

# Informe de Laboratorio N

## Tema: Docker

Nota

Estudiante	Escuela	Asignatura
Chambilla Perca Ricardo Mauricio rchambillap@unsa.edu.pe	Escuela Profesional de Ingeniería de Sistemas	Programación Web 2 Semestre: III Código: 1701213

Laboratorio	Tema	Duración
1	Docker	08

Semestre académico	Fecha de inicio	Fecha de entrega
2024 - A	Del 8 de Mayo 2024	Al 15 de Mayo 2024

## 1 Ejercicios propuestos

Crear un contenedor en Docker basado en ubuntu 20.04: Especificaciones del Lab 01

### Instale el servidor web Apache HTTP server 2.x

En mi caso, el trabajo final de programación web 1, fue realizado con el servidor NGINX, que simular a Apache, despliega contenido estático y puede ejecutar CGIS. Este software lo instalo en mi imagen a través de el archivo de DockerFile

```
RUN apt-get update
RUN apt-get install -y nginx
RUN apt-get install -y fcgiwrap
```

#### 1.1 Instale cualquiera de estos lenguajes de programación: PHP, Perl, Python.

Configure el servidor web para que interprete uno de los lenguajes de programación.

De la misma forma, para el lenguaje de programación que utilizamos fue Perl, y en el dockerfile en el que estoy mi contenedor saca la imagen se encuentra las instrucciones para la instalación de estas dependencias

```
RUN apt-get install -y perl
RUN apt-get install -y cpanminus
# Lets make this inside the container
# RUN cpan CGI
# RUN cpan DBI
# RUN cpan JSON
# RUN cpan CGI::Session
```

Como se puede ver en el comentario la instalación de los módulos de Perl que utilizamos para realizar la página web.

## 1.2 Instale cualquiera de los servidores de base de datos: MySQL, MariaDB, PostgreSQL.

Instale el servidor Open SSH Server. Envíe archivos al servidor: imágenes, css, js, etc.

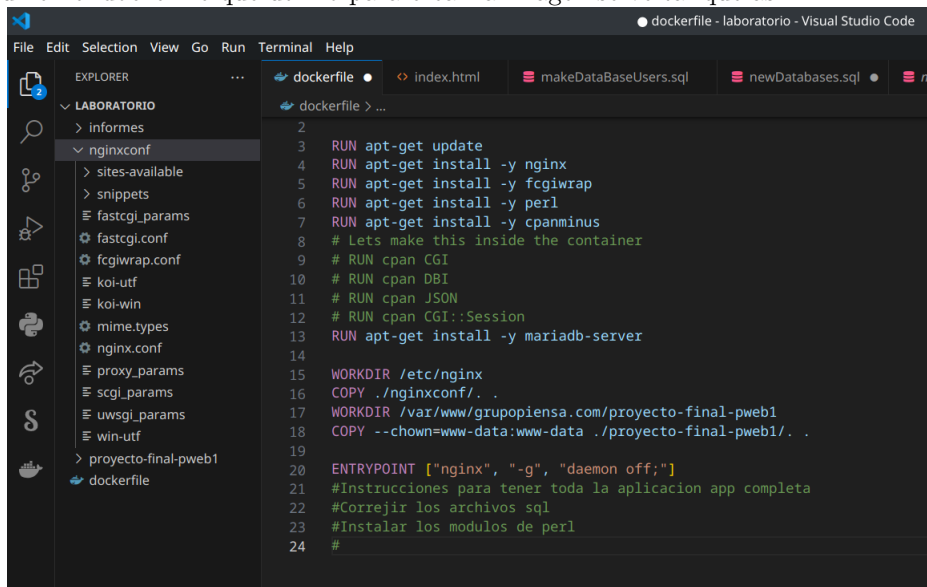
En el dockerfile también se instala la base de datos MariaDB.

```
RUN apt-get install -y mariadb-server
```

## 1.3 Cree un usuario pweb2 con contraseña: 12345678.

Otorgue permisos al usuario para acceder a la aplicación web. (Read-/Write)

Esta operación se realizó a través del terminal del contenedor una vez creada la imagen y corriendo. En resumen el dockerfile que utilizo para crear la imagen se ve tal que así:



```
2
3 RUN apt-get update
4 RUN apt-get install -y nginx
5 RUN apt-get install -y fcgiwrap
6 RUN apt-get install -y perl
7 RUN apt-get install -y cpanminus
8 # Lets make this inside the container
9 # RUN cpan CGI
10 # RUN cpan DBI
11 # RUN cpan JSON
12 # RUN cpan CGI::Session
13 RUN apt-get install -y mariadb-server
14
15 WORKDIR /etc/nginx
16 COPY ./nginxconf/. .
17 WORKDIR /var/www/grupopiensa.com/proyecto-final-pweb1
18 COPY --chown=www-data:www-data ./proyecto-final-pweb1/. .
19
20 ENTRYPOINT ["nginx", "-g", "daemon off;"]
21 #Instrucciones para tener toda la aplicación app completa
22 #Corregir los archivos sql
23 #Instalar los módulos de perl
24 #
```

