



Universidad de Guadalajara
Centro Universitario
de
Ciencias Exactas
e
Ingenierías
(CUCEI)

Materia: COMPUTACION TOLERANTE A FALLAS

Nombre:
Jonathan Aceves López

Código:
217481363
NRC: 179961

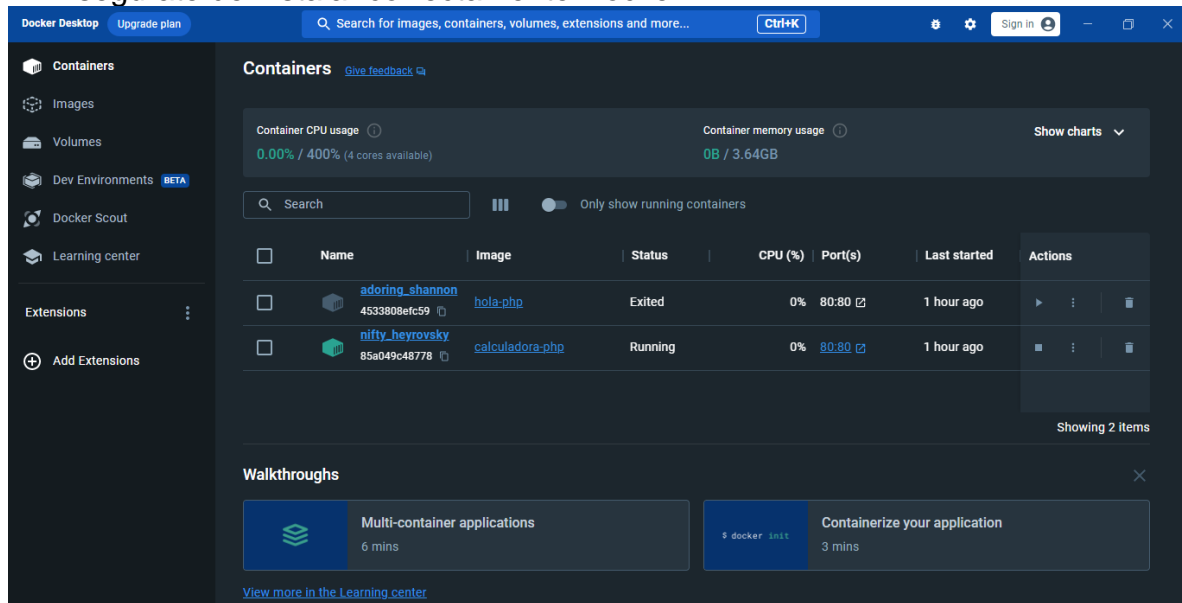
Ejemplo utilizando Docker.

En esta actividad se utilizó ciertas herramientas como Docker, CMD y block de notas para crear una imagen con PHP y Apache para hacer funcionar una calculadora básica en entorno local (Localhost).

Para poder crear y hacer funcionar debemos seguir los siguientes pasos:

*Advertencia, todo esto se realizó en una computadora con sistema operativo Windows, más en específico Windows 11 por lo que puede que algunas cosas no funcionen en entornos Linux y IOS.

1.- Asegúrate de instalar correctamente Docker.



2.- Crearemos por CMD un directorio con nuestro código fuente.

“mkdir src” (Sin las “” además que en mi caso, ya existe pero debería funcionar.)

```
C:\Users\jonat>mkdir src
Ya existe el subdirectorio o el archivo src.
```

