S.I.G.D.

SISTEMA OPERATIVOS III

GFORZE

| ROL | APELLIDO | NOMBRE | C.I | CORREO | TELÉFONO |
| --- | --- | --- | --- | --- | --- |
| Coordinador | Mujica | Martin | 5.523.189-7 | mujica0106@gmail.com | 095003085 |
| Subcoordinador | Farías | Facundo | 5.332.076-0 | j.facuwebmaster@gmail.com | 093448935 |
| Integrante 1 | González | Sheila | 6.392.488-5 | sheilagn2003@gmail.com | 096247975 |
| Integrante 2 | Vallejos | Diego | 5.493.546-5 | micezedie@gmail.com | 097302419 |

Docente: Fagúndez, Luis

| Fecha de culminación  19/07/2022 |
| --- |

| PRIMERA ENTREGA |
| --- |

**Índice**

[**Estudio de los diferentes roles de los usuarios del sistema.**](#_heading=h.kjx9kiu5q40j) **4**

[**Usuarios Cliente**](#_heading=h.e16qe95wfl07) **4**

[**Usuarios Base de Datos**](#_heading=h.9dxdno6f2hg3) **4**

[**Usuarios Servidor**](#_heading=h.8pa6ox553h0o) **4**

[**Servidor de Respaldo**](#_heading=h.weln2whnbgt7) **4**

[**Relevamiento y justificación del Sistema Operativo a utilizar tanto en las terminales de los usuarios como en el servidor.**](#_heading=h.uge1bvvztwm8) **5**

[**Manual de instalación del Sistema Operativo en el servidor.**](#_heading=h.onzl0mj6fo3j) **8**

[Paso 1:](#_heading=h.flyv3niksa8e) 8

[Paso 2:](#_heading=h.h8dktxdv5bkm) 9

[Paso 3:](#_heading=h.lp017um1kojw) 10

[Paso 4:](#_heading=h.k5xwtinr27p1) 11

[Paso 5:](#_heading=h.mf2tsjz1g5ns) 12

[Paso 6:](#_heading=h.xjn2v1a4icbt) 13

[Paso 6:](#_heading=h.uxwrj5mf0qwv) 14

[Paso 7:](#_heading=h.fyusdus3yi4v) 16

[Paso 8:](#_heading=h.ir0jvu3dyzb6) 16

[**Instalación de Docker dentro de la máquina virtual con el sistema operativo a utilizar. (WEB)**](#_heading=h.mcb1ftb22lfc) **20**

[Paso 1:](#_heading=h.8s6q1b657m5l) 20

[Paso 2:](#_heading=h.hprnw1u75z65) 20

[Paso 3:](#_heading=h.9n93wmgsyta5) 21

[Paso 4:](#_heading=h.q8xe9hvl495x) 21

[Paso 5:](#_heading=h.oc4g9qxdmp7y) 22

[**Instalación de un servidor LAMP en la máquina virtual (usando docker o local).(web)**](#_heading=h.367kai5kn6rk) **23**

[LOCAL](#_heading=h.y5xm9aolm47a) 23

[PASO 0:](#_heading=h.6u937ylcanm6) 23

[PASO 1:](#_heading=h.r43gqdvksgbu) 23

[PASO 2:](#_heading=h.h5ozj5wihi) 24

[PASO 3:](#_heading=h.u3ken5ojd3ps) 25

[PASO 4:](#_heading=h.qhwmq3dvtwg4) 28

[PASO 5:](#_heading=h.ox2nvroz6ygu) 28

[**Manual de Instalación de MySQL/MariaDB junto a sus requisitos para trabajar con Docker.**](#_heading=h.lgne9bwm2url) **29**

[Paso 0:](#_heading=h.319p6jhg810z) 29

[Paso 1:](#_heading=h.9vmyuegbdbcr) 29

[Paso 2:](#_heading=h.3hhban2wn2n) 29

[Paso 3:](#_heading=h.8zcbjcxg0xmw) 30

[Paso 4:](#_heading=h.iawm30hlb3ch) 30

[Paso 5:](#_heading=h.4bq3iqcs116y) 30

[Paso 6:](#_heading=h.wxhdge2siv6o) 31

### Estudio de los diferentes roles de los usuarios del sistema.

#### Usuarios Cliente

Invitado: Ingresando la C.I. puede cargar(mostrar) un perfil desde cualquier terminal.

Administrador del Sitio:Tiene todos los permisos.

Árbitro:Incidencias. Verifica si los jugadores están habilitados.Registra la veracidad de las incidencias registradas en el encuentro (Anotaciones, Sanciones, Faltas).

DT(Director Tecnico):Genera equipos. Carga la ficha de los Jugadores previamente cargados por el Administrativo.

Administrativo:Carga las fichas. Asigna los roles del Sitio(le otorga los permisos a cada usuario).

Analista:Verifica y sube los datos completos de los partidos.

#### Usuarios Base de Datos

Root

Test:Crear usuarios para verificar que funcione correctamente.

#### Usuarios Servidor

Root(Default)

Administrador de Respaldo:Genera un respaldo de todos los datos subidos.

Administrador de Paquetes:Mantiene actualizado el servidor.

Administrador de Base de Datos:Chequea los datos de la base de datos.

Administrador de Logs:Chequea si hay error en el Servidor.

Todos estos usuarios van a estar en /etc/sudoers.

#### Servidor de Respaldo

Administrador de Respaldo

Root

### Relevamiento y justificación del Sistema Operativo a utilizar tanto en las terminales de los usuarios como en el servidor.

Nuestra empresa GFORZE, una empresa de desarrollo de software, realizó una puesta en común con el equipo de desarrollo, sobre la utilización de un sistema operativo que se adapte a las necesidades de un servidor.En base a los requerimientos que nos proporciona el cliente en base al estudio de la especificación de requerimientos llegamos a la conclusión de usar Fedora Server porque se adapta a nuestras necesidades. A continuación se detalla un punteo de los beneficios de aplicar Fedora Server al software:

-Gratuito y flexible.

-No requiere muchas especificaciones para su funcionamiento.

-Es seguro, ya que es robusto y evita el ingreso de intrusos, aplica SELinux( Security-Enhanced-Linux) que es un módulo de seguridad en el linux kernel.

-Es un software libre, dicho de otro modo, código abierto.

-Ofrece un soporte constante.

-Administrar de forma favorable la memoria.

