

**SISTEMA INFORMATICO DE GESTION DE ENTRENAMIENTO**

Team Tryhard 3BC

Sistemas Operativos III

**Primera entrega**

**Docente**  **Integrantes**

Daniel Garcia Mathias Diaz

Alejo Tabares

Cristian Carpio

Brayan Rivero

Dylan Arrua

Montevideo **15 de julio de 2024**

[**Introducción 3**](#_jhvggmtxb9p9)

[**Marco Teórico 3**](#_63hh4732fjbg)

[**Objetivos 5**](#_l5bxwqqbl7fu)

[**Desarrollo 6**](#_h6uruk6hj9d4)

[Análisis Roles de usuario - Gestión de usuarios 6](#_28viiedjwhmb)

[Relevamiento y justificación del Sistema Operativo 8](#_6pk0928pai3)

[Manual de instalación del Sistema Operativo Fedora Server 40 8](#_3z1w72299cbm)

[Manual de Instalación de Docker 16](#_q8l0kcyotvkv)

[Manual de Instalación de LAMP 17](#_lyxvkuhfbicu)

[Configuración servicios ssh 20](#_v8s9myz179ge)

[**Anexos 21**](#_g7deet1hxkl8)

[**Bibliografia 21**](#_20cnyeqncfsv)

# Introducción

La presente documentación aborda varios aspectos fundamentales para la configuración y gestión de un servidor. Siguiendo los lineamientos y requisitos establecido por el docente de la materia Sistemas Operativos III, se realiza un estudio detallado de los diferentes roles de los usuarios del servidor, también se lleva a cabo el relevamiento y la justificación del Sistema Operativo a utilizar, tanto en las terminales de los usuarios como en el servidor.

Asimismo, se incluye un manual de instalación del Sistema Operativo en el servidor, contemplando opciones como SUSE Linux Enterprise Server, Red Hat Enterprise Linux, Fedora Server o CentOS. Se procederá con la instalación de Docker dentro de una máquina virtual con el sistema operativo seleccionado, específicamente para el desarrollo web.

Posterior a esto, se instalará un servidor LAMP en la máquina virtual, utilizando Docker. También se elaborará un manual de instalación de MySQL/MariaDB, detallando sus requisitos para trabajar con Docker en Desarrollo Web.

Finalmente, se presentará la primera versión del script de gestión de usuarios y grupos del sistema operativo, proporcionando una herramienta esencial para la administración eficiente del sistema.

# Marco Teórico

Esta documentación se basa en varios conceptos clave relacionados con la configuración y gestión de servidores, el uso de sistemas operativos Linux, y la implementación de contenedores Docker para el desarrollo web.

**1- Configuración y Gestión de Servidores**

Es un punto fundamental en el campo de la administración de sistemas. Implica la instalación, configuración y mantenimiento de hardware y software de servidor para asegurar que las aplicaciones y servicios se ejecuten de manera eficiente y segura. Este proceso incluye la gestión de recursos, administración de usuarios y la implementación de políticas de seguridad.

**2- Roles de Usuarios en el Servidor**

La gestión de usuarios y sus roles es crucial para garantizar la seguridad y eficiencia en un entorno de servidor. Los usuarios pueden tener diferentes niveles de acceso y permisos, lo que les permite realizar diversas tareas según su rol. Los roles comunes incluyen administradores, desarrolladores y usuarios finales. La correcta asignación y gestión de estos roles ayuda a proteger los recursos del sistema y a mantener un entorno operativo seguro.

**3- Elección del Sistema Operativo**

La elección del sistema operativo para un servidor depende de varios factores, como la estabilidad, la seguridad, el soporte y las necesidades específicas de la aplicación. La elección adecuada del sistema operativo puede mejorar significativamente el rendimiento y la seguridad del servidor.

**4- Implementación de DOCKER**

Docker es una plataforma de software que permite crear, probar y desplegar aplicaciones rápidamente mediante el uso de contenedores. Los contenedores son entornos aislados que contienen todo lo necesario para ejecutar una aplicación, lo que facilita la portabilidad y la escalabilidad. Docker se utiliza ampliamente en el desarrollo web para crear entornos de desarrollo consistentes y replicables.

**5- Servidor LAMP**

Un servidor LAMP es un conjunto de software de código abierto utilizado para hospedar aplicaciones web. LAMP es un acrónimo que representa Linux (sistema operativo), Apache (servidor web), MySQL/MariaDB (sistema de gestión de bases de datos) y PHP (lenguaje de programación). La implementación de un servidor LAMP en un contenedor Docker permite la gestión más flexible y eficiente de las aplicaciones web.

**6- Gestión de Usuarios y Grupos en Linux**

La gestión de usuarios y grupos en sistemas Linux es esencial para mantener la seguridad y la organización del sistema. Los scripts de gestión de usuarios y grupos pueden facilitar esta tarea, permitiendo la creación, modificación y eliminación de usuarios y grupos de manera eficiente.

**7- Configuración servicios SSH**

SSH, o secure shell, es un protocolo TCP que permite la administración remota de equipos mediante terminales, utilizando una arquitectura cliente-servidor (donde el cliente es el administrador y el servidor el administrador) y poniendo un especial énfasis en la seguridad.

# Objetivos

El objetivo general del proyecto es desarrollar un sistema web que permita a diferentes instituciones deportivas gestionar sus clientes, ya sean deportistas o pacientes de fisioterapia. Para ello Sistemas Operativos plantea objetivos más específicos, estos son:

* Estudio de los diferentes roles de los usuarios del servidor.
* Relevamiento y justificación del Sistema Operativo a utilizar tanto en las terminales de los usuarios como en el servidor.
* Manual de instalación del Sistema Operativo en el servidor, por ejemplo, SUSE Linux Enterprise
* Server, Red Hat Enterprise Linux, Fedora Server o CentOS.
* Instalación de Docker dentro de la máquina virtual con el sistema operativo a utilizar. (solamente Desarrollo Web).
* Instalación de un servidor LAMP en la máquina virtual (usando docker para DESARROLLO WEB y local para Desarrollo y Soporte)
* Manual de Instalación de MySQL/MariaDB junto a sus requisitos para trabajar con Docker para Desarrollo Web e instalación local para Desarrollo y Soporte.
* Primera versión del script de gestión de usuarios y grupos del sistema operativo.

