
	<p>UNIVERSIDAD NACIONAL DE SAN AGUSTIN FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMA</p>	
<p>Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación</p>		
<p>Aprobación: 2022/03/01</p>	<p>Código: GUIA-PRLE-001</p>	<p>Página: 1</p>

INFORME DE LABORATORIO

(formato estudiante)

INFORMACIÓN BÁSICA					
ASIGNATURA:	<i>programacion web 2</i>				
TÍTULO DE LA PRÁCTICA:	<i>Docker</i>				
NÚMERO DE PRÁCTICA:	<i>01</i>	AÑO LECTIVO:	<i>2024</i>	NRO. SEMESTRE:	<i>A</i>
FECHA DE PRESENTACIÓN	<i>11/5/24</i>	HORA DE PRESENTACIÓN			
INTEGRANTE (s) <i>JOSE MARIA TICONA SAURE</i>				NOTA (0-20)	
DOCENTE(s): <i>LINO JOSE PINTO OPPE</i>					

RESULTADOS Y PRUEBAS

Informe de Laboratorio

JOSE MARIA TICONA SAURE

11 de mayo de 2024

Siendo Docker una herramienta que agiliza el desarrollo, prueba y despliegue de aplicaciones mediante el uso de contenedores, los cuales contienen todo lo necesario para ejecutar una aplicación de manera uniforme en diversos entornos. Gracias a esto ya no existe la famosa frase 'En mi computadora si me funciona'; ya que, Docker crea un entorno ya configurado con las herramientas correspondientes como mysql, etc. A continuación explicar los pasos que realicé para lograr cumplir con el laboratorio que trata sobre subir mi proyecto a Docker y a Docker Hub; y hacer un video sobre de que trata el contenedor Docker:

1. Instalación de la Herramientas a Usar: Para subir el proyecto de PW1 a Docker se requiere de herramientas que ayuden en la ejecución de este.
 - Docker: Localice la página oficial de Docker e instale el installer, no me dio ningún problema alguno.



Figura 1: Página oficial Docker

Al tener instalado coloque el siguiente comando para crear un contenedor:

```
docker run --name pw2_lab01 -p 8084:80 -p 8085:3306 -p 8086:22 -it ubuntu:20.04 /bin/bash
```

Figura 2: Primer comando

Esto nos permite entrar en el cmd del contenedor, donde colocaremos los comandos que nos ayudarán a configurar ese entorno. Para la instalación de las siguientes herramientas seguimos los pasos dados por el Ingeniero en el documento, que son comandos que logran instalar las siguientes herramientas e iniciarlas en algunos casos:

- Apache:


```
apt-get install apache2
/etc/init.d/apache2 start
```
- Perl(cgi):


```
apt-get install perl
a2enmod cgi
apt-get install libapache2-mod-fcgid
```
- mysql:


```
apt-get install mariadb-server
/etc/init.d/mysql start
```
- Open SSH Server:


```
apt-get install openssh-server
/etc/init.d/ssh start
```
- VIM:


```
apt-get install vim
```

2. Creación de los archivos html y pl: Para la creación de los archivos html y pl usamos el editor vim, logramos crearlos en el directorio correspondiente.

- Para html: vim /var/www/html/
- Para pl: vim /usr/lib/cgi-bin

3. Transferencia de archivos como imágenes, js y css(uso de File-Zilla): En este paso mediante la herramienta FileZilla logramos conectar el servidor creado en Docker al local, donde pudimos transferir archivos como imágenes, css y js.

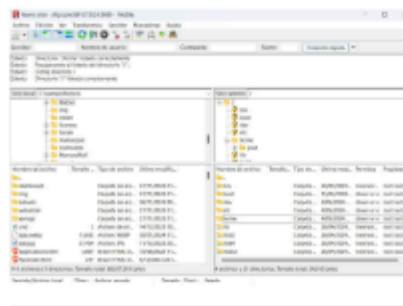


Figura 3: Herramienta para trasladar archivos

4. Conexión de los archivos en ese entorno: Uno de los problemas encontrados fue lograr que estos se conecten con las rutas, debido a que el mysql estaba descongurado, esto me ocasiono ciertas complicaciones. Primero se uso comandos sql para configurar el mysql o mariaDB. Comando aprendidos en pw1 como 'CREATE DATABASE wikipedia;' de esta manera creamos la base de datos y con 'CREATE TABLE' implementamos las tablas y la cabecera correspondiente. No olvidemos darle todos los permisos al usuario quien cre la DATABASE. En mi caso use el siguiente comando para obtener los permisos correspondientes:

- `GRANT ALL PRIVILEGES ON *.* TO 'pw2'@'localhost' WITH GRANT OPTION;`

Otro problema que se me presento fueron los perls estos no se ejecutaban en el entorno web, intentando logre averiguar que estos no ten an permisos de ejecución de tal manera us el siguiente comando:

