Despliegue de Aplicaciones WEB

Daniel Pérez, Claudia Trabanco y Aitor Merino

Instalación y configuración de base de datos

1. Actualizamos los paquetes dentro de la maquina de Ubuntu.

```
sudo apt-get update
```

2. Instalamos mysql-server y mysql-client.

```
sudo apt-get install mysql-server mysql-client
```

3. Configuramos mysql_secure_installation.

```
sudo  mysql_secure_installation
```

4. Entramos como admin en mysql.

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

- Comprobamos las bases de datos ya creadas:

```
master@daw-143:~$ sudo mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 17
Server version: 5.7.37-0ubuntu0.18.04.1 (Ubuntu)

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

mysql>
```

5. Creamos la base de datos.

```
create database test_virtual;
```

- Comprobamos que se ha creado:

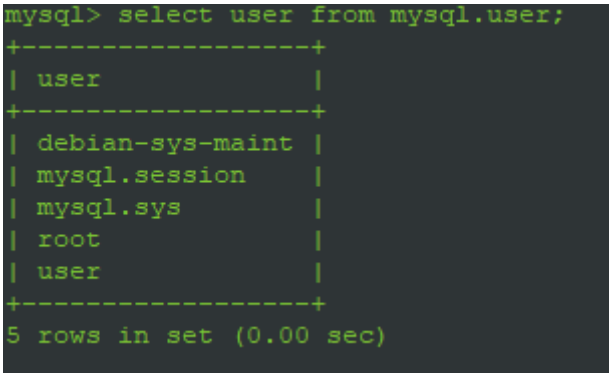
```
mysql> create database test_virtual;
Query OK, 1 row affected (0.00 sec)

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

6. Creamos el usuario al que llamamos `user` y le ponemos la misma contraseña.

```
create user 'user'@'localhost' identified by
'user';
```

- Comprobación de todos los usuarios creados



7. Otorgamos todos los privilegios al usuario `user` en la base de datos `test_virtual`.

```
grant all privileges on test_virtual.* to
'user'@'localhost' with grant option;
```

8. Forzamos al servidor a recargar los privilegios asegurándonos de que dichos cambios se noten sin necesidad de reiniciar.

```
flush privileges;
```

Instalación de recursos necesarios y vsftpd

1. Instalamos maven.

```
sudo apt-get install maven
```

2. Instalamos openjdk8.

```
sudo apt-get install openjdk8
```

- Comprobamos la versión de Java:

```
master@daw-143:~$ java -version
openjdk version "11.0.13" 2021-10-19
OpenJDK Runtime Environment (build 11.0.13+8-Ubuntu-0ubuntu1.18.04)
OpenJDK 64-Bit Server VM (build 11.0.13+8-Ubuntu-0ubuntu1.18.04, mixed mode)
master@daw-143:~$
```

3. Instalamos el vsftpd.

