



Trabajo Pila Lamp

Jesús Sánchez Sánchez ASIR2

[Creación de VagrantFile](#)

[Creación de provision Apache](#)

[Instalación manual Apache2](#)

[Creación de provision Mysql](#)

[Instalación manual Mysql](#)

[Comprobación del funcionamiento](#)

Creación de VagrantFile

Para crear un vagrant file primero que nada nos situaremos en el directorio que queramos usar, lo recomendable es crear uno nuevo y trabajar en él. En mi caso crearé y trabajaré en uno llamado `PilaLamp`

Para crearlo ejecutamos el comando `mkdir nombre`

Una vez creado nos entramos dentro de un directorio sino tenemos instalado vagrant lo instalaremos primero, yo lo he instalado en Fedora ya que es el sistema operativo que uso. Para instalarlo ejecutamos los siguientes comandos:

- `sudo dnf install -y dnf-plugins-core`
- `sudo dnf config-manager --add-repo https://rpm.releases.hashicorp.com/fedora/hashicorp.repo`
- `sudo dnf -y install vagrant`

Una vez instalado y situados en la carpeta ejecutamos `vagrant init` para crear el fichero de configuración.

Una vez creado añadiremos las siguientes lineas:

```
Vagrant.configure("2") do |config|

  config.vm.box = "ubuntu/jammy64"
  config.vm.define "JesusSanApache" do |apache|
    apache.vm.hostname = "JesusSanApache"
    apache.vm.network "forwarded_port", guest:80, host: 8080
    apache.vm.network "public_network"
    apache.vm.network "private_network", ip: "172.31.1.10", virtualbox____intnet: "redserver"
    apache.vm.provision "shell", path: "provapache.sh"

  end
```

```

config.vm.define "JesusSanMysql" do |sql|
  sql.vm.hostname = "JesusSanMysql"
  sql.vm.network "public_network"
  sql.vm.network "private_network", ip: "172.31.1.11", virtualbox____intnet: "redserver"
  sql.vm.provision "shell", path: "provmysql.sh"
end

```

Ahora ejecutamos `vagrant up` para comprobar que todo funciona.

Una vez que hemos comprobado que funciona vamos a realizar el provisión para automatizar todo.

Creación de provision Apache

```

#!/bin/bash
# Actualiza la lista de paquetes disponibles
sudo apt update
# Instala el servidor web Apache
sudo apt install apache2 -y
# Instala PHP y el módulo para Apache para interpretar PHP
sudo apt install php libapache2-mod-php php-mysql -y
# Reinicia el servicio Apache para aplicar los cambios
sudo systemctl restart apache2
# Cambia al directorio raíz de los archivos web
cd /var/www/html
# Crea un nuevo directorio para la aplicación GestAlumnos
sudo mkdir GestAlumnos
# Entra en el directorio GestAlumnos
cd GestAlumnos
# Clona el repositorio de la práctica LAMP en el directorio actual
sudo git clone https://github.com/josejuansanchez/iaw-practica-lamp.git
# Mueve todos los archivos del repositorio clonado al directorio actual
sudo mv iaw-practica-lamp/* .
# Elimina el directorio del repositorio clonado ya que ya no es necesario
sudo rm -r iaw-practica-lamp/
# Elimina el directorio 'db' que no es necesario para el despliegue
sudo rm -r db/
# Elimina cualquier archivo README que no sea necesario
sudo rm README*
# Elimina la página de inicio predeterminada de Apache
sudo rm /var/www/html/index.html
# Deshabilita el sitio por defecto de Apache
sudo a2dissite 000-default.conf
# Copia la configuración del sitio por defecto para crear una nueva configuración para GestAlumnos
sudo cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-available/gest alumnos.conf
# Sobrescribe el archivo de configuración del nuevo sitio con la configuración específica para el sitio
# Se usa EOL pero se podría utilizar cualquiera palabra que tenga un principio y luego al final
cat <<EOL > /etc/apache2/sites-available/gest alumnos.conf
<VirtualHost *:80>
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html/GestAlumnos/src

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combine
</VirtualHost>
EOL
# Crea el archivo de configuración de la base de datos para la aplicación
cat <<EOL > /var/www/html/GestAlumnos/src/config.php