- `chmod +x prueba.pl`

Posteriormente arreglando las rutas de los archivos html, pl, css, js y las imágenes logre que mi proyecto de pw1 funcione:

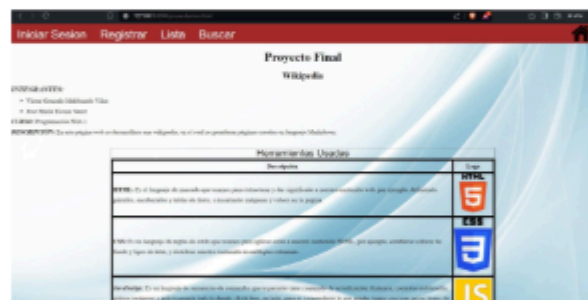


Figura 4: Proyecto Final

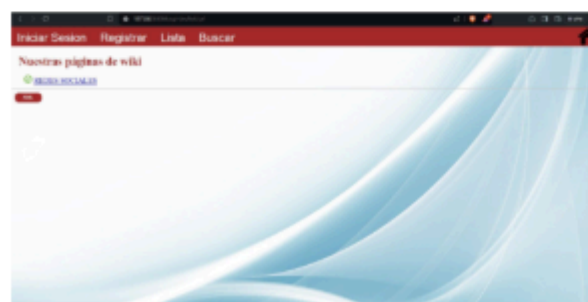


Figura 5: Proyecto Final

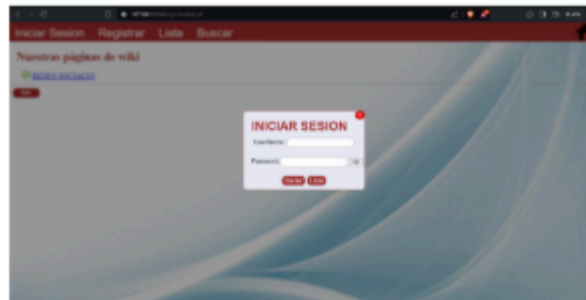


Figura 6: Proyecto Final

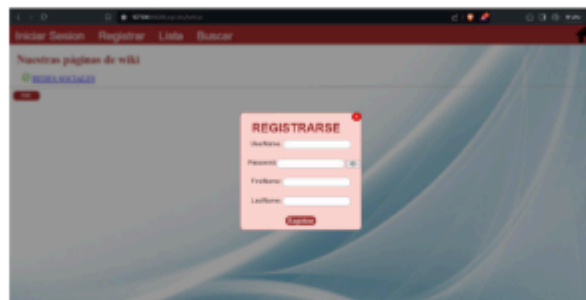


Figura 7: Proyecto Final

5. Subir mi contenedor a Docker Hub: Para ello tuve que investigar, tuve que observar videos, leer documentos e incluso me ayude de IAs. Donde segu los siguientes pasos:

- a) Crear una cuenta en Docker Hub:



Figura 8: Docker Hub

- b) Logearme en mi pc con el comando: docker login

c) seguir los siguientes comandos:

```
docker commit abcdef123456 joseticona/pweb2_lab1: latest
docker tag pweb2_lab1:v1 joseticona/pweb2_lab1:v1
docker push joseticona/pweb2_lab1:v1
```

Figura 9: Comandos

d) Verificar en la pagina Docker Hub:

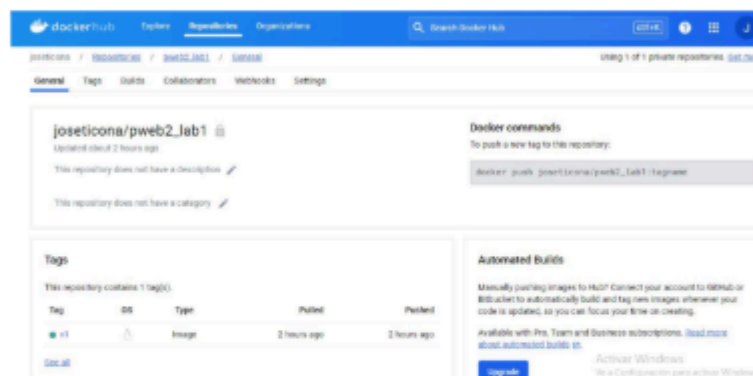


Figura 10: Contenedor subido

6. Un contenedor es?: Con todo lo aprendido se puede decir que un contenedor es un entorno informático aislado y eficiente que contiene todos los elementos necesarios para ejecutar una aplicación de forma independiente y rápida.

CONCLUSIONES:

- Docker es una herramienta poderosa para la virtualización de aplicaciones, permitiendo la creación y ejecución de contenedores de manera eficiente y rápida.
- Docker Hub ofrece un repositorio centralizado de imágenes Docker, facilitando la colaboración y distribución de aplicaciones y servicios.
- Docker facilita la colaboración entre equipos de desarrollo al estandarizar los entornos de trabajo y eliminar las diferencias entre sistemas locales.