luego al final de documento y dentro de la página en blanco habilitada para cada captura, se realiza el "pegado" o "paste". Si fuera necesario se ajustará el tamaño.
  - **[DIBnn]**: Son esquemas que se solicitan en el enunciado de la práctica y pueden ser realizadas con cualquier herramienta dependiendo del tipo de dibujo, incluso escanear la imagen de un dibujo hecho a mano. También pueden ser completar tablas, escribir texto.

Utilizar los espacios habilitados al final de este documento para la inserción de las capturas y dibujos.

El documento entregado tendrá el siguiente nombre:

ICC0008-UF3-PR01-"username".pdf "username" = nombre de usuario del alumno en la plataforma Ejemplo: ICC0008-UF3-PR01-raulgarciaflores.pdf
---

**[DIB01]: 0,5 puntos reservados para pulcritud y correcta redacción de las explicaciones.**

Puntuación: 0,5

## Ejercicio #1 Docker

Durante las sesiones de esta UF, hemos montado un servidor Nginx en un contenedor de Docker.

## Ejercicio #1 - Tarea #1

Ahora es tu turno.

Monta un servidor en Apache dentro de un contenedor de Docker. Dicho servidor tiene que poder accederse desde el cliente de Ubuntu.

- El servidor tiene que servir una web tuya (puede ser básica o avanzada)
- Documenta totalmente el proceso.
- Explica los comandos que tiene el fichero Dockerfile de tu contenedor.
- Haz que tu web sea accesible desde el puerto 8080
- Mata el contenedor y borra el contenedor (no borres la imagen personalizada)

### [DIB01..x]: Capturas y explicaciones del proceso

Para instalar docker en el servidor de ubuntu he realizado los siguientes comandos:

- Apt install docker: instalar docker
- Docker pull ubuntu: para bajar imagen ubuntu del repositorio docker hub
- Me bajo la imagen de httpd con Docker pull httpd
- Después compruebo que docker funciona en la maquina virtual con el siguiente comando

```
root@anaserver:/home/anaserver# docker run -it ubuntu /bin/bash
root@f3b506011d62:/# ls
bin    dev    home  lib32  libx32  mnt    proc  run    srv    tmp    var
boot  etc    lib   lib64  media   opt    root  sbin   sys    usr
root@f3b506011d62:/# _
```

Le doy a exit, y creo el contenedor para comprobar que el puerto 80 este libre

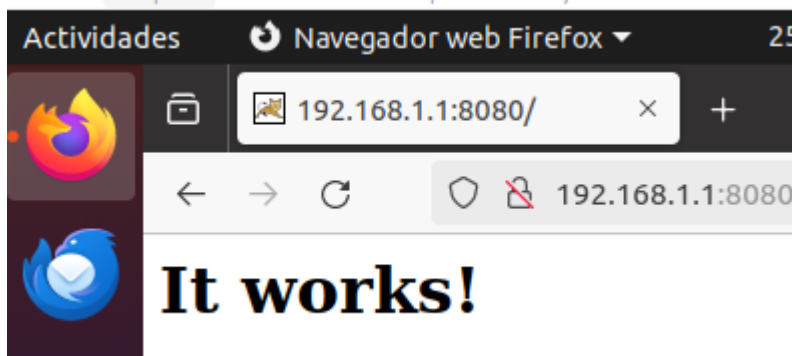
Me descargo la imagen de apache httpd

```
root@anaserver:/home/anaserver# docker run -d --name anadocker -p 8080:80 httpd
Unable to find image 'httpd:latest' locally
latest: Pulling from library/httpd
b0a0cf830b12: Extracting 28.02MB/29.15MB
```

Arranco contenedor Docker, y me error . porque el puerto 80 hace conflicto con tomcat, por lo tanto, apago tomcat y vuelvo a ejecutar y ya funciona e inicio el contenedor anadocker

```
root@anaserver:/home/anaserver# systemctl stop tomcat
root@anaserver:/home/anaserver# docker start anadocker
anadocker
root@anaserver:/home/anaserver# _
```

Y voy a la maquina cliente al puerto 8080 y compruebo que funciona



Hago un index.html en home

```
GNU nano 6.2
<html>
<head>
<title>Mi docker web</title>
</head>
<body>
<h1>Bienvenidos a mi docker web</h1>
<p>hola que tal</p>
</body>
</html>
```

```
root@anaserver:/home/anaserver# nano Dockerfile _
```

En el dockerfile, configuro que copio la pagina web y expongo el puerto 80 de httpd, para poder conectarnos y ver la pagina web

```
GNU nano 6.2
FROM httpd
COPY index.html /usr/local/apache2/htdocs/_
EXPOSE 80
```