```
sudo apt-get install vsftpd
```

Despliegue del back

1. Descomentamos el atributo `write_enable=YES` del archivo de configuración del vsftpd:

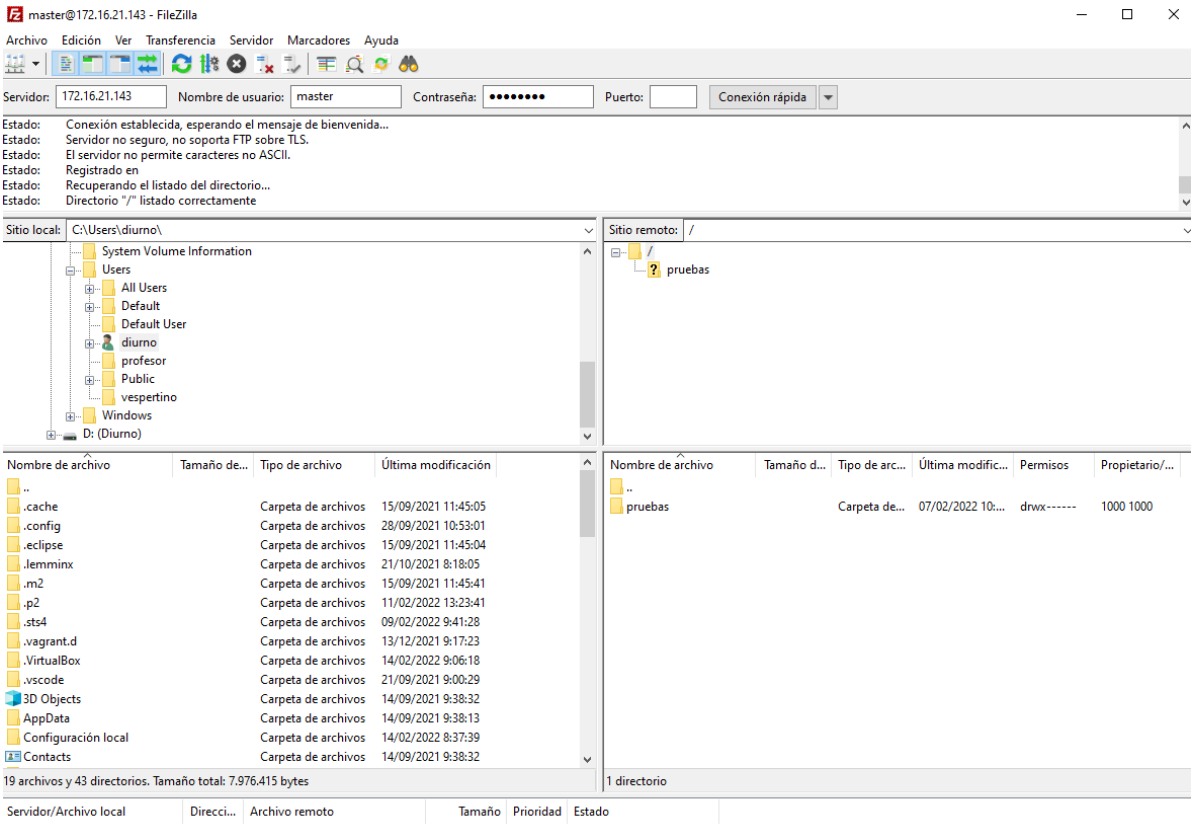
```
# Example config file /etc/vsftpd.conf
#
# The default compiled in settings are fairly paranoid. This sample file
# loosens things up a bit, to make the ftp daemon more usable.
# Please see vsftpd.conf.5 for all compiled in defaults.
#
# READ THIS: This example file is NOT an exhaustive list of vsftpd options.
# Please read the vsftpd.conf.5 manual page to get a full idea of vsftpd's
# capabilities.
#
#
# Run standalone?  vsftpd can run either from an inetd or as a standalone
# daemon started from an initscript.
listen=YES
#
# This directive enables listening on IPv6 sockets. By default, listening
# on the IPv6 "any" address (::) will accept connections from both IPv6
# and IPv4 clients. It is not necessary to listen on *both* IPv4 and IPv6
# sockets. If you want that (perhaps because you want to listen on specific
# addresses) then you must run two copies of vsftpd with two configuration
# files.
listen_ipv6=NO
#
# Allow anonymous FTP? (Disabled by default).
anonymous_enable=YES
#
# Uncomment this to allow local users to log in.
local_enable=YES
#
# Uncomment this to enable any form of FTP write command.
write_enable=YES
```

2. Reiniciamos el vsftpd.service y nos aseguramos de que está `enabled`:

```
systemctl restart vsftpd.service
systemctl is-enabled vsftpd.service
```

```
master@daw-143:~$ systemctl is-enabled vsftpd.service
enabled
master@daw-143:~$
```

3. Entramos en FileZilla usando el usuario master y añadimos el back dentro de Home:



4. Limpiamos el mvn y modificamos el pom.xml del virtualBACK:

```
<finalName>virtual</finalName>
```

```
[INFO] Installing /home/virtualBACK/target/virtual.jar to /root/.m2/repository/com/cavanosa/virtual/0.0.1-SNAPSHOT/vi
rtual-0.0.1-SNAPSHOT.jar
[INFO] Installing /home/virtualBACK/pom.xml to /root/.m2/repository/com/cavanosa/virtual/0.0.1-SNAPSHOT/virtual-0.0.1
-SNAPSHOT.pom
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  42.981 s
[INFO] Finished at: 2022-02-14T09:20:41Z
[INFO] ------------------------------------------------------------------------
```

```
        </dependencies>

        <build>
                <plugins>
                        <plugin>
                                <groupId>org.springframework.boot</groupId>
                                <artifactId>spring-boot-maven-plugin</artifactId>
                                <configuration>
                                        <executable>true</executable>
                                        <finalName>virtual</finalName>
                                </configuration>
                        </plugin>
                </plugins>
        </build>

</project>
```

5. Instalamos el mvn.