```

```
<?php

define('DB_HOST', '172.31.1.11');
define('DB_NAME', 'lamp_db');
define('DB_USER', 'lamp_user');
define('DB_PASSWORD', '1234-Lamp');

\mysqli = mysqli_connect(DB_HOST, DB_USER, DB_PASSWORD, DB_NAME);

?>
EOL
# Opcional: Establecer permisos para los archivos y cambiar el propietario al usuario de Apache
#sudo chmod -R 755 /var/www/GestAlumnos/
#sudo chown -R www-data:www-data /var/www/GestAlumnos/
# Habilita el sitio GestAlumnos en Apache
sudo a2ensite gestalumnos.conf
# Reinicia Apache para aplicar la nueva configuración
sudo systemctl restart apache2
```

Instalación manual Apache2

El anterior script para el provisión nos permite levantar la maquina totalmente operativa y funcional, sin embargo es importante saber que hace y porque lo estamos haciendo. Por lo tanto voy a explicar como sería una configuración manual.

Una vez que hemos hecho el `vagrant up` nos conectamos a la maquina por ssh, vagrant permite hacerlo de manera muy sencilla. Simplemente ejecutamos el comando → `vagrant ssh nombre` en nombre pondremos el nombre que le hemos puesto a la maquina de apache, en mi caso se llama JesusSanApache, por lo tanto quedaría así `vagrant ssh JesusSanApache`.

Una vez que estamos conectados a la maquina lo primero que haremos será actualizar los paquetes y librerías con `sudo apt update`.

Una vez actualizado instalaremos apache usando el comando `sudo apt install apache2 -y`

Después de tener apache2 instalado vamos instalar php y los módulos correspondientes para conectar apache2 y mysql con php. Esto lo haremos ejecutando el comando `sudo apt install php libapache2-mod-php php-mysql -y`

Una vez instalado nos iremos con `cd` a la ruta `/var/www/html` la cual es la ruta por defecto donde estarán alojados el código y las configuraciones de la página. Ahora vamos a crear un directorio que será donde vamos a instalar nuestra pagina, en mi caso la he llamado `GestAlumnos`

Nos entramos en el directorio con `cd` y descargamos con `git clone` el siguiente enlace:

<https://github.com/josejuansanchez/iaw-practica-lamp.git>

Ahora moveremos todo lo que hay dentro al directorio actual y eliminaremos lo que no necesitamos con:

- `sudo mv iaw-practica-lamp/* .`
- `sudo rm -r iaw-practica-lamp/`
- `sudo rm -r db/`
- `sudo rm README*`
- `sudo rm /var/www/html/index.html`

Una vez que hemos eliminado lo que no necesitamos nos vamos a ir a ir al directorio de configuración de apache que esta en la ruta `/etc/apache2` y dentro de aquí nos iremos a `sites-available`, dentro de este directorio lo que encontramos son los sitios que están disponibles pero no habilitados por lo que podemos aprovechar eso y copiar el sitio por defecto para crear nosotros uno nuevo.

Copiamos el fichero de configuración y lo renombramos con `sudo cp 000-default.conf gestalumnos`

Ahora lo editaremos con nano, cuando lo abrimos nos encontramos algo como esto:

```
<VirtualHost *:80>
    ServerAdmin webmaster@localhost
```

```
DocumentRoot /var/www/html

ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combine
</VirtualHost>
```

Veremos que la ruta que viene por defecto está apuntado a html y la nuestra tendrá que apuntar al directorio que hemos creado y al directorio donde se encuentra todo que sería `/GestAlumnos/src`

Por ello el fichero editado quedaría así:

```
<VirtualHost *:80>
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html/GestAlumnos/src

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combine
</VirtualHost>
```

Ahora en el directorio de src concretamente en `/var/www/html/GestAlumnos/src`, encontraremos un fichero llamado `config.php`, dentro de este definiremos el host de la base de datos(IP del servidor mysql), el nombre, el usuario y su contraseña. En mi caso el fichero quedará así:

```
<?php

define('DB_HOST', '172.31.1.11');
define('DB_NAME', 'lamp_db');
define('DB_USER', 'lamp_user');
define('DB_PASSWORD', '1234-Lamp');

'$mysqli' = mysqli_connect(DB_HOST, DB_USER, DB_PASSWORD, DB_NAME);

?>
```

—Punto importante— Si lo vamos añadir para el provision importante que en `'$mysqli'` habría que añadir una barra invertida, así: `\'$mysqli'` para que a la hora de ejecutarlo el provision no lo tome como una variable ya que sino no lo pondrá en el fichero.

Guardamos el fichero y habilitamos el sitio con → `sudo a2ensite gestalumnos.conf`

Ahora reiniciaremos apache2 para que se ejecuten las nuevas modificaciones:

```
sudo systemctl restart apache2
```

Con esto ya habríamos acabado la parte de apache2.

Ahora pasaremos a configurar Mysql.

Creación de provision Mysql

```
#!/bin/bash
# Actualiza la lista de paquetes disponibles
sudo apt update
# Instala el servidor MySQL
sudo apt install mysql-server -y
# Clona el repositorio de la práctica LAMP
sudo git clone https://github.com/josejuansanchez/iaw-practica-lamp.git
# Mueve los archivos del repositorio al directorio actual
sudo mv iaw-practica-lamp/* .
# Elimina el directorio clonado ya que los archivos se han movido
sudo rm -r iaw-practica-lamp/
```



```
# Elimina el directorio 'src/' (si existe)
sudo rm -r src/
# Elimina los archivos README* (si existen)
sudo rm README*
# Cambia al directorio 'db/' que contiene el archivo de la base de datos
cd db/
# Cambia al usuario root para ejecutar los comandos MySQL
sudo su
# Carga la base de datos desde el archivo SQL usando MySQL
mysql -u root < database.sql
# Crea un usuario para la base de datos y le otorga privilegios
mysql -u root <<EOF
CREATE USER IF NOT EXISTS 'lamp_user'@'172.31.1.10' IDENTIFIED BY '1234-Lamp';
GRANT ALL PRIVILEGES ON lamp_db.* TO 'lamp_user'@'172.31.1.10';
FLUSH PRIVILEGES;
EOF
# Cambia la configuración de MySQL para permitir conexiones desde una IP específica
sed -i 's/^bind-address.*/bind-address = 172.31.1.11/' /etc/mysql/mysql.conf.d/mysqld.cnf
# Reinicia el servicio MySQL para aplicar los cambios de configuración
sudo systemctl restart mysql
# Elimina la ruta por defecto del sistema
sudo ip route del default
```

Instalación manual Mysql

Primero que nada nos conectaremos por ssh a la maquina, para ello como hemos hecho con el servidor de apache2 haremos `vagrant ssh` y el nombre que en mi caso es `JesusSanMysql`

Una vez estamos conectados a la maquina haremos un `sudo apt update` para actualizar las librerías.

Ahora instalaremos mysql-server con `sudo apt install mysql-server -y`

Una vez instalado pasaremos a descargar los repositorios como en el apache:

```
sudo git clone https://github.com/josejuansanchez/iaw-practica-lamp.git
```

Ahora borraremos todo menos la base de datos con:

- `sudo rm -r iaw-practica-lamp/`
- `sudo rm -r src/`
- `sudo rm README*`

Ahora nos entraremos en el directorio llamado `db` y allí encontraremos un fichero de nombre `database.sql` cuyo contenido es el siguiente:

```
DROP DATABASE IF EXISTS lamp_db;
CREATE DATABASE lamp_db CHARSET utf8mb4;
USE lamp_db;

-- Create the users table
CREATE TABLE users (
  id INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,
  name VARCHAR(100) NOT NULL,
  age INT UNSIGNED NOT NULL,
  email VARCHAR(100) UNIQUE NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Aquí encontraremos un script de creación de la base de datos, para poder crear usando la base de datos usando el .sql, elevamos privilegios como root con `sudo su` y ejecutamos el siguiente comando: `mysql -u root < database.sql`

Ahora para asegurarnos de que funciona nos entraremos a la base de datos como root con `mysql -u root`. Una vez que estamos dentro de mysql podemos comprobar que la base de datos está haciendo `show databases;`, si el script se ejecutó

bien ahora deberíamos de tener una base de datos llamada lamp_db.

Ahora vamos a crear un usuario y a darles privilegios, este usuario es el que usaremos para poder conectar la base de datos y apache, que es el que hemos añadido en el `config.php` anteriormente.

Para ello primero creamos el usuario con:

