

# MANUAL INSTALACIÓN PROYECTO PR101 - Fase 1



**DAW-IT00168-00-Proyecto PR101 -  
Fase 1**

**Autor: Andrés Cortés Escobedo**

# INDICE

<b>Requisitos del sistema .....</b>	<b>1</b>
<b>Servidor usado para estar en la nube.....</b>	<b>1</b>
<b>Scripts que hay que ejecutar .....</b>	<b>9</b>
<b>Instalar Apache, Php 7.4 y mysql.....</b>	<b>12</b>
<b>PHP 8.0 en apache y PHP 7.4 en la línea de comandos por defecto. ....</b>	<b>13</b>
<b>INSTALAR PHPMYADMIN .....</b>	<b>14</b>
<b>Clave pública de nuestro usuario jdaw, para que se conecte a nuestro servidor.....</b>	<b>15</b>
<b>Referencias.....</b>	<b>18</b>

## Requisitos del sistema

No necesitamos ningún requisito del sistema, solamente un dispositivo que tenga internet y se pueda conectar a la página web.

## Servidor usado para estar en la nube


Lo que he usado para instalar la aplicación web es un servidor de '**Amazon Web Services**', con una cuenta de 'aws educate' <https://www.awseducate.com/student/s/>, seleccionamos la Ubuntu server 20.04 LTS, que es de 64 bits.


(esta parte corresponde mas bien a la parte del desarrollador/programador parte privada)


Tras iniciar sesión, en mi caso voy a la classroom que tengo y hacemos clic en 'Go to classroom'

Course Name ⓘ	Description	Educator ⓘ	Course End Date ⓘ	Credit Allocated Per Student ⓘ	Status
DAW-primero	Primero de DAW	José Carlos Álvarez González	06/30/2021	\$50	Accepted <a href="#">Go to classroom</a> ↗

## Your AWS Account Status

**Active**  
full access ( )

**\$48.29**  
remaining credits (estimated)

**2:55**  
session time

[Account Details](#) [AWS Console](#)

Please use AWS Educate Account responsibly. Remember to shut down your instances when make the best use of your credits. And, don't forget to logout once you are done with your w

Le damos a 'AWS Console '

## MANUAL USUARIO PROYECTO PR101 FASE 1 DAW-IT00168-00-PROYECTO PR101 - FASE 1


Hacemos clic en ‘Ejecute una máquina virtual’ y Seleccionamos esta máquina.

Paso 1: Elegir una imagen de Amazon Machine (AMI)

**¿Utiliza S3 para el almacenamiento?**  
Amazon S3 está diseñado para ofrecer una durabilidad de los datos del 99,999999999% (11 nueves), lo que significa que los datos están disponibles cuando son necesarios y están protegidos contra errores, errores y amenazas.

Pruébalo

Ocultar

**SUSE Linux Enterprise Server 15 SP2 (HVM), SSD Volume Type** - ami-0fde50fcbcd46f2f7 (64 bits x86) / ami-05f2f5f76d89313bb (64 bits Arm)


Apto para la capa

Seleccionar

SUSE Linux Enterprise Server 15 Service Pack 2 (HVM), EBS General Purpose (SSD) Volume Type. Amazon EC2 AMI Tools preinstalled: Apache 2.2, MySQL 5.5, PHP 5.3, and Ruby 1.8.7 available.

64 bits (x86)

64 bits (ARM)

**Ubuntu Server 20.04 LTS (HVM), SSD Volume Type** - ami-042e8287309f5df03 (64 bits x86) / ami-0b75998a97c952252 (64 bits Arm)

Apto para la capa

Seleccionar

Ubuntu Server 20.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (http://www.ubuntu.com/cloud/services).

64 bits (x86)

64 bits (ARM)

Ahora toca seleccionar el tipo de instancia, una instancia en este caso es una maquina virtual alojada en este caso en AWS, contiene aplicaciones instaladas predeterminadas y configuraciones del sistema operativo.

