Guía Final de Tercer Corte

Guía elaborada por: ***Carlos David Páez Ferreira***

Guía Liderada por: ***Harvey Nicolás Echavarria Ortiz***

Fecha: ***17 Mayo 2022***

Apartado 1: Docker

Verificar la instalación de Docker

```
docker --version
```

```
C:\Users\carlo>docker --version
Docker version 20.10.14, build a224086

C:\Users\carlo>
```

Instalar imágenes necesarias

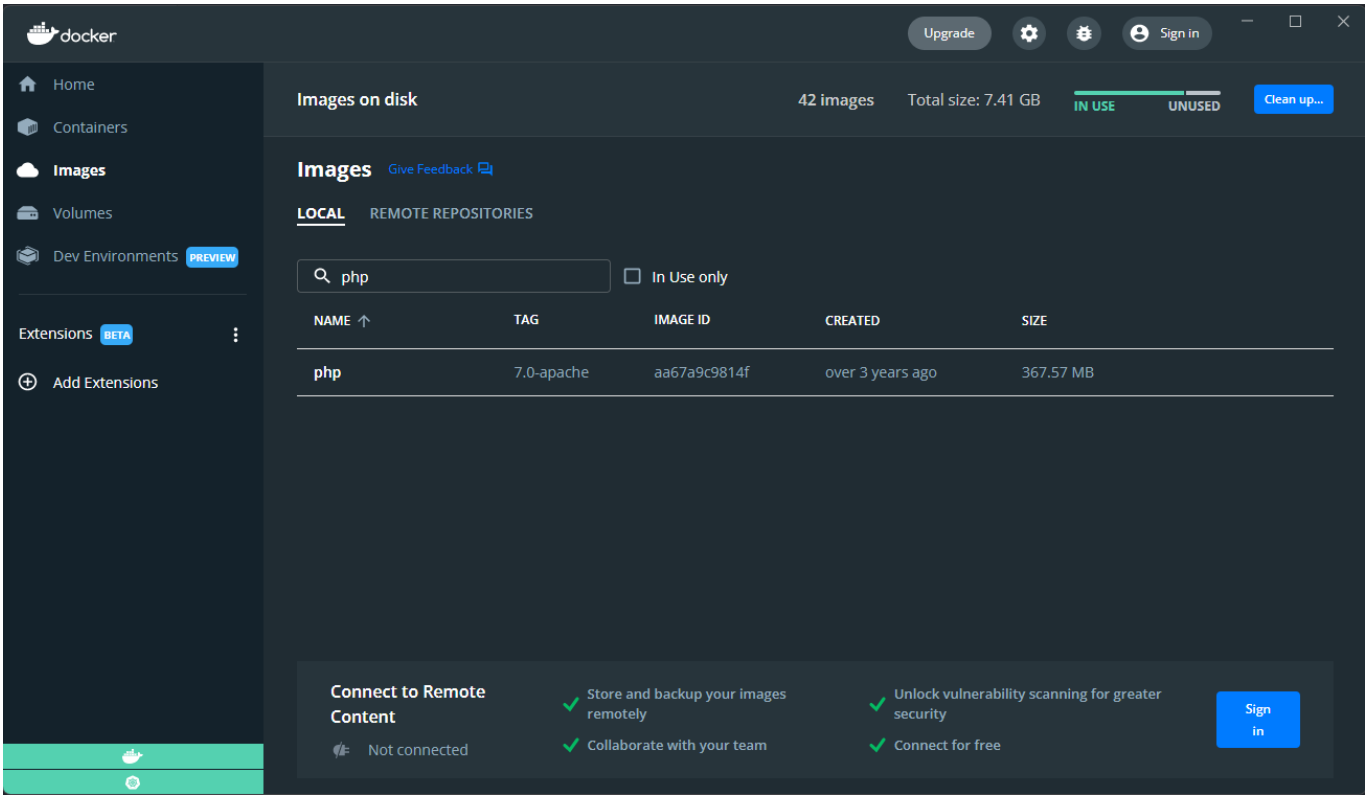
Instalar las imágenes de MySQL 5.7 y PHP 7.0 junto Apache.

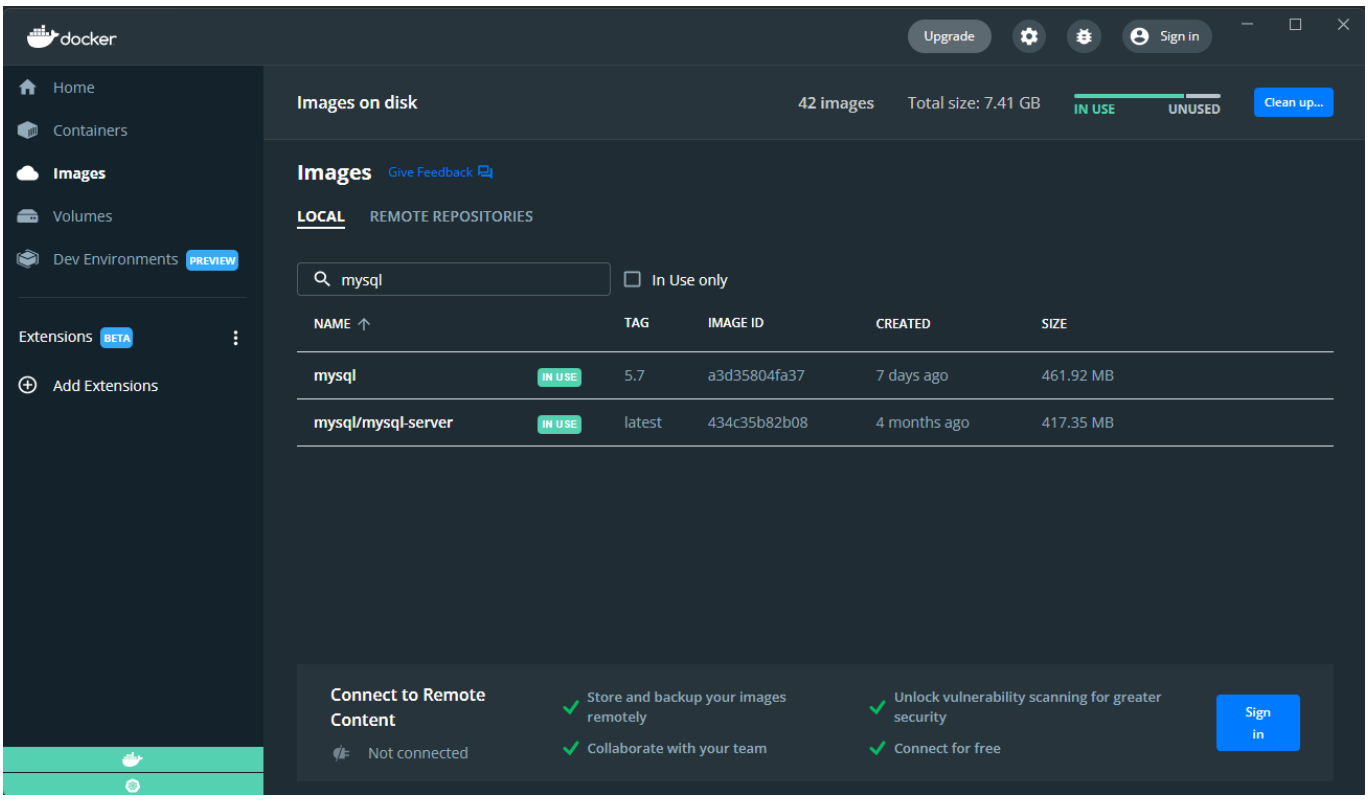
```
docker pull mysql:5.7 & docker pull php:7.0-apache
```

```
C:\Users\carlo>docker pull mysql:5.7 & docker pull php:7.0-apache
5.7: Pulling from library/mysql
Digest: sha256:16e159331007eccc069822f7b731272043ed572a79a196a05ffa2ea127caaf67
Status: Image is up to date for mysql:5.7
docker.io/library/mysql:5.7
7.0-apache: Pulling from library/php
177e7ef0df69: Pull complete
9bf89f2eda24: Pull complete
350207dcf1b7: Pull complete
a8a33d96b4e7: Pull complete
c0421d5b63d6: Pull complete
f76e300fbe72: Pull complete
af9ff1b9ce5b: Pull complete
d9f072d61771: Pull complete
37007e292198: Pull complete
8ba923990f24: Pull complete
98af8902979a: Pull complete
f1548c2cd376: Pull complete
e1062fd0605a: Pull complete
Digest: sha256:1d34b2e491a02ba7a8d26478132015e197a5ffea37f0a93b42621d11cfe042cc
Status: Downloaded newer image for php:7.0-apache
docker.io/library/php:7.0-apache

C:\Users\carlo>
```