```
CREATE USER IF NOT EXISTS 'lamp_user'@'172.31.1.10' IDENTIFIED BY '1234-Lamp';
```

Si nos fijamos la ip que le ponemos es la ip del servidor de apache y no la del servidor de mysql, mucho cuidado con eso.

Ahora le daremos los permisos necesarios con:

```
GRANT ALL PRIVILEGES ON lamp_db.* TO 'lamp_user'@'172.31.1.10';
```

Y aplicamos los permisos con:

```
FLUSH PRIVILEGES;
```

Ahora tendremos que editar en el fichero `mysqld.cnf` la ip del `bind-address`, aquí le tendremos que indicar la ip del servidor mysql, en mi caso es `172.31.1.11`.

La ruta donde esta este fichero es `/etc/mysql/mysql.conf.d/mysqld.cnf` mucho ojo ya que hay dos archivos uno acabado en d y otro no, tenemos que editar el que acaba en d.

Ahora lo ultimo que quedaría es reiniciar mysql con `sudo systemctl restart mysql` y para una mejor seguridad le quitaremos la puerta de enlace de la nat para que no tenga salida a internet, esto lo haremos con `sudo ip route del default`

Comprobación del funcionamiento

```
rv25@fedora:~/Asir/WEB/prueba3apachesql$ ls
provapache.sh  provmysql.sh  Vagrantfile
rv25@fedora:~/Asir/WEB/prueba3apachesql$ mkdir ../pruebaPilaLamp
rv25@fedora:~/Asir/WEB/prueba3apachesql$ cp * ../pruebaPilaLamp/
rv25@fedora:~/Asir/WEB/prueba3apachesql$ cd ../pruebaPilaLamp/
rv25@fedora:~/Asir/WEB/pruebaPilaLamp$ ls
provapache.sh  provmysql.sh  Vagrantfile
rv25@fedora:~/Asir/WEB/pruebaPilaLamp$
```

```
rv25@fedora:~/Asir/WEB/pruebaPilaLamp$ bat provapache.sh

File: provapache.sh

1  #!/bin/bash
2  sudo apt update
3  sudo apt install apache2 -y
4  sudo apt install php libapache2-mod-php php-mysql -y
5  sudo systemctl restart apache2
6  cd /var/www/html
7  sudo mkdir GestAlumnos
8  cd GestAlumnos
9  sudo git clone https://github.com/josejuansanchez/iaw-practica-lamp.git
10 sudo mv iaw-practica-lamp/* .
11 sudo rm -r iaw-practica-lamp/
12 sudo rm -r db/
13 sudo rm README*
14 sudo rm /var/www/html/index.html
15 sudo a2disssite 000-default.conf
16 sudo cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-available/gestalumnos.conf
17 cat <<EOL > /etc/apache2/sites-available/gestalumnos.conf
18 <VirtualHost *:80>
19     ServerAdmin webmaster@localhost
20     DocumentRoot /var/www/html/GestAlumnos/src
21
22     ErrorLog ${APACHE_LOG_DIR}/error.log
23     CustomLog ${APACHE_LOG_DIR}/access.log combine
24 </VirtualHost>
25 EOL
26 cat <<EOL > /var/www/html/GestAlumnos/src/config.php
27 <?php
28
29 define('DB_HOST', '172.31.1.11');
30 define('DB_NAME', 'lamp_db');
31 define('DB_USER', 'lamp_user');
32 define('DB_PASSWORD', '1234-Lamp');
33
34 \ $mysqli = mysqli_connect(DB_HOST, DB_USER, DB_PASSWORD, DB_NAME);
35
36 ?>
37 EOL
38 #sudo chmod -R 755 /var/www/html/GestAlumnos/
39 #sudo chown -R www-data:www-data /var/www/html/GestAlumnos/
40 sudo a2ensite gestalumnos.conf
41 sudo systemctl restart apache2
```