-Arranque del sistema de forma completa y sin errores

**Terminal:**

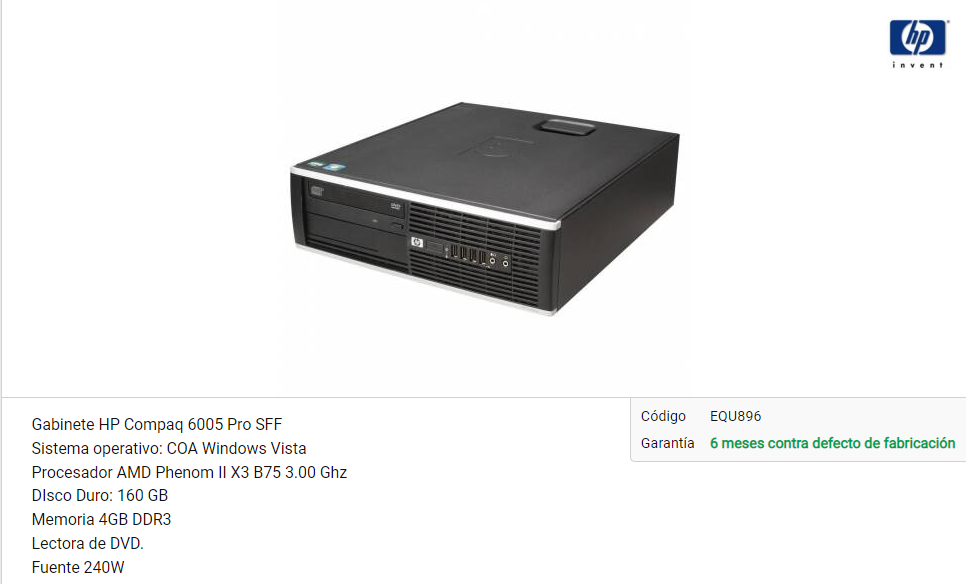
Para comprar las terminales vamos a dejarles 2 presupuestos diferentes, uno de bajos costos y uno de costos medios.

Les ofrecemos estos 2 presupuestos ya que básicamente la terminal lo único que va a hacer será abrir google o su navegador preferido para inicializar la pagina, lo unico que cambiaria seria el SO que prefiera utilizar para ejecutar dicha operación.

**Primer presupuesto:**

De la empresa CDR Medios hemos visualizado distintas computadoras y hemos elegido para este presupuesto de precio bajo:

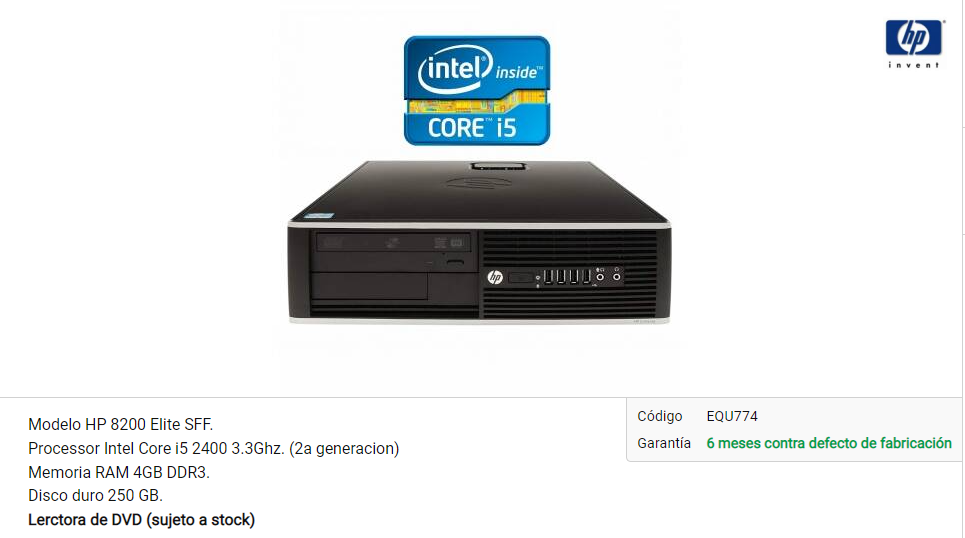
Equipo HP AMD 3.0Ghz, 4GB, 160GB, DVD que valdría 45 U$D, esta computadora tendría que utilizar un SO de linux ya que lo requerimientos no nos permite utilizar WINDOW.



**Segundo presupuesto:**

De la empresa CDR Medios hemos visualizado distintas computadoras y hemos elegido para este presupuesto de precio bajo:

Equipo HP Core i5 3.3Ghz, 4GB, 250GB que valdria 125 U$D, esta computadora puede utilizar cualquier SO hasta Window 10 Home.

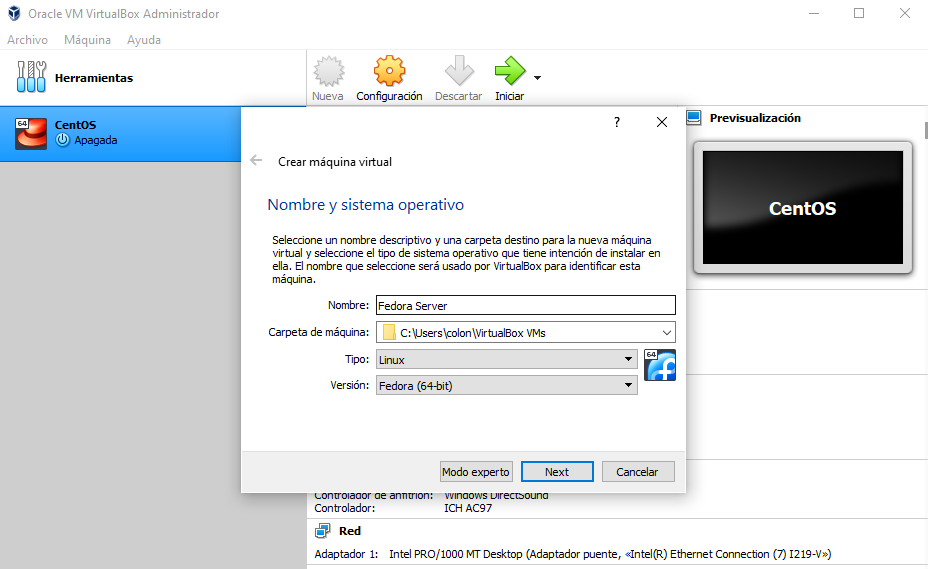


### Manual de instalación del Sistema Operativo en el servidor.

#### Paso 1:

Primero le debemos dar a Maquina/Nuevo, luego tendremos que escribir Fedora Server y el programa automáticamente te pondrá los datos de Tipo y Versión.

