

ACTIVITATS AMPLIACIÓ-RE- FORÇ – UD2 SISTEMES OPERATIUS – INSTAL·LACIÓ

Activitats Pràctiques

Exercici 1 – Instal·lar docker a Ubuntu 22.04

Instalar Docker en Ubuntu 22.04

Hoy toca jugar y vamos a ver como instalar Docker en Ubuntu 22.04, no hay diferencia en la instalación con respecto a versiones anteriores, pero bueno, para que sepáis que esta soportada y que funciona correctamente.

Os dejo este otro post, donde instalaba Docker Engine en una VM con Debian 11.

Vamos a configurar los repositorios de Docker e instalar desde ellos, ya que facilita mucho Instalar Docker en Ubuntu o Debian. Además, es la manera más recomendable.

Empezamos actualizando los repositorios con **sudo apt update**.

A terminal window titled 'gorka@gorka-Docker: ~' with a dark background. The prompt is 'gorka@gorka-Docker:~\$'. The command 'sudo apt update' has been entered. The output shows the password prompt '[sudo] contraseña para gorka:', followed by four repository objects (Obj:1 to Obj:4) for jammy-security, jammy, jammy-updates, and jammy-backports. It then shows 'Leyendo lista de paquetes... Hecho', 'Creando árbol de dependencias... Hecho', 'Leyendo la información de estado... Hecho', and 'Todos los paquetes están actualizados.' The prompt returns to 'gorka@gorka-Docker:~\$' with a cursor.

```
gorka@gorka-Docker:~$ sudo apt update
[sudo] contraseña para gorka:
Obj:1 http://security.ubuntu.com/ubuntu jammy-security InRelease
Obj:2 http://es.archive.ubuntu.com/ubuntu jammy InRelease
Obj:3 http://es.archive.ubuntu.com/ubuntu jammy-updates InRelease
Obj:4 http://es.archive.ubuntu.com/ubuntu jammy-backports InRelease
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Todos los paquetes están actualizados.
gorka@gorka-Docker:~$
```

Una vez instalados, instalaremos los paquetes necesarios, con **sudo apt install ca-certificates curl gnupg lsb-release**

```
gorka@gorka-Docker: ~
gorka@gorka-Docker: $ sudo apt install ca-certificates curl gnupg lsb-release
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
ca-certificates ya está en su versión más reciente (20211016).
fijado ca-certificates como instalado manualmente.
gnupg ya está en su versión más reciente (2.2.27-3ubuntu2).
fijado gnupg como instalado manualmente.
lsb-release ya está en su versión más reciente (11.1.0ubuntu4).
fijado lsb-release como instalado manualmente.
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
  chromium-codecs-ffmpeg-extra gstreamer1.0-vaapi libgstreamer-plugins-bad1.0-0 libva-wayland2
Utilice «sudo apt autoremove» para eliminarlos.
Se instalarán los siguientes paquetes NUEVOS:
  curl
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 194 kB de archivos.
Se utilizarán 453 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [Y/n] y
Des:1 http://es.archive.ubuntu.com/ubuntu jammy-updates/main amd64 curl amd64 7.81.0-1ubuntu1.2 [194 kB]
Descargados 194 kB en 1s (226 kB/s)
Seleccionando el paquete curl previamente no seleccionado.
(Leyendo la base de datos ... 179445 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../curl_7.81.0-1ubuntu1.2_amd64.deb ...
Desempaquetando curl (7.81.0-1ubuntu1.2) ...
Configurando curl (7.81.0-1ubuntu1.2) ...
Procesando disparadores para man-db (2.10.2-1) ...
gorka@gorka-Docker: $
```

Una vez instalado añadiremos la clave GPG oficial de Docker con el siguiente comando:

curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg

```
gorka@gorka-Docker: ~
gorka@gorka-Docker: $ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
gorka@gorka-Docker: $
```

Con el siguiente comando configuraremos el repositorio estable

echo «deb [arch=\$(dpkg --print-architecture) signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu \$(lsb_release -cs) stable» | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null Nota: al loro si copiáis el comando desde aquí, las comillas dobles aparecen con otro carácter, cosas de WordPress.

```
gorka@gorka-Docker: ~
gorka@gorka-Docker: $ echo "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
gorka@gorka-Docker: $
```

Si listamos el directorio con **ls -l /etc/apt/sources.list.d/** , veremos que el nuevo repositorio estable de docker se ha añadido.

```
gorka@gorka-Docker: $ ls -l /etc/apt/sources.list.d/
total 4
-rw-r--r-- 1 root root 128 may 15 19:54 docker.list
gorka@gorka-Docker: $ cat /etc/apt/sources.list.d/docker.list
deb [arch=amd64 signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu jammy stable
gorka@gorka-Docker: $
```