# Desarrollo

## Análisis Roles de usuario - Gestión de usuarios

Se comenzó trabajando con el estudio de los diferentes roles de los usuarios del servidor y se pensó de tal manera que la gestión de usuarios y recursos en el sistema operativo sea eficiente. Además este análisis sirve para poder diferenciar bien los permisos que los distintos usuarios y grupos tendrán dentro del sistema a crear.

| ROL | NOMBRE | GRUPO | PERMISOS |
| --- | --- | --- | --- |
| Administrador | admin | group0, group1, wheel | Tiene permisos de lectura, escritura y ejecución sobre todo es “ROOT” |
| Gestión y uso de docker | manager | group0, group2, gdocker | tiene todos los permisos sobre su script |
| inspector de logs | inspector | group0, group2, glogs | el inspector tiene permisos RWX (lectura, escritura y ejecución) sobre su script |
| respaldo | backup | group0, group2, gbackup | El backup tiene RWX sobre su script |

Estos roles están diseñados para la gestión eficiente de usuarios y recursos en el sistema operativo. Cada rol tiene asignado un nombre, pertenece a ciertos grupos y tiene permisos específicos sobre sus respectivos scripts o tareas asignadas. Esto asegura que cada usuario tenga acceso solo a las funciones y recursos necesarios para llevar a cabo sus responsabilidades, siguiendo el principio de privilegios mínimos.

El administrador, con el rol de "Administrador", tiene el mayor nivel de acceso y permisos sobre el sistema operativo. Puede realizar tareas de gestión de usuarios, firewall, SSH, administrar el servidor LAMP en Docker, ver logs y realizar respaldos. Este usuario pertenece a los grupos group0, group1 y wheel, lo que le otorga permisos adicionales sobre ciertos recursos.

El usuario responsable de la gestión y uso de Docker tiene el rol de "Gestión y uso de Docker" y pertenece a los grupos group0, group2 y gdocker. Este usuario tiene todos los permisos sobre su script, lo que le permite manejar Docker de manera eficiente y segura.

El Inspector de Logs tiene acceso a los registros del sistema y del servidor utilizando un script designado. Este usuario pertenece a los grupos group0, group2 y glogs, y tiene permisos de lectura, escritura y ejecución sobre su script para poder analizar los registros de manera efectiva.

Por último, el usuario encargado de los respaldos tiene el rol de "Respaldo" y pertenece a los grupos group0, group2 y gbackup. Este usuario tiene permisos de lectura, escritura y ejecución sobre su script, lo que le permite realizar y gestionar los respaldos del sistema de manera adecuada.

## 

## Relevamiento y justificación del Sistema Operativo

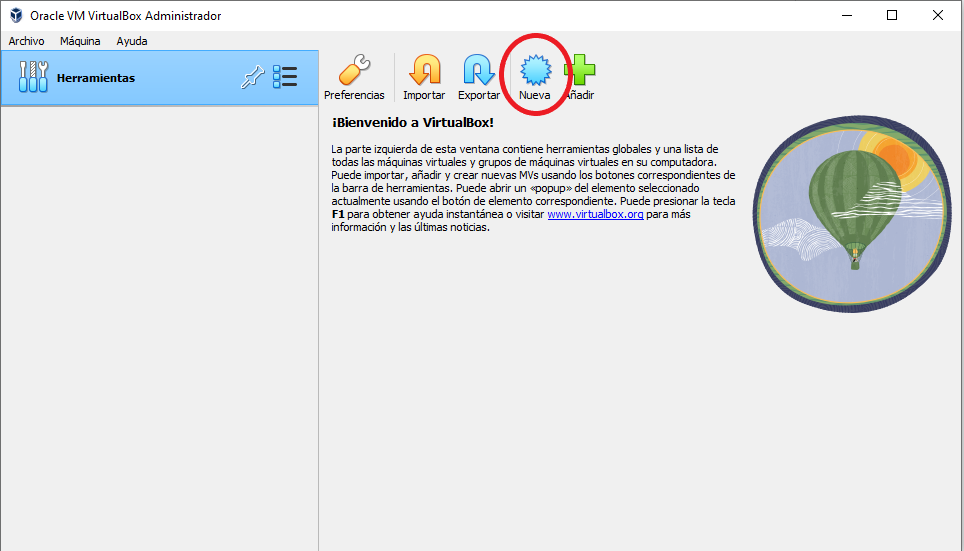
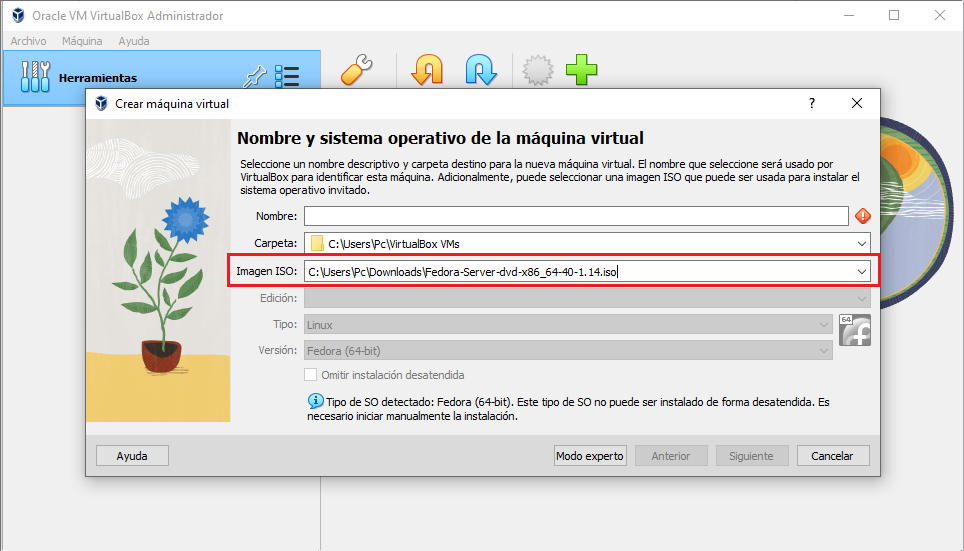
Siguiendo la recomendación de nuestro profesor en Sistemas Operativos, Daniel García, optamos por utilizar Fedora Server. Este sistema operativo de servidor, respaldado por la comunidad y de ciclo de vida corto, ofrece a los administradores de sistemas experimentados la posibilidad de aprovechar las últimas tecnologías disponibles en el ámbito del código abierto, sin importar su experiencia previa en otros sistemas operativos.