```
rv25@fedora:~/Asir/WEB/pruebaPilaLamp$ bat provmysql.sh

File: provmysql.sh

1  #!/bin/bash
2  sudo apt update
3  sudo apt install mysql-server -y
4  sudo git clone https://github.com/josejuansanchez/iaw-practica-lamp.git
5  sudo mv iaw-practica-lamp/* .
6  sudo rm -r iaw-practica-lamp/
7  sudo rm -r src/
8  sudo rm README*
9  cd db/
10 sudo su
11 mysql -u root < database.sql
12 mysql -u root <<EOF
13 CREATE USER IF NOT EXISTS 'lamp_user'@'172.31.1.10' IDENTIFIED BY '1234-Lamp';
14 GRANT ALL PRIVILEGES ON lamp_db.* TO 'lamp_user'@'172.31.1.10';
15 FLUSH PRIVILEGES;
16 EOF
17 sed -i 's/^bind-address.*/bind-address = 172.31.1.11/' /etc/mysql/mysql.conf.d/mysqld.cnf
18 sudo systemctl restart mysql
19 sudo ip route del default
```

```
File: Vagrantfile

1 # -*- mode: ruby -*-
2 # vi: set ft=ruby :
3
4 # All Vagrant configuration is done below. The "2" in Vagrant.configure
5 # configures the configuration version (we support older styles for
6 # backwards compatibility). Please don't change it unless you know what
7 # you're doing.
8 Vagrant.configure("2") do |config|
9   # The most common configuration options are documented and commented below.
10   # For a complete reference, please see the online documentation at
11   # https://docs.vagrantup.com.
12
13   # Every Vagrant development environment requires a box. You can search for
14   # boxes at https://vagrantcloud.com/search.
15   config.vm.box = "ubuntu/jammy64"
16
17   config.vm.define "JesusSanApache" do |apache|
