

# MANUAL DE USUARIO

## Implementación Docker

Elaboró: Oscar Jaime González

Ultima revisión 18 de septiembre de 2023

## Contenido

OBJETIVO .....	3
INTRODUCCIÓN .....	4
REPOSITORIO GITHUB .....	5
CREAR CONTENEDORES DOCKER .....	6
BASE DE DATOS.....	7
VISUALIZACION WEB .....	9
BIBLIOGRAFÍA .....	11

## OBJETIVO

Este manual de usuario esta elaborado para guiar los pasos requeridos para el uso de los contenedores Docker de Servicio Web y Bases de Datos, este manual te orientará sobre las tecnologías necesarias para su uso y las configuraciones necesarias para su visualización final.

Al concluir este manual se podrá utilizar esta implementación en el desarrollo de aplicaciones web.

## INTRODUCCIÓN

Para la creación de los contenedores utilizaremos un repositorio Github para obtener los archivos necesarios. Al crear los contenedores Docker, se obtendrá acceso al servidor web y utilizaremos un código PHP para verificar esta conexión.

Verificaremos la conexión a la base de datos y utilizando un manejador de bases de datos se creará una tabla para confirmar la conexión entre contenedores.

Como resultado visualizaremos una página web con una consulta a una base de datos.

## REPOSITORIO GITHUB

Accedemos al [Repositorio Git](#), damos click en “Code” y descargamos en ZIP la carpeta necesaria.

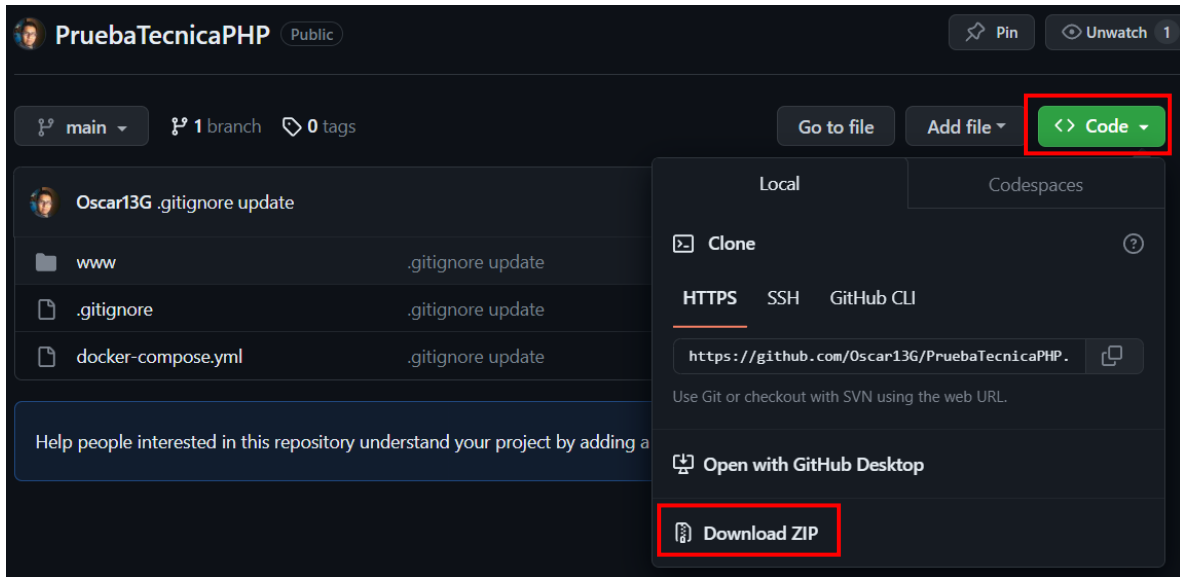


Imagen 1. Repositorio GitHub.

Extraemos la carpeta en un directorio conocido.

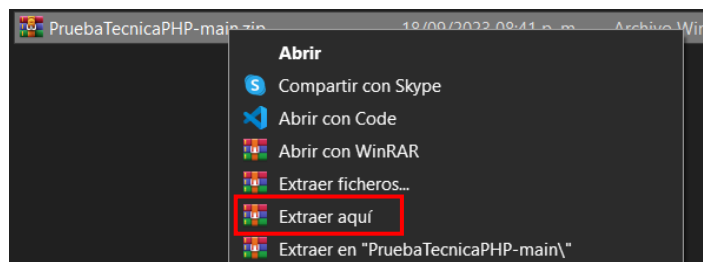


Imagen 2. Archivo Zip

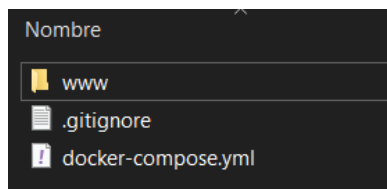


Imagen 3. Directorio “PruebaTecnicaPHP”.