Fedora Server, respaldado por Red Hat, cuenta con una amplia comunidad de usuarios y se mantiene constantemente a la vanguardia de la innovación. Es utilizado principalmente por desarrolladores como un entorno de desarrollo y prueba, donde pueden experimentar con los últimos paquetes de software antes de su distribución en otros sistemas.

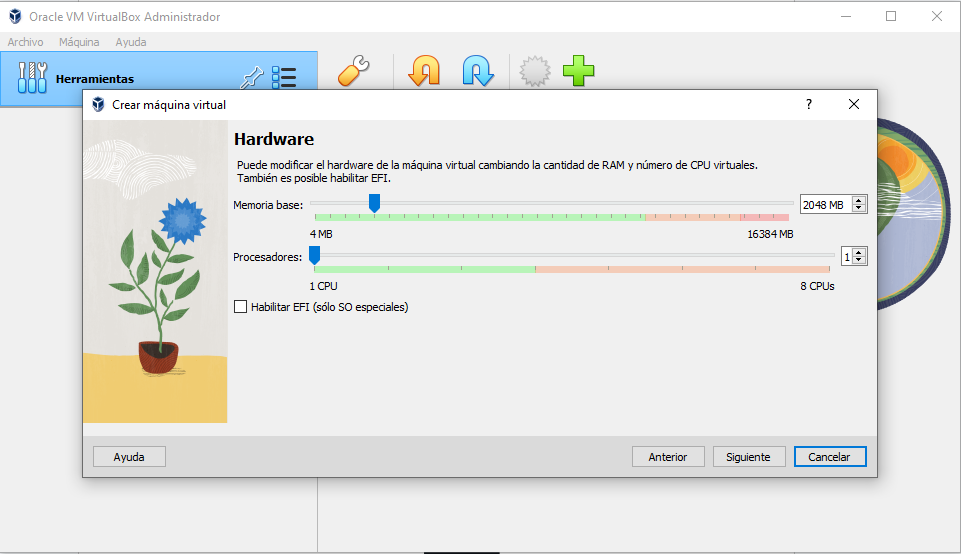
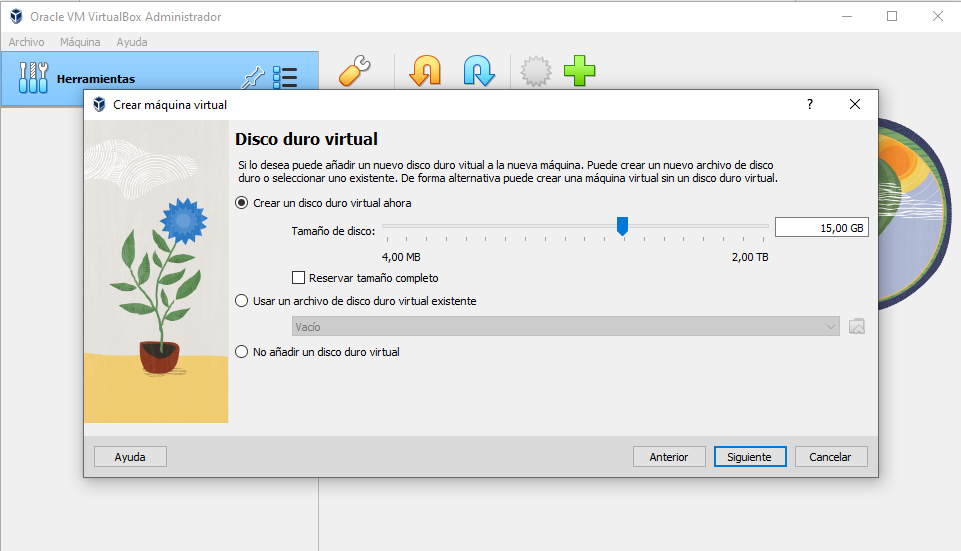
La elección de Fedora Server no solo nos brinda acceso a un conjunto robusto de herramientas y servicios para nuestras necesidades de servidor, sino que también nos sitúa en el epicentro de la innovación en el mundo del código abierto. Con esta plataforma, estamos seguros de estar preparados para abordar cualquier desafío tecnológico que se nos presente, mientras nos mantenemos al día con las últimas tendencias y avances en el panorama informático.