1. Elige AMI2. Elige tipo de instancia3. Configurar la instancia4. Añadir el almacenamiento5. Agregar etiquetas6. Página Config Security Group7. Revisar

**Paso 2: Página Choose an Instance Type**  
Amazon EC2 proporciona una amplia selección de tipos de instancias optimizadas para adaptarse a diferentes casos de uso. Las instancias son servidores virtuales que pueden ejecutar aplicaciones. Tienen distintas combinaciones de CPU, memoria, almacenamiento y capacidad de red, lo que proporciona una gran flexibilidad para elegir la combinación de recursos adecuada para las aplicaciones. [Más información](#) acerca de los tipos de instancias y cómo pueden satisfacer sus necesidades de computación.

Filtrar por: Todas las familias de instanciasGeneración actualMostrar/ocultar columnas

Seleccionada actualmente: t2.micro (1 vCPU, 2.5 GiB RAM, 1 GiB memoria SSD solo)

	Familia	Tipo	vCPU	Memoria (GiB)	Almacenamiento de la instancia (GiB)	Optimizado para EBS disponible	Uso compartido de la red	Compatibilidad con IPv6
<input type="checkbox"/>	t2	t2.nano	1	0.5	EBS solo	-	De bajo a moderado	SI
<input checked="" type="checkbox"/>	t2	t2.micro	1	1	EBS solo	-	De bajo a moderado	SI
<input type="checkbox"/>	t2	t2.small	1	2	EBS solo	-	De bajo a moderado	SI
<input type="checkbox"/>	t2	t2.medium	2	4	EBS solo	-	De bajo a moderado	SI
<input type="checkbox"/>	t2	t2.large	2	8	EBS solo	-	De bajo a moderado	SI
<input type="checkbox"/>	t2	t2.xlarge	4	16	EBS solo	-	Moderada	SI
<input type="checkbox"/>	t2	t2.2xlarge	8	32	EBS solo	-	Moderada	SI
<input type="checkbox"/>	t3	t3.nano	2	0.5	EBS solo	SI	Hasta 5 gbps	SI
<input type="checkbox"/>	t3	t3.micro	2	1	EBS solo	SI	Hasta 5 gbps	SI
<input type="checkbox"/>	t3	t3.small	2	2	EBS solo	SI	Hasta 5 gbps	SI
<input type="checkbox"/>	t3	t3.medium	2	4	EBS solo	SI	Hasta 5 gbps	SI
<input type="checkbox"/>	t3	t3.large	2	8	EBS solo	SI	Hasta 5 gbps	SI

Cancelar

Anterior

Revisar y lanzar

Siguiente: Página Configuración de los detalles de la instancia

Seleccionamos esta instancia.

2

## MANUAL USUARIO PROYECTO PR101 FASE 1 DAW-IT00168-00-PROYECTO PR101 - FASE 1

**Paso 7: Página Review Instance Launch**  
Revise los detalles de lanzamiento de su instancia. Recuerda para editar los cambios de cada sección, haga clic en **Lanzar** para asignar un par de claves a la instancia y completar el proceso de lanzamiento.

**Mejore la seguridad de su instancia.** Su grupo de seguridad, **launch-wizard-1**, está abierto a todo el mundo.  
Su instancia puede estar accesible desde cualquier dirección IP. Le recomendamos que actualice las reglas de su grupo de seguridad para permitir el acceso únicamente desde direcciones IP conocidas.  
También puede abrir puertos adicionales en su grupo de seguridad para facilitar el acceso a la aplicación o el servicio que está ejecutando, por ejemplo, HTTP (80) para los servidores web. [Editar grupos de seguridad](#)

**Detalles de la AMI** [Editar AMI](#)

**Ubuntu Server 20.04 LTS (HVM), SSD Volume Type - ami-042e82873095d4f03**  
Ubuntu Server 20.04 LTS (HVM), 64-bit, General Purpose (M5) Instance Type, Support available from Canonical (<http://www.ubuntu.com/cloud/instances>)  
Tipo de dispositivo raíz: ebs Tipo de almacenamiento: hvm

**Tipo de instancia** [Editar tipo de instancia](#)

Tipo de instancia	ECU	vCPU	Memoria (GiB)	Almacenamiento de la instancia (GiB)	Optimizado para EBS disponible	Desempeño de la red
t2.micro	-	1	1	EBS solo	-	Low to Moderate

**Grupos de seguridad** [Editar grupos de seguridad](#)

Nombre del grupo de seguridad	Descripción
launch-wizard-1	launch-wizard-1 created 2021-05-03T12:58:41.794+01:00

Tipo (1)	Protocolo (1)	Rango de puertos (1)	Origen (1)	Descripción (1)
SSH	TCP	22	0.0.0.0/0	

**Detalles de la instancia** [Editar detalles de la instancia](#)

Número de instancias: 1 Opción de compra: **On-Demand**

Red: **vpc-1b0c4f5d**

[Cancelar](#) [Anterior](#) [Lanzar](#)

La lanzamos.

Con ello nos pedirá si queremos un par de claves de rsa, para que nos de la clave pública y con ello conseguir conectarnos a nuestra instancia.

**Seleccione un par de claves existente o cree un nuevo par de claves**

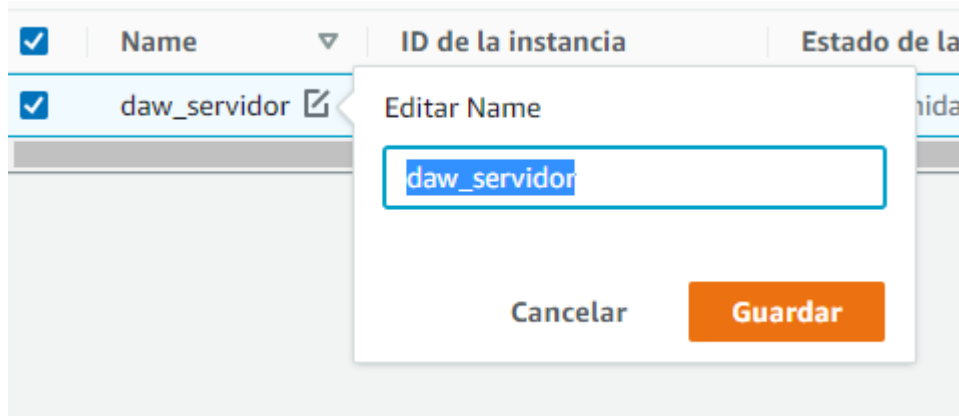
Un par de claves consta de una **clave pública** que AWS almacena y un **archivo de claves privadas** que usted almacena. Juntos, le permiten conectarse a su instancia de forma segura. Para las AMI de Windows, el archivo de claves privadas es necesario para obtener la contraseña usada para iniciar sesión en la instancia. Para las AMI de Linux, el archivo de claves privadas le permite realizar una conexión SSH segura con su instancia.

Nota: El par de claves seleccionado se añadirá al conjunto de claves autorizadas para esta instancia. Obtenga más información sobre [cómo eliminar pares de claves existentes de una AMI pública](#).

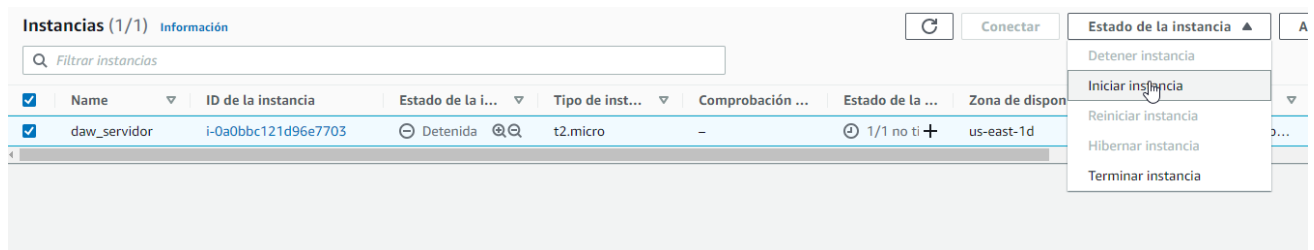
Tiene que descargar el **archivo de claves privadas** (archivo \*.pem) para poder continuar. **Guárdelo en un lugar seguro y accesible.** No podrá descargar el archivo de nuevo después de crearlo.

[Cancelar](#) [Lanzar instancias](#)

Con esto ya si podemos lanzar nuestra instancia, después de esto hacemos clic en ver instancias.

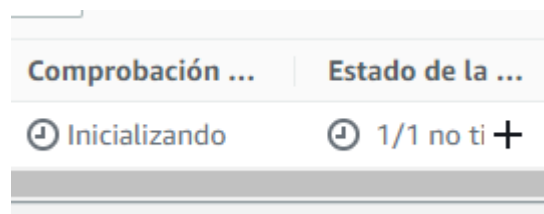


Nos saldrá nuestro panel y le damos a instancias, después ello veremos las instancias que tenemos en este caso solamente tenemos una, ponemos el ratón encima del nombre y se lo cambiamos para poder identificarla.