Verificar las imágenes instaladas






```
docker images
```

```
C:\Users\carlo>docker images
REPOSITORY
mysql
hubproxy.docker.internal:5000/docker/desktop-kubernetes
k8s.gcr.io/kube-apiserver
k8s.gcr.io/kube-proxy
k8s.gcr.io/kube-scheduler
k8s.gcr.io/kube-controller-manager
k8s.gcr.io/etcd
p2t1_node_mongo_web
<none>
<none>
<none>
<none>
<none>
hellonode
mongo
hubproxy.docker.internal:5000/docker/desktop-kubernetes
dpage/pgadmin4
k8s.gcr.io/pause
postgres
gcr.io/k8s-minikube/kicbase
B
mysql/mysql-server
hubproxy.docker.internal:5000/docker/desktop-kubernetes-apiserver
k8s.gcr.io/kube-apiserver
hubproxy.docker.internal:5000/docker/desktop-kubernetes-proxy
k8s.gcr.io/kube-proxy
hubproxy.docker.internal:5000/docker/desktop-kubernetes-scheduler
B
k8s.gcr.io/kube-scheduler
B
hubproxy.docker.internal:5000/docker/desktop-kubernetes-controller-manager
k8s.gcr.io/kube-controller-manager
k8s.gcr.io/coredns/coredns
B
hubproxy.docker.internal:5000/docker/desktop-kubernetes-etcd
k8s.gcr.io/etcd
hubproxy.docker.internal:5000/docker/desktop-kubernetes-coredns
B
k8s.gcr.io/coredns/coredns
B
mariadb/server
hubproxy.docker.internal:5000/docker/desktop-vpnkit-controller
docker/desktop-vpnkit-controller
docker/desktop-storage-provisioner
B
hubproxy.docker.internal:5000/docker/desktop-storage-provisioner
B
hubproxy.docker.internal:5000/docker/desktop-kubernetes-pause
k8s.gcr.io/pause
php

C:\Users\carlo>
```

Correr imagen de MySQL

Vamos a ejecutar diferentes banderas al momento de correr la imagen de MySQL:

- `-p`: Indica el puerto por el cual escuchará la aplicación y el puerto del contenedor.

- `--name`: Indica el nombre del proceso o contenedor que se iniciará.
- `-v`: Indica que se creará un volumen, el cual permite almacenar datos de la DB en una carpeta con el fin de salvarlos en caso de que el contenedor se apague o se caiga. La primera rta hace referencia a la carpeta que se creará en nuestro equipo, la segunda hace referencia a la dirección del contenedor por defecto.
- `-e`: Indica la contraseña
- `-d`, para que MySQL corra en segundo plano.

```
docker run -p 3307:3306 --name database -v
C:/Users/carlo/.../P3T1_Guia_Final:/var/lib/mysql -e MYSQL_ROOT_PASSWORD=password
-d mysql:5.7
```

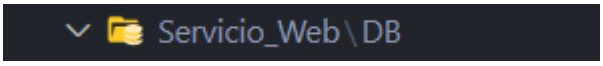
```
C:\Users\carlo>docker run -p 3307:3306 --name database -v C:/Users/carlo/Documents/Noveno_Semestre/DevOps/Tercer_Corte/P3T1_Guia_Final:/var/lib/mysql -e MYSQL_ROOT_PASSWORD=password -d mysql
:5.7
12ce148768567463f54d1af81fdb23d959d5f10f0c886f141a7ecb9288fd8d5a
```

Verificamos que este corriendo nuestro contenedor:

```
docker ps
```

```
C:\Users\carlo>docker ps
CONTAINER ID   IMAGE                COMMAND                  CREATED          STATUS          PORTS                               NAMES
428ddaf9b4f0   mysql:5.7            "docker-entrypoint.s…"   43 seconds ago   Up 42 seconds   33060/tcp, 0.0.0.0:3307->3306/tcp   database
cc96093816f3   p2t1_node_mongo_web  "docker-entrypoint.s…"   6 weeks ago      Up 2 hours      0.0.0.0:5000->3000/tcp              exampleapp
21b2aad70c24   postgres             "docker-entrypoint.s…"   2 months ago     Up 2 hours      0.0.0.0:5432->5432/tcp              src_postgres_1
```

Dentro del directorio que hemos escogido, creamos una carpeta para almacenar una página web, y dentro de este directorio, añadimos un nuevo folder para la base de datos.



Correr la imagen de PHP

Vamos a correr la imagen de PHP con las siguientes banderas:

- `-p`: Indicar el puerto
- `-name`: Indicar el nombre del proceso o contenedor
- `-v`: Indicar la creación de un volumen en el que almacenaremos los archivos de nuestro proyecto.
- `-d`: Correr en segundo plano
- `--link`: Comunicar entre el contenedor de PHP y la base de datos.

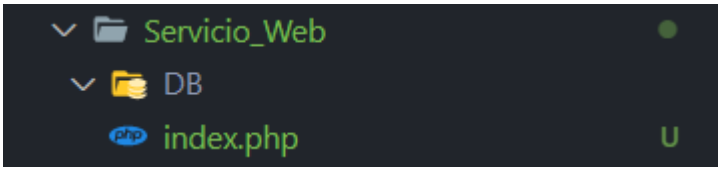
```
docker run -p 1000:80 -v
C:/Users/carlo/.../P3T1_Guia_Final/Servicio_Web:/var/www/html --name servidorphp -
d --link database php:7.0-apache
```

```
C:\Users\carlo>docker run -p 1000:80 -v C:/Users/carlo/                    /P3T1_Guia_Final/Servicio_Web:/var/www/html --name servidorphp -d --link database php:7.0-
apache
695a031207378fb71baecbdbb39d5c632b205e2e549a93d65a225eb1ee6693cf
```

```
docker ps
```

```
C:\Users\carlo>docker ps
CONTAINER ID   IMAGE                COMMAND                  CREATED              STATUS              PORTS                                NAMES
5a67bfeac68c   php:7.0-apache       "docker-php-entrypoi…"   About a minute ago   Up About a minute   0.0.0.0:1000->80/tcp                 servidorphp
428ddaf9b4f0   mysql:5.7            "docker-entrypoint.s…"   4 minutes ago        Up 4 minutes        33060/tcp, 0.0.0.0:3307->3306/tcp    database
cc96093816f3   p2ti_node_mongo_web  "docker-entrypoint.s…"   6 weeks ago          Up 2 hours          0.0.0.0:5000->3000/tcp               exampleapp
21b2aad70c24   postgres             "docker-entrypoint.s…"   2 months ago         Up 2 hours          0.0.0.0:5432->5432/tcp               src_postgres_1
```