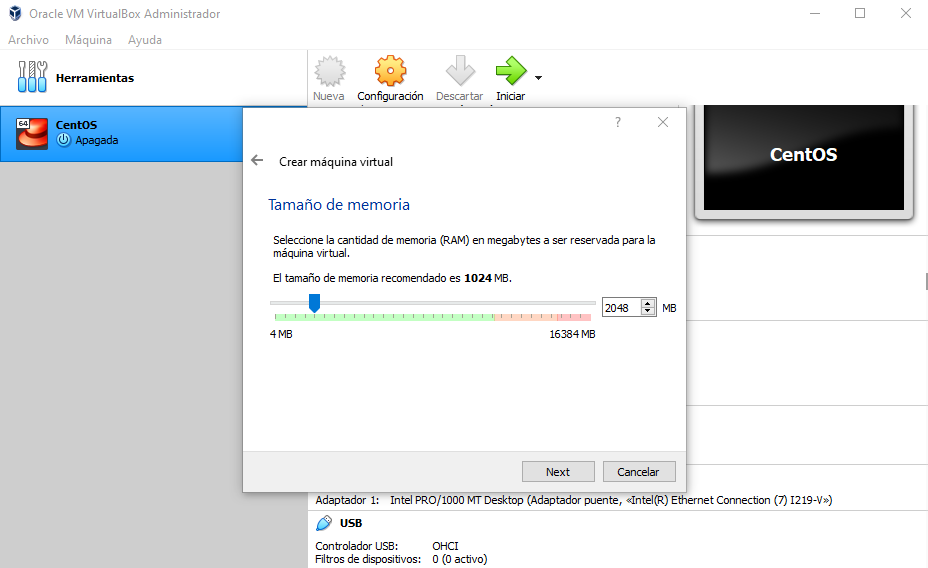
Luego le daremos a next y pasaremos al siguiente paso.



#### Paso 2:

Luego de darle a next nos aparecerá esta opción. Aquí tendremos que seleccionar el tamaño de la memoria ram que virtualbox utiliza para reproducir nuestro Fedora Server.

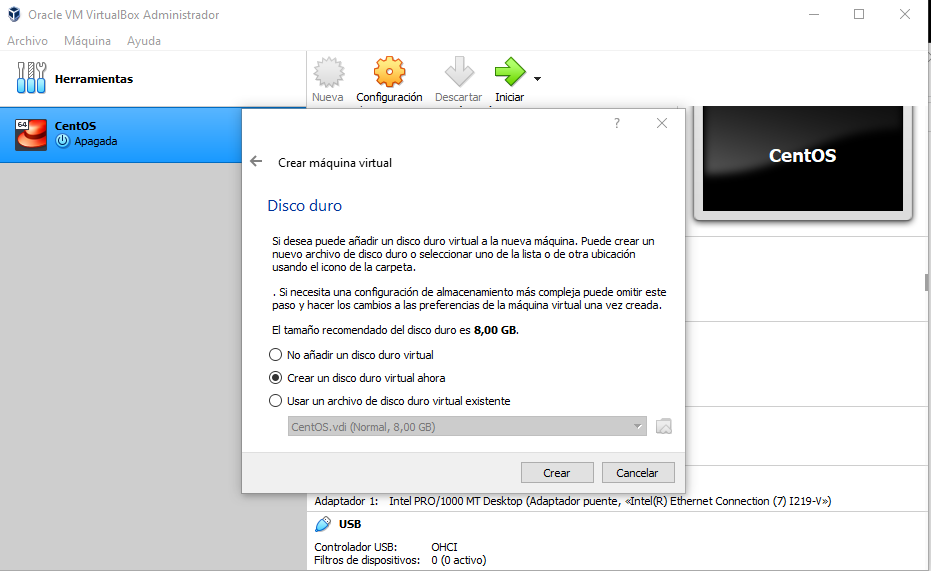
En este caso pondremos 2GB de memoria RAM pero si quieres le puedes poner más, dependiendo de cuanta memoria ram tengas disponible. Por último le daremos next y procederemos el siguiente paso



#### 

#### Paso 3:

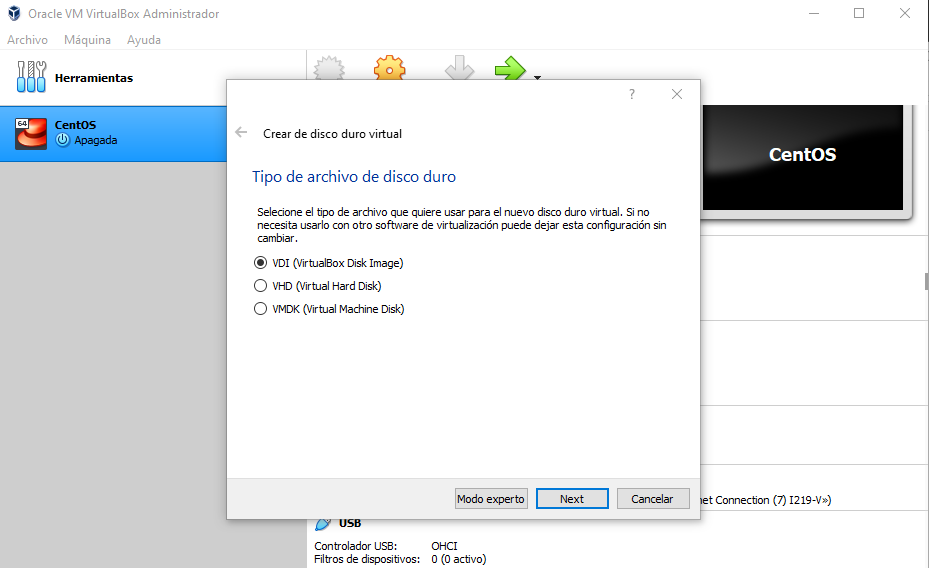
En este paso debemos de seleccionar un disco duro para que la máquina virtual pueda utilizar, en este caso si o si debemos de poner la opción “Crea un disco duro virtual ahora”.  
Por último le daremos a next.



#### 

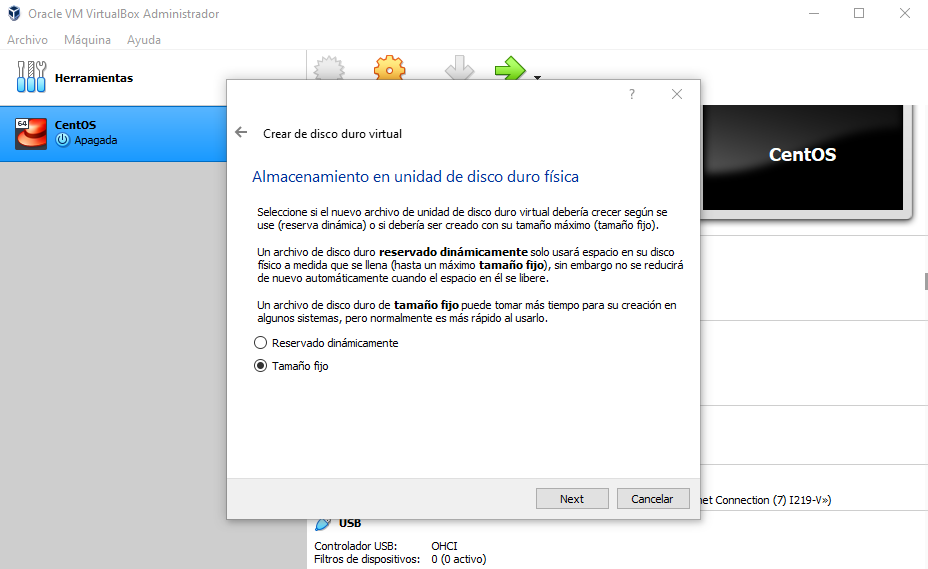
#### Paso 4:

Aquí seleccionaremos el tipo de disco duro virtual, este básicamente seleccionamos VDI ya que nosotros tenemos un archivo .iso para instalar la máquina virtual.