Una vez adentro del terminal del contenedor puedo comenzar los procesos de MariaDB así como el servicio de fcgiwrap el cual me permite ejecutar cgi en servidores estáticos administrados por nginx.

```
docker run -dp 8000:80 nginxserver
docker exec -it mystifying_ellis /bin/bash
root@1a30749290f4:/var/www/grupopiensa.com/proyecto-final-pweb1# /etc/init.d/mariadb
start
* Starting MariaDB database server mariadbd

[ OK ]
root@1a30749290f4:/var/www/grupopiensa.com/proyecto-final-pweb1# /etc/init.d/fcgiwrap
start
* Starting FastCGI wrapper fcgiwrap

[ OK ]
root@1a30749290f4:/var/www/grupopiensa.com/proyecto-final-pweb1# mysql
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 33
Server version: 10.6.16-MariaDB-0ubuntu0.22.04.1 Ubuntu 22.04

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> create database pweb1
-> ;
Query OK, 1 row affected (0.001 sec)
MariaDB [(none)]> GRANT ALL PRIVILEGES ON pweb1.* TO 'alumno'@'%' IDENTIFIED BY 'pweb1';
Query OK, 0 rows affected (0.015 sec)

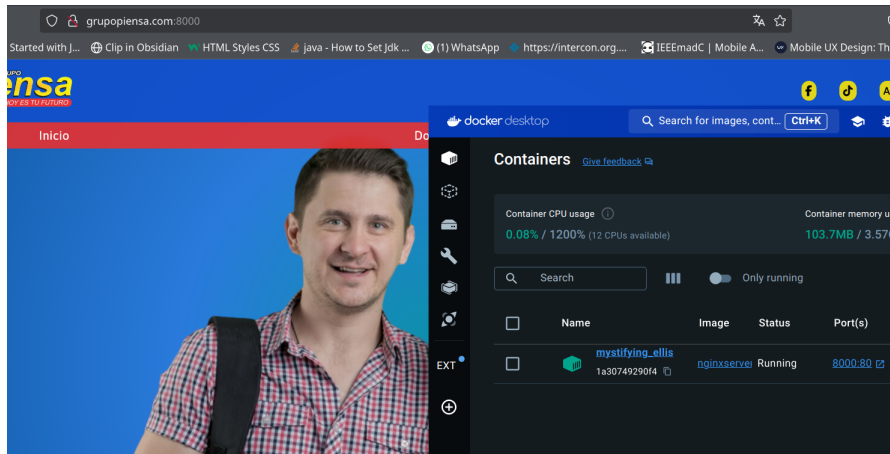
MariaDB [(none)]> exit
Bye
```

#### 1.4 Finalmente implemente el trabajo final del curso de pw1 en ese contenedor.

Elabore un informe paso a paso para donde explique funcionalmente el proyecto demostrando que se trata de un contenedor docker.

Adjunte la URL de un video donde muestre que se trata de un contenedor Docker.

Entonces para probar mi contenedor de docker que este correctamente sirviendo los servidores de nginx voy a entrar al buscador de firefox y buscar la pagina web grupopiensa, el cual esta mapeado al localhost 127.0.0.1 en el puerto 8000, el cual tiene un port forwarding al puerto 80 del contenedor



Aquí dejo el enlace al video de YT en el que se muestra la veracidad del uso de Docker para hostear paginas web: <https://youtu.be/ERklT6IMV7E>

## 2 Rúbricas

### 2.1 Entregable Informe

Tabla 1: Tipo de Informe

<b>Informe</b>	
<b>Latex</b>	El informe está en formato PDF desde Latex, con un formato limpio (buena presentación) y fácil de leer.

### 2.2 Rúbrica para el contenido del Informe y demostración

- El alumno debe marcar o dejar en blanco en celdas de la columna **Checklist** si cumple con el ítem correspondiente.
- Si un alumno supera la fecha de entrega, su calificación será sobre la nota mínima aprobada, siempre y cuando cumpla con todos los ítems.
- El alumno debe autocalificarse en la columna **Estudiante** de acuerdo a la siguiente tabla:

Tabla 2: Niveles de desempeño

<b>Puntos</b>	Nivel			
	Insatisfactorio 25%	En Proceso 50%	Satisfactorio 75%	Sobresaliente 100%
<b>2.0</b>	0.5	1.0	1.5	2.0
<b>4.0</b>	1.0	2.0	3.0	4.0

Tabla 3: Rúbrica para contenido del Informe y demostración

Contenido y demostración		Puntos	Checklist	Estudiante	Profesor
<b>1. GitHub</b>	Hay enlace URL activo del directorio para el laboratorio hacia su repositorio GitHub con código fuente terminado y fácil de revisar.	2	X	2	
<b>2. Commits</b>	Hay capturas de pantalla de los commits más importantes con sus explicaciones detalladas. (El profesor puede preguntar para refrendar calificación).	4	X	4	
<b>3. Código fuente</b>	Hay porciones de código fuente importantes con numeración y explicaciones detalladas de sus funciones.	2	X	1	
<b>4. Ejecución</b>	Se incluyen ejecuciones/pruebas del código fuente explicadas gradualmente.	2	X	1	
<b>5. Pregunta</b>	Se responde con completitud a la pregunta formulada en la tarea. (El profesor puede preguntar para refrendar calificación).	2	X	2	
<b>6. Fechas</b>	Las fechas de modificación del código fuente están dentro de los plazos de fecha de entrega establecidos.	2	X	0	
<b>7. Ortografía</b>	El documento no muestra errores ortográficos.	2	X	2	
<b>8. Madurez</b>	El Informe muestra de manera general una evolución de la madurez del código fuente, explicaciones puntuales pero precisas.	4	X	4	
<b>Total</b>		20		16	

### 3 Referencias

- [https://drive.google.com/file/d/19hjCsMtViypEj3c0PRF\\_cflH1r8r9vBn/view?usp=sharing](https://drive.google.com/file/d/19hjCsMtViypEj3c0PRF_cflH1r8r9vBn/view?usp=sharing)