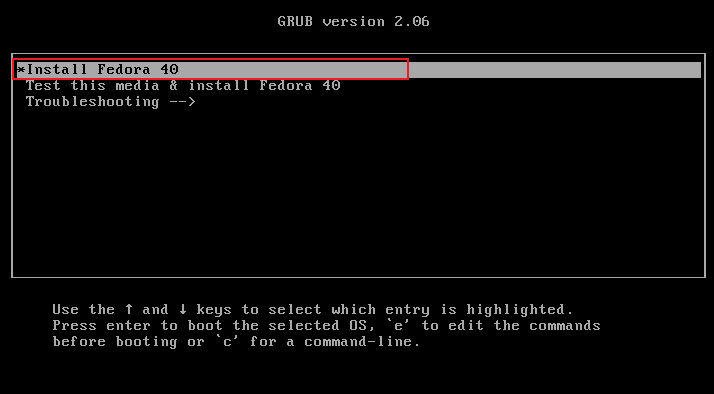
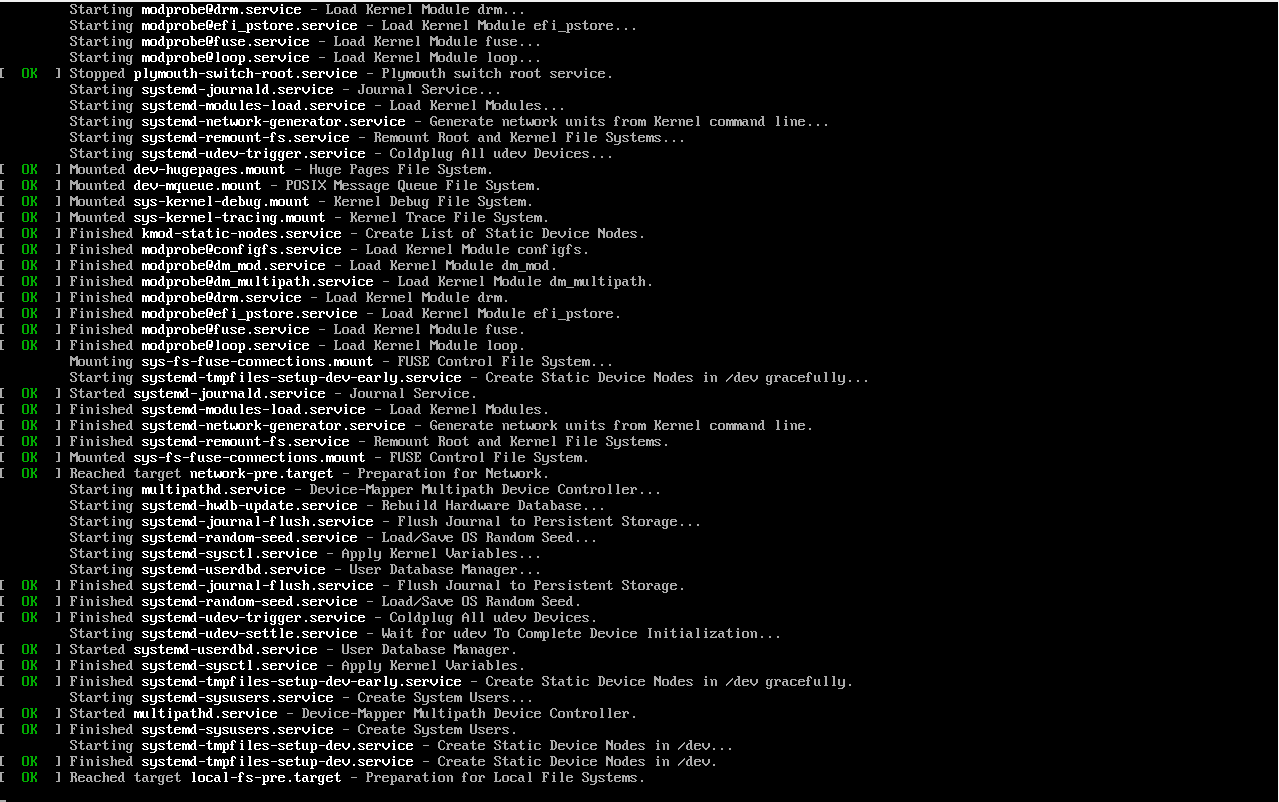
## Manual de instalación del Sistema Operativo Fedora Server 40

Para instalar el sistema operativo debemos descargar la imagen de Fedora Server en la versión 40. Dicha imagen se encuentra disponible en <https://fedoraproject.org/es/server/download>. Una vez teniendo la imagen debemos de instalar virtualbox a través del siguiente link: <https://www.virtualbox.org/wiki/Downloads>. Al abrir esta herramienta haremos click en el icono de “Nueva”.

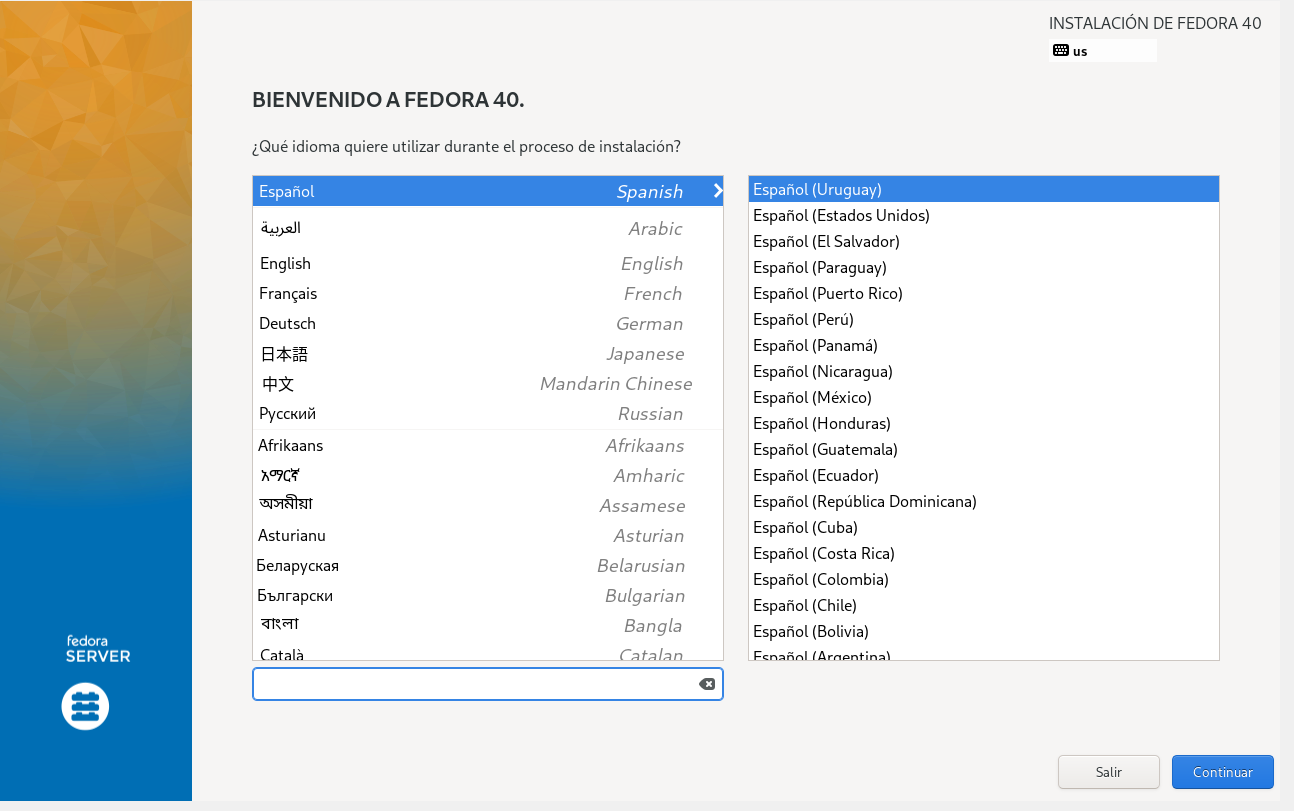
En ese momento se abrirá una nueva ventana donde debemos de seleccionar la imagen del sistema operativo seleccionado previamente.

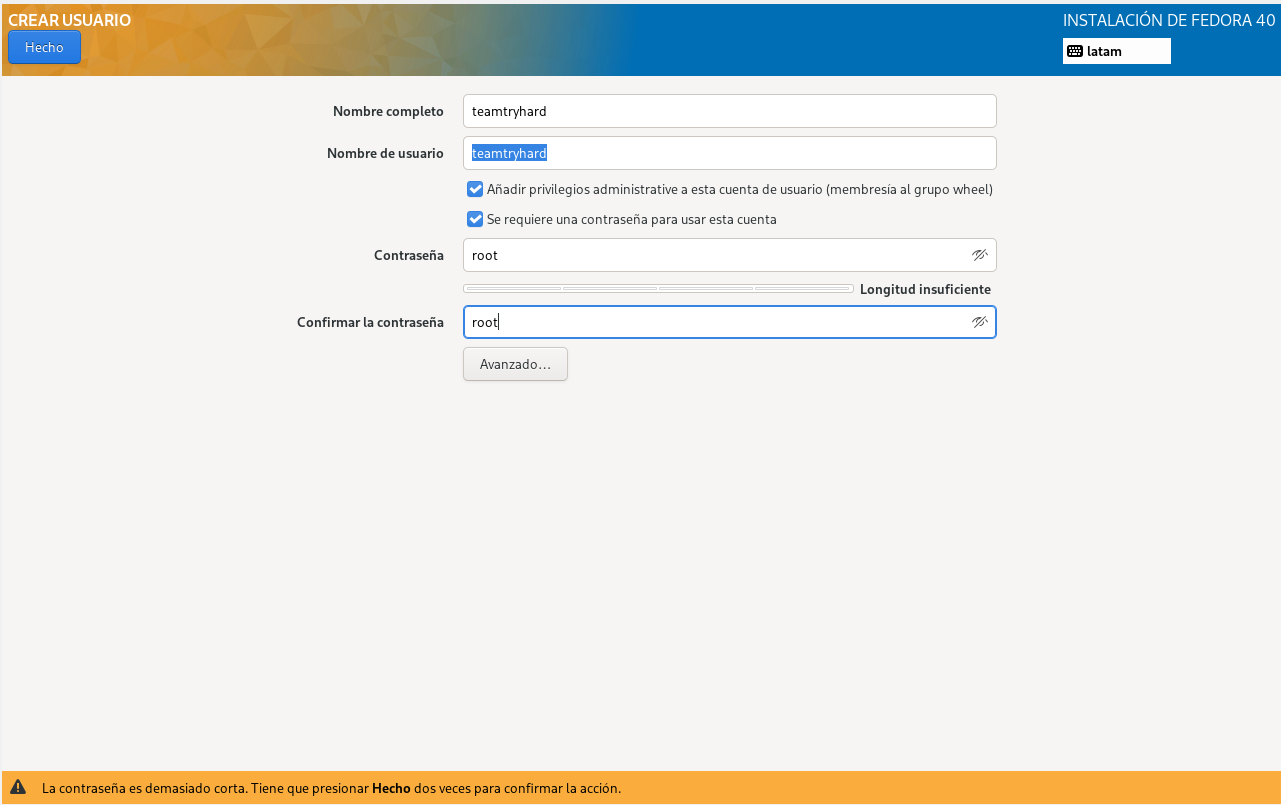
Una vez ahí debemos de asignarle MEMORIA RAM y cantidad de CPUs a utilizar en la máquina virtual. Posterior a esto debemos asignar qué partición del disco duro utilizará y darle una cantidad X de GBs para el sistema.

Luego damos a siguiente e iniciamos la máquina virtual. Cuando se inicie, el sistema abrirá una terminal y debemos de movernos con las flechas del teclado y hacer ENTER en la opción “Install Fedora 40”.