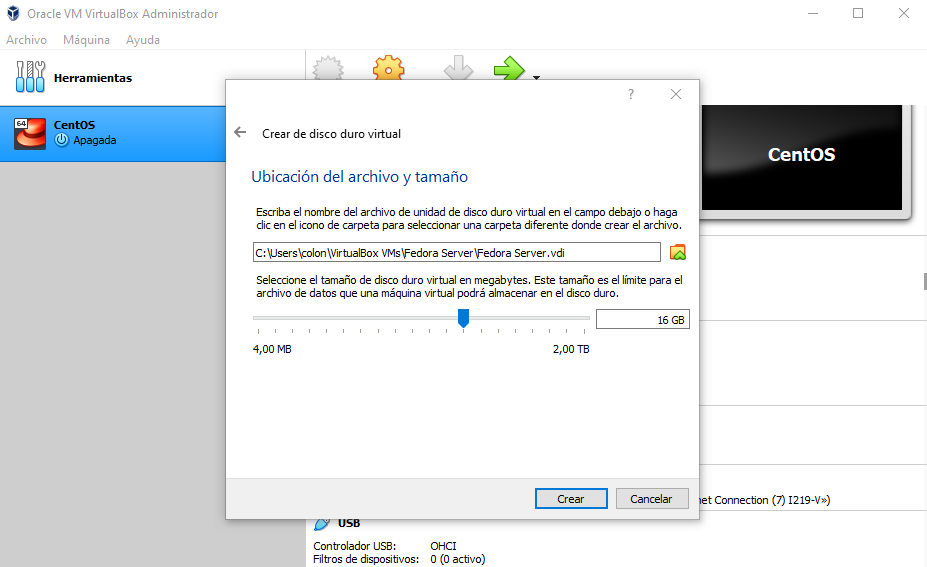
#### Paso 5:

Aquí seleccionamos tamaño fijo para que si al superar la cantidad de contenido no nos abarca más espacio del definido anteriormente, si tienes espacio de sobra igual puede ponerlo reservado dinámicamente pero eso es a tu elección.



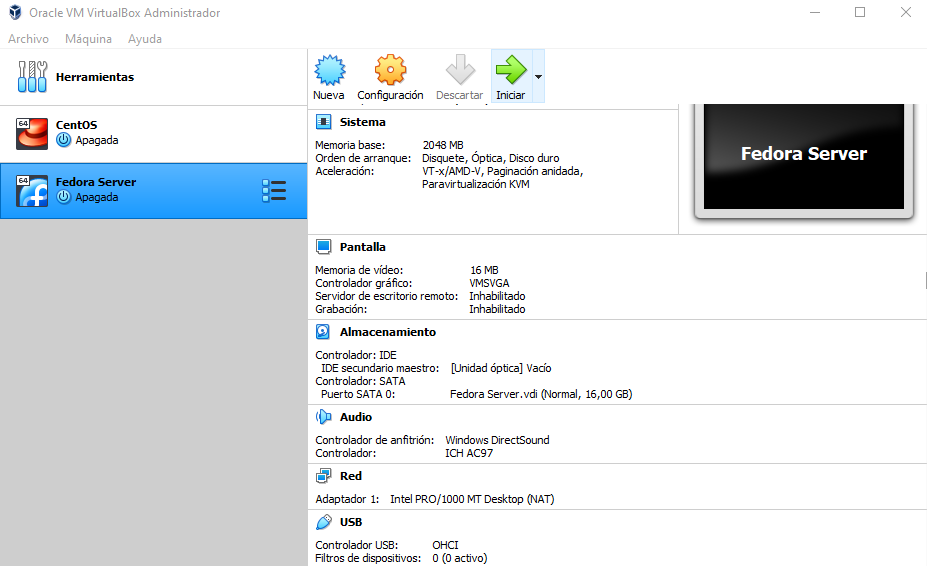
#### Paso 6:

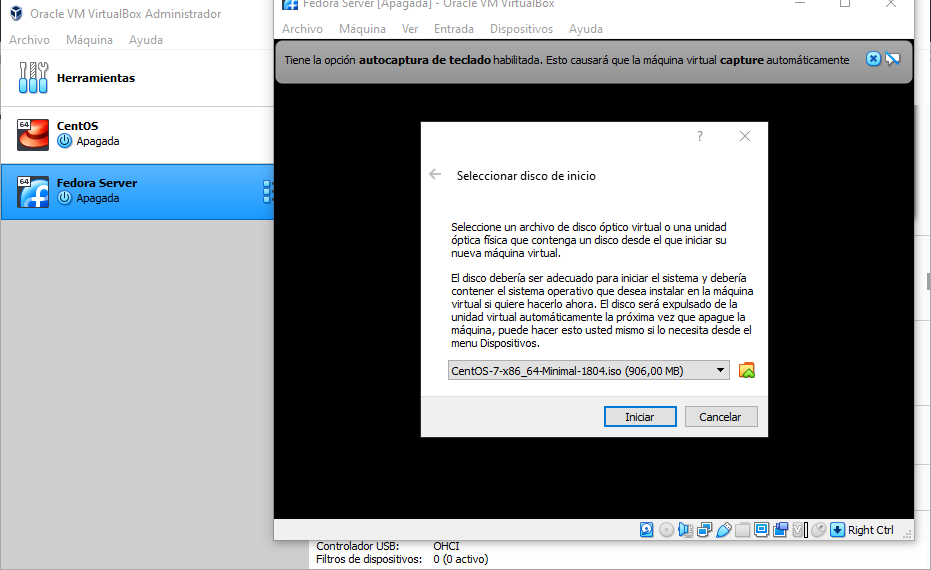
Aquí seleccionamos la cantidad de espacio que queremos que tenga Fedora Server para utilizar el mínimo es 8GB pero a su gusto pueden agregar más.

