



Desarrolla en local y súbelo a la nube con Docker

Josep Maria Labarta



¿Qué es Docker?

¿Cómo instalar Docker en Windows-10?

¿Cómo instalar WordPress en Docker?

¿Cómo trabajar con Docker?

¿Que es Docker?

Docker es un programa que permite construir e implementar nuevas aplicaciones de forma más rápida en «**contenedores**» y ejecutarlo en máquinas **virtuales**.

La ventaja de esta forma de proceder es que los contenedores se ejecutan en entornos aislados, de forma que aumenta la seguridad del sistema.

¿Que es Docker?

Los contenedores Docker tienen todo el software y sus dependencias en una unidad estandarizada que incluye todo lo que necesitan para ejecutar: código, herramientas del sistema, librerías y ficheros de configuración, con lo que no hay software innecesario.

Esto garantiza que su aplicación siempre se ejecutará igual y hace que la colaboración sea tan simple como compartir una imagen del contenedor.

¿Cómo instalar Docker en Windows-10?

Debemos tener Windows-10 Pro o Enterprise

Las características de Windows de Hyper-V y contenedores deben estar habilitadas.

Si no tenemos una de estas versiones habrá que instalar Virtual Box, pero entonces posiblemente no merece ya la pena instalar Docker.

¿Cómo instalar Docker en Windows-10?

Para proceder a la instalación debemos descargar Docker desde:

<https://docs.docker.com/docker-for-windows/install/>

¿Cómo instalar WordPress en Docker?

Ahora que ya tenemos **Docker** corriendo, llega el momento de instalar WordPress en Docker con Windows-10, en local, usando «Docker-Compose».

Para ello primero vamos a crear una carpeta que voy a llamar Docker (tú puedes poner el nombre que quieras) en C:\Users\xxxx (o donde decidas).

¿Cómo instalar WordPress en Docker?

Dentro de esta carpeta vamos a tener que poner un archivo con el nombre «docker-compose.yml» (este si que debe nombrarse tal cual) y dentro de este archivo deberá ir todo el código para hacer la instalación de WordPress + Mysql + PhhMyAdmin.

Este es el código en cuestión:


```
version: '3.1'
services:
  wordpress:
    container_name: meetup-wp
    image: wordpress:latest
    restart: always
    ports:
      - "80:80"
    volumes:
      - ./wordpress:/var/www/html
    environment:
      WORDPRESS_DB_HOST: db:3306
      WORDPRESS_DB_USER: admin
      WORDPRESS_DB_PASSWORD: admin1234
      WORDPRESS_DB_NAME: wordpress
  db:
    container_name: meetup-db
    image: mariadb:10.3
  restart: always
  environment:
    MYSQL_DATABASE: wordpress
    MYSQL_USER: admin
    MYSQL_PASSWORD: admin1234
    MYSQL_RANDOM_ROOT_PASSWORD: '1'
  phpmyadmin:
    container_name: meetup-php
    depends_on:
      - db
    image: phpmyadmin/phpmyadmin
    links:
      - db
    ports:
      - "8080:80"
    restart: always
```

Puedes cambiar las claves de usuario y password en WordPress y en Mysql.

Encontrarás en internet muchos ejemplo de ficheros «docker-compose.yml» y algunos bastante diferentes al que yo he puesto, por supuesto todos son válidos y puedes probar los que quieras y quedarte con el que te funcione mejor o más te guste.

Los hay con diferentes versiones de WordPress, o de Mysql. También algunos sin la opción de phpMyAdmin.

Si te interesa puedes buscarlo con Mariadb. En fin que tienes donde escoger e ir probando.

Ya con todo esto preparado vamos a «PowerShell» y escribes «cd docker» para entrar en la carpeta, luego «docker-compose up» para instalar WordPress

Versiones:

- Docker 1.24.1
- mariadb:10.4.8 / mysql:8.0.18
- apache:2.4.41 / nginx:1.17.5
- php:7.3.11
- wp:5.2.4

phpmyadmin/phpmyadmin:

¿ Cómo trabajar con Docker?

Tenemos básicamente tres formas de hacerlo:

- Windows PowerShell (utilizando comandos)
- Kitematic (incluido en Docker , es visual y muy básico)
- Con un editor de código

Cuando ya tenemos los contenedores creados:

Arrancar solo los contenedores que nos hagan faltan.

Por ello es conveniente darles un nombre y número que nos indiquen para que son.
Los contenedores se arrancan de forma muy fácil desde un editor de código.