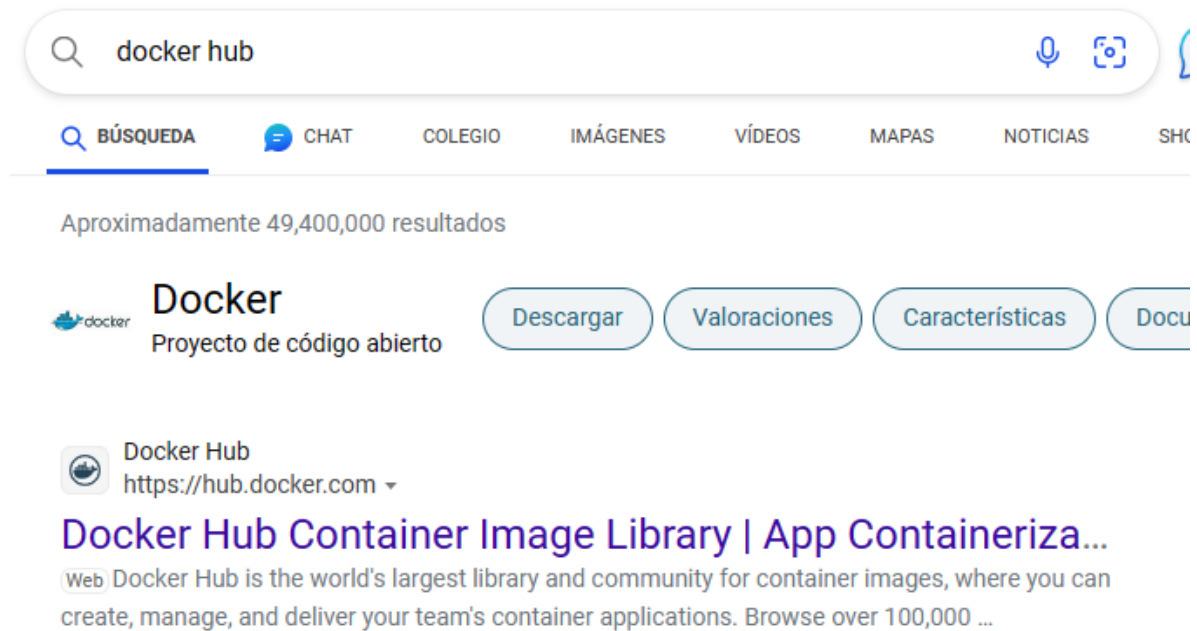
3.- Entra en el directorio con el comando cd.

“cd src”

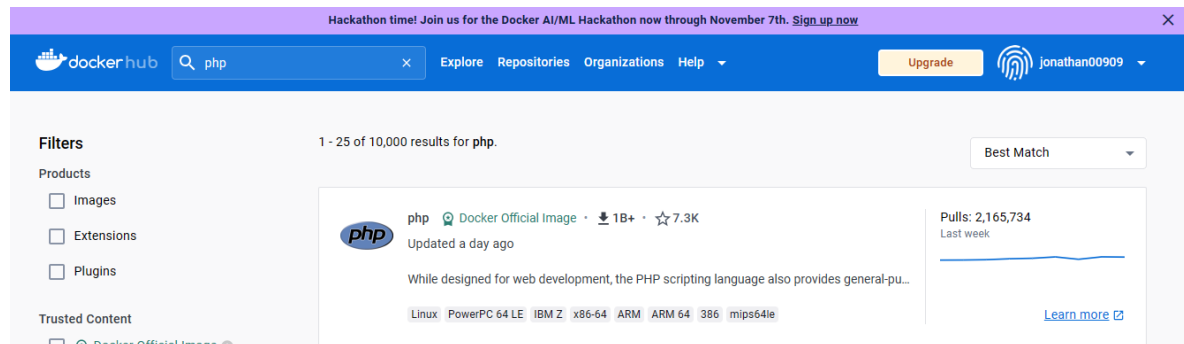
```
C:\Users\jonat>cd src/
C:\Users\jonat\src>
```

4.- Ahora entra a la carpeta src desde el explorador de archivos Windows en mi caso como pueden ver en la imagen anterior del punto 3. “C:\Users\jonat\src”

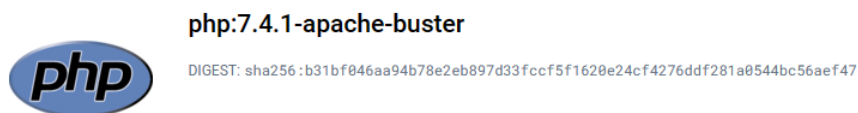
5.- Ahora abre tu navegador de confianza y busca “Docker hub” y se meten a la primera opción.



