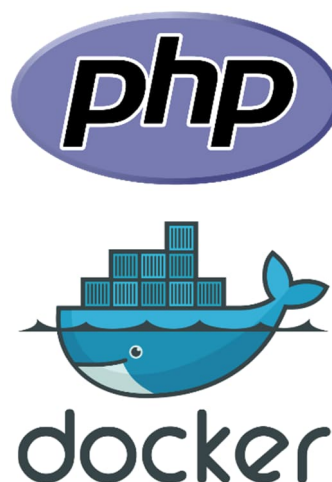


# PHP con Docker

Instrucciones sobre como integrar proyectos con PHP en PLCnext mediante contenedores en Docker



©PHOENIX CONTACT AR 2021-11-30

## 1 Descripción

Este documento describe el procedimiento de integrar proyectos con PHP en PLCnext mediante contenedores en Docker.

## 2 Requisitos de hardware

- AXC 1152
- AXC 2152

## 3 Requisitos de software

- WinSCP
- PuTTY

## Tabla de contenido

1	Descripción .....	1
2	Requisitos de hardware .....	1
3	Requisitos de software .....	1
4	Integrar Docker en PLCnext .....	2
5	Integrar PHP y Apache .....	2
5.1	Crear Dockerfile .....	2
5.2	Crear imagen .....	3
5.3	Crear contenedor .....	3
5.4	Acceso desde el navegador .....	4



Encontrará más información sobre la implementación de Docker en PLCnext en:  
[https://github.com/PLCnext/Docker\\_GettingStarted](https://github.com/PLCnext/Docker_GettingStarted)

## 4 Integrar Docker en PLCnext

- Con el objetivo de integrar Docker en el controlador PLCnext será necesario abrir una sesión en PuTTY como usuario root (ver "Documentación referida al tema") y colocar la siguiente instrucción en la línea de comandos:

```
git clone https://github.com/PLCnext/Docker_GettingStarted.git
```

- Una vez finalizada la descarga, se debe ingresar a la carpeta Docker\_GettingStarted a través del comando `cd`:

```
cd Docker_GettingStarted
```

- Luego se le da permisos a un script de instalación (`setup.sh`) que permanece dentro de la carpeta a la cual se ingresó recientemente.

```
chmod +x setup.sh
```

- A continuación, se ejecuta dicho script con la siguiente instrucción:

```
./setup.sh
```

- Durante la ejecución del script de instalación se tiene la opción de seleccionar Docker o balenaEngine.

```
Please choose your container runtime:
Install balenaEngine(recommend): 1
Install Docker: 2
```

Fig. 1 Opciones de descarga

- Además, será posible integrar docker-compose en caso de que se requiera.

```
Do you wish to install docker-compose?
```

Fig. 2 Opciones de descarga

## 5 Integrar PHP y Apache

- En el sistema de archivos del controlador PLCnext será necesario agregar una carpeta que va a contener al Dockerfile y a otra carpeta con los archivos PHP del proyecto en cuestión. Esto es posible realizarlo mediante una sesión de WinSCP.
- En el ejemplo que se muestra a continuación se creó una carpeta denominada "Docker\_tests" en la dirección `/opt/plcnext` del sistema de archivos. La carpeta "src" contiene los archivos PHP del proyecto.

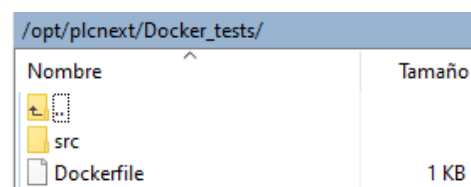


Fig. 3 Contenido de la carpeta

### 5.1 Crear Dockerfile

- Mediante el uso de una herramienta como Notepad++ creamos el siguiente archivo en donde se especifica lo siguiente:

```
1 FROM php:7.4-apache
2
3 COPY /src /var/www/html
4
5 EXPOSE 80
```

Fig. 4 Dockerfile

- Con el comando FROM se especifica la imagen y la versión, la cual se utilizará como base de la imagen que se está creando.
- Luego mediante COPY se copia el contenido ubicado en la carpeta "src" en la carpeta "html" ubicada en la dirección `/var/www` del contenedor que se crea con la imagen.

- Con EXPOSE se define una referencia del puerto en el que se ejecuta el servidor.
- Es necesario aclarar que a la hora de guardar el archivo Dockerfile es necesario especificar el tipo: "All types".

Nombre:	Dockerfile
Tipo:	All types

Fig. 5 Tipo de archivo

## 5.2 Crear imagen

- Es necesario abrir una sesión en PuTTY y dirigirse a la ubicación del Dockerfile mediante el comando `cd`. Siguiendo con el ejemplo:

```
cd /opt/plcnext/Docker_tests
```

- Una vez realizado esto se coloca la siguiente instrucción en la línea de comandos:

```
docker build -t hello-php .
```

- El comando `build` leerá el archivo Dockerfile y mediante las instrucciones especificadas en él crea una imagen.
- Mediante `-t` se le coloca un nombre a la imagen, en este caso `"hello-php"`.
- Finalmente se coloca un `.` para especificar que el archivo Dockerfile se encuentra en la ubicación actual.
- Con el comando `docker images` es posible verificar las imágenes incluidas:

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
hello-php	latest	877ec1d1cf73	2 hours ago	336MB
php	7.4-apache	e223f1ded436	11 days ago	336MB

Fig. 6 Imágenes

## 5.3 Crear contenedor

- Mediante el siguiente comando se creará un contenedor, el cual va a alojar a la imagen que se ha creado anteriormente, es decir que PHP y el servidor web apache se ejecutarán en dicho contenedor.

```
docker run -p 5000:80 -d -v /opt/plcnext/Docker_tests/src:/var/www/html/ hello-php
```

El comando `run` es el cual va a permitir crear un contenedor a partir de la imagen especificada, en este caso `"hello-php"`.

La instrucción `-p` permite especificar el puerto por el cual se requiere acceder al servidor ubicado en el contenedor. En este caso de ejemplo el servidor alojado en el contenedor utiliza el puerto 80 y mediante la PC se va a poder acceder al mismo mediante el 5000.

El comando `-d` permite ejecutar al contenedor en modo "detach", es decir que lo ejecuta en segundo plano sin la necesidad de tener la sesión de PuTTY activa.

Para que los cambios realizados en el código de los archivos ubicados en la carpeta `src` se vean reflejados será necesario utilizar un concepto de Docker denominado "volumes"

Los "volumes" permiten copiar contenido desde el host, en este caso el sistema de archivos del controlador PLCnext, al contenido del contenedor.

Con el comando `v-` es posible utilizar los "volumes". Se debe especificar la ubicación de la carpeta `src` y de `:` la dirección de la carpeta `html` del contenedor en cuestión.

## 5.4 Acceso desde el navegador

- Con tal de visualizar el contenido especificado en la carpeta mencionada anteriormente es necesario colocar en el navegador la dirección IP del controlador PLCnext y especificar el puerto que se seleccionó al momento de crear el contenedor.

*`https://%PlcAddress%:%Port%`*