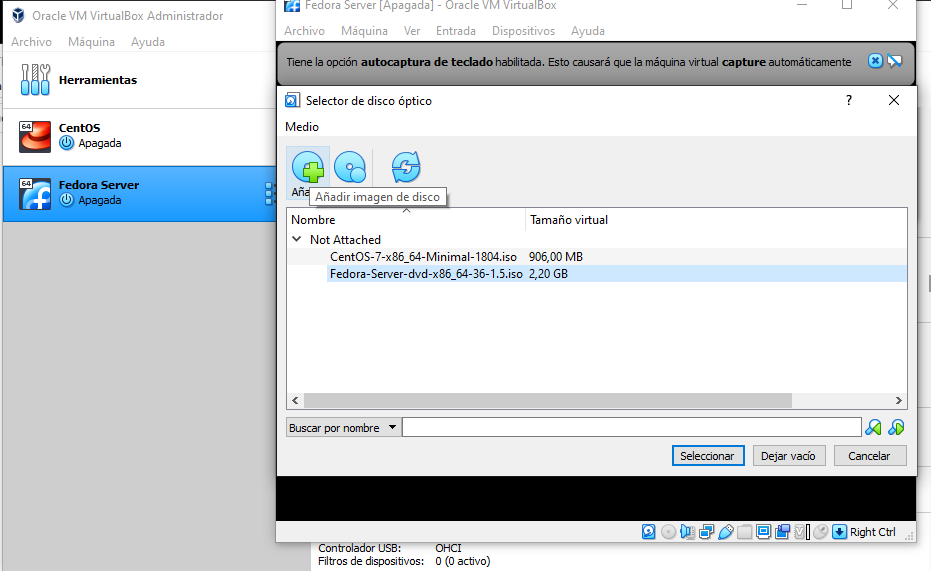


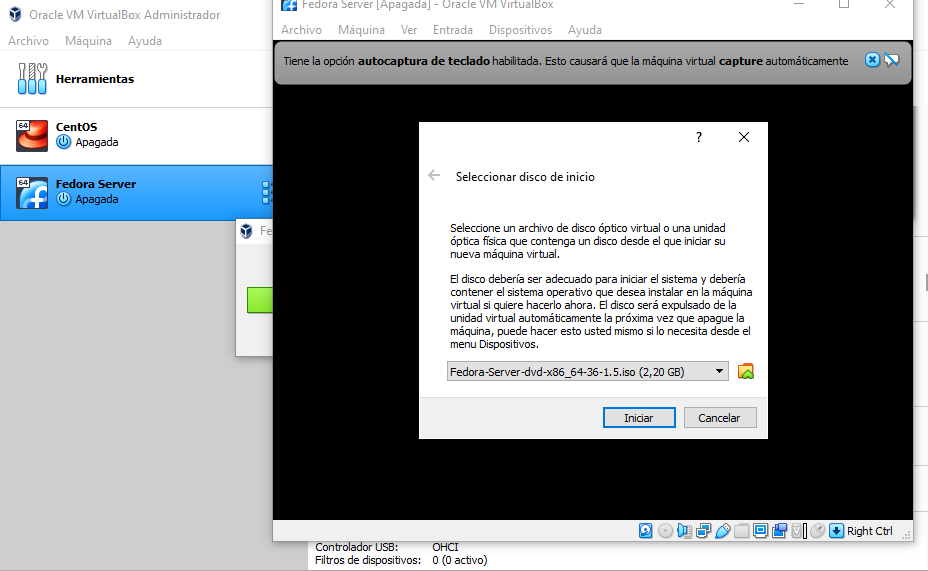
#### Paso 6:

En este paso le daremos a iniciar Fedora Server seleccionaremos el archivo .iso de Fedora Server y le daremos a iniciar.



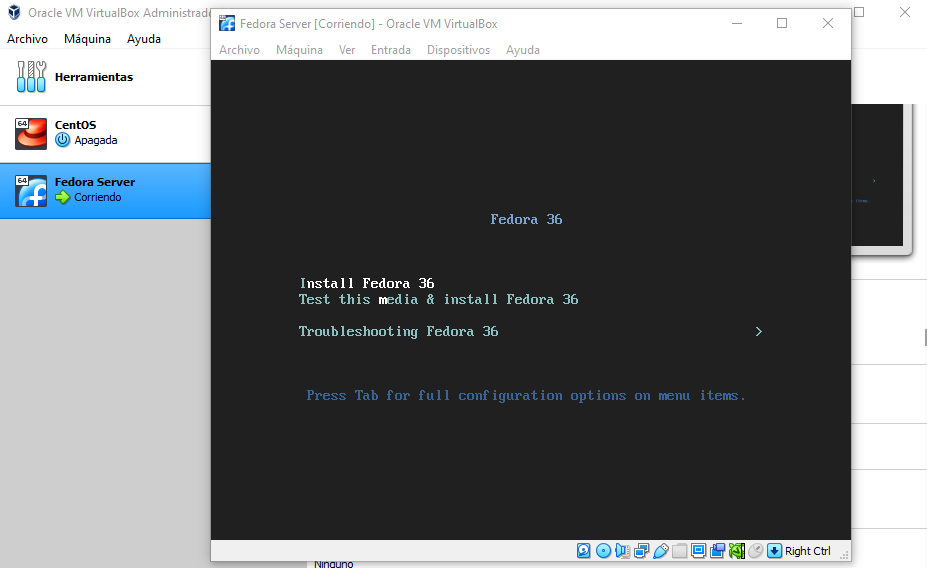






#### Paso 7:

Aquí con el teclado (las flechas) seleccionamos la opción install fedora.



#### Paso 8:

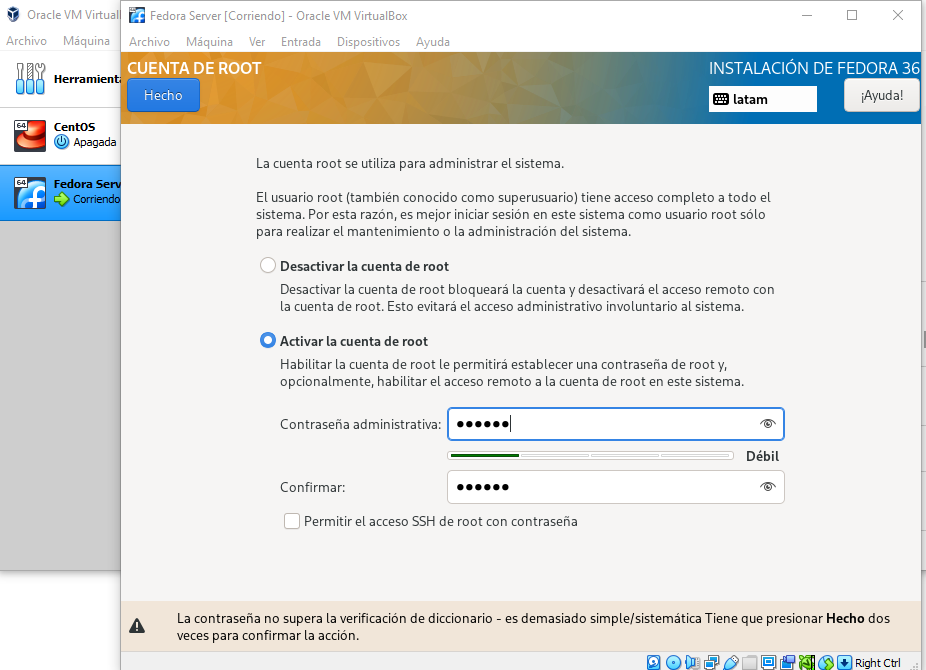
Aquí pasaremos a la instalación de Fedora Server, Seleccionamos idioma, el destino de instalación (Básicamente seleccionar el disco duro), seleccionar el usuario root, activar el usuario y poner una contraseña.