Con esto ya tenemos los archivos necesarios para crear nuestros contenedores.

## CREAR CONTENEDORES DOCKER

Para la creación de los contenedores seguiremos los siguientes pasos:

A continuación, abrimos una ventana terminal y utilizaremos el comando **cd** y **dir** para movernos al directorio “PruebaTecnicaPHP” y verificar que el archivo “Docker-compose.yml” está en el directorio.

```
Microsoft Windows [Versión 10.0.19045.3448]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Kimba>cd C:\Users\Kimba\Desktop\PruebaTecnica\PruebaPHP

C:\Users\Kimba\Desktop\PruebaTecnica\PruebaPHP>dir
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: 84F1-F336

Directorio de C:\Users\Kimba\Desktop\PruebaTecnica\PruebaPHP

18/09/2023  01:21 p. m.  <DIR>          .
18/09/2023  01:21 p. m.  <DIR>          ..
18/09/2023  05:35 p. m.  <DIR>          db
18/09/2023  06:11 p. m.             649 docker-compose.yml
18/09/2023  01:21 a. m.  <DIR>          www
                1 archivos             649 bytes
                4 dirs  33,098,522,624 bytes libres

C:\Users\Kimba\Desktop\PruebaTecnica\PruebaPHP>
```

Imagen 4. CMD, ejecución de comandos cd y dir

Ejecutamos un comando de Docker, “docker compose up -d”. Docker comenzara a descargar las imágenes y a crear los contenedores y al finalizar los iniciara.

```
C:\Users\Kimba\Desktop\PruebaTecnica\PruebaPHP>docker compose up -d
[+] Running 25/18
  db-mysql 10 layers [██████████] 0B/0B Pulled 46.5s
  serv-apache 13 layers [██████████████] 0B/0B Pulled 3.1s
```

Imagen 5. Descarga de imágenes y creación de contenedores.

```
[+] Running 3/3
  Network pruebaphp_default Created 0.1s
  Container db-mysql Started 5.3s
  Container serv-apache Started 0.1s
```

Imagen 6. Inicio de los contenedores.

Con estas acciones tenemos los servicios arriba y funcionando.

## BASE DE DATOS

Para verificar la base de datos utilizaremos un manejador de bases de datos, para fines ilustrativos de este manual se utiliza DBeaver. Este manejador se puede descargar en este [link](#).

Configuramos una conexión con los siguientes datos:

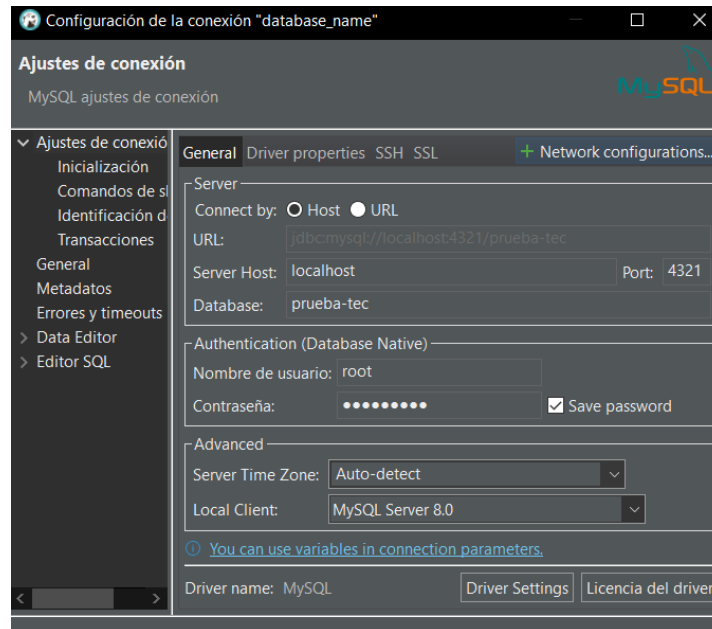


Imagen 7. Configuración de conexión a MySQL.

Para la contraseña utilizamos “pswdummy1”

Adicionalmente seleccionamos la pestaña “Drivers properties” y cambiamos el valor de “allowPublicKeyRetrieval” a “True”.

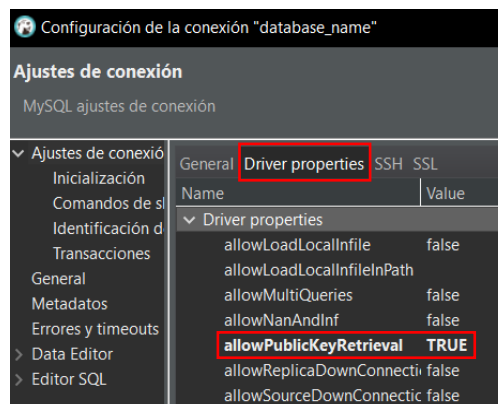


Imagen 8. Configuración de conexión a MySQL.