Acto seguido, actualizaremos los repositorios con **sudo apt update**, donde ya veremos que coger del repositorio de Docker.

```
gorka@gorka-Docker: ~
gorka@gorka-Docker: $ sudo apt update
Des:1 https://download.docker.com/linux/ubuntu jammy InRelease [48,9 kB]
Obj:2 http://es.archive.ubuntu.com/ubuntu jammy InRelease
Obj:3 http://security.ubuntu.com/ubuntu jammy-security InRelease
Des:4 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages [5.109 B]
Obj:5 http://es.archive.ubuntu.com/ubuntu jammy-updates InRelease
Obj:6 http://es.archive.ubuntu.com/ubuntu jammy-backports InRelease
Descargados 54,0 kB en 1s (66,0 kB/s)
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Todos los paquetes están actualizados.
gorka@gorka-Docker: $
```

Instalado y actualizado, pasaremos a instalar Docker CE, con **sudo apt install docker-ce docker-ce-cli containerd.io docker-compose-plugin**

```
gorka@gorka-Docker: ~
gorka@gorka-Docker: $ sudo apt-get install docker-ce docker-ce-cli containerd.io docker-compose-plugin
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
  chromium-codecs-ffmpeg-extra gstreamer1.0-vaapi libgstreamer-plugins-bad1.0-0 libva-wayland2
Utilice «sudo apt autoremove» para eliminarlos.
Se instalarán los siguientes paquetes adicionales:
  docker-ce-rootless-extras docker-scan-plugin git git-man liberror-perl libslirp0 pigz slirp4netns
Paquetes sugeridos:
  aufs-tools cgroupfs-mount | cgroup-lite git-daemon-run | git-daemon-sysvinit git-doc git-email git-gui gitk gitweb git-cvs
  git-mediawiki git-svn
Se instalarán los siguientes paquetes NUEVOS:
  containerd.io docker-ce docker-ce-cli docker-ce-rootless-extras docker-compose-plugin docker-scan-plugin git git-man
  liberror-perl libslirp0 pigz slirp4netns
0 actualizados, 12 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 112 MB de archivos.
Se utilizarán 469 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 https://download.docker.com/linux/ubuntu jammy/stable amd64 containerd.io amd64 1.6.4-1 [28,1 MB]
Des:2 http://es.archive.ubuntu.com/ubuntu jammy/universe amd64 pigz amd64 2.6-1 [63,6 kB]
Des:3 http://es.archive.ubuntu.com/ubuntu jammy/main amd64 liberror-perl all 0.17029-1 [26,5 kB]
Des:4 http://es.archive.ubuntu.com/ubuntu jammy-updates/main amd64 git-man all 1:2.34.1-1ubuntu1.2 [952 kB]
Des:5 http://es.archive.ubuntu.com/ubuntu jammy-updates/main amd64 git amd64 1:2.34.1-1ubuntu1.2 [3.130 kB]
9% [5 git 1.148 kB/3.130 kB 37%] [1 containerd.io 3.706 kB/28,1 MB 13%]
```

Una vez instalados, comprobamos la versión que nos ha instalado de Docker, en este caso es la última. [Docker release notes](#)


```

gorka@gorka-Docker: ~
gorka@gorka-Docker:~$ sudo docker -v
Docker version 20.10.16, build aa7e414
gorka@gorka-Docker:~$
gorka@gorka-Docker:~$ sudo docker --help

Usage:  docker [OPTIONS] COMMAND

A self-sufficient runtime for containers

Options:
  --config string      Location of client config files (default "/root/.docker")
  -c, --context string  Name of the context to use to connect to the daemon (overrides DOCKER_HOST env var and default
                        context set with "docker context use")
  -D, --debug           Enable debug mode
  -H, --host list       Daemon socket(s) to connect to
  -l, --log-level string Set the logging level ("debug"|"info"|"warn"|"error"|"fatal") (default "info")
  --tls                Use TLS; implied by --tlsverify
  --tlscacert string    Trust certs signed only by this CA (default "/root/.docker/ca.pem")
  --tlscert string       Path to TLS certificate file (default "/root/.docker/cert.pem")
  --tlskey string        Path to TLS key file (default "/root/.docker/key.pem")
  --tlsverify           Use TLS and verify the remote
  -v, --version         Print version information and quit

Management Commands:
  app*      Docker App (Docker Inc., v0.9.1-beta3)
  builder   Manage builds
  buildx*   Docker Buildx (Docker Inc., v0.8.2-docker)
  compose*  Docker Compose (Docker Inc., v2.5.0)
  config    Manage Docker configs
  container Manage containers
  context    Manage contexts
  image     Manage images
  manifest  Manage Docker image manifests and manifest lists
  network   Manage networks
  node      Manage Swarm nodes
  plugin    Manage plugins
  scan*     Docker Scan (Docker Inc., v0.17.0)