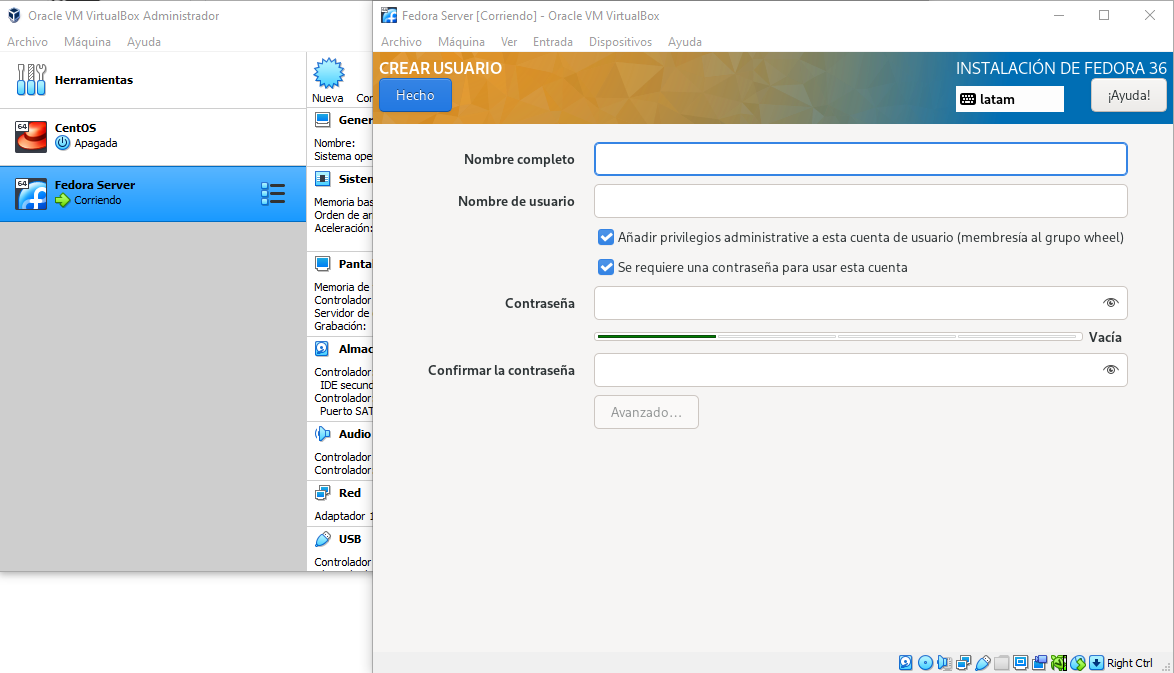
Por último seleccionamos el usuario y llenamos las opción(Nombre y contraseña), por ultimo le damos a COMENZAR INSTALACIÓN y esperamos a que nos instale Fedora Server.

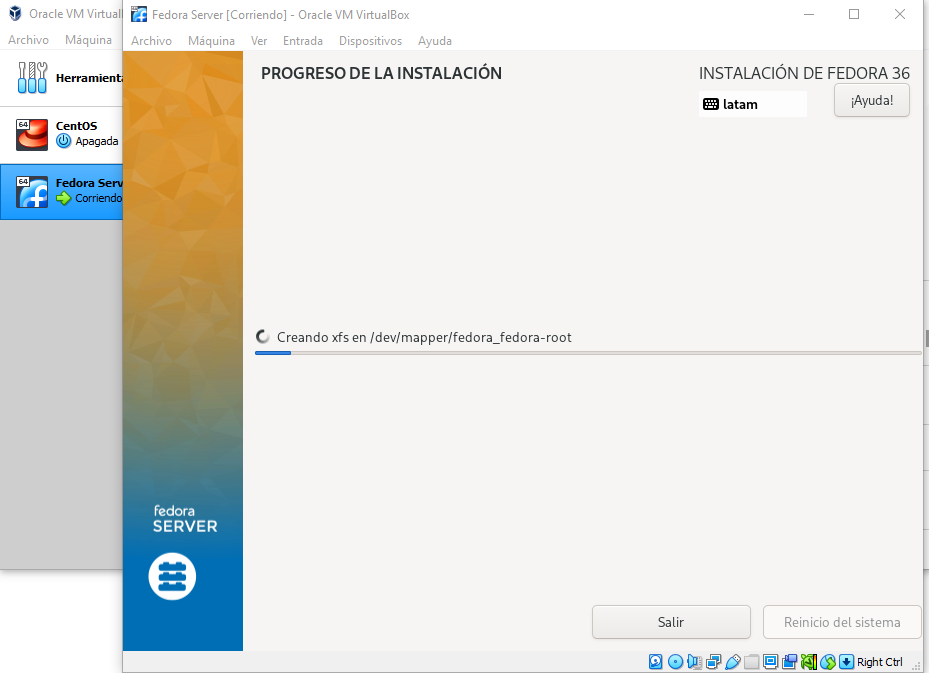










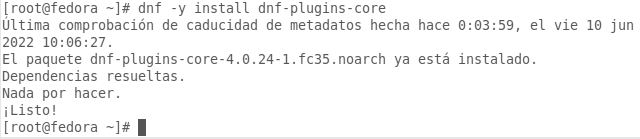


### Instalación de Docker dentro de la máquina virtual con el sistema operativo a utilizar. (WEB)

#### Paso 1:

Aquí básicamente estamos instalando los plugins necesarios para instalar el docker.

– # dnf -y install dnf-plugins-core



#### Paso 2:

En este paso se agrega el repositorio docker.