18     apache.vm.hostname = "JesusSanApache"
19     apache.vm.network "forwarded_port", guest:80, host: 8080
20     #apache.vm.network "forwarded_port", guest:443, host: 443
21     apache.vm.network "public_network"
22     apache.vm.network "private_network", ip: "172.31.1.10", virtualbox____intnet: "redserver"
23     apache.vm.provision "shell", path: "provapache.sh"
24   end
25
26   config.vm.define "JesusSanMysql" do |sql|
27     sql.vm.hostname = "JesusSanMysql"
28     sql.vm.network "public_network"
29     sql.vm.network "private_network", ip: "172.31.1.11", virtualbox____intnet: "redserver"
30     sql.vm.provision "shell", path: "provmysql.sh"
31   end
32 end
```

Homepage

127.0.0.1:8080

User Communities Red Hat Free Content Albarregas Moodle Nodo313 - Más que un... Skills for All with Cisco...

Simple LAMP web app

Demo app

Add New Data

Name	Age	Email	Update
------	-----	-------	--------

vagrant@JesusSanApache:~\$ sudo systemctl status apache2

● apache2.service - The Apache HTTP Server

Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)

Active: active (running) since Sat 2024-11-02 22:53:15 UTC; 14min ago

Docs: https://httpd.apache.org/docs/2.4/

Process: 10660 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)

Main PID: 10664 (apache2)

Tasks: 7 (limit: 1102)

Memory: 12.4M

CPU: 81ms

CGroup: /system.slice/apache2.service

└─10664 /usr/sbin/apache2 -k start

└─10665 /usr/sbin/apache2 -k start

└─10666 /usr/sbin/apache2 -k start