Hay varios editores que tienen complementos para Docker:

- Eclipse
- Visual Studio Code
- Sublime Text 3
- Atom
- Emacs
- Vim

COMANDOS DOCKER-COMPOSE

Instalar: `docker-compose up`

Instalar: `docker-compose up -d` (para que se ejecute en segundo plano)

Parar: `docker-compose stop`

Arrancar: `docker-compose start`

Pausar: `docker-compose pause`

COMANDOS DOCKER-COMPOSE

Ver que hay: `docker ps -a`

Ver containers: `docker-container ls --all`

Ver imágenes: `docker images`

Eliminar: `docker-compose down -v` (elimina todo)

Eliminar todos los contenedores: `docker rm $(docker ps -a -q)`

Eliminar container : `docker rm Idcontainer`

Eliminar imágenes: `docker rmi ID-imagen`

WORDPRESS:

Puertos

Hay que utilizar diferentes puertos si queremos instalar 5 WordPress en local

localhost:80 = wordpress

localhost:8080 = phpMyAdmin

localhost:82 = wordpress2

localhost:8082 = phpMyAdmin-2

localhost:83 = wordpress3

localhost:8083 = phpMyAdmin-3

localhost:84 = wordpress4

localhost:8084 = phpMyAdmin-4

localhost:85 = wordpress5

localhost:8085 = phpMyAdmin-5

URLs:

Docker:

<https://www.docker.com/>

<https://www.docker.com/products/docker-desktop>

<https://hub.docker.com/>

<https://docs.docker.com/compose/wordpress/>

Docker-compose:

<https://github.com/nezhar/wordpress-docker-compose>

<https://github.com/algolia/docker-compose-wordpress/blob/master/docker-compose.yml>

INFORMACION SOBRE DOCKER:

- <https://labarta.es/instalar-docker-en-windows-10/>
- <https://labarta.es/instalar-wordpress-en-docker-con-windows-10/>
- <https://labarta.es/trabajar-con-docker-en-visual-studio-code>

Y Ahora al por faena

Vamos a poner en practica todo lo que hemos visto

En vivo y en directo



¡ Muchas gracias por vuestra atención !

Link para descargar esta presentación:

<https://labarta.es/go/wordpress-con-docker/>


Prueba para ver si funciona Docker

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

PS C:\Users\José> docker version
Client: Docker Engine - Community
 Version:           18.09.2
 API version:       1.39
 Go version:        go1.10.8
 Git commit:        6247962
 Built:             Sun Feb 10 04:12:31 2019
 OS/Arch:           windows/amd64
 Experimental:      false

Server: Docker Engine - Community
 Engine:
  Version:           18.09.2
  API version:       1.39 (minimum version 1.12)
  Go version:        go1.10.6
  Git commit:        6247962
  Built:             Sun Feb 10 04:13:06 2019
  OS/Arch:           linux/amd64
  Experimental:      false
PS C:\Users\José> █
```


> Windows 10 (C:) > Usuarios > José > Docker > wordpress3

Nombre	Fecha de modifica...	Tipo
 docker-compose	30/06/2019 13:13	Archivo YML

Instalación de WordPress con Docker Compose

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.





















Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/powershell

PS C:\Users\José> cd docker
PS C:\Users\José\docker> cd wordpress
PS C:\Users\José\docker\wordpress> docker-compose up -d
```

```
Windows PowerShell

PS C:\Users\José\docker\wordpress> docker-compose up -d
Pulling db (mysql:8.0.16)...
8.0.16: Pulling from library/mysql
fc7181108d40: Already exists
787a24c80112: Pull complete
a08cb039d3cd: Pull complete
4f7d35eb5394: Pull complete
5aa21f895d95: Pull complete
a742e211b7a2: Pull complete
0163805ad937: Pull complete
87f18876c3ff: Pull complete
78082f25f167: Pull complete
0a510f055c17: Pull complete
312b0999e433: Pull complete
f864cfdc0264: Pull complete
Creating wp-1 ... done
Creating mysql-1 ... done
Creating phpmyadmin-1 ... done
PS C:\Users\José\docker\wordpress>
```