6.- Una vez dentro de “Docker hub” buscan php.



Y se meten a la imagen que más les convenga, en este caso php 7.4.1-apache-buster



7.- Volviendo a la carpeta “src” crea un archivo de texto dentro llamado “Dockerfile”. (Asegúrese de llamarlo así, tarde 30 minutos en encontrar el error)

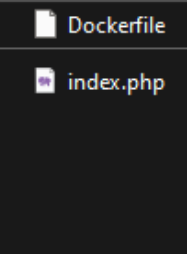
8.- Dentro de este escribiremos.

```
FROM php:7.4.1-apache-buster
COPY . /var/www/html/
EXPOSE 80
```

9.-Una vez echo, asegúrate de que el archivo “Dockerfile” no contenga la extensión .txt o cualquier otro que hubieras utilizado.

Bien{  }	Mal{  }
---	---

10.- Ahora dentro de la carpeta src crea tu archivo php con tu código, en esté caso es una calculadora.

 index.php	<pre><!DOCTYPE html> <html> <head> <title>Calculadora Básica</title> </head> <body> <h1>Calculadora Básica</h1> <form method="post"> <input type="text" name="num1" placeholder="Número 1" required> <select name="operation"> <option value="add">Sumar (+)</option> <option value="subtract">Restar (-)</option> <option value="multiply">Multiplicar (*)</option> <option value="divide">Dividir (/)</option> </select> <input type="text" name="num2" placeholder="Número 2" required> <input type="submit" value="Calcular"> </form> <?php if (\$_SERVER["REQUEST_METHOD"] == "POST") { \$num1 = \$_POST["num1"]; \$num2 = \$_POST["num2"]; \$operation = \$_POST["operation"]; if (!is_numeric(\$num1) !is_numeric(\$num2)) { echo "Por favor, ingresa números válidos."; } else { switch (\$operation) {</pre>
---	---

	<pre> case "add": \$result = \$num1 + \$num2; break; case "subtract": \$result = \$num1 - \$num2; break; case "multiply": \$result = \$num1 * \$num2; break; case "divide": if (\$num2 == 0) { echo "No es posible dividir por cero."; } else { \$result = \$num1 / \$num2; } break; default: echo "Operación no válida."; break; } echo "Resultado: \$result"; } } ?> </body> </html> </pre>
--	--

11.- De nuevo en el CMD escribiremos “dir” para asegurarnos de que estamos en la carpeta y archivos correctos:

```

C:\Users\jonat\src>dir
El volumen de la unidad C es Windows-SSD
El número de serie del volumen es: 86CF-D62E

Directorio de C:\Users\jonat\src

29/10/2023  02:07 p. m.      <DIR>          .
29/10/2023  11:34 a. m.      <DIR>          ..
29/10/2023  12:10 p. m.                  62 Dockerfile
29/10/2023  12:35 p. m.          1,776 index.php
                2 archivos          1,838 bytes
                2 dirs  348,657,504,256 bytes libres

```

12.- Si todo está correcto usaremos el siguiente comando “`docker build -t calculadora-php .`”

