

Fase 4 - Activitat 6.3: Test de funcionament en execució d'una aplicació PHP utilitzant PHPUnit

0- Identificació del grup i activitat:

Curs: ASIX2

Projecte: GP2 DevOps i Cloud Computing

Fase: 4

Activitat: 6.3

Grup:

Membres:

1- Introducció i objectius de l'activitat 6.3

- Lectura de les especificacions de l'activitat
- Creació d'un Docker per desplegar una aplicació que tingui el framework PHPUnit necessari per fer tests de funcionament.
- Execució dels tests de funciomanet de l'aplicació i comprovació de resultats.

2- Descarregant i instanciant una imatge de Docker amb Apache i PHP 7.4

a) Accedeix a la màquina virtual **gp1f4a6-grupXX** (a on **XX** és el teu número de grup) que vas crear amb **Vagrant** a l'activitat **gp1f4a6.1**. Accedeix a la carpeta **projectes**. Clona el projecte <https://github.com/globproj2/gp1f4a6.3.git>. Comprova que s'ha creat la carpeta **gp1f4a6.3**.

b) Accedeix a la carpeta **gp1f4a6.3**. Esborra el directori **.git**. Executa: `rm -rf .git`. Amb això s'evita la sincronització amb el dipòsit **gp1f4a6.3** de l'usuari **globproj2**.

c) Comprova que dins **gp1f4a6.3** hi ha una carpeta de nom **app** que té una aplicació web. Pel moment no cal accedir-hi.

d) Dins de **gp1f4a6.3** crea un fitxer **Dockerfile** amb el següent contingut:

```
vagrant@gp1f4a6:~/projectes/gp1f4a6.3$ cat Dockerfile
FROM php:7.4-apache
RUN apt-get update
RUN apt-get install -y zip
RUN apt-get install -y unzip
RUN curl -sS https://getcomposer.org/installer | php -- --install-dir=/usr/local/bin --filename=composer
WORKDIR /var/www/html
COPY app .
EXPOSE 80
```

i a continuació executa:

```
docker build -t calculadora:1.0 .
```

Aquesta ordre munta una imatge de contenidor que:

- S'anomena **calculadora**
- Utilitzant una imatge ja existent a **Docker Hub** de nom **php:7.4-apache** que té un sistema **Linux** mínim, **Apache2** i **PHP7.4** com a base per crear una imatge nova.
- El directori per defecte (**WORKDIR**) de la imatge és **/var/www/html** de manera que l'ordre **COPY** copiarà la carpeta **app** dins del **/var/www/html**.
- A més a més, s'exposarà el port **80/tcp** del contenidor i per tant, es podrà accedir al port **80/tcp** del contenidor **Docker** des de la màquina virtual.
- La imatge serà desada com la versió **1.0**.
- S'executaran (**RUN**) les ordres necessàries per poder afegir **composer** dins de la imatge.

e) Atura el contenidor **gp1f4a6** si està en marxa.

f) Instancia la imatge i crea un contenidor de nom **calculadora**. Amb les opcions **-i -t -d** indica que s'executi en **2n terme** i que es pugui accedir més tard via **bash** al contenidor. Indica que el port **80/tcp** del contenidor s'exposi com a port **80/tcp** de la **màquina virtual**. Executa des de dins de la carpeta **gp1f4a6.3**:

```
docker run --name calculadora -i -t -d -p 80:80 calculadora:1.0
```

g) Comprova que el contenidor s'ha instanciat que està **actiu (up)**, que exposa els **ports** demanats a la màquina host, que el seu **nom** és correcte i que comparteix la carpeta correcta. Executa:

```
docker ps -a
```

h) Comprova que pots accedir a l'interpret d'ordres **bash** del contenidor via **terminal** executant:

```
docker exec -it calculadora bash
```

i) Comprova que dins **/var/www/html** està l'aplicació.