```

Comprobada la versió, descarquem i llançem el primer contenidor de prova

```

gorka@gorka-Docker: ~
gorka@gorka-Docker:~$ sudo docker run -ti -d --name myubuntu ubuntu
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
125a6e411906: Pull complete
Digest: sha256:26c68657ccce2cb0a31b330cb0be2b5e108d467f641c62e13ab40cbec258c68d
Status: Downloaded newer image for ubuntu:latest
f59c5d6b5e66157de3e290282e95a8453dbc7a910a834681b3a85d439014e87f

```

I comprovem l'estat del nou contenidor amb `sudo docker ps -a`

```

gorka@gorka-Docker: ~
gorka@gorka-Docker:~$ sudo docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
f59c5d6b5e66   ubuntu   "bash"    36 seconds ago    Up 35 seconds           myubuntu

```

Exercici 2 – Instal·lar Wordpress en docker sobre Ubuntu 22.04

Como instalar wordpress usando Docker

Si quieres hacer una instalación local de wordpress, los pasos a seguir usando Docker son los siguientes.

NOTA: La instalación de Docker está pensada para Ubuntu, pero pueden replicarse en windows Mac u otra distribucion Linux.

Instalar Docker

Descubrir docker es algo que te cambia la vida. Finalmente puedes hacer instalaciones de sistemas de forma rápida. Ya no tienes miedo de desconfigurar tu maquina principal. Y puedes hacer cuantas pruebas requieras en la configuración de los servidores. Sin embargo, aun debes crear las imágenes y lanzar los containers de forma manual. Montar volúmenes, configurar las redes internas, etc. Para evitar ese trabajo manual es recomendable usar docker-compose. Docker-compose es una herramienta que interpreta un archivo de texto docker-compose.yml y ejecuta por ti todos los comandos docker correspondientes. En este caso, para instalar Docker, Docker-compose y todas sus dependencias necesarias ejecutamos el comando:

sudo apt install docker-compose

Una vez terminado, estamos listos para instalar nuestro wordpress.

Instalar Wordpress

Creamos una carpeta donde residirá el proyecto:

mkdir proyecto-wordpress

En dicha carpeta creamos un archivo ***docker-compose.yml*** :

version: '2' services: web: image: wordpress volumes: - ./wp-content:/var/www/html/wp-content/ ports: - 80: 80 depends_on: - db links: - db : db db: image: mysql: 5.7 environment: - MYSQL_ALLOW_EMPTY_PASSWORD=true - MYSQL_DATABASE=wordpress - MYSQL_USER=root

El archivo en cuestión le indica al sistema que se descargue la última versión de wordpress, que ejecute el servidor en el puerto 80 de nuestro localhost, y que la proyecto-wordpress/wp-content contenga toda la información que se sube a wordpress. Ahora procedemos a ejecutar el servicio de wordpress ejecutando el comando