Una vez dado “Enter” el sistema ejecutará distintas acciones para realizar la instalación del Sistema Operativo, esto se vé algo así: 

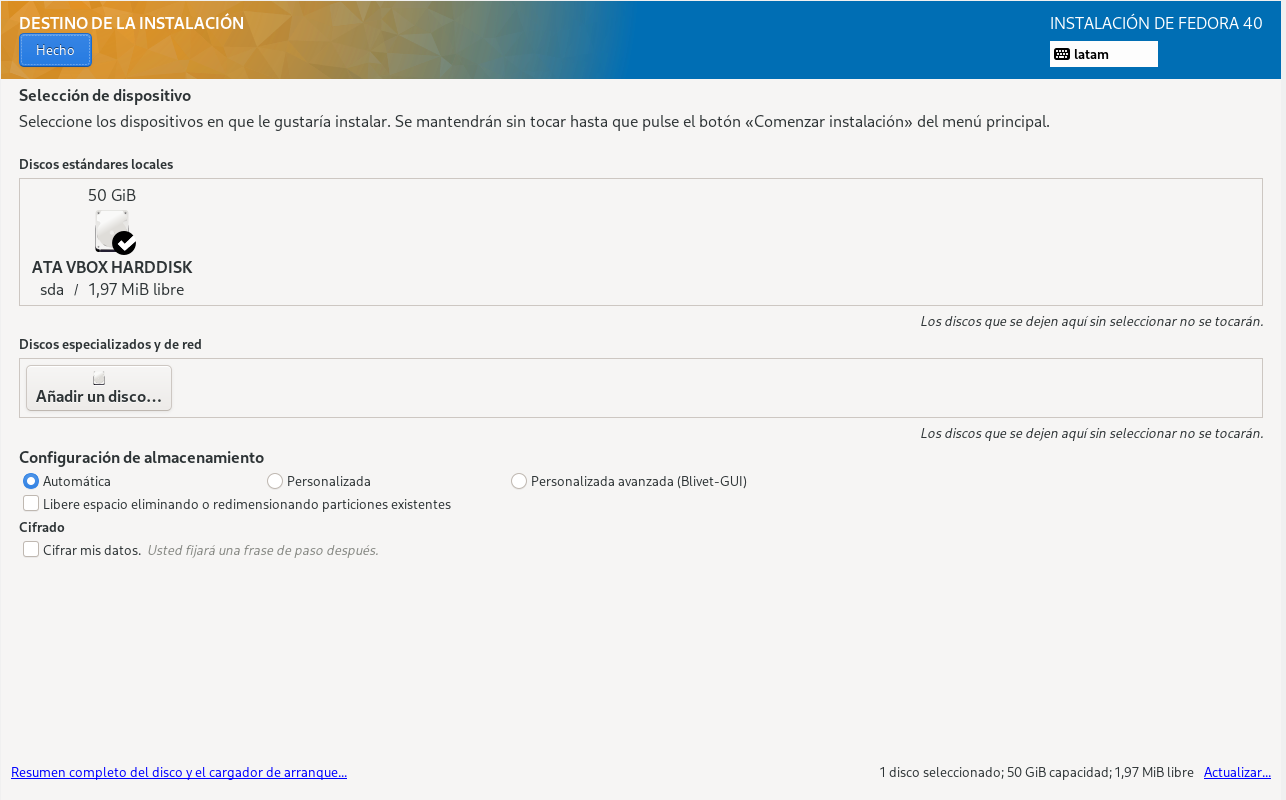
Posterior a esto se abrirá una pantalla donde seleccionaremos el país en el que estamos ubicados y el idioma que hablamos.

Una vez hecho esto, crearemos el usuario, para ello debemos ir a la opción “Creación de usuario”. 

Aquí crearemos el usuario, asignándole un nombre de usuario y una contraseña.

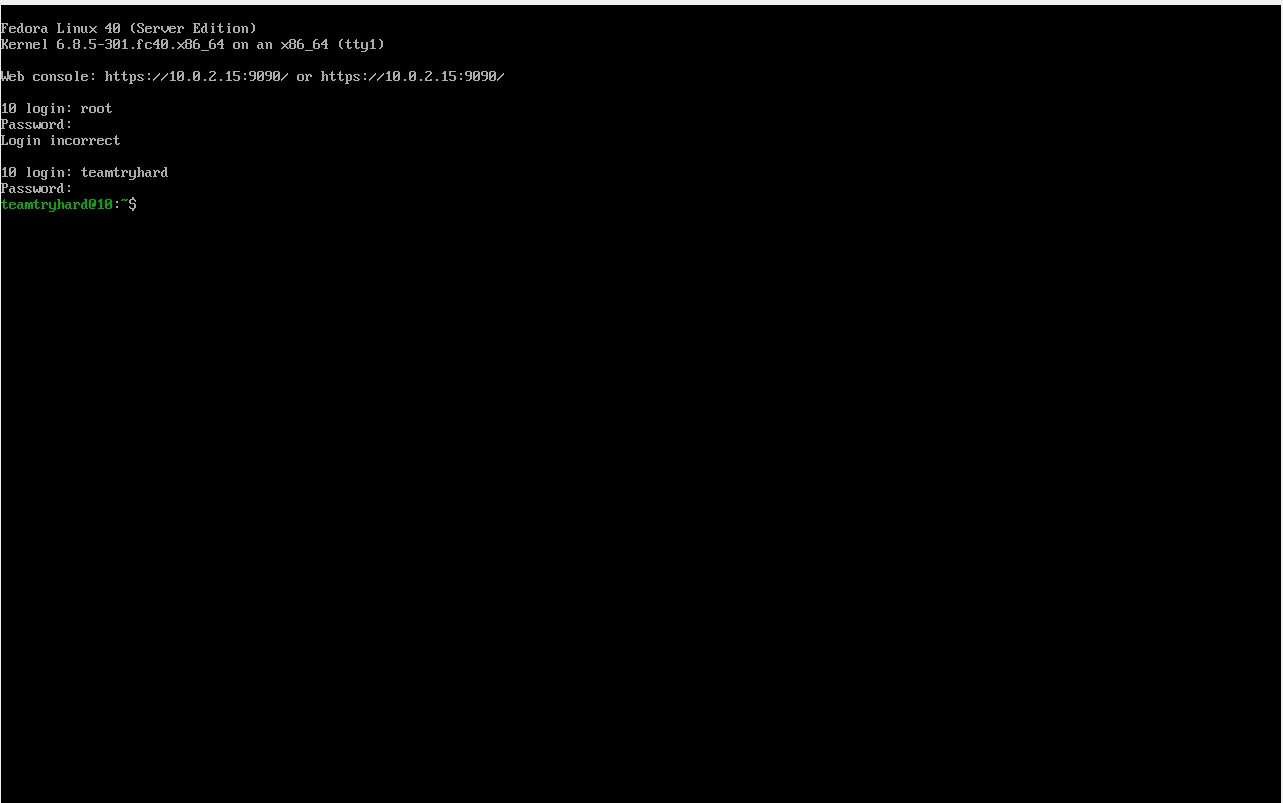
Posterior a esto iremos a la opción “Destino de la instalación” y allí seleccionaremos el disco duro que le asignamos anteriormente en VirtualBox.

# 

Hacemos clic en “Hecho” y en pantalla principal damos clic en “Comenzar la instalación”.

Una vez en este punto esperamos a que el sistema se instale y damos a “Reinicio del sistema”.

# 

Iniciamos la máquina virtual de vuelta e iniciamos sesión en el sistema.

## 

## Manual de Instalación de Docker

Se comienza abriendo el Sistema FEDORA SERVER 40, para luego continuar con la instalación de DOCKER. Para ello comenzaremos con la instalación del paquete para configurar el repositorio.

Se comienza instalado el paquete que que proporciona los comandos para administrar sus repositorios DNF:

1. **sudo dnf -y install dnf-plugins-core**
2. **sudo dnf config-manager --add-repo** [**https://download.docker.com/linux/fedora/docker-ce.repo**](https://download.docker.com/linux/fedora/docker-ce.repo)

Luego se instala el motor Docker, para ello se ejecuta:

1. **sudo dnf install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin**

Dicho comando instala Docker, pero no inicia Docker. También crea un dockergrupo, sin embargo, no agrega ningún usuario al grupo de forma predeterminada.

Para iniciar docker utilizaremos:

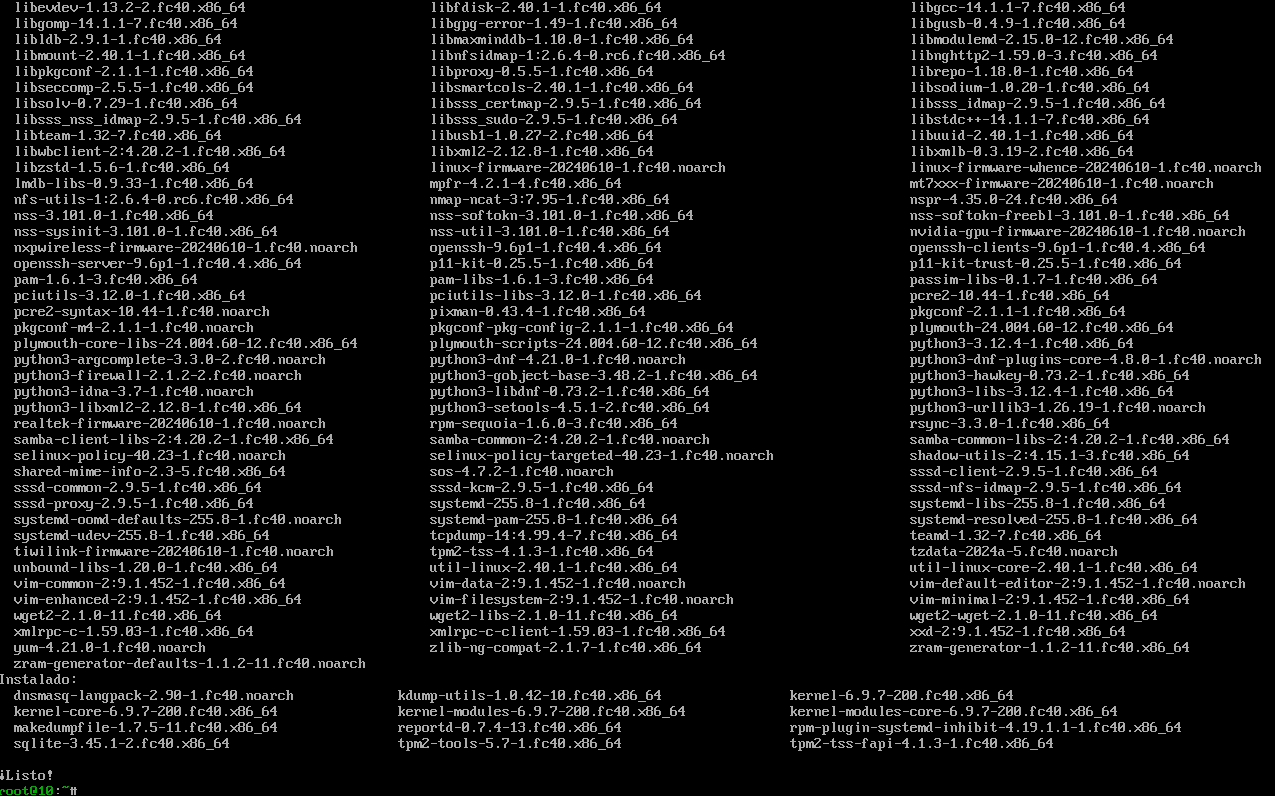
1. **sudo systemctl start docker**

Verifique que la instalación de Docker Engine sea exitosa ejecutando la hello-world imagen para comprobar que se haya instalado correctamente.

1. **sudo docker run hello-world**

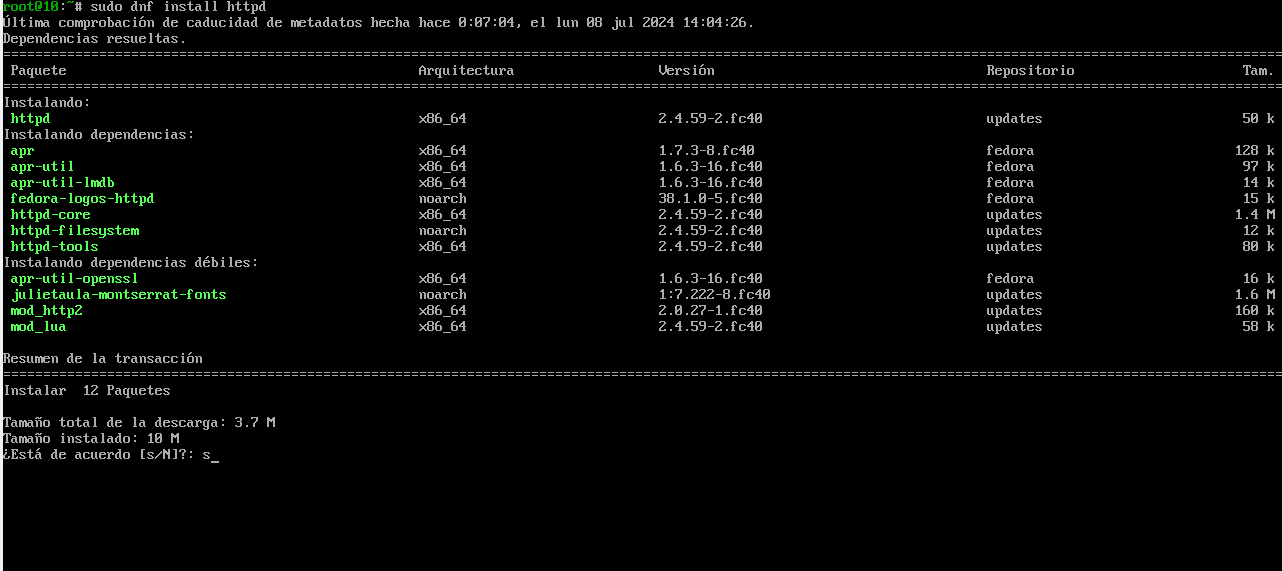
## Manual de Instalación de LAMP

1. Antes de comenzar con la instalación de LAMP, actualice los repositorios de paquetes del sistema y los paquetes instalados a sus últimas versiones usando el siguiente comando:

sudo dnf update

2. Instale el servidor web Apache usando el siguiente comando:

sudo dnf install httpd



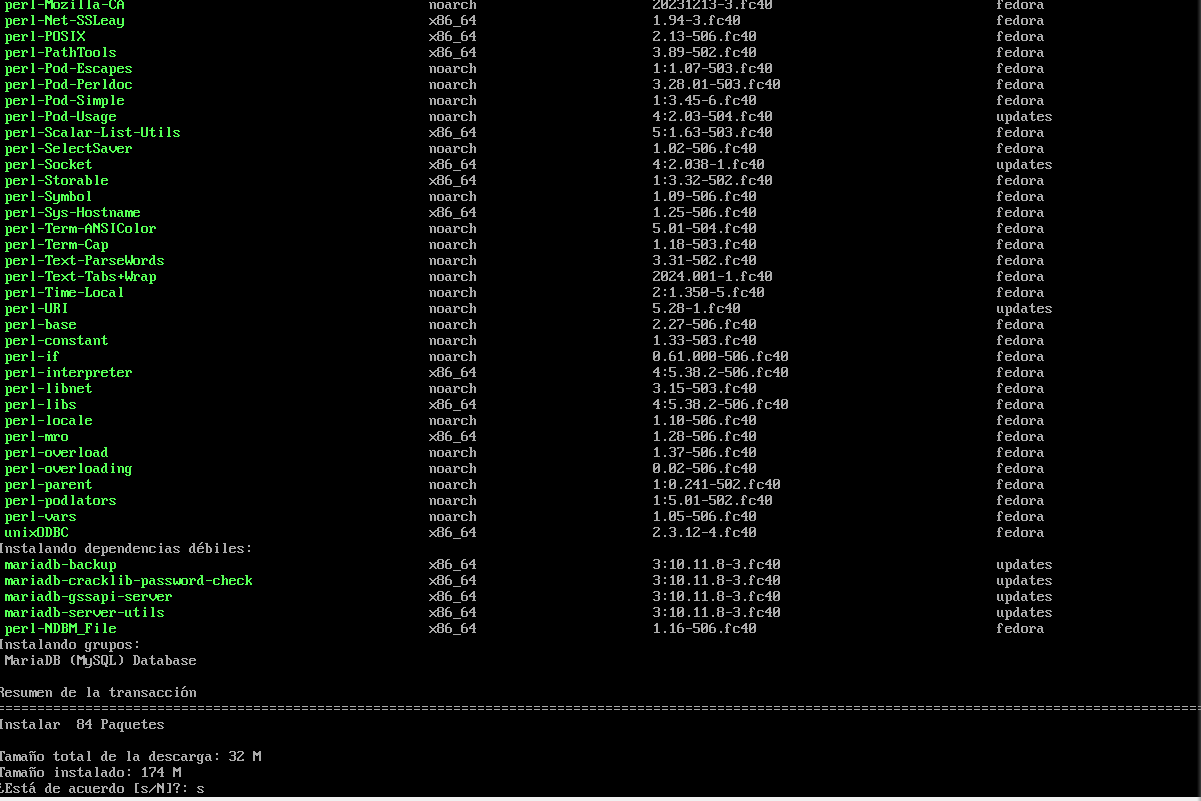
3. Iniciar y habilitar Apache: una vez que Apache esté instalado, inicie el servicio y habilítelo para que se inicie en el arranque usando los siguientes comandos:

sudo systemctl start httpd

sudo systemctl enable httpd

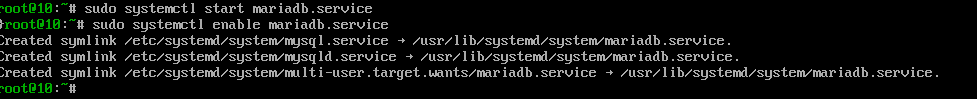


4. Instalar MySQL: Instale el servidor de base de datos MySQL usando el siguiente comando:

sudo dnf install @mysql