La seleccionamos y en estado de la instancia le damos a iniciar instancia esperamos unos minutos actualizamos página y ya veremos que esta iniciada.

Antes de que se inicie hará las comprobaciones necesarias y cuando este 2/2 significa que ha hecho las comprobaciones y en el estado veremos iniciada.



Ahora mismo esta en estado de inicialización aún tenemos que esperar.

<input type="checkbox"/>	Name ▾	ID de la instancia	Estado de la i... ▾	Tipo de inst... ▾	Comprobación ...	Estado de la ...	Zona de dispon... ▾	DNS de IPv4 pública ▾
<input type="checkbox"/>	daw_servidor	i-0a0bbc121d96e7703	En ejecución	t2.micro	2/2 comprobaci...	1/1 no ti +	us-east-1d	ec2-3-223-61-50.comp...

Como vemos las comprobaciones ya están hechas, y el estado de la instancia esta ya en ejecución, podemos ya entrar en nuestro servidor virtual.

**IMPORTANTE : Cuando hayamos terminado de usar la máquina y la queremos apagar para que no nos quite el total del saldo si no esta en uso, debemos darle a DETENER INSTANCIA, si le damos a terminar instancia será eliminada.**

Si nuestro ordenador usa sistema operativo Windows podemos usar git bash ya que usamos Git si tuvieses Linux como sistema no tendrías que instarte ningún programa de terceros, bueno tendrías que instalarte git si no lo tuvieses también.

Antes de entrar mediante SSH, vamos hacer unas pequeñas configuraciones a nuestro servidor virtual.

Como nuestro servidor virtual de Ubuntu, su uso va a ser de servidor Web tendremos que darle unas reglas para que los usuarios puedan acceder mediante http y https, para conectarnos nosotros median ssh y poder hacernos ping para comprobar por ejemplo si tenemos conexión a internet.

<input checked="" type="checkbox"/>	Name ▾	ID de la instancia	Estado de la i... ▾	Tipo de inst... ▾	Comprobación ...	Estado de la ...
<input checked="" type="checkbox"/>	daw_servidor	i-0a0bbc121d96e7703	En ejecución	t2.micro	-	1 alarma +

Instancia: i-0a0bbc121d96e7703 (daw\_servidor)

Detalles

Seguridad

Redes

Almacenamiento

Comprobaciones de estado

Monitoreo

Etiquetas

▼ Detalles de seguridad

Rol de IAM

-

Grupos de seguridad

sg-03ca7be17c1250273 (launch-wizard-1)

ID del propietario

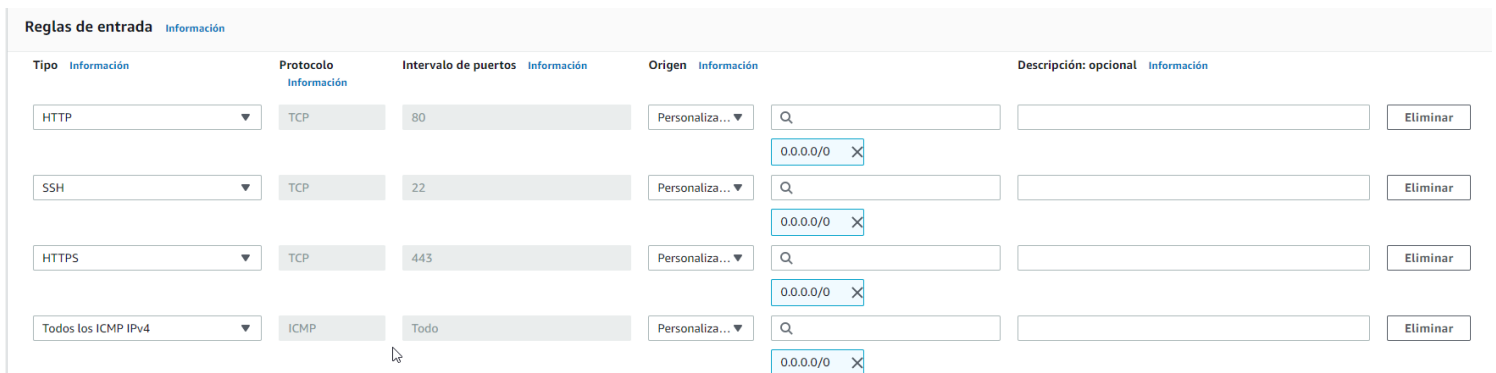
257914319538

Donde estábamos actualmente en nuestras instancias si bajamos hacia abajo vemos que pone id y nuestra id mas el nombre del servidor entre paréntesis, tenemos que dale a la pestaña de seguridad y hacer clic en el grupo de seguridad, que estará de color azul.



Reglas de entrada (4)					Editar reglas de entrada
Tipo	Protocolo	Intervalo de puertos	Origen	Descripción: opcional	
HTTP	TCP	80	0.0.0.0/0	-	
SSH	TCP	22	0.0.0.0/0	-	
HTTPS	TCP	443	0.0.0.0/0	-	
Todos los ICMP IPv4	ICMP	Todo	0.0.0.0/0	-	

Aquí podemos ver que yo ya las reglas de entrada ya las tengo hechas, para hacerlas tenemos que hacer clic en 'Editar reglas de entrada'



Tipo	Protocolo	Intervalo de puertos	Origen	Descripción: opcional	
HTTP	TCP	80	Personaliza... 0.0.0.0/0		Eliminar
SSH	TCP	22	Personaliza... 0.0.0.0/0		Eliminar
HTTPS	TCP	443	Personaliza... 0.0.0.0/0		Eliminar
Todos los ICMP IPv4	ICMP	Todo	Personaliza... 0.0.0.0/0		Eliminar

Tendríamos que seleccionar los protocolo que seria HTTP, SSH, HTTPS Y Todos los ICMP IPv4 protocolo en todos tcp el numero de puerto y el origen 0.0.0.0, para que las aceptemos todas y guardamos reglas.

Ahora solos nos queda darle una ip elástica , ya que no podemos darle una ip fija por que nos costaría dinero, con esto conseguimos darle una dirección ip pública a la que se pueda tener acceso desde internet y con ello asociar nuestra dirección ip a esta, por lo que nunca cambiaría esta dirección ip.



Etiquetas

Límites

▼ Instancias

**Instancias** New

Tipos de instancia

Plantillas de lanzamiento

Solicitudes de spot

Savings Plans

Instancias

reservadas New

Hosts dedicados

Instancias programadas

Reservas de capacidad

▼ Imágenes

AMI

▼ Elastic Block Store

Volúmenes

Instantáneas

Administrador del ciclo  
de vida

▼ Red y seguridad

Security Groups New

Direcciones IP  
elásticas New

Grupos de ubicación

Nos vamos a `Direcciones IP elásticas`.



Hacemos clic en `Asignar la dirección IP elástica`.

En el siguiente apartado lo dejamos todo como esta y le damos a asignar, seleccionamos nuestra máquina virtual y ya tendríamos nuestra dirección IP elástica.

Nos vamos otra vez a Instancias, podemos ver que en el apartado IP elástica ya tenemos una dirección IP, que la podemos usar tanto para hacer las conexiones de ssh o http/s y ping con esa IP, ahora solo nos hace falta darle un DNS.

En mi caso como tengo cuenta Github students, tengo un dominio gratis en [.Tech](#).

Después de haberle asignado un DNS a nuestro servidor ahora toca entrar en el y hacer las siguientes configuraciones.

## Scripts que hay que ejecutar

Que se meta en el enlace de la página web (<http://proyectos-andres.tech/pr101>).

Si eres desarrollador/programador estos son los siguientes scripts que he hecho para la configuración de la máquina.

Tras habernos dado AWS el par de claves.

Nos habrá dado la clave pública de nuestro servidor para conectarnos a ella tenemos que ir a la carpeta de nuestro usuario, si estamos en git bash o en una consola Unix haciendo un `cd ~`, nos lleva directamente a él.

Creamos un directorio que se llame `.ssh` (con el punto al principio significa que esta en oculto, por lo que hay que hacer el comando `ls -a` para ver los directorios ocultos )

Nos copiamos la llave pública y la dejamos dentro de este directorio, guardar bien esta llave ya que es muy importante para entrar en el servidor.