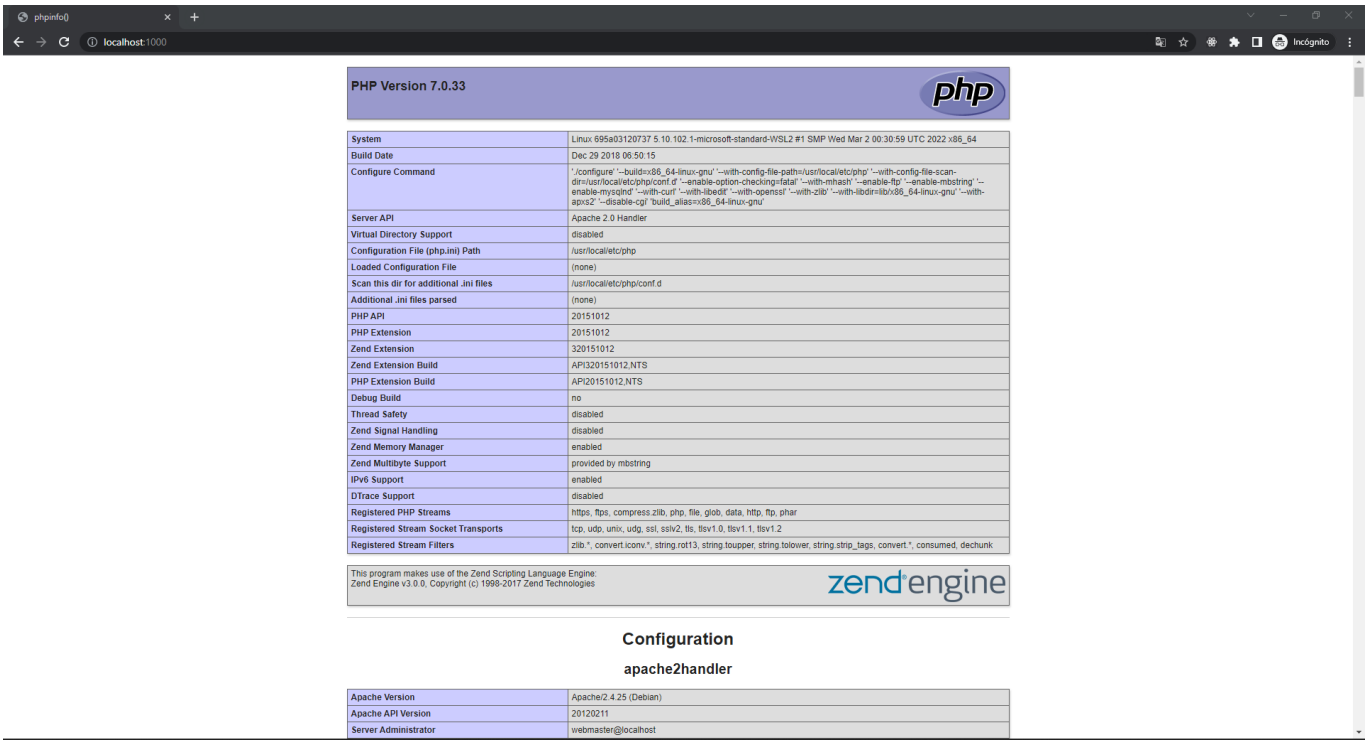
Creamos un archivo llamado `index.php` dentro de la carpeta de `Servicio_Web`:



Dentro de nuestro nuevo archivo, escribimos lo siguiente:

```
<?php
phpinfo();
?>
```

Luego, en el navegador ingresamos la siguiente ruta: `http://localhost:1000` y debemos obtener lo siguiente:



Comprobar que el servidor MySQL funciona

Entrar a la consola interactiva del contenedor de mysql:

```
docker exec -i -t database bash
```

```
C:\Users\carlo>docker exec -i -t database bash
root@e2749101e305:/#
```

Ejecutar MySQL:

```
mysql -u root -p
```

```
root@e2749101e305:/# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.7.38 MySQL Community Server (GPL)

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

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

mysql>
```

Listar las bases de datos:

```
show databases;
```

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
4 rows in set (0.00 sec)
```

Crear base de datos de usuarios:

```
create database usuarios;
```

```
mysql> create database usuarios;
Query OK, 1 row affected (0.01 sec)

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
| usuarios           |
+--------------------+
5 rows in set (0.00 sec)
```

Usar la base de datos recién creada:

```
use usuarios;
```

```
mysql> use usuarios;
Database changed
mysql>
```

Crear tabla tabla de clientes:

```
create table clientes(username varchar(20) primary key not null, nombre
varchar(30), correo varchar(50), contra varchar(20));
```

```
mysql> create table clientes(username varchar(20) primary key not null, nombre varchar(30), correo varchar(50), contra varchar(20));
Query OK, 0 rows affected (0.05 sec)

mysql> show tables;
+-------------------+
| Tables_in_usuarios |
+-------------------+
| clientes          |
+-------------------+
1 row in set (0.00 sec)
```

Insertar datos dentro de la tabla:

```
insert into clientes values ('neo', 'neo', 'neo@gmail.com', 'neo_password');
```

```
mysql> insert into clientes values ('neo', 'neo', 'neo@gmail.com', 'neo_password');
Query OK, 1 row affected (0.01 sec)

mysql> select * from clientes;
+----------+--------+---------------+--------------+
| username | nombre | correo        | contra       |
+----------+--------+---------------+--------------+
| neo      | neo    | neo@gmail.com | neo_password |
+----------+--------+---------------+--------------+
1 row in set (0.00 sec)
```

Conectar PHP con la base de datos

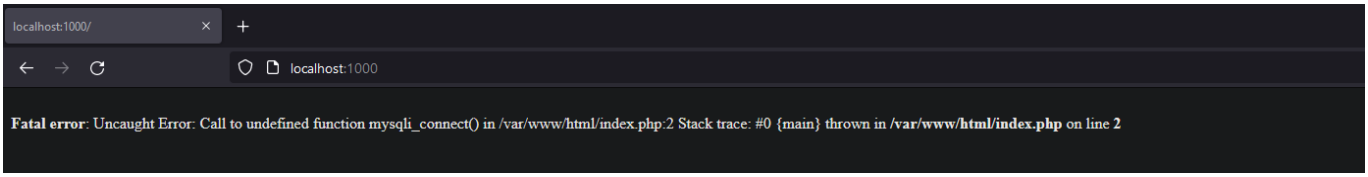
Dentro del archivo `index.php` añadimos las siguientes líneas:

```
<?php
$conn = mysqli_connect("database:3306", "root", "password", "usuarios");

if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}

echo "Connected successfully";
?>
```

Cuando regresamos a la ruta dentro del navegador, vamos a observar el siguiente error:



Esto se debe a que la función `mysqli_connection()` no se encuentra definida, puesto que el servidor que tenemos instalado de php no tiene la extensión para conectarse a mysql. Para solucionar dicho error debemos seguir estos pasos:

- Abrir el contenedor de PHP por medio del siguiente comando:

  ```
  docker exec -i -t servidorphp /bin/bash
  ```

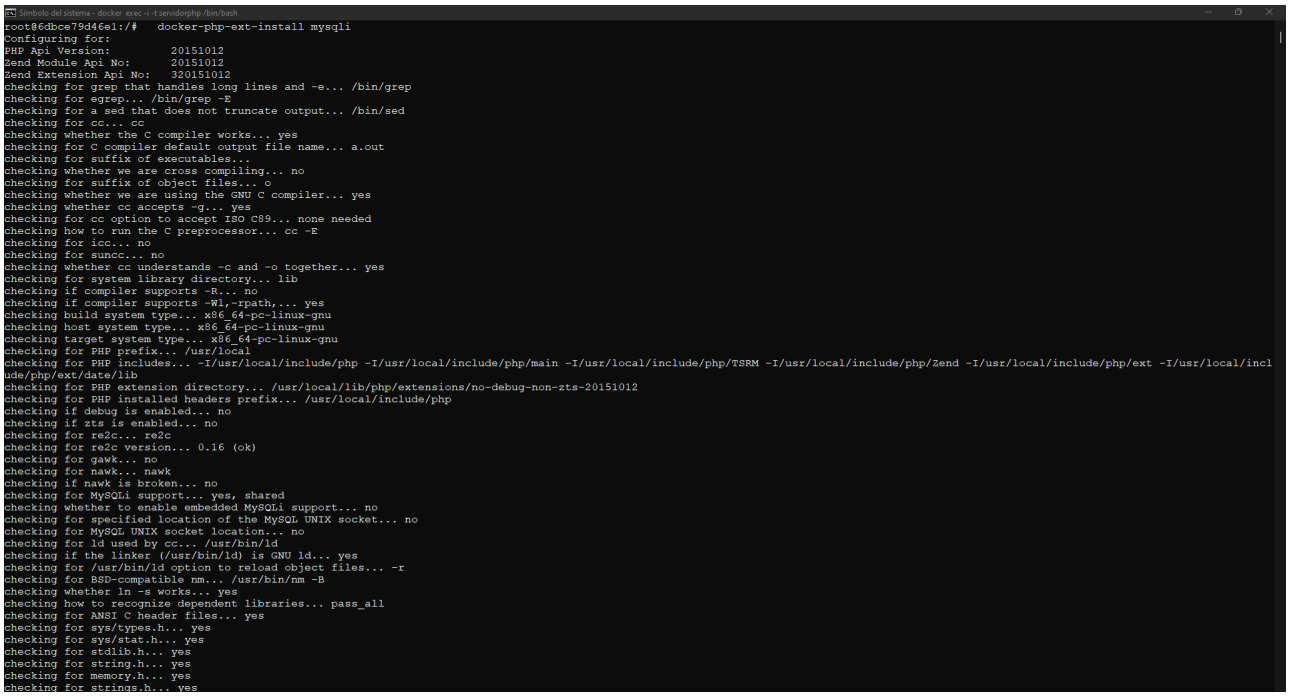
  ```
  C:\Users\carlo>docker exec -i -t servidorphp /bin/bash
  root@6dbce79d46e1:/var/www/html#
  ```

- Ir al directorio `/` y listar los archivos y directorios:

  ```
  root@6dbce79d46e1:/var/www/html# cd /
  root@6dbce79d46e1:/# ls
  bin  boot  dev  etc  home  lib  lib64  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
  root@6dbce79d46e1:/#
  ```

- Usamos el siguiente comando para instalar la extensión necesaria:

  ```
  docker-php-ext-install mysqli
  ```



Debemos copiar la ruta que aparece en la sección de `Installing shared extensions` al momento de terminar la instalación de la extensión. (`/usr/local/lib/php/extensions/no-debug-non-zts-20151012/`)

```
Build complete.
Don't forget to run 'make test'.