Vista de los ficheros de WordPress en nuestra instalación local con Docker

Nombre	Fecha de modificación	Tipo	Tamaño
 wp-admin	18/06/2019 19:50	Carpeta de archivos	
 wp-content	03/07/2019 22:07	Carpeta de archivos	
 wp-includes	18/06/2019 19:50	Carpeta de archivos	
 .htaccess	03/07/2019 22:06	Archivo HTACCESS	1 KB
 index	01/12/2017 0:11	Archivo PHP	1 KB
 license	01/01/2019 21:37	Documento de tex...	20 KB
 readme	09/04/2019 0:59	Chrome HTML Do...	8 KB
 wp-activate	12/01/2019 7:41	Archivo PHP	7 KB
 wp-blog-header	01/12/2017 0:11	Archivo PHP	1 KB
 wp-comments-post	21/01/2019 2:34	Archivo PHP	3 KB
 wp-config	03/07/2019 22:03	Archivo PHP	4 KB
 wp-config-sample	03/07/2019 22:03	Archivo PHP	3 KB
 wp-cron	09/01/2019 9:37	Archivo PHP	4 KB
 wp-links-opml	16/01/2019 6:29	Archivo PHP	3 KB
 wp-load	01/12/2017 0:11	Archivo PHP	4 KB
 wp-login	10/06/2019 15:34	Archivo PHP	39 KB
 wp-mail	01/12/2017 0:11	Archivo PHP	9 KB
 wp-settings	28/03/2019 20:04	Archivo PHP	19 KB
 wp-signup	16/01/2019 17:51	Archivo PHP	31 KB
 wp-trackback	01/12/2017 0:11	Archivo PHP	5 KB

About Docker Desktop

Settings

Check for Updates

Troubleshoot

Switch to Windows containers...

Docker Hub

Documentation

Kitematic

Sign in / Create Docker ID...

Repositories

Kubernetes

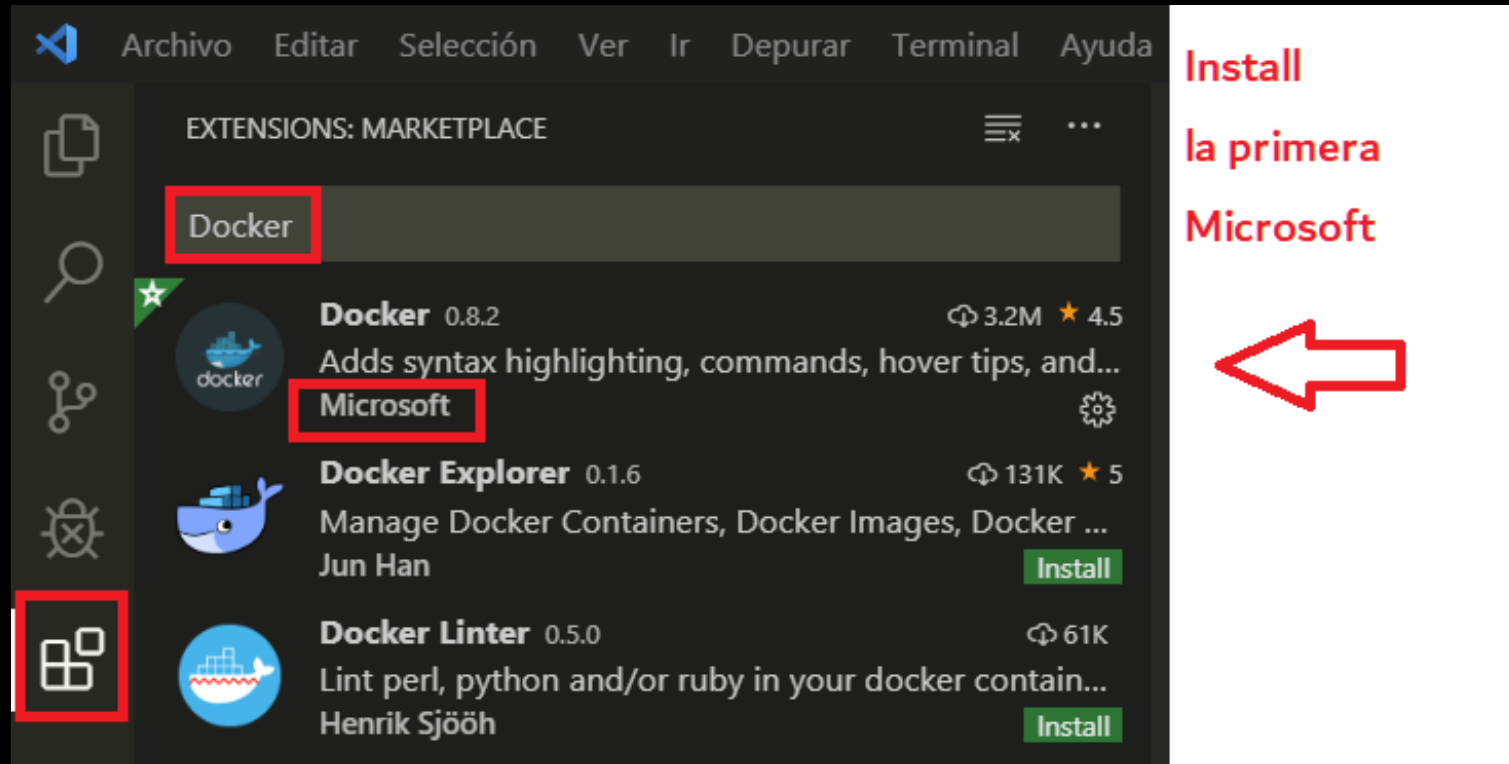
Restart...