Para conectarnos mediante ssh :

`Eval `ssh-agent -s`` → para crear el proceso, luego nos dará el numero de que el proceso se ha creado.

`Ssh-add clave publica` → para añadirla a ese proceso .

`Ssh usuario del servidor@ip/dns del servidor` para conectarnos a él.

```
andre@DESKTOP-9NCF627 MINGW64 ~/.ssh
$ eval `ssh-agent -s`
Agent pid 1411

andre@DESKTOP-9NCF627 MINGW64 ~/.ssh
$ ssh-add daw_server_2022.pem
Identity added: daw_server_2022.pem (daw_server_2022.pem)
```

```
$ ssh ubuntu@proyectos-andres.tech
The authenticity of host 'proyectos-andres.tech (3.223.61.50)' can't be established.
ECDSA key fingerprint is SHA256:Tsz7PzhF0cJNXa5CP6rs9hYFH28/0DrRXyYopzSMAZg.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'proyectos-andres.tech,3.223.61.50' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.4.0-1038-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of mar 16 mar 2021 11:19:28 UTC

System load:  0.04               Processes:           108
Usage of /:   35.9% of 7.69GB    Users logged in:    0
Memory usage: 54%               IPv4 address for eth0: 172.31.38.126
Swap usage:   0%

 * Introducing self-healing high availability clusters in MicroK8s.
   Simple, hardened, Kubernetes for production, from RaspberryPi to DC.

   https://microk8s.io/high-availability

8 updates can be installed immediately.
0 of these updates are security updates.
To see these additional updates run: apt list --upgradable

Last login: Wed Mar 10 11:05:47 2021 from 80.24.185.208
ubuntu@ip-172-31-38-126:~$
```

Tras habernos conectado actualizaremos nuestro repositorios y nuestra instancia :

Sudo apt-get update && sudo apt-get upgrade -y

Instalamos los paquetes básicos en español :

Sudo apt-get install language-pack-es -y

Configuramos nuestro locale para trabajar en español :

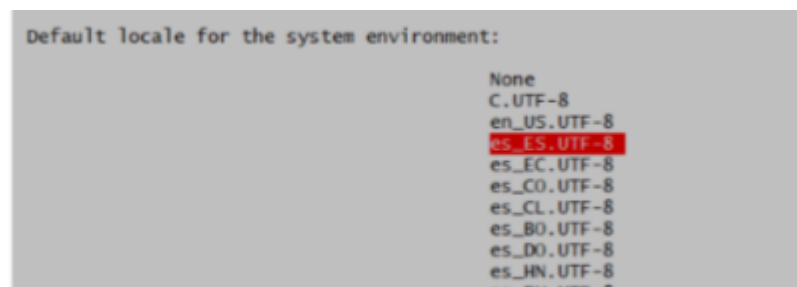
Sudo dpkg-reconfigure locales

Dejamos seleccionado el locale en\_US.UTF-8 UTF-8

Seleccionamos con el espacio el de Español España es\_ES.UTF-8 UTF\_8 :



Seleccionamos el locale por defecto es\_ES.UTF-8



Para comprobar que se está aplicando el locale que queremos, ejecutamos el comando `locale`

Veremos que no lo ha hecho aún para ello lo que debemos hacer es salir mediante `ssh` haciendo un `exit`

Volvemos a entrar y ya estará todo correctamente.

Para configurar nuestra zona horaria :

```
Sudo dpkg-reconfigure tzdata
```

Escogemos Europa y en zona horaria Madrid.

Y si ejecutamos `date`

Vemos que estaría la hora correcta

## Instalar Apache, Php 7.4 y mysql

Ejecutamos el siguiente comando :

```
Sudo apt-get install lamp-server^
```

Si ejecutamos

`Php -v` → veremos la versión instalada de php

`Mysql --version` → veremos la versión de mysql instalada

`Apache2ctl -v` → veremos la versión de apache

Reiniciamos Apache2

```
Systemctl restart apache2
```

Comprobamos si funciona Apache metiéndonos en nuestra dirección IP /DNS, en nuestro navegador web.



Funciona.

Para agregar php en apache tenemos que hacer lo siguiente

```
Sudo a2enmod php7.4
```

Tenemos que añadir el modulo de php en este caso como hemos instalado la versión 7.4 por ello ponemos php7.4.