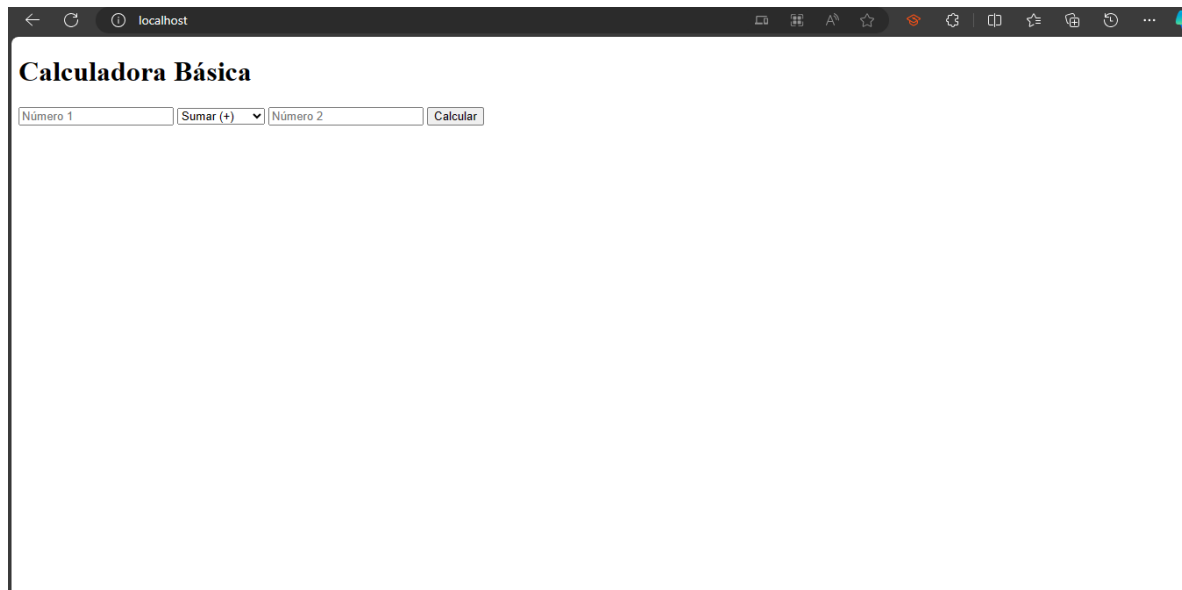
```
C:\Users\jonat\src>docker build -t calculadora-php .
[+] Building 2.0s (7/7) FINISHED                                docker:default
=> [internal] load build definition from Dockerfile              0.0s
=> => transferring dockerfile: 99B                               0.0s
=> [internal] load .dockerignore                                0.0s
=> => transferring context: 2B                                    0.0s
=> [internal] load metadata for docker.io/library/php:7.4.1-apache-buster 1.8s
=> [internal] load build context                                0.0s
=> => transferring context: 60B                                   0.0s
=> CACHED [1/2] FROM docker.io/library/php:7.4.1-apache-buster@sha256:adeca522dc9cc66693e89ce9 0.0s
=> [2/2] COPY . /var/www/html/                                  0.0s
=> exporting to image                                           0.0s
=> => exporting layers                                           0.0s
=> => writing image sha256:45da9edd84c6cfd29c6939244334c2c274dd4d9fb389a13e7fbc8bb2328324a 0.0s
=> => naming to docker.io/library/calculadora-php              0.0s

What's Next?
View a summary of image vulnerabilities and recommendations → docker scout quickview
```

13.- Si todo sale bien debería salirte esto:

```
C:\Users\jonat\src>docker run -p 80:80 calculadora-php
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 172.17.0.2. Set the 'ServerName' directive globally to suppress this message
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 172.17.0.2. Set the 'ServerName' directive globally to suppress this message
[Sun Oct 29 20:17:45.961838 2023] [mpm_prefork:notice] [pid 1] AH00163: Apache/2.4.38 (Debian) PHP/7.4.1 configured -- resuming normal operations
[Sun Oct 29 20:17:45.961993 2023] [core:notice] [pid 1] AH00094: Command line: 'apache2 -D FOREGROUND'
```

14.- Lo único que queda es ir a <http://localhost/> desde tu navegador y comprobar que funciona y que tu código de php funcione también.



Extra: En la aplicación de Docker saldrá la imagen activa también:

Containers

Images

Volumes

Dev Environments BETA

Docker Scout

Learning center

Extensions

Add Extensions

Images

Local

Hub

Artifactory

EARLY ACCESS

414.01 MB / 414.01 MB in use

1 Images

Last refresh: 4 minutes ago

Search

	Name	Tag	Status	Created	Size	Actions
	calculadora-php	latest	In use	2 minutes ago	414.01 MB	<div></div> <div></div> <div></div>

Showing 1 item