Quit Docker Desktop

KITEMATIC

The screenshot displays the Kitematic interface within Docker Desktop. On the left, a sidebar lists various containers, including 'meetup_db_1' (mysql:8.0.18), 'meetup_wordpress...' (wordpress:fatest), and three instances of 'mysql-1', 'mysql-2', and 'mysql-3' (all mysql:8.0.16). Below these are three instances of 'phpmyadmin-1', 'phpmyadmin-2', and 'phpmyadmin-3' (all phpmyadmin), and three instances of 'wp-1', 'wp-2', and 'wp-3' (all wordpress:fatest). The 'Containers' tab is selected and highlighted with a red box. At the bottom left, the 'DOCKER CLI' button is also highlighted with a red box. The main area shows the 'meetup_db_1' container, which is currently 'STOPPED'. Below the container name are controls for 'START', 'RESTART', 'EXEC', and 'DOCS'. The 'CONTAINER LOGS' section shows '0 No logs for this container.' A terminal window is open in the foreground, displaying the Windows PowerShell prompt and the following text: 'Copyright (C) Microsoft Corporation. Todos los derechos reservados. Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6 PS C:\Users\José\Desktop>'. The terminal window title is 'C:\WINDOWS\System32\WindowsPowerShell\v1.0\powershell.exe'.

Visual Studio Code – añadir aplicación Docker



The screenshot shows the Visual Studio Code interface with the Extensions Marketplace open. The search bar at the top of the marketplace contains the word "Docker". Below the search bar, the first extension listed is "Docker" by Microsoft, version 0.8.2, with 3.2M downloads and a 4.5 star rating. The "Microsoft" publisher name is highlighted with a red box. A red arrow points from the text on the right towards the "Docker" extension entry. Other extensions listed include "Docker Explorer" by Jun Han and "Docker Linter" by Henrik Sjööh.

Archivo Editar Selección Ver Ir Depurar Terminal Ayuda

EXTENSIONS: MARKETPLACE

Docker

Docker 0.8.2 3.2M ★ 4.5
Adds syntax highlighting, commands, hover tips, and...
Microsoft

Docker Explorer 0.1.6 131K ★ 5
Manage Docker Containers, Docker Images, Docker ...
Jun Han **Install**

Docker Linter 0.5.0 61K
Lint perl, python and/or ruby in your docker contain...
Henrik Sjööh **Install**

Install la primera Microsoft

Visual Studio Code

The screenshot displays the Visual Studio Code interface with the following elements:

- Menu Bar:** Archivo, Editar, Selección, Ver, Ir, Depurar, Terminal, Ayuda.
- Left Panel (EXPLORER):**
 - DOCKER:** A red box highlights the DOCKER icon.
 - CONTAINERS:** A list of containers with a context menu open over the first one. The menu items are: Nuevo archivo, Abrir carpeta..., Agregar carpeta de área de trabajo..., View Logs, Inspect, Start (highlighted with a red arrow), and Remove....
 - IMAGES:** docker4w/nsenter-dockerd, mysql, php, phpmyadmin/phpmyadmin, wordpress.
 - REGISTRIES, NETWORKS, VOLUMES:** Lists of Docker resources.
- Center Panel:** A "Bienvenido" (Welcome) page with sections for Inicio (New file, Open folder, Add workspace folder), Reciente (Recent), and Ayuda (Help).
- Right Panel (PERSONALIZAR):** A "Personalizar" (Customize) page with sections for Herramientas y lenguajes (Tools and languages), Configuración y enlaces de teclado (Keyboard configuration and shortcuts), Tema de color (Color theme), and Más información (More information).
- Terminal Panel:** A red circle highlights the TERMINAL icon. The terminal shows a PowerShell prompt: PS C:\Users\José>