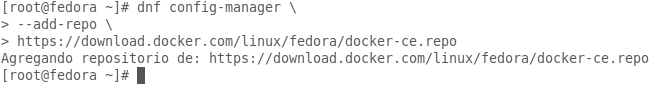
– # dnf config-manager \

–

--add-repo \

–

<https://download.docker.com/linux/fedora/docker-ce.repo>

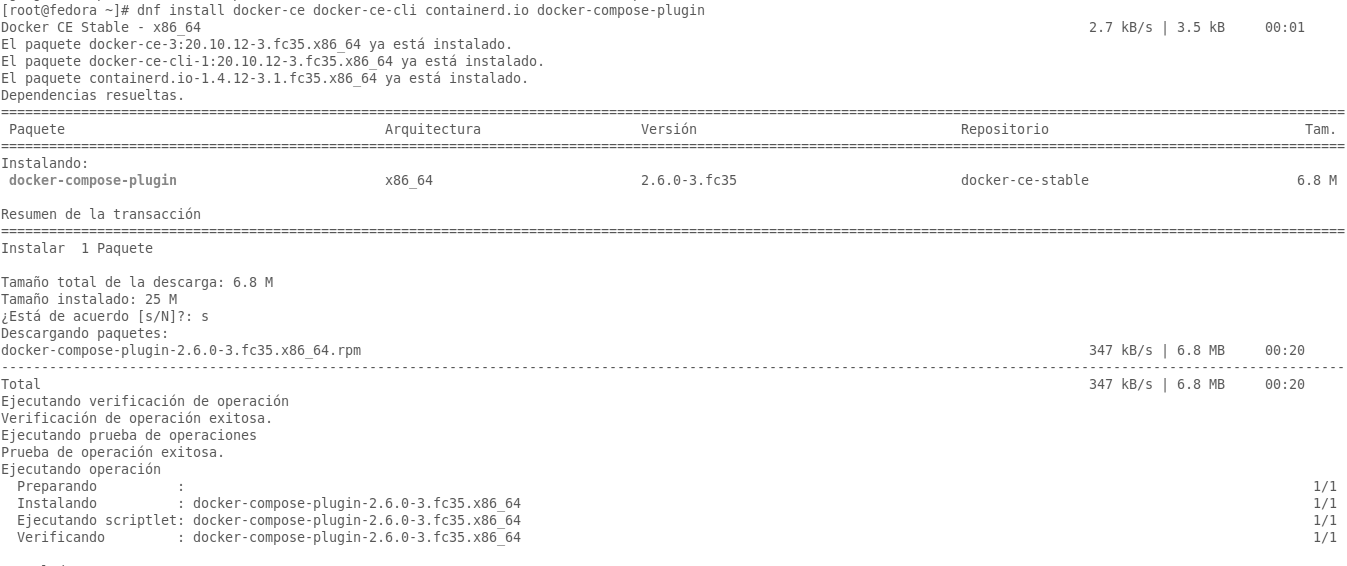


#### Paso 3:

En este paso instalamos el docker.

– # dnf install docker-ce docker-ce-cli containerd.io docker-

compose-plugin





#### Paso 4:

En este paso iniciamos el docker.

– # systemctl start docker

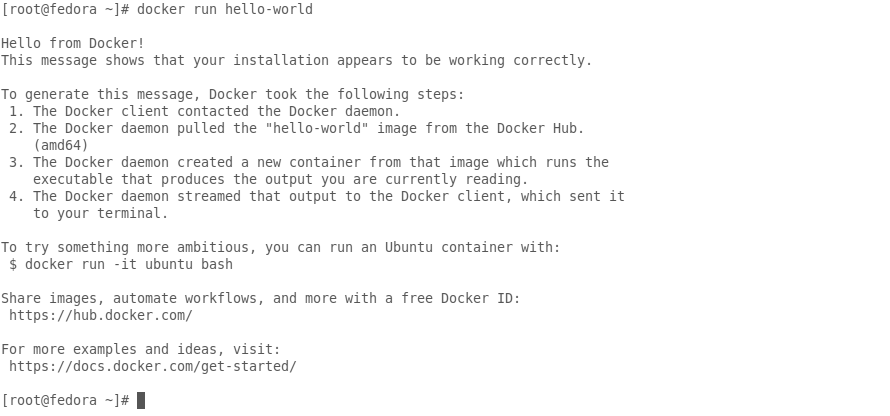
– # systemctl enable docker



#### Paso 5:

Y finalmente con este comando vemos si está correctamente instalado.

– # docker run hello-world



### Instalación de un servidor LAMP en la máquina virtual (usando docker o local).(web)

#### LOCAL

Primeros pasos

LAMP, significado:

– L – Linux → Sería el kernel que estamos utilizando.

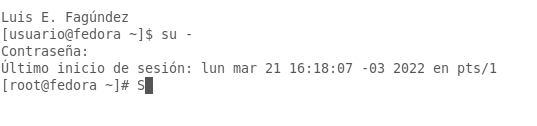
– A – Apache → Es el servidor que se encarga de interpretar a HTTP/HTTPS.

– M – MySQL/MariaDB → Seria el servidor de BD encargado.

– P – PHP → Lenguaje de programación que vamos a utilizar en este curso.

##### **PASO 0:**

* La consola nos inicia como USUARIO.
* Se debe cambiar a usuario ROOT para poder configurar nuestro FEDORA SERVER.



##### PASO 1:

* Actualizamos FEDORA SERVER:

Con el comando → # dnf -y update

* El comando dnf sirve para administrar los paquetes, en este caso solicitamos

que se actualicen todos los paquetes de nuestro servidor

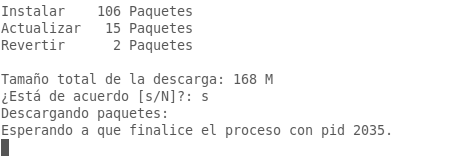


* Instalar herramientas básicas para desarrollo:

→ # yum groupinstall "Development Tools" "Development Libraries"



* Se pedirá una instalación [s/n] (si / no).



##### **PASO 2:**

* Instalamos el WEB SERVER (Apache Linux en éste caso junto a

PHP en su última versión disponible)

→ # dnf groupinstall "Web Server"



* Se pedirá una confirmación de instalación [s/n] (si / no).