Installing shared extensions:     /usr/local/lib/php/extensions/no-debug-non-zts-20151012/
Installing header files:          /usr/local/include/php/
find . -name \*.gcno -o -name \*.gcda | xargs rm -f
find . -name \*.lo -o -name \*.o | xargs rm -f
find . -name \*.la -o -name \*.a | xargs rm -f
find . -name \*.so | xargs rm -f
find . -name .libs -a -type d|xargs rm -rf
rm -f libphp.la       modules/* libs/*
```

- Nos ubicamos dentro de la ruta `/usr/local/etc/php/` y listamos los archivos que se encuentran dentro de dicho directorio

```
root@6dbce79d46e1:/# cd /usr/local/etc/php/
root@6dbce79d46e1:/usr/local/etc/php# ls
conf.d  php.ini-development  php.ini-production
root@6dbce79d46e1:/usr/local/etc/php#
```

- Instalamos nano con los siguientes comandos:

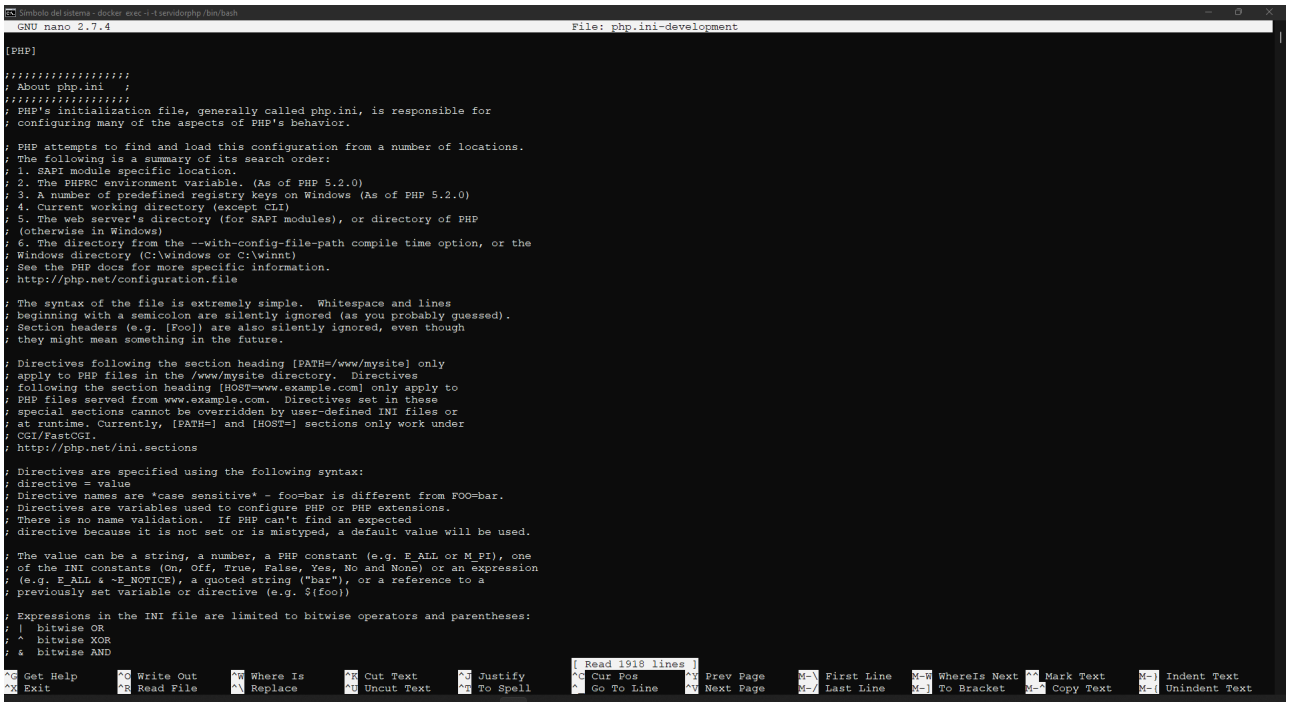
```
apt-get update
```

```
apt-get install nano
```

```
root@6dbce79d46e1:/usr/local/lib/php# apt-get update
Hit:1 http://security.debian.org/debian-security stretch/updates InRelease
Ign:2 http://deb.debian.org/debian stretch InRelease
Hit:3 http://deb.debian.org/debian stretch-updates InRelease
Hit:4 http://deb.debian.org/debian stretch Release
Reading package lists... Done
root@6dbce79d46e1:/usr/local/lib/php# apt-get install nano
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  spell
The following NEW packages will be installed:
  nano
0 upgraded, 1 newly installed, 0 to remove and 64 not upgraded.
Need to get 485 kB of archives.
After this operation, 2092 kB of additional disk space will be used.
Get:1 http://deb.debian.org/debian stretch/main amd64 nano amd64 2.7.4-1 [485 kB]
Fetched 485 kB in 0s (1093 kB/s)
debconf: unable to initialize frontend: Dialog
debconf: (No usable dialog-like program is installed, so the dialog based frontend cannot be used. at /usr/share/perl5/Debconf/FrontEnd/Dialog.pm line 76, <> line 1.)
debconf: falling back to frontend: Readline
Selecting previously unselected package nano.
(Reading database ... 18637 files and directories currently installed.)
Preparing to unpack .../nano_2.7.4-1_amd64.deb ...
Unpacking nano (2.7.4-1) ...
Setting up nano (2.7.4-1) ...
update-alternatives: using /bin/nano to provide /usr/bin/editor (editor) in auto mode
update-alternatives: warning: skip creation of /usr/share/man/man1/editor.1.gz because associated file /usr/share/man/man1/nano.1.gz (of link group editor) doesn't exist
update-alternatives: using /bin/nano to provide /usr/bin/pico (pico) in auto mode
update-alternatives: warning: skip creation of /usr/share/man/man1/pico.1.gz because associated file /usr/share/man/man1/nano.1.gz (of link group pico) doesn't exist
root@6dbce79d46e1:/usr/local/lib/php#
```

- Abrimos el archivo `php.ini-development` mediante el siguiente comando:

```
nano php.ini-development
```



Luego, buscamos el apartado de rutas dinámicas y añadimos la ruta que teníamos copiada, y salimos del editor con `Ctrl + X`, `y` y `Enter`:

```
;;;;;;;;;;;;;;;;;;;;;
; Dynamic Extensions ;
;;;;;;;;;;;;;;;;;;;;;

; If you wish to have an extension loaded automatically, use the following
; syntax:
;
;   extension=modulename.extension
;
; For example, on Windows:
;
;   extension=msql.dll
;
; ... or under UNIX:
;
;   extension=msql.so
;
; ... or with a path:
;
;   extension=/path/to/extension/msql.so

/usr/local/lib/php/extensions/no-debug-non-zts-20151012/
```

- Aplicamos el mismo procedimiento para el archivo de `php.init-production`:

```
root@6dbce79d46e1:/usr/local/etc/php# nano php.ini-development
root@6dbce79d46e1:/usr/local/etc/php# nano php.ini-production
root@6dbce79d46e1:/usr/local/etc/php#
```

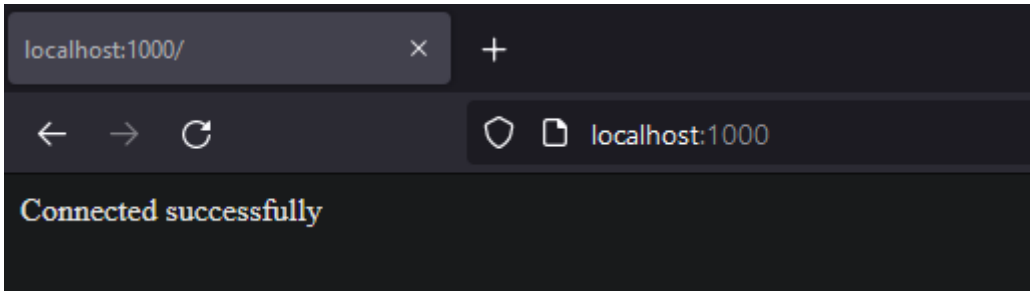
- Reiniciamos el contenedor de PHP con el siguiente comando:

```
docker restart servidorphp
```

```
C:\Users\carlo>docker restart servidorphp
servidorphp
```

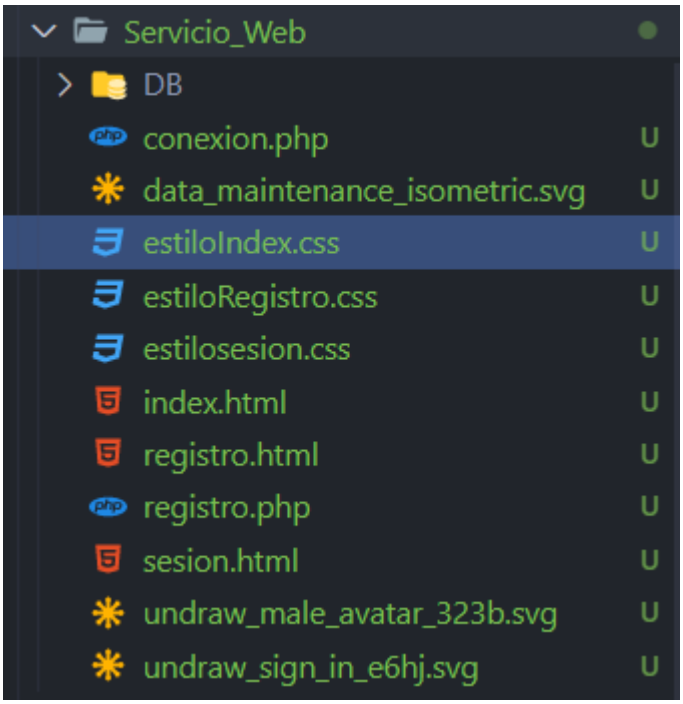
```
C:\Users\carlo>docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED         STATUS          PORTS                               NAMES
6dbce79d46e1   php:7.0-apache "docker-php-entrypoi…"   15 hours ago    Up 39 seconds   0.0.0.0:1000->80/tcp                servidorphp
e2749101e305   mysql:5.7      "docker-entrypoint.s…"   15 hours ago    Up 4 hours      33060/tcp, 0.0.0.0:3307->3306/tcp   database
```

- Recargamos el navegador, y esto será lo que debe aparecer:



Página Web

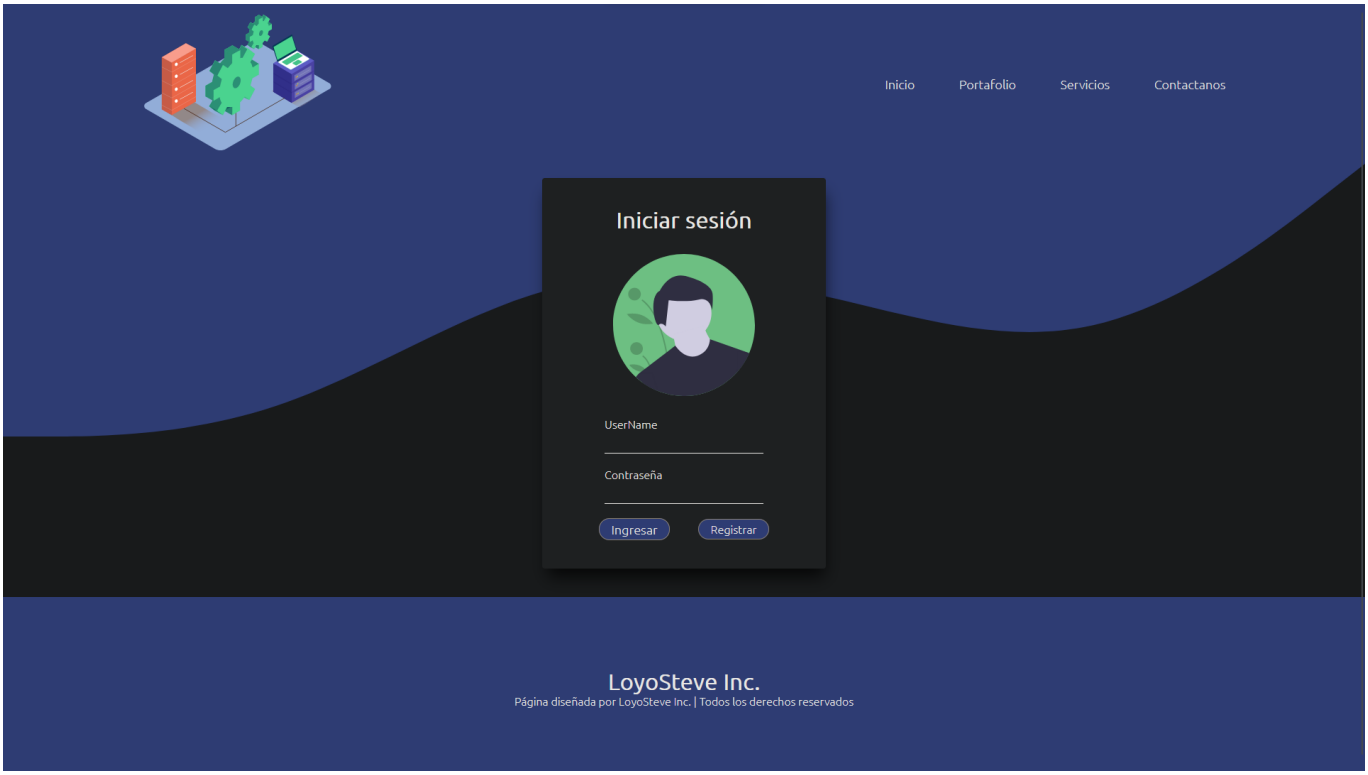
Para efectos del taller vamos a añadir algunos archivos a nuestro directorio `Servicio_Web`:

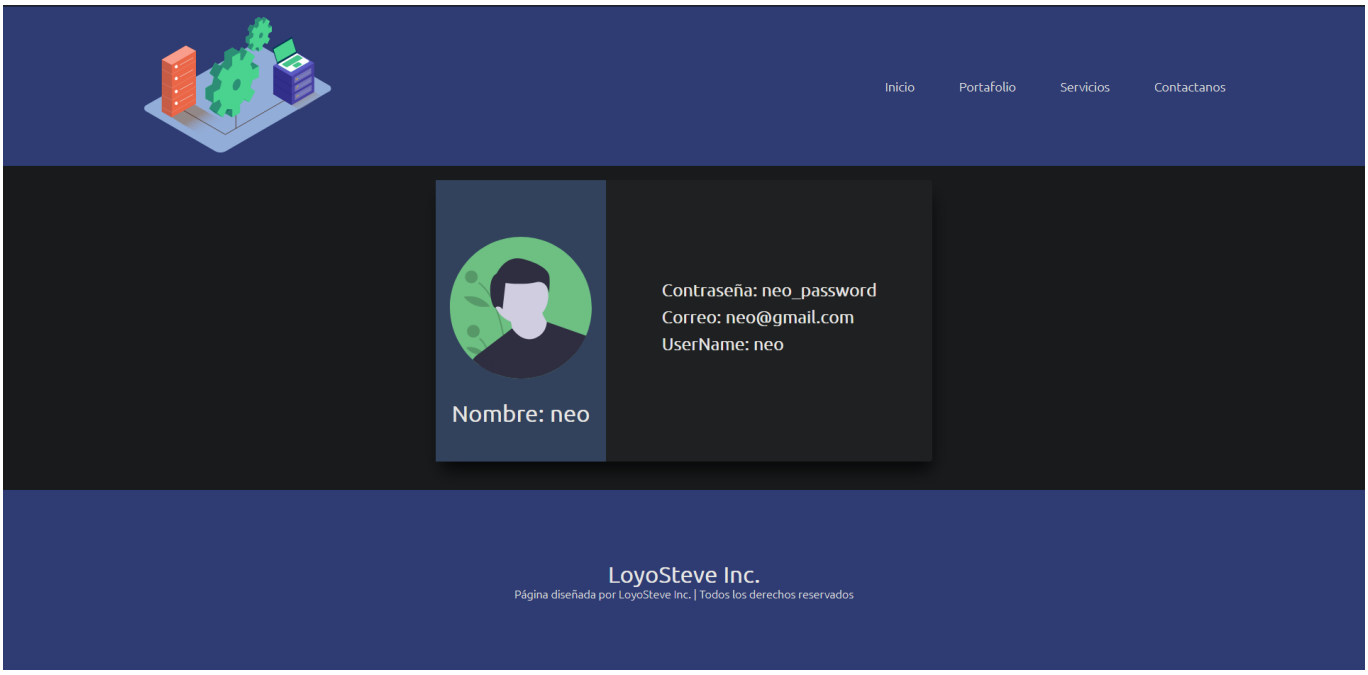


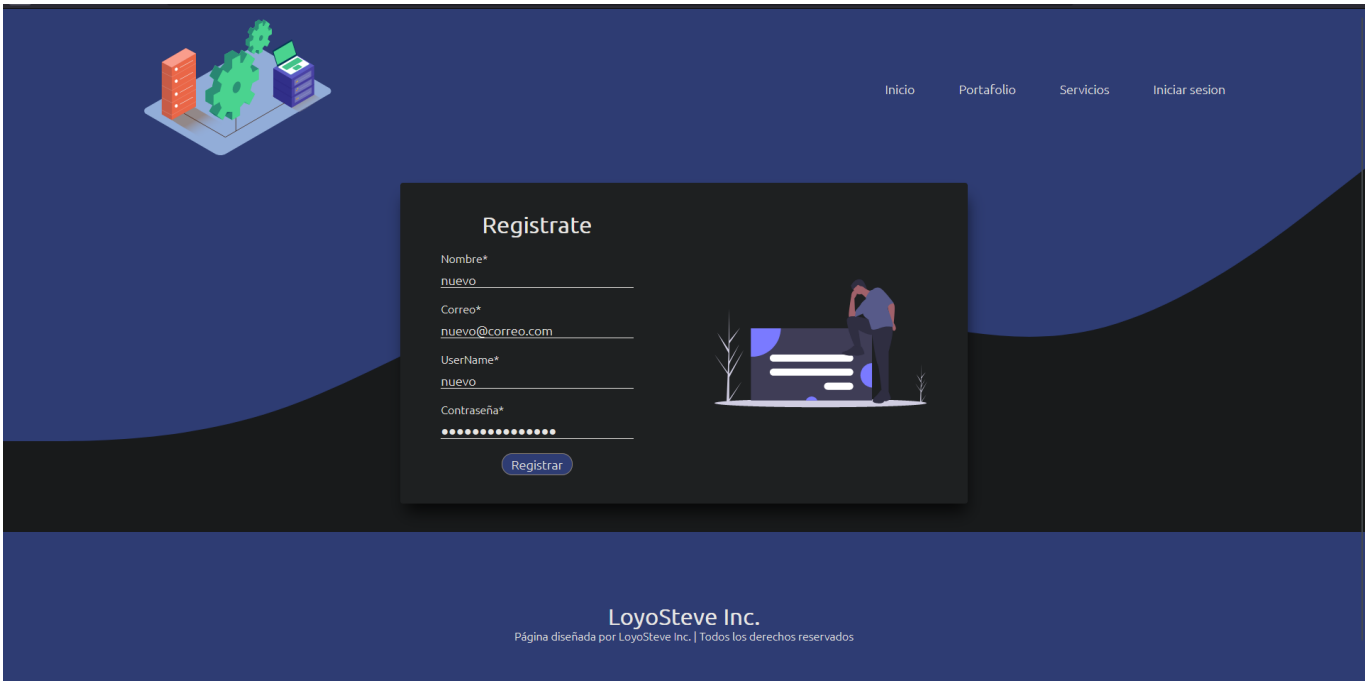
Dentro del archivo `conexion.php` y `registro.php` hacemos los siguientes cambios en la variable de conexión a la base de datos:

```
$con = mysqli_connect("database:3306","root","password","usuarios");
```

Si volvemos al navegador en la dirección `http://localhost:1000`, vamos a observar lo siguiente:







Si vamos al contenedor de la base de datos, podemos verificar que se ha registrado el usuario:

```
docker exec -i -t database bash
```

```
mysql -u root -p
```

```
use usuarios;
```

```
select * from clientes;
```

```
C:\Users\carlo>docker exec -i -t database bash
root@e2749101e305:/# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 5
Server version: 5.7.38 MySQL Community Server (GPL)

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

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

mysql> use usuarios;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> select * from clientes;
+----------+--------+------------------+-----------------+
| username | nombre | correo           | contra          |
+----------+--------+------------------+-----------------+
| neo      | neo    | neo@gmail.com    | neo_password    |
| nuevo    | nuevo  | nuevo@correo.com | xFEPjtGgs9WKJpq |
+----------+--------+------------------+-----------------+
2 rows in set (0.00 sec)

mysql>
```

Docker-Compose

Dentro del directorio `Servicio_Web` creamos el archivo `docker-compose.yml`, dentro del cual tendremos la siguiente información:

```
version: '3'

services:
    mysql:
        image: mysql:5.7
        container_name: docker-mysql
        environment:
            MYSQL_DATABASE: usuarios
            MYSQL_ROOT_USER: user_docker
            MYSQL_USER: user_docker
```

```
            MYSQL_PASSWORD: password
            MYSQL_ROOT_PASSWORD: password
        ports:
            - "3307:3306"
        restart: always

    web:
        image: php:7.0-apache
        container_name: docker-php
        ports:
            - "1000:80"
        volumes:
            - ./www:/var/www/html
        links:
            - mysql
```

Vamos a bajar nuestro contenedores:

```
docker stop database & docker stop servidorphp
```

```
C:\Users\carlo>docker stop database & docker stop servidorphp
database
servidorphp
```

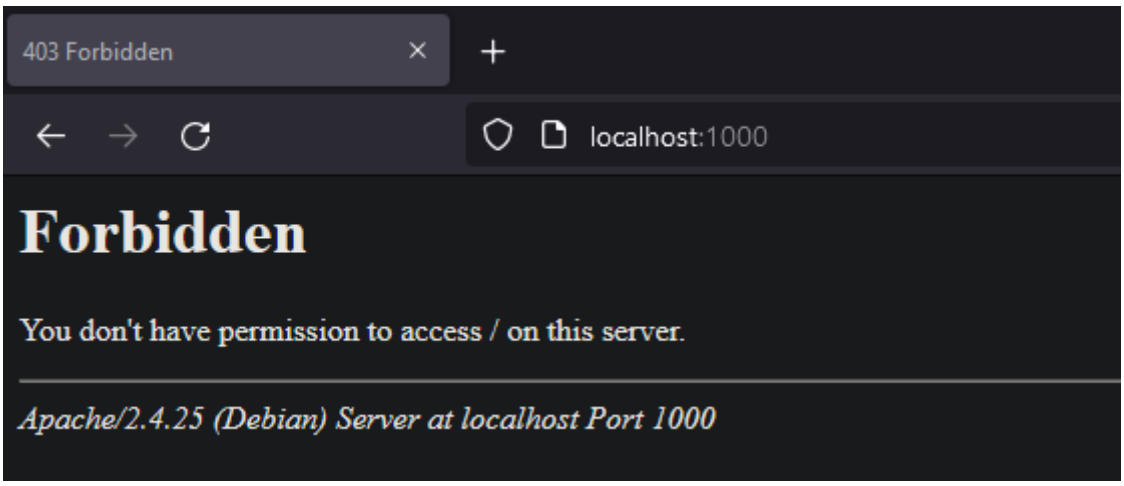
Y ahora usamos el siguiente comando dentro del directorio `Servicio_Web`:

```
docker compose-up
```

```
C:\Users\carlo\Documents\Noveno_Semestre\DevOps\Tercer_Corte\P3T1_Guia_Final\Servicio_Web>docker-compose up
Recreating docker-mysql ... done
Recreating docker-php   ... done
Attaching to docker-mysql, docker-php
docker-mysql | 2022-05-18 18:57:41+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 5.7.38-1debian10 started.
docker-mysql | 2022-05-18 18:57:41+00:00 [Note] [Entrypoint]: Switching to dedicated user 'mysql'
docker-mysql | 2022-05-18 18:57:41+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 5.7.38-1debian10 started.
docker-mysql | 2022-05-18 18:57:41+00:00 [Note] [Entrypoint]: Initializing database files
docker-mysql | 2022-05-18T18:57:41.844473Z 0 [Warning] TIMESTAMP with implicit DEFAULT value is deprecated. Please use --explicit_defaults_for_timestamp server option (see documentation for
more details).
docker-mysql | 2022-05-18T18:57:42.051006Z 0 [Warning] InnoDB: New log files created, LSN=45790
docker-mysql | 2022-05-18T18:57:42.096368Z 0 [Warning] InnoDB: Creating foreign key constraint system tables.
docker-mysql | 2022-05-18T18:57:42.108565Z 0 [Warning] No existing UUID has been found, so we assume that this is the first time that this server has been started. Generating a new UUID: 653
9e74e-d6dc-11ec-8777-0242ac130002.
docker-mysql | 2022-05-18T18:57:42.110121Z 0 [Warning] Gtid table is not ready to be used. Table 'mysql.gtid_executed' cannot be opened.
docker-mysql | 2022-05-18T18:57:42.365370Z 0 [Warning] A deprecated TLS version TLSv1 is enabled. Please use TLSv1.2 or higher.
docker-mysql | 2022-05-18T18:57:42.365421Z 0 [Warning] A deprecated TLS version TLSv1.1 is enabled. Please use TLSv1.2 or higher.
docker-mysql | 2022-05-18T18:57:42.366532Z 0 [Warning] CA certificate ca.pem is self signed.
docker-mysql | 2022-05-18T18:57:42.422885Z 1 [Warning] root@localhost is created with an empty password ! Please consider switching off the --initialize-insecure option.
docker-php | AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 172.19.0.3. Set the 'ServerName' directive globally to suppress this message
docker-php | AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 172.19.0.3. Set the 'ServerName' directive globally to suppress this message
docker-php | [Wed May 18 18:57:42.561691 2022] [mpm_prefork:notice] [pid 1] AH00163: Apache/2.4.25 (Debian) PHP/7.0.33 configured -- resuming normal operations
docker-php | [Wed May 18 18:57:42.561936 2022] [core:notice] [pid 1] AH00094: Command line: 'apache2 -D FOREGROUND'
docker-mysql | 2022-05-18 18:57:44+00:00 [Note] [Entrypoint]: Database files initialized
docker-mysql | 2022-05-18 18:57:44+00:00 [Note] [Entrypoint]: Starting temporary server
docker-mysql | 2022-05-18 18:57:44+00:00 [Note] [Entrypoint]: Waiting for server startup
docker-mysql | 2022-05-18T18:57:44.940735Z 0 [Warning] TIMESTAMP with implicit DEFAULT value is deprecated. Please use --explicit_defaults_for_timestamp server option (see documentation for
more details).
docker-mysql | 2022-05-18T18:57:44.942082Z 0 [Note] mysqld (mysqld 5.7.38) starting as process 78 ...
docker-mysql | 2022-05-18T18:57:44.944538Z 0 [Note] InnoDB: PUNCH HOLE support available
docker-mysql | 2022-05-18T18:57:44.944584Z 0 [Note] InnoDB: Mutexes and rw_locks use GCC atomic builtins
docker-mysql | 2022-05-18T18:57:44.944589Z 0 [Note] InnoDB: Uses event mutexes
docker-mysql | 2022-05-18T18:57:44.944592Z 0 [Note] InnoDB: GCC builtin __atomic_thread_fence() is used for memory barrier
docker-mysql | 2022-05-18T18:57:44.944593Z 0 [Note] InnoDB: Compressed tables use zlib 1.2.11
docker-mysql | 2022-05-18T18:57:44.944595Z 0 [Note] InnoDB: Using Linux native AIO
docker-mysql | 2022-05-18T18:57:44.944884Z 0 [Note] InnoDB: Number of pools: 1
docker-mysql | 2022-05-18T18:57:44.945020Z 0 [Note] InnoDB: Using CPU crc32 instructions
docker-mysql | 2022-05-18T18:57:44.946540Z 0 [Note] InnoDB: Initializing buffer pool, total size = 128M, instances = 1, chunk size = 128M
docker-mysql | 2022-05-18T18:57:44.953148Z 0 [Note] InnoDB: Completed initialization of buffer pool
docker-mysql | 2022-05-18T18:57:44.955465Z 0 [Note] InnoDB: If the mysqld execution user is authorized, page cleaner thread priority can be changed. See the man page of setpriority().
docker-mysql | 2022-05-18T18:57:44.967290Z 0 [Note] InnoDB: Highest supported file format is Barracuda.
docker-mysql | 2022-05-18T18:57:44.975298Z 0 [Note] InnoDB: Creating shared tablespace for temporary tables
docker-mysql | 2022-05-18T18:57:44.975398Z 0 [Note] InnoDB: Setting file './ibtmp1' size to 12 MB. Physically writing the file full; Please wait ...
docker-mysql | 2022-05-18T18:57:44.988764Z 0 [Note] InnoDB: File './ibtmp1' size is now 12 MB.
docker-mysql | 2022-05-18T18:57:44.989313Z 0 [Note] InnoDB: 96 redo rollback segment(s) found. 96 redo rollback segment(s) are active.
docker-mysql | 2022-05-18T18:57:44.989347Z 0 [Note] InnoDB: 32 non-redo rollback segment(s) are active.
docker-mysql | 2022-05-18T18:57:44.990280Z 0 [Note] InnoDB: 5.7.38 started; log sequence number 2749976
docker-mysql | 2022-05-18T18:57:44.990436Z 0 [Note] InnoDB: Loading buffer pool(s) from /var/lib/mysql/ib_buffer_pool
docker-mysql | 2022-05-18T18:57:44.991048Z 0 [Note] Plugin 'FEDERATED' is disabled.
docker-mysql | 2022-05-18T18:57:44.991752Z 0 [Note] InnoDB: Buffer pool(s) load completed at 220518 18:57:44
docker-mysql | 2022-05-18T18:57:44.996290Z 0 [Note] Found ca.pem, server-cert.pem and server-key.pem in data directory. Trying to enable SSL support using them.
docker-mysql | 2022-05-18T18:57:44.996331Z 0 [Note] Skipping generation of SSL certificates as certificate files are present in data directory.
docker-mysql | 2022-05-18T18:57:44.996336Z 0 [Warning] A deprecated TLS version TLSv1 is enabled. Please use TLSv1.2 or higher.
docker-mysql | 2022-05-18T18:57:44.996338Z 0 [Warning] A deprecated TLS version TLSv1.1 is enabled. Please use TLSv1.2 or higher.
docker-mysql | 2022-05-18T18:57:44.996871Z 0 [Warning] CA certificate ca.pem is self signed.
```

```
docker-mysql | 2022-05-18T18:57:48.954423Z 0 [Note] Shutting down plugin 'mysql_native_password'
docker-mysql | 2022-05-18T18:57:48.954542Z 0 [Note] Shutting down plugin 'binlog'
docker-mysql | 2022-05-18T18:57:48.955099Z 0 [Note] mysqld: Shutdown complete
docker-mysql |
docker-mysql | 2022-05-18 18:57:49+00:00 [Note] [Entrypoint]: Temporary server stopped
docker-mysql |
docker-mysql | 2022-05-18 18:57:49+00:00 [Note] [Entrypoint]: MySQL init process done. Ready for start up.
docker-mysql |
docker-mysql | 2022-05-18T18:57:49.899023Z 0 [Warning] TIMESTAMP with implicit DEFAULT value is deprecated. Please use --explicit_defaults_for_timestamp server option (see documentation for
more details).
docker-mysql | 2022-05-18T18:57:49.900161Z 0 [Note] mysqld (mysqld 5.7.38) starting as process 1 ...
docker-mysql | 2022-05-18T18:57:49.902644Z 0 [Note] InnoDB: PUNCH HOLE support available
docker-mysql | 2022-05-18T18:57:49.902736Z 0 [Note] InnoDB: Mutexes and rw_locks use GCC atomic builtins
docker-mysql | 2022-05-18T18:57:49.902744Z 0 [Note] InnoDB: Uses event mutexes
docker-mysql | 2022-05-18T18:57:49.902748Z 0 [Note] InnoDB: GCC builtin __atomic_thread_fence() is used for memory barrier
docker-mysql | 2022-05-18T18:57:49.902750Z 0 [Note] InnoDB: Compressed tables use zlib 1.2.11
docker-mysql | 2022-05-18T18:57:49.902753Z 0 [Note] InnoDB: Using Linux native AIO
docker-mysql | 2022-05-18T18:57:49.903118Z 0 [Note] InnoDB: Number of pools: 1
docker-mysql | 2022-05-18T18:57:49.903559Z 0 [Note] InnoDB: Using CPU crc32 instructions
docker-mysql | 2022-05-18T18:57:49.905519Z 0 [Note] InnoDB: Initializing buffer pool, total size = 128M, instances = 1, chunk size = 128M
docker-mysql | 2022-05-18T18:57:49.912005Z 0 [Note] InnoDB: Completed initialization of buffer pool
docker-mysql | 2022-05-18T18:57:49.913980Z 0 [Note] InnoDB: If the mysqld execution user is authorized, page cleaner thread priority can be changed. See the man page of setpriority().
docker-mysql | 2022-05-18T18:57:49.925397Z 0 [Note] InnoDB: Highest supported file format is Barracuda.
docker-mysql | 2022-05-18T18:57:49.932393Z 0 [Note] InnoDB: Creating shared tablespace for temporary tables
docker-mysql | 2022-05-18T18:57:49.932500Z 0 [Note] InnoDB: Setting file './ibtmp1' size to 12 MB. Physically writing the file full; Please wait ...
docker-mysql | 2022-05-18T18:57:49.945966Z 0 [Note] InnoDB: File './ibtmp1' size is now 12 MB.
docker-mysql | 2022-05-18T18:57:49.946485Z 0 [Note] InnoDB: 96 redo rollback segment(s) found. 96 redo rollback segment(s) are active.
docker-mysql | 2022-05-18T18:57:49.946517Z 0 [Note] InnoDB: 32 non-redo rollback segment(s) are active.
docker-mysql | 2022-05-18T18:57:49.947246Z 0 [Note] InnoDB: 5.7.38 started; log sequence number 12660054
docker-mysql | 2022-05-18T18:57:49.947505Z 0 [Note] InnoDB: Loading buffer pool(s) from /var/lib/mysql/ib_buffer_pool
docker-mysql | 2022-05-18T18:57:49.947831Z 0 [Note] Plugin 'FEDERATED' is disabled.
docker-mysql | 2022-05-18T18:57:49.950253Z 0 [Note] InnoDB: Buffer pool(s) load completed at 220518 18:57:49
docker-mysql | 2022-05-18T18:57:49.952455Z 0 [Note] Found ca.pem, server-cert.pem and server-key.pem in data directory. Trying to enable SSL support using them.
docker-mysql | 2022-05-18T18:57:49.952500Z 0 [Note] Skipping generation of SSL certificates as certificate files are present in data directory.
docker-mysql | 2022-05-18T18:57:49.952506Z 0 [Warning] A deprecated TLS version TLSv1 is enabled. Please use TLSv1.2 or higher.
docker-mysql | 2022-05-18T18:57:49.952508Z 0 [Warning] A deprecated TLS version TLSv1.1 is enabled. Please use TLSv1.2 or higher.
docker-mysql | 2022-05-18T18:57:49.953008Z 0 [Warning] CA certificate ca.pem is self signed.
docker-mysql | 2022-05-18T18:57:49.953056Z 0 [Note] Skipping generation of RSA key pair as key files are present in data directory.
docker-mysql | 2022-05-18T18:57:49.953499Z 0 [Note] Server hostname (bind-address): '*'; port: 3306
docker-mysql | 2022-05-18T18:57:49.953586Z 0 [Note] IPv6 is available.
docker-mysql | 2022-05-18T18:57:49.953743Z 0 [Note]   - '::' resolves to '::';
docker-mysql | 2022-05-18T18:57:49.953777Z 0 [Note] Server socket created on IP: '::'.
docker-mysql | 2022-05-18T18:57:49.955691Z 0 [Warning] Insecure configuration for --pid-file: Location '/var/run/mysqld' in the path is accessible to all OS users. Consider choosing a differ
ent directory.
docker-mysql | 2022-05-18T18:57:49.964403Z 0 [Note] Event Scheduler: Loaded 0 events
docker-mysql | 2022-05-18T18:57:49.964717Z 0 [Note] mysqld: ready for connections.
docker-mysql | Version: '5.7.38'  socket: '/var/run/mysqld/mysqld.sock'  port: 3306  MySQL Community Server (GPL)
```

Volvemos a intentar correr nuestra aplicación dentro del navegador, y recibiremos el siguiente mensaje debido a que no se configuro los directorio indicies (`index.php`, `index.html`).




```
docker-mysql | 2022-05-18T18:57:49.964403Z 0 [Note] Event Scheduler: Loaded 0 events
docker-mysql | 2022-05-18T18:57:49.964717Z 0 [Note] mysqld: ready for connections.
docker-mysql | Version: '5.7.38'  socket: '/var/run/mysqld/mysqld.sock'  port: 3306  MySQL Community Server (GPL)
docker-php | 172.19.0.1 - - [18/May/2022:18:59:36 +0000] "GET /estiloIndex.css HTTP/1.1" 404 507 "http://localhost:1000/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:101.0) Gecko/20100101
Firefox/101.0"
docker-php | [Wed May 18 18:59:54.788676 2022] [:error] [pid 18] [client 172.19.0.1:54742] script '/var/www/html/conexion.php' not found or unable to stat, referer: http://localhost:1000/
docker-php | 172.19.0.1 - - [18/May/2022:18:59:54 +0000] "POST /conexion.php HTTP/1.1" 404 504 "http://localhost:1000/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:101.0) Gecko/20100101 Fi
refox/101.0"
docker-php | [Wed May 18 19:00:17.721398 2022] [:error] [pid 19] [client 172.19.0.1:54744] script '/var/www/html/registro.php' not found or unable to stat, referer: http://localhost:1000/reg
istro.html
docker-php | 172.19.0.1 - - [18/May/2022:19:00:17 +0000] "POST /registro.php HTTP/1.1" 404 504 "http://localhost:1000/registro.html" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:101.0) Geck
o/20100101 Firefox/101.0"
docker-php | 172.19.0.1 - - [18/May/2022:19:00:25 +0000] "GET /registro.html HTTP/1.1" 404 505 "http://localhost:1000/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:101.0) Gecko/20100101 Fi
refox/101.0"
docker-php | 172.19.0.1 - - [18/May/2022:19:00:26 +0000] "GET /registro.html HTTP/1.1" 404 504 "http://localhost:1000/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:101.0) Gecko/20100101 Fi
refox/101.0"
docker-php | 172.19.0.1 - - [18/May/2022:19:00:29 +0000] "GET / HTTP/1.1" 403 502 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:101.0) Gecko/20100101 Firefox/101.0"
docker-php | [Wed May 18 19:00:29.395629 2022] [autoindex:error] [pid 20] [client 172.19.0.1:54746] AH01276: Cannot serve directory /var/www/html/: No matching DirectoryIndex (index.php,inde
x.html) found, and server-generated directory index forbidden by Options directive
docker-php | [Wed May 18 19:01:21.925245 2022] [autoindex:error] [pid 21] [client 172.19.0.1:54748] AH01276: Cannot serve directory /var/www/html/: No matching DirectoryIndex (index.php,inde
x.html) found, and server-generated directory index forbidden by Options directive
docker-php | 172.19.0.1 - - [18/May/2022:19:01:21 +0000] "GET / HTTP/1.1" 403 503 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:101.0) Gecko/20100101 Firefox/101.0"
```

Aparatado 2: Kubernetes