Luego dentro de 'var/www/html/' nos creamos un archivo php y escribimos la función de php

```
Phpinfo();
```

Y veremos que funciona.

### **PHP 8.0 en apache y PHP 7.4 en la línea de comandos por defecto.**

Instalamos php 8.0

Instalamos las utilidades para poder instalar un repositorio con firma :

```
Sudo apt install ca-certificates apt-transport-https software-properties-common
```

Agregamos el repositorio ondrej/php :

```
Sudo add-apt-repository ppa:ondrej/php
```

Instalamos php 8.0

```
Sudo apt-get install php8.0 libapache2-mod-php8.0 php8.0-mysql
```

Añadimos a apache el modulop php8.0

```
Sudo a2ismod php7.4
```

```
Sudo a2enmod php8.0
```

Reiniciamos Apache2

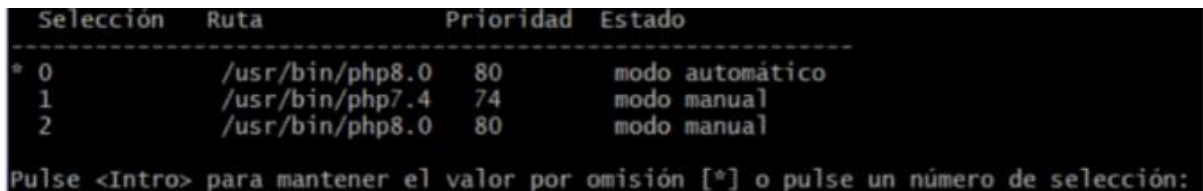
```
Systemctl restart apache2
```

Comprobamos como esta el servidor de apache

```
Sudo systemctl status apache2
```

Ahora toca configurar php7.4 en línea de comandos

```
Sudo update-alternatives --config php
```



Selección	Ruta	Prioridad	Estado
* 0	/usr/bin/php8.0	80	modo automático
1	/usr/bin/php7.4	74	modo manual
2	/usr/bin/php8.0	80	modo manual

Pulse <Intro> para mantener el valor por omisión [\*] o pulse un número de selección:

Seleccionamos en este caso la 1 ya que es la versión 7.4 para línea de comandos.

Si hacemos un php -v podemos ver que en la línea de comandos esta instalada la versión 7.4

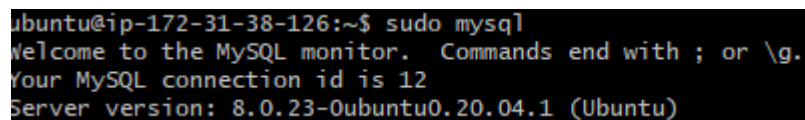
## INSTALAR PHPMYADMIN

Instalamos phpmyadmin mas sus dependencias y otros paquetes necesarios

```
sudo apt install phpmyadmin php-mbstring php-zip php-gd php-json php-curl  
-y
```

Seleccionamos apache2, le damos contraseña a phpmyadmin y estaría listo.

Ahora entrar en sql y añadir usuarios en este caso vamos añadirle una contraseña a Jose Carlos.



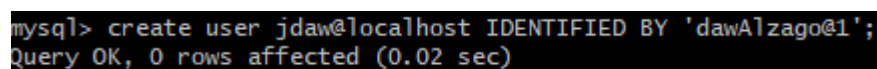
```
ubuntu@ip-172-31-38-126:~$ sudo mysql  
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 12  
Server version: 8.0.23-0ubuntu0.20.04.1 (Ubuntu)
```

Ejecutamos el siguiente código :

```
Sudo mysql
```

Ya estaríamos dentro de mysql

Ahora creamos el usuario.



```
mysql> create user jdaw@localhost IDENTIFIED BY 'dawAlzago@1';  
Query OK, 0 rows affected (0.02 sec)
```

Usuario → jdaw

Contraseña → dawAlzago@1



Le damos privilegios para todo

```
mysql> GRANT ALL PRIVILEGES ON *.* TO jdaw@localhost WITH GRANT OPTION  
-> ;  
Query OK, 0 rows affected (0.01 sec)
```

Ahora el usuario jdaw puede entrar en phpmyadmin y hacer lo que quiera ya que tiene todos los permisos.