* Con el comando systemctl le indicamos que inicie el servidor

web y lo configure para iniciar en el arranque del sistema:

→ systemctl start httpd



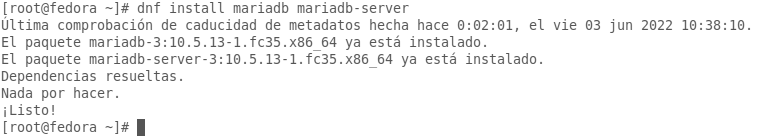
→ systemctl enable httpd



##### **PASO 3:**

* Instalamos el motor de Bases de Datos MariaDB-Server

→ # dnf install mariadb mariadb-server



* Usamos el comando systemctl para proceder igual que apache

.→ systemctl start mariadb.service



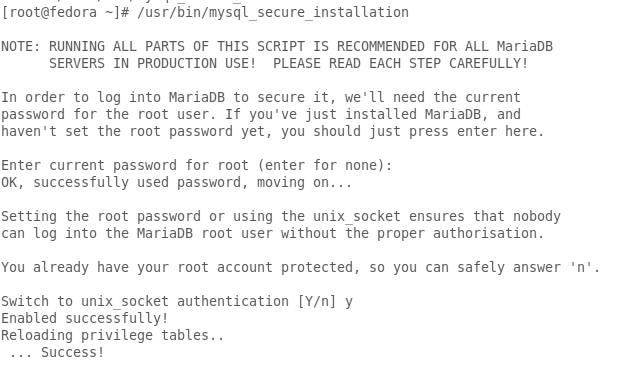
→ systemctl enable mariadb.service

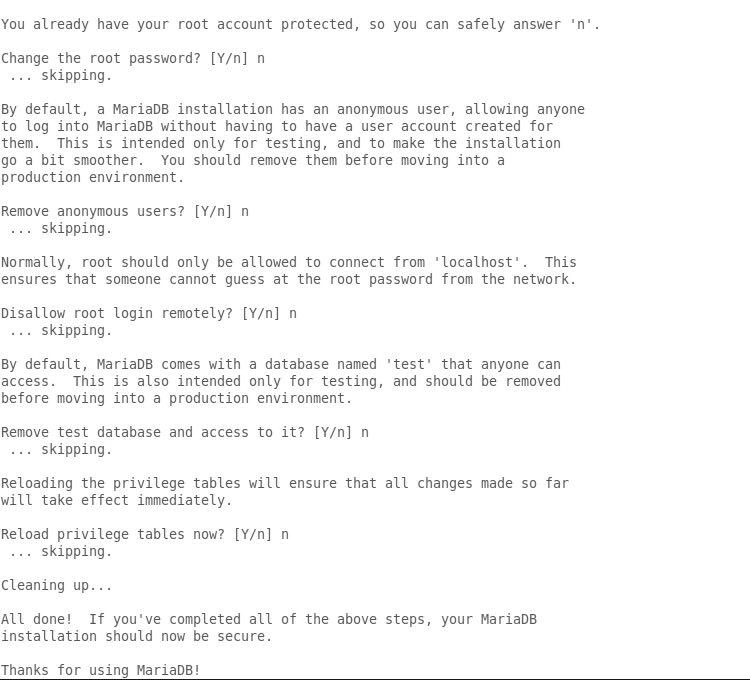


* Realizamos las configuraciones del usuario root de MariaDB

ejecutando el siguiente comando

→ /usr/bin/mysql\_secure\_installation





##### PASO 4:

* Vamos a probar que todo funciona correctamente, para eso

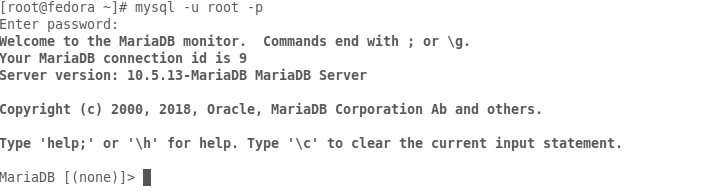
desde la consola vamos a testear el motor de la base de datos

usando el siguiente comando:

→ $ mysql -u root -p

* Luego de ejecutar este comando nos pedirá el password del root que

colocamos en el anterior comando.



##### PASO 5:

* Para probar que PHP funciona junto a APACHE nos basta

simplemente con crear un archivo .php llamado index.php y lo

guardamos en /var/www/html/prueba/index.php, el archivo

contendrá el siguiente código:



→ <?php

phpinfo();

→ ?>



* Para ejecutarlo simplemente vamos a ejecutar en un

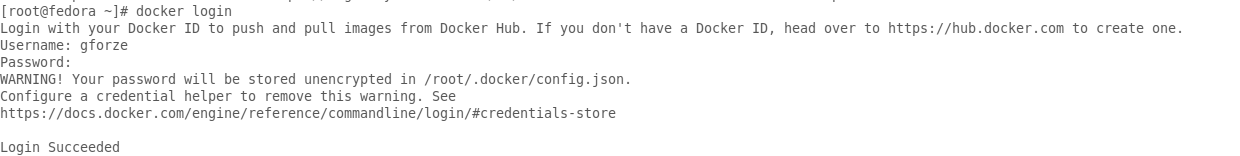
navegador web localhost/prueba

### Manual de Instalación de MySQL/MariaDB junto a sus requisitos para trabajar con Docker.

#### Paso 0:

En este paso procederemos a logear con nuestra cuenta de [Docker Hub](https://hub.docker.com/) , utilizando el siguiente comandos:

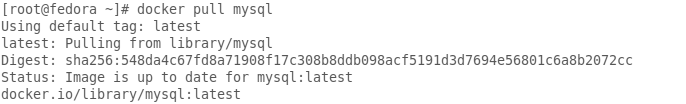
-$ docker login



#### Paso 1:

Vamos a descargar la última imagen oficial de MySQL desde docker hub. Para eso ejecutamos:

-$ docker pull mysql



#### Paso 2:

Ahora que tenemos mysql instalado en docker vamos a crear un volumen. En este caso vamos a usar el mismo guión creado previamente.

-$ docker volume create proyecto\_mysql



#### 

#### Paso 3:

Con el siguiente comando podemos visualizar los contenedores con sus correspondientes nombres.

-$docker ps



#### Paso 4:

Ejecutamos el contenedor

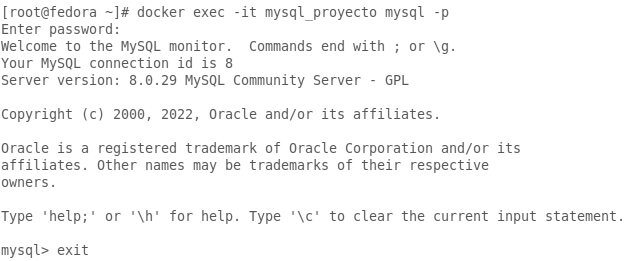
-$ docker run -d -p 33060:3306 --name mysql\_proyecto -e MYSQL\_ROOT\_PASSWORD=1234 --mount src=mysql\_proyecto,dst=/var/lib/mysql mysql



#### Paso 5:

Para acceder al contenedor ejecutamos:

-$ docker exec -it mysql\_proyecto mysql -p



#### 

#### Paso 6:

Estos comandos se usan básicamente para iniciar el contenedor, para detener el contenedor y eliminar el proceso que ejecutamos.

-$ docker start mysql\_proyecto

-$ docker stop mysql\_proyecto

-$ docker rm mysql\_proyecto