docker-compose up -d

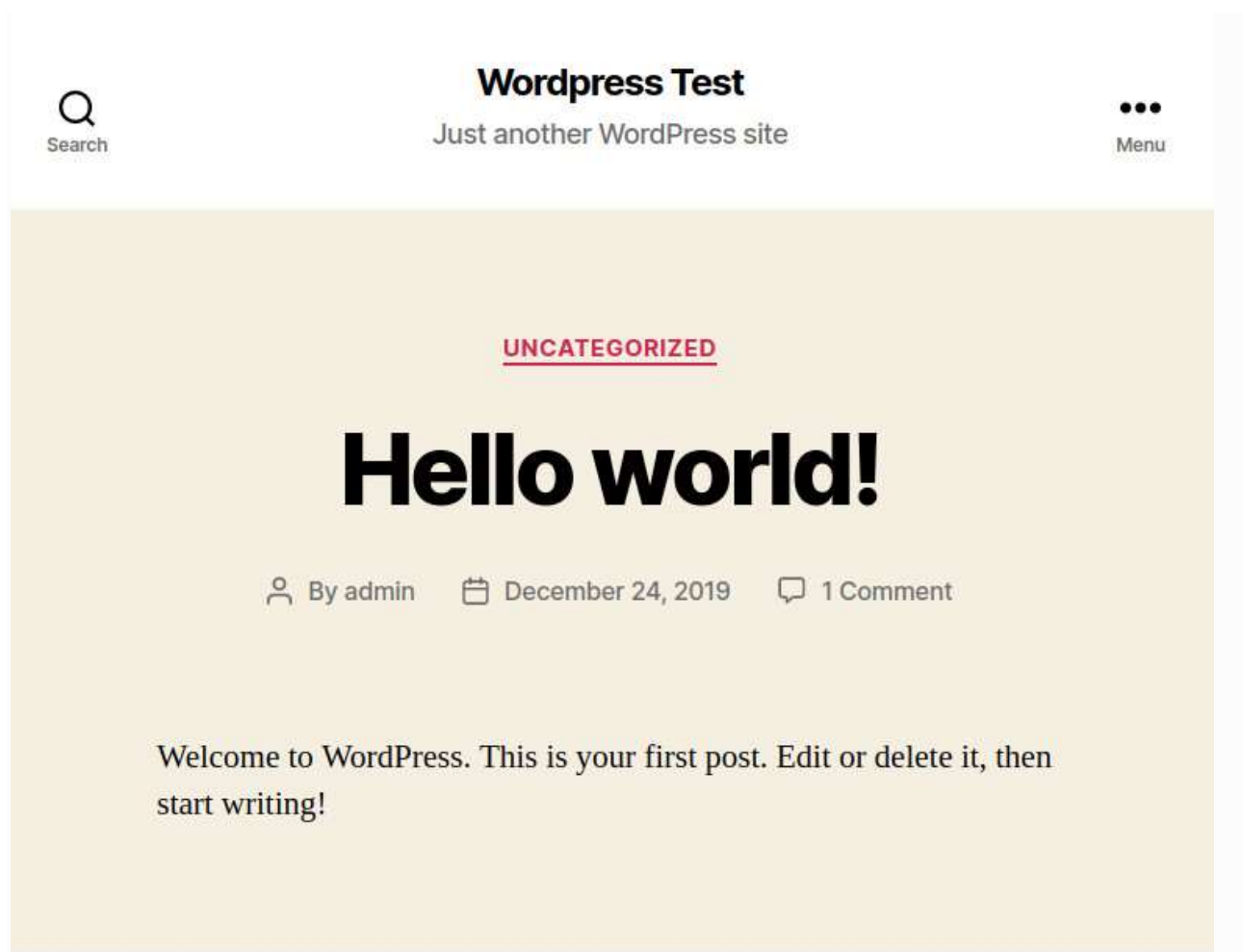
Al realizar la configuración inicial de wordpress a traves de localhost, debemos tomar en cuenta, que al usar docker el **Database Host** ya no sera localhost sino **db**. (En el archivo docker-compose.yml, en la seccion services => web => links => le pusimos el nombre db al servidor de la base de datos).



Below you should enter your database connection details. If you're not sure about these, contact your host.

Database Name	<input type="text" value="wordpress"/>	The name of the database you want to use with WordPress.
Username	<input type="text" value="root"/>	Your database username.
Password	<input type="password"/>	Your database password.
Database Host	<input type="text" value="db"/>	You should be able to get this info from your web host, if localhost doesn't work.
Table Prefix	<input type="text" value="wp_"/>	If you want to run multiple WordPress installations in a single database, change this.

Instalación Finalizada Exitosamente ;)



Comandos de Docker-Compose

El servicio de docker no es algo que debe estar permanentemente en memoria. Podemos prenderlo y apagarlo cuando lo necesitamos.

Ejecutar Wordpress

```
docker-compose up -d
```

Detener Wordpress

```
docker-compose stop
```

Borrar la instalación

```
docker-compose down
```

Revisar los logs

```
docker-compose logs web
```


Nota:

- Has d'elaborar un informe on expliques com has realitzat els diferents apartats o revisat els diferents punts sol·licitats en els exercicis. Realitza captures de pantalla, i edita-les (requadres, fletxes, etc.) de manera que et servisca de suport a l'explicació. Elabora l'informe com si es tractara d'un tutorial per a un usuari que no l'ha fet mai (no tan detallat com un manual, però sí amb text escrit donant suport a les captures de pantalla). Recorda que es valorarà positivament l'estètica i l'ordre (separació de la resposta per apartats, figures numerades, amb títol i referenciades en el text, etc.).