[phpmyadmin](#)

## Clave pública de nuestro usuario jdaw, para que se conecte a nuestro servidor.

Tras habernos dado la clave pública de nuestro usuario jdaw, toca añadirla al servidor a su usuario, para ello usamos el comando scp que es un comando ssh para subir o bajar archivos, lo tenemos que hacer con nuestro usuario y subirla.

```
andre@DESKTOP-9NCF627 MINGW64 ~/.ssh  
$ scp jca_daw_rsa.pub ubuntu@proyectos-andres.tech:/tmp/
```

En nuestro servidor lo guardamos en el directorio temporal.

Nos conectamos mediante ssh y vamos hacia ese directorio

```
ubuntu@ip-172-31-38-126:/tmp$ ls jc*  
jca_daw_rsa.pub
```

Vemos que ahí se encuentra ahora creamos un usuario

```
ubuntu@ip-172-31-38-126:/home$ adduser jdaw  
adduser: Sólo root puede añadir un usuario o un grupo al sistema.  
ubuntu@ip-172-31-38-126:/home$ sudo adduser jdaw  
Añadiendo el usuario 'jdaw' ...  
Añadiendo el nuevo grupo 'jdaw' (1002) ...  
Añadiendo el nuevo usuario 'jdaw' (1002) con grupo 'jdaw' ...  
Creando el directorio personal '/home/jdaw' ...  
Copiando los ficheros desde '/etc/skel' ...  
Nueva contraseña:  
Vuelva a escribir la nueva contraseña:  
passwd: contraseña actualizada correctamente  
Cambiando la información de usuario para jdaw  
Introduzca el nuevo valor, o presione INTRO para el predeterminado  
Nombre completo []:  
Número de habitación []:  
Teléfono del trabajo []:  
Teléfono de casa []:  
Otro []:  
¿Es correcta la información? [S/n] s
```

Nombre del usuario en el servidor → jdaw

Contraseña del usuario en el servidor → daw

Ahora lo añadimos al grupo sudo para que tenga permisos de administrador, ya que Jose Carlos es un administrador bueno.

```
ubuntu@ip-172-31-38-126:/home$ sudo adduser jdaw sudo
Añadiendo al usuario 'jdaw' al grupo 'sudo' ...
Añadiendo al usuario jdaw al grupo sudo
Hecho.
```

Ahora en el directorio de home de jdaw, creamos un directorio llamado .ssh y dentro de el guardamos la clave publica y le cambiamos el nombre a authorized\_keys

```
ubuntu@ip-172-31-38-126:/tmp$ mv jca_daw_rsa.pub /home/jdaw/.ssh/
```

```
ubuntu@ip-172-31-38-126:/home/jdaw$ sudo cp jca_daw_rsa.pub ~/.ssh/authorized_keys
```

Usuario y contraseña en apache para que pueda entrar en la parte privada de nuestra aplicación web.

Nos tenemos que al directorio donde queramos que solo accedan por usuario y contraseña en mi caso sería dentro del directorio pr101/parte\_privada/ que se encuentra en var/www/html/ para ello ejecutamos lo siguiente :

```
Sudo htpasswd -c /var/www/html/pr101/parte_privada/.htpasswd jdaw
```

Contraseña → daw

Si queremos añadir otro usuario sería el mismo comando sin '-c' ya que nos borraría el archivo y jdaw desaparece.

```
Sudo htpasswd /var/www/html/pr101/parte_privada/.htpasswd otro usuario
```

Para que esto surja efecto tenemos entrar en la configuración de apache y crearnos un virtualhost :

```
sudo nano /etc/apache2/sites-enabled/000-default.conf
```

Escribimos lo siguiente, <directory aquí va donde queremos que cuando un usuario entre le pida usuario y contraseña>

```
<VirtualHost *:80>
    <Directory "var/www/html/pr101/parte_privada">
        AuthType Basic
        AuthName "Restricted Content"
        AuthUserFile /var/www/html/pr101/parte_privada/.htpasswd
        Require valid-user
    </Directory>
</VirtualHost>
```

Reiniciamos apache y listo.

## **Referencias**

- Alumno Andrés Cortés de 080 formación
- Documento “IT00179- Configurar AWS EC2 ubuntu 20.04” autor José Carlos Álvarez