Creamos una tabla con nombre “empleado”.

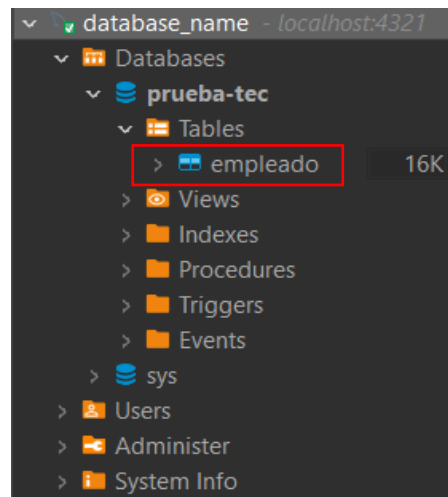


Imagen 9. Base de Datos y tabla empleado.

Para la verificación es necesario tener al menos un registro en la tabla. No importan los atributos mientras existan registros. Entonces creamos registros para la tabla.

empleado				
Propiedades Datos Diagrama ER				
empleado Enter a SQL expression to filter results (use Ctrl+Space)				
	empleado_id	nombre	apellido	
1	1	Oscar	Glz	
2	2	Vicente	Pantaleon	

Con estos elementos comprobamos la conexión a la Base de Datos y adicionalmente creamos archivos necesarios para la visualización final.



## VISUALIZACION WEB

Para verificar la conexión se adiciono un Código PHP que genera una conexión a la base de Datos y el servidor Web. Muestra una página y realiza una consulta de la base de datos.

Accedemos a un navegador web y en la barra de dirección escribimos <http://localhost:1234/> lo que nos abrirá la página y nos mostrará la imagen siguiente.



Imagen 10. visualización Web con error.

Este error esta asociado a PHP y una extensión de Mysql. Para eliminar el error ejecutaremos un script de Linux que realiza la instalación de la extensión necesaria.

Abrimos CMD y ejecutamos el comando “docker exec -ti serv-apache bash”, este comando nos permite conectarse al servidor apache y realizar modificaciones en la configuración de PHP.



Imagen 11. Acceso al servidor Apache.

Adicionalmente ejecutamos el comando “bash extension-mysql.sh”, este comando instala y modifica archivos necesarios para completar la configuración.

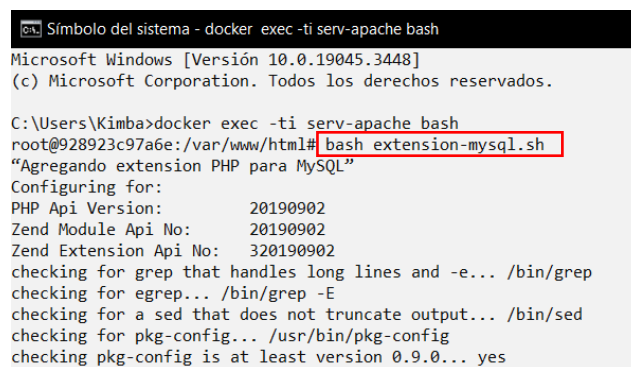


Imagen 12. instalación y modificación de archivos.

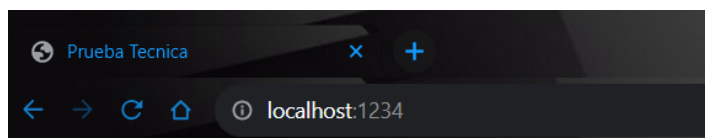
Al finalizar nos mostrara un mensaje de completado y podremos salir del servidor Apache escribiendo “exit”.

```
warning: mysqli (mysqli.so) is already loaded!

find . -name \*.gcno -o -name \*.gcda | xargs rm -f
find . -name \*.lo -o -name \*.o | xargs rm -f
find . -name \*.la -o -name \*.a | xargs rm -f
find . -name \*.so | xargs rm -f
find . -name .libs -a -type d|xargs rm -rf
rm -f libphp.la      modules/* libs/*
"Completado"
root@928923c97a6e:/var/www/html# exit_
```

*Imagen 13. Script completado.*

Para finalizar refrescamos el navegador web donde accedimos a la dirección <http://localhost:1234/> y visualizaremos la página web y un mensaje de conexión a la base de datos. Además de mostrar el numero de registros en la tabla “empleado”.



## Prueba tecnica con Docker

### Conexion a DB Existosa

**La Consulta devolvió 2 filas.**

*Imagen 14. Conexión exitosa.*

## BIBLIOGRAFÍA

*DBEaver Community / Free Universal Database Tool.* (s. f.). <https://dbeaver.io/>

«*Home*». (2023, 22 agosto). Docker Documentation. <https://docs.docker.com/>

<https://github.com/Oscar13G/PruebaTecnicaPHP>