j) Per sortir del del contenidor executa des del bash del contenidor. Executa: **exit**

3- Afegint a l'aplicació el framework PHPUnit per poder realitzar tests de funcionament.

a) Afegeix dins del directori **/var/www/html** del contenidor Docker (a on està l'aplicació) la versió **9** del framework **PHPUnit**. Això ens permetrà executar tests de funcionament de l'aplicació. Executa des de la màquina virtual:

```
docker exec -it calculadora composer require --dev phpunit/phpunit ^9
```

Aquesta ordre fa que dins del contenidor s'executi l'ordre **composer require --dev phpunit/phpunit ^9** i per tant, dins del contenidor ara tenim l'eina PHPUnit per fer proves de funcionament d'aplicacions PHP.

b) Ara executa des de la màquina virtual: **docker exec -it calculadora ls** i comprova que dins del contenidor tens els fitxers **composer.json**, **composer.lock** i el directori **vendor** que formen part del framework **PHPUnit**.

4- Creació de tests de funcionament de l'aplicació amb PHPUnit

a) Comprova que dins del directori **app** de la màquina virtual tens un fitxer preparat per crear tests de funcionament de l'aplicació de nom **operacionsTest.php** que també es troba a **/var/www/html** del contenidor.

b) Des de la màquina virtual executa la següent ordre per instal·lar **nano** dins del contenidor:

```
docker exec -it calculadora apt-get -y install nano
```

c) Des de la màquina virtual, completa el fitxer **operacionsTest.php** (que ja està preparat per fer els tests de suma) per poder fer tests de resta, multiplicació, divisió i potenciació amb el següent joc de tests:

- Executa **docker exec -it calculadora nano operacionsTest.php**
- Modifica la prova de test per la Resta:
 - operand1 = 0, operand2 = 0
 - operand1 = 1, operand2 = 4
 - operand1 = 5, operand2 = 2
- Modifica la prova de test per la Multiplicació:
 - operand1 = 0, operand2 = 0
 - operand1 = 2, operand2 = 4
 - operand1 = 5, operand2 = -2
 - operand1 = -2, operand2 = -3
- Modifica la prova de test per la Divisió:
 - operand1 = 8, operand2 = 2
 - operand1 = 6, operand2 = -2
 - operand1 = -12, operand2 = -4
- Modifica la prova de test per la Potenciació:
 - operand1 = 2, operand2 = 3
 - operand1 = 2, operand2 = -1
 - operand1 = -4, operand2 = -2

c) Executa des de la màquina virtual:

```
docker exec -it calculadora ./vendor/phpunit/phpunit/phpunit /var/www/html/operacionsTest.php
```

que executarà els tests definits dins **OperacionsTest.php**.

d) Comprova que tots els tests funcionen correctament. El resultat hauria de ser:

```
OK (1 test, 16 assertions)
```

e) Si tot funciona correctament, avisa'm per poder fer es comprovacions.

Lliurament de l'activitat - Part tècnica

a) Comprovació que funciona el test demanat. El resultat hauria de ser l'indicat a l'apartat **d)** del punt **4**.

b) Data límit per obtenir el 100% de la nota: **dilluns 24-10-23** a les **17.45**. Posteriorment la nota serà menor.

Lliurament de l'activitat - Part Documentació i planificació

a) Documenta i planifica l'activitat de documentació:

- Heu de fer un document **PDF** amb totes les ordres executades i els resultats obtinguts a cadascun dels punts i apartats de la pràctica.
- Heu d'afegir a la documentació el codi final del vostre fitxer **OperacionsTest.php**.
- Heu d'afegir una planificació de la creació de la documentació, indicant el temps que espereu necessitar per dur-la a terme. Indiqueu els possibles desviaments respecte de la planificació realitzada i finalment expliqueu els motius per aquest desviament.

b) Presentació i lliurament de la documentació: **divendres 27-10-23** a les **19.10**. Posteriorment la nota serà menor.