└─10667 /usr/sbin/apache2 -k start

└─10668 /usr/sbin/apache2 -k start

└─10669 /usr/sbin/apache2 -k start

└─10670 /usr/sbin/apache2 -k start

Nov 02 22:53:15 JesusSanApache systemd[1]: Starting The Apache HTTP Server...

Nov 02 22:53:15 JesusSanApache apachectl[10663]: AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.0.1 instead. See the error log for more details.

Nov 02 22:53:15 JesusSanApache systemd[1]: Started The Apache HTTP Server.

lines 1-21/21 (END)

vagrant@JesusSanMysql:~\$ sudo systemctl status mysql

● mysql.service - MySQL Community Server

Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: enabled)

Active: active (running) since Sat 2024-11-02 22:54:41 UTC; 13min ago

Process: 3280 ExecStartPre=/usr/share/mysql/mysql-systemd-start pre (code=exited, status=0/SUCCESS)

Main PID: 3288 (mysqld)

Status: "Server is operational"

Tasks: 38 (limit: 1102)

Memory: 374.9M

CPU: 2.449s

CGroup: /system.slice/mysql.service

└─3288 /usr/sbin/mysqld

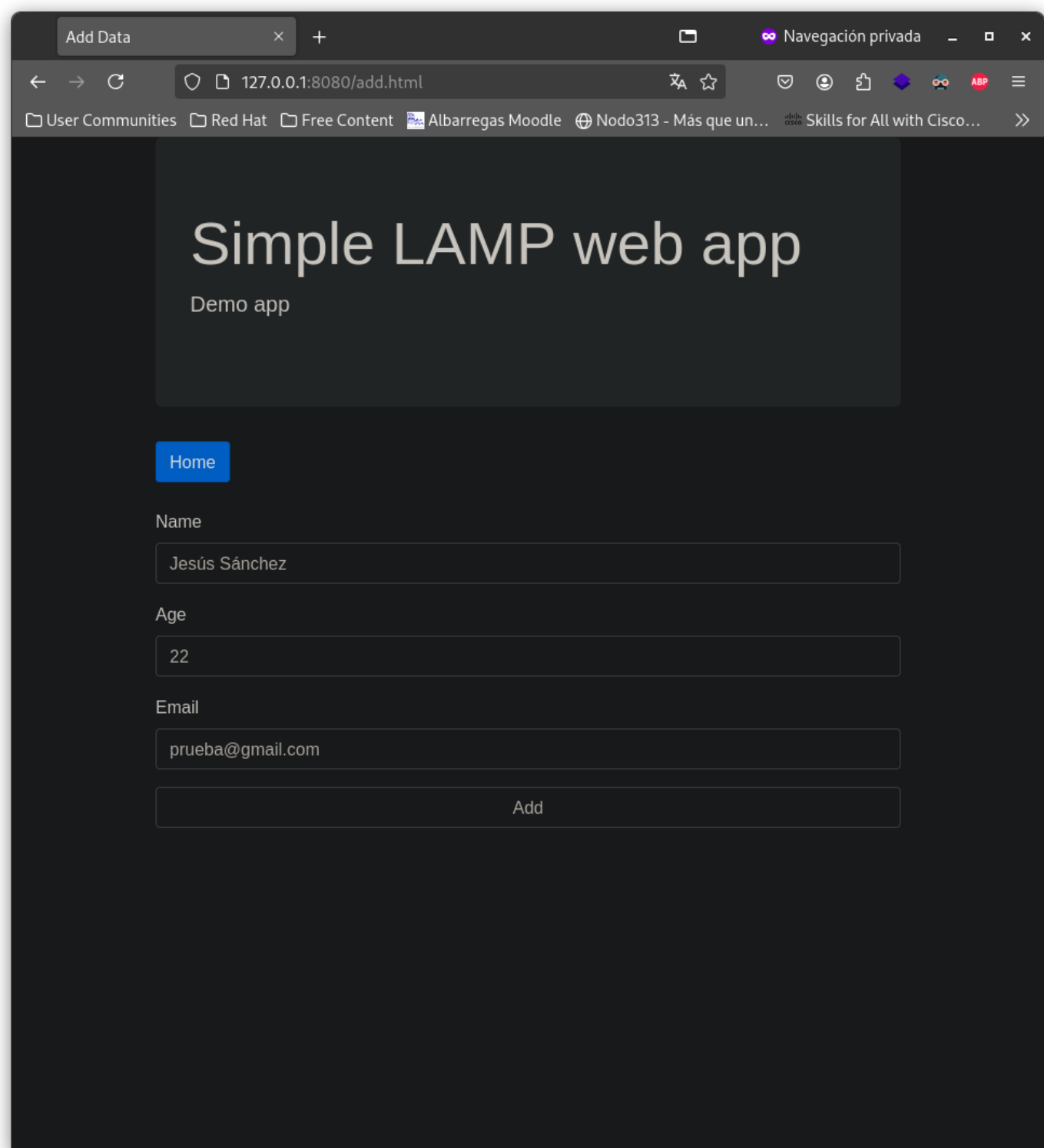
Nov 02 22:54:40 JesusSanMysql systemd[1]: Starting MySQL Community Server...

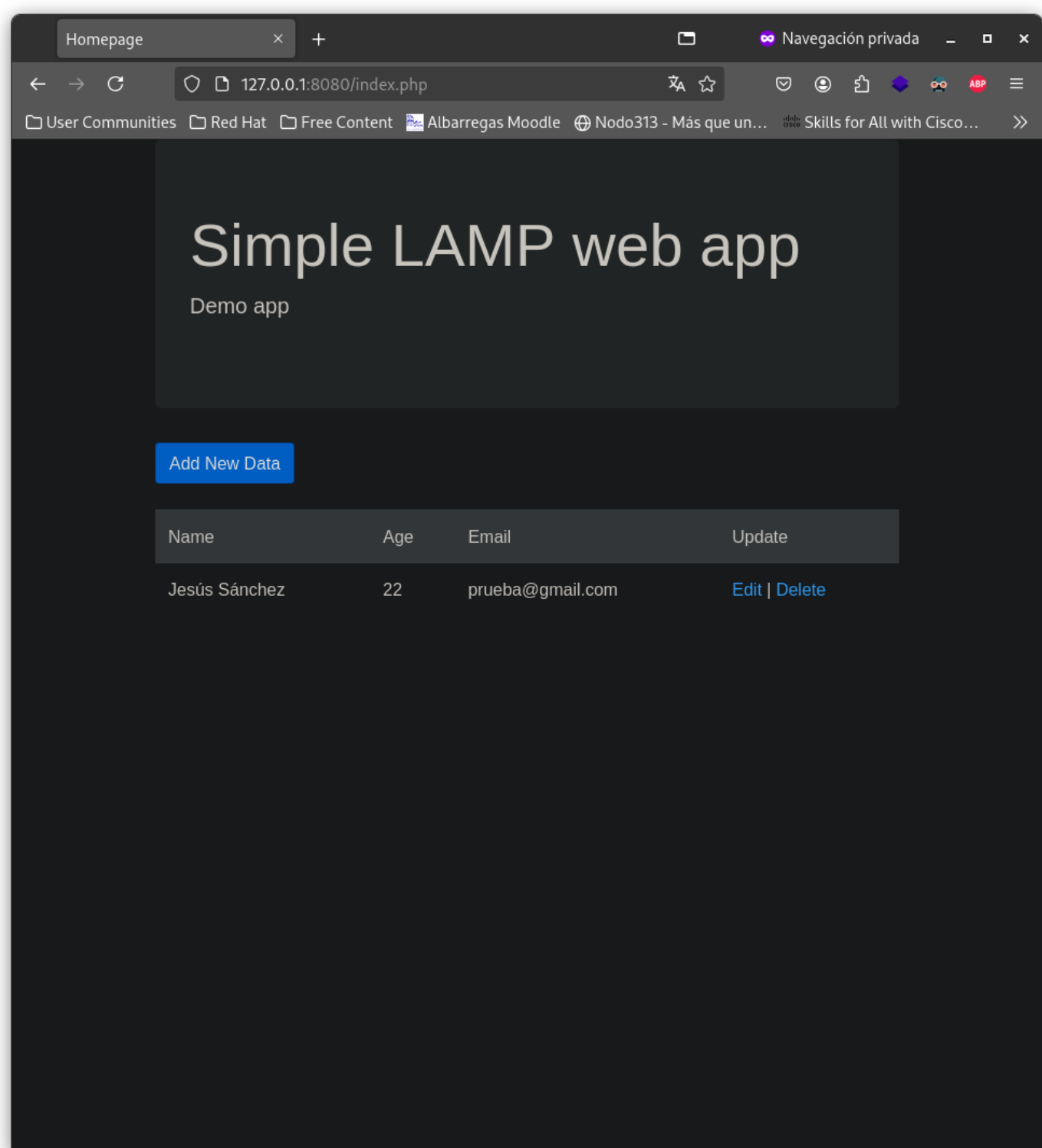
Nov 02 22:54:41 JesusSanMysql systemd[1]: Started MySQL Community Server.

vagrant@JesusSanMysql:~\$

Trabajo Pila Lamp

8





```
mysql> select * from users;
+----+-----+-----+-----+
| id | name          | age | email                |
+----+-----+-----+-----+
|  1 | Jesús Sánchez |  22 | prueba@gmail.com    |
+----+-----+-----+-----+
1 row in set (0.00 sec)
```