```
mvn install
```

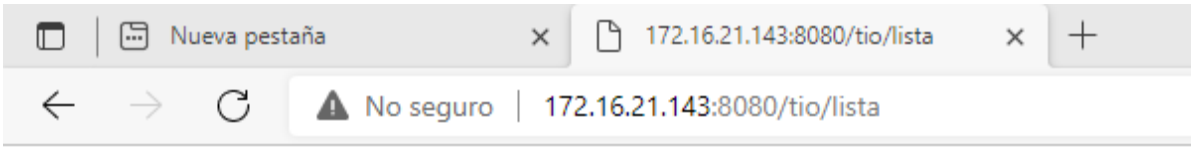
6. Ejecutamos el archivo jar.

```
java -jar target/virtual.jar
```

- Encontramos algunos problemas al iniciar la aplicación Spring Boot ya que el puerto 8080 estaba ocupado.

7. Accedemos a la URL.

```
172.16.21.143:8080/tio/lista
```



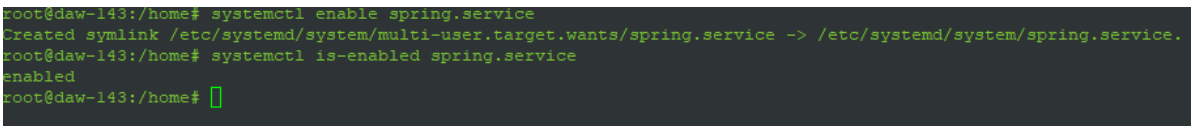
[]

8. Creamos el servicio y cambiamos algunos detalles.

```
nano etc/systemd/system/spring.service
```

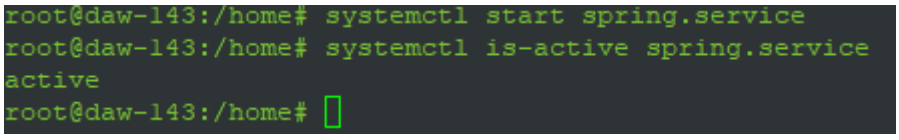
9. Comprobamos si el servicio está funcionando.

```
systemctl enable spring.service
systemctl is-enabled spring.service
```



10. Iniciamos el servicio y lo comprobamos.

```
systemctl start spring.servicio
systemctl is-active spring.service
```



Despliegue del front

1. Instalamos Apache.

```
sudo apt-get install Apache2
```

2. Copiamos index.html a /home.

```
mv /var/www/html/index.html /home/
```

- Comprobamos que Apache se muestra:

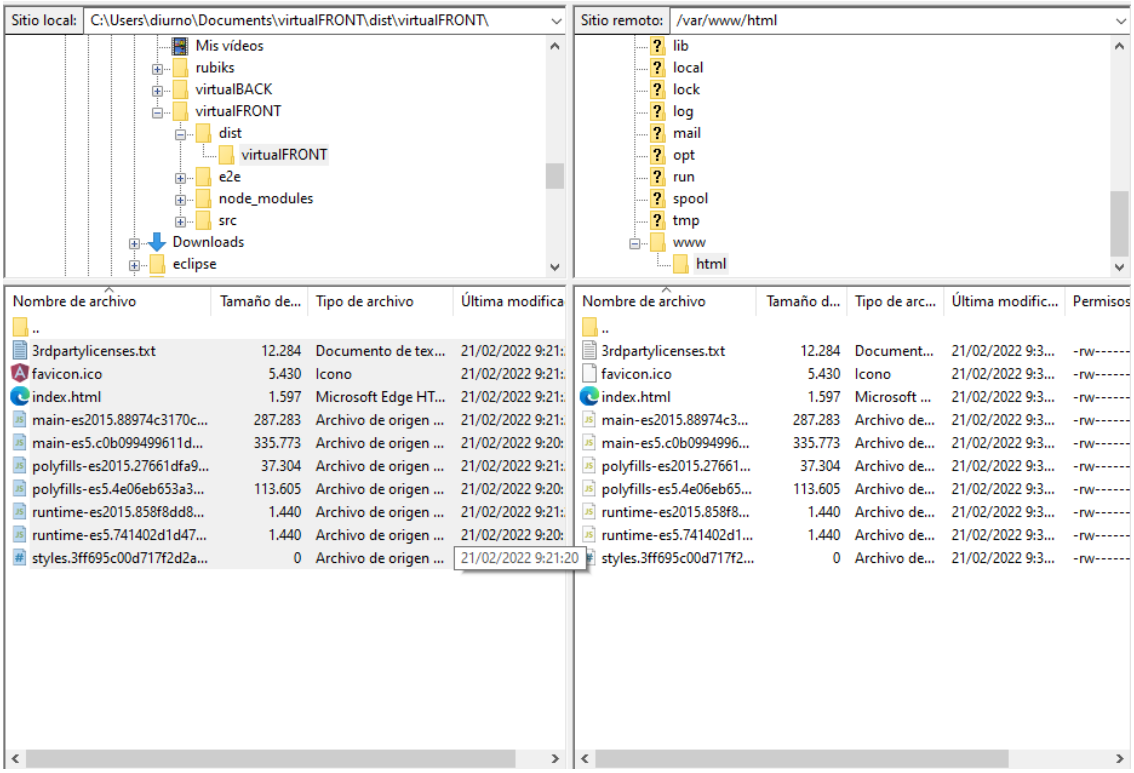
3. Nos bajamos una extensión de Angular para crear el ejecutable tras instalar node js en Visual Studio Code.

```
npm install -g @angular/cli
```

4. Creamos el proyecto del front.

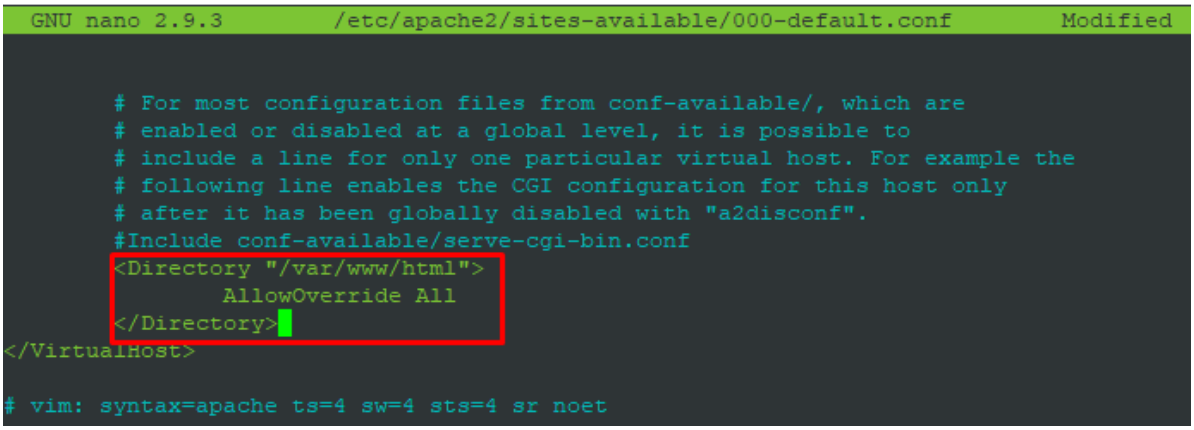
```
ng build --prod
```

5. Subimos los archivos a la carpeta /var/www/html como vemos en la imagen:



6. Cambiamos los permisos para poder acceder al servidor y algunas directivas:

```
chmod -R 755 /var/www/html
```



Resultados de la tarea

- Ya se puede acceder al menú, hacer inserciones, editar y eliminar elementos: