

Tarea 4

Asignatura: Despliegue de aplicaciones web.

Ciclo Superior Desarrollo de Aplicaciones Web.



Condiciones de entrega.

La forma de entregar los ejercicios será en un fichero ZIP con el formato **ApellidosNombreTarea4.pdf**.

Puntuación.

El boletín está puntuado sobre 10 puntos. En caso de no entregarlo en fecha, hay una prórroga de 3 días en las que se puede entregar con penalización de 1 punto por día hasta un mínimo de 3.5. A partir del tercer día la tarea será puntuada con un cero.

El ejercicio debe funcionar cuando se ejecute.

La nota mínima para considerar aprobado el boletín es de 3.5 puntos.

Ejercicios.

Para hacer estos ejercicios debemos utilizar la máquina virtual de Docker.

La forma de entregar los ejercicios será en un PDF donde vayáis explicando, mediante pantallazos, cómo vais haciendo el ejercicio, y cada comando que utilizáis.

1. Ejercicio 1 (3 puntos)

- a. Crearemos un contenedor **Ubuntu** y ejecutaremos el comando “ls -l” dentro de él.

```
C:\Users\...>docker run -it ubuntu
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
423ae2b273f4: Pull complete
de83a2304fa1: Pull complete
f9a83bce3af0: Pull complete
b6b53be908de: Pull complete
Digest: sha256:04d48df82c938587820d7b6006f5071dbbffc7ca01d2814f81857c631d44d6f
Status: Downloaded newer image for ubuntu:latest
root@9230edb704c5:/# ls -l
bash: ls-l: command not found
root@9230edb704c5:/# ls -l
total 64
drwxr-xr-x  2 root root 4096 Feb 19 01:17 bin
drwxr-xr-x  2 root root 4096 Apr 24  2018 boot
drwxr-xr-x  5 root root 360 Mar  6 00:53 dev
drwxr-xr-x  1 root root 4096 Mar  6 00:53 etc
drwxr-xr-x  2 root root 4096 Apr 24  2018 home
drwxr-xr-x  8 root root 4096 May 23  2017 lib
drwxr-xr-x  2 root root 4096 Feb 19 01:15 lib64
drwxr-xr-x  2 root root 4096 Feb 19 01:14 media
drwxr-xr-x  2 root root 4096 Feb 19 01:14 mnt
drwxr-xr-x  2 root root 4096 Feb 19 01:14 opt
dr-xr-xr-x 125 root root  0 Mar  6 00:53 proc
drwx----- 2 root root 4096 Feb 19 01:17 root
drwxr-xr-x  1 root root 4096 Feb 21 22:20 run
drwxr-xr-x  1 root root 4096 Feb 21 22:20/sbin
drwxr-xr-x  2 root root 4096 Feb 19 01:14/srv
dr-xr-xr-x 13 root root  0 Mar  6 00:53 sys
drwxrwxrwt  2 root root 4096 Feb 19 01:17 tmp
drwxr-xr-x  1 root root 4096 Feb 19 01:14 usr
drwxr-xr-x  1 root root 4096 Feb 19 01:17 var
root@9230edb704c5:/#
```

- b. Comprobamos qué imagen se ha descargado

```
C:\Users\...>docker images
REPOSITORY          TAG          IMAGE ID          CREATED           SIZE
ubuntu              latest       72300a873c2c     13 days ago      64.2MB
C:\Users\...>
```

- c. Comprobamos qué contenedores hay activos

```
C:\Users\...>docker ps
CONTAINER ID   IMAGE          COMMAND           CREATED        STATUS        PORTS          NAMES
9230edb704c5   ubuntu        "/bin/bash"      11 minutes ago Up 11 seconds          gallant_grothendieck
```

- d. Eliminamos el contenedor

```
C:\Users\...>docker ps
CONTAINER ID   IMAGE          COMMAND           CREATED        STATUS        PORTS          NAMES
9230edb704c5   ubuntu        "/bin/bash"      15 minutes ago Up 4 minutes          gallant_grothendieck

C:\Users\...>docker rm -f 923
923

C:\Users\...>docker ps
CONTAINER ID   IMAGE          COMMAND           CREATED        STATUS        PORTS          NAMES
```

2. Ejercicio 2 (3 puntos)

- a. Crearemos un contenedor **Ubuntu**, con nombre **Ejercicio2** y ejecutaremos **de manera interactiva** la instrucción “bin/bash”.

```
C:\Users\c...>docker run -it --name Ejercicio2 ubuntu /bin/bash
root@423a558273a9:/#
```

- b. Comprobamos qué contenedores hay activos

```
C:\Users\c...>docker run -it --name Ejercicio2 ubuntu /bin/bash
root@40865a851d0c:/#
C:\Users\c...>docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
40865a851d0c	ubuntu	"/bin/bash"	47 seconds ago	Up 46 seconds		Ejercicio2

- c. Salimos de ese contenedor, **PARÁNDOLO**
- d. Comprobamos que el contenedor Ejercicio2 **NO** está activo
(Están c y d en la misma imagen)

```
C:\Users\c...>docker attach 408
root@40865a851d0c:/# exit
exit
C:\Users\c...>docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
--------------	-------	---------	---------	--------	-------	-------

- e. Arrancamos ese contenedor
- f. Comprobamos que el contenedor **ESTÁ ACTIVO**
(Están e y f en la misma imagen)

```
C:\Users\c...>docker start 408
408
C:\Users\c...>docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
40865a851d0c	ubuntu	"/bin/bash"	5 minutes ago	Up 2 seconds		Ejercicio2

- g. Borrarnos el contenedor **SIN PARARLO**

```
C:\Users\c...>docker rm -f 408
408
C:\Users\c...>docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
--------------	-------	---------	---------	--------	-------	-------

3. Ejercicio 3 (3 puntos)

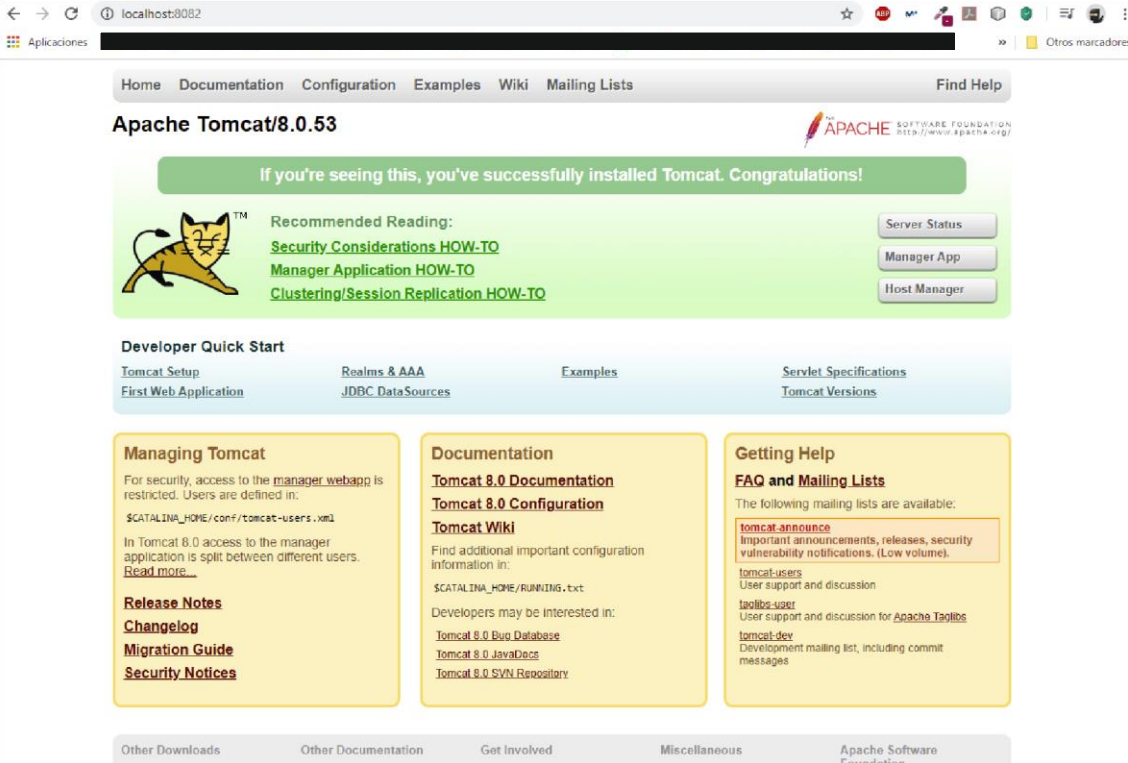
Queremos montar un servidor Tomcat para que nos sirva de servidor de aplicaciones. Para ello, lanzaremos un contenedor **Tomcat:8.0** con las siguientes características:

- Nombre: Ejercicio3
- Autor: Nombre del alumno
- Puerto público: 8082
- Puerto del contenedor: 8080

Nos conectaremos desde nuestro navegador al localhost:8082 y mostramos el resultado obtenido.

```
C:\Users\>docker run --name Ejercicio3 -e AUTHOR="Alberto" -d -p 8082:8080 tomcat:8.0
Unable to find image 'tomcat:8.0' locally
8.0: Pulling from library/tomcat
f189db1b88b3: Pull complete
3d06cf2f1b5e: Pull complete
edd0da9e3091: Pull complete
eb7768aae14e: Pull complete
e2780f585e0f: Pull complete
e5ed720afeba: Pull complete
d9e134700cfc: Pull complete
e4804b33d02a: Pull complete
b9df0c24315e: Pull complete
49fdae8aaa20: Pull complete
1aea3d9a32e6: Pull complete
Digest: sha256:8ecb10948deb32c34aeadf7bf95d12a93fbd3527911fa629c1a3e7823b89ce6f
Status: Downloaded newer image for tomcat:8.0
2eb077f4d02ab3fe833fd31158680f3b4aabfe5d115a0f703dfd606b5e13219f

C:\Users\>docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
2eb077f4d02a        tomcat:8.0         "catalina.sh run"   17 seconds ago     Up 16 seconds      0.0.0.0:8082->8080/tcp   Ejercicio3
```



4. Ejercicio 4 (1 puntos)

- a. Crearemos una imagen con el contenedor del Ejercicio3

```
C:\Users\>docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
2eb077f4d02a        tomcat:8.0         "catalina.sh run"   17 seconds ago     Up 16 seconds      0.0.0.0:8082->8080/tcp   Ejercicio3

C:\Users\>docker commit 2eb077f4d02a imagentomcatalberto
sha256:fd1b31d228ba7d9e5dbf794d785bf024c399c3776b3fc44595f19f9f06a0079f
```

- b. Ejecutaremos esa nueva imagen

```
C:\Users\ >docker run -d imagentomcatalberto
6902cd96b7000391deb8f2d4f8aa48f6a696a5d36cee6a4110d02fbe9ecbedf0

C:\Users\ >docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
6902cd96b700	imagentomcatalberto	"catalina.sh run"	11 seconds ago	Up 10 seconds	8080/tcp	inspiring_murdock
2eb077f4d02a	tomcat:8.0	"catalina.sh run"	18 minutes ago	Up 18 minutes	0.0.0.0:8082->8080/tcp	Ejercicio3

- c. Subiremos esa imagen a nuestro repositorio de Docker Hub

```
C:\Users\ >docker tag fd1b31d228ba albertoblancoabalde/imagentomcatalberto:firsttry

C:\Users\ >docker push albertoblancoabalde/imagentomcatalberto
The push refers to repository [docker.io/albertoblancoabalde/imagentomcatalberto]
79fff440ff00: Pushed
d0f3f4011f28: Pushed
583dc95d65c9: Pushed
f26731984f9b: Pushed
9f052711b40a: Pushed
81242e1e644e: Pushed
39a6e47c4ae6: Pushed
fc6174f0df4a: Pushed
425325c72d90: Pushed
c596d5191368: Pushed
daf45b2cad9a: Pushed
8c466bf4ca6f: Pushed
firsttry: digest: sha256:73bd569eb404d552d23ea15c1e241ee78ddaa903f8f3a109894d5c855f33ccae size: 2834
```