Construyo mi propia imagen a partir de mi dockerfile

```

root@anaserver:/home/anaserver# docker build -t httpdana .
[+] Building 0.4s (7/7) FINISHED                                docker:default
=> [internal] load build definition from Dockerfile              0.0s
=> => transferring dockerfile: 103B                             0.0s
=> [internal] load .dockerignore                                0.0s
=> => transferring context: 2B                                    0.0s
=> [internal] load metadata for docker.io/library/httpd:latest  0.0s
=> [internal] load build context                                0.0s
=> => transferring context: 32B                                   0.0s
=> CACHED [1/2] FROM docker.io/library/httpd                   0.0s
=> [2/2] COPY index.html /usr/local/apache2/htdocs/           0.1s
=> exporting to image                                           0.1s
=> => exporting layers                                           0.1s
=> => writing image sha256:7397a2eb2e860ba4c54932443cd16b8ab6bc11e76b7a026832f1f56601b20a5e 0.0s
=> => naming to docker.io/library/httpdana                     0.0s
root@anaserver:/home/anaserver# _

```

Primero paro el contenedor , después lo elimino y ejecutamos un contenedor con el mismo nombre a partir de la imagen que he generado

```

root@anaserver:/home/anaserver# docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS
8a1b21400897   a11289bacbbd  "httpd-foreground"     7 minutes ago  Up 7 minutes  0.0.0.0:8080->80/t
cp, :::8080->80/tcp  anadocker
root@anaserver:/home/anaserver# docker stop anadocker
anadocker
root@anaserver:/home/anaserver# docker rm 8a1
8a1
root@anaserver:/home/anaserver# docker run -d --name anadocker -p 8080:80 httpdana
87fde53e694ba33e965b224da3dd1a57fd101f692a70c30ec0b5fa24f74bf183
root@anaserver:/home/anaserver#

```

En la maquina cliente ya podemos acceder a la web que esta en nuestro contenedor docker



Finalmente, mato y borro el contenedor

```

root@anaserver:/home/anaserver# docker stop anadocker
anadocker
root@anaserver:/home/anaserver# docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS
87fde53e694b   httpdana      "httpd-foreground"     3 minutes ago  Exited (0) 8 seconds ago
anadocker
root@anaserver:/home/anaserver# docker rm 87f
87f
root@anaserver:/home/anaserver# docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS        NAMES
root@anaserver:/home/anaserver# _

```

Puntuación: 6,75

[https://hub.docker.com/\\_/httpd](https://hub.docker.com/_/httpd)

## Ejercicio #1 - Tarea #2

Busca en Docker Hub una imagen que te parezca interesante. Puede ser cualquiera. Haz un pull de la imagen y crea el contenedor. Explica lo que hace el contenedor.

[DIB02.y]: Capturas del contenedor y explicación de lo que hace

Descargo la imagen de nginx de dockerhub

```
anacontainer
root@anaserver:/home/anaserver# docker pull nginx
Using default tag: latest
latest: Pulling from library/nginx
b0a0cf830b12: Already exists
8ddb1e6cdf34: Extracting 27.69MB/41.82MB
5858b8061118: Download complete
```

Compruebo el funcionamiento de docker

```
anacontainer
root@anaserver:/home/anaserver# docker run -it ubuntu /bin/bash
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
3c645031de29: Pull complete
Digest: sha256:1b8d8ff4777f36f19bfe73ee4df61e3a0b789caeff29caa019539ec7c9a57f95
Status: Downloaded newer image for ubuntu:latest
root@806406ad341e:/# ls
bin  dev  home  lib32  libx32  mnt  proc  run  srv  tmp  var
boot  etc  lib  lib64  media  opt  root  sbin  sys  usr
root@806406ad341e:/#
```

Creo el contenedor con la imagen de nginx

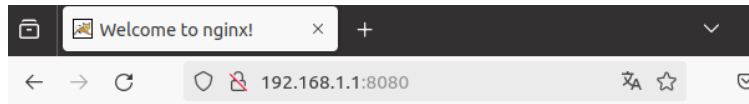
```
root@anaserver:/home/anaserver# docker run -d --name anacontainer -p 8080:80 nginx
4fcf213cdd4529b24f351c24e0859945b2a8303bfac1c1de847a649e90f7d14
root@anaserver:/home/anaserver# _
```

Apago el programa que pueda estar interfiriendo en el mismo puerto e Inicio el contenedor anacontainer

```
root@anaserver:/home/anaserver# systemctl stop tomcat
```

```
root@anaserver:/home/anaserver# docker start anacontainer
anacontainer
```

La página conecta correctamente en la maquina cliente



## Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to [nginx.org](http://nginx.org).  
Commercial support is available at [nginx.com](http://nginx.com).

*Thank you for using nginx.*

Con la imagen de nginx, el contenedor ejecutará una instancia de nginx en un entorno aislado, lo que significa que se puede comenzar a usar nginx para una aplicación sin necesidad de configurar y mantener una instalación de nginx en el sistema operativo anfitrión.

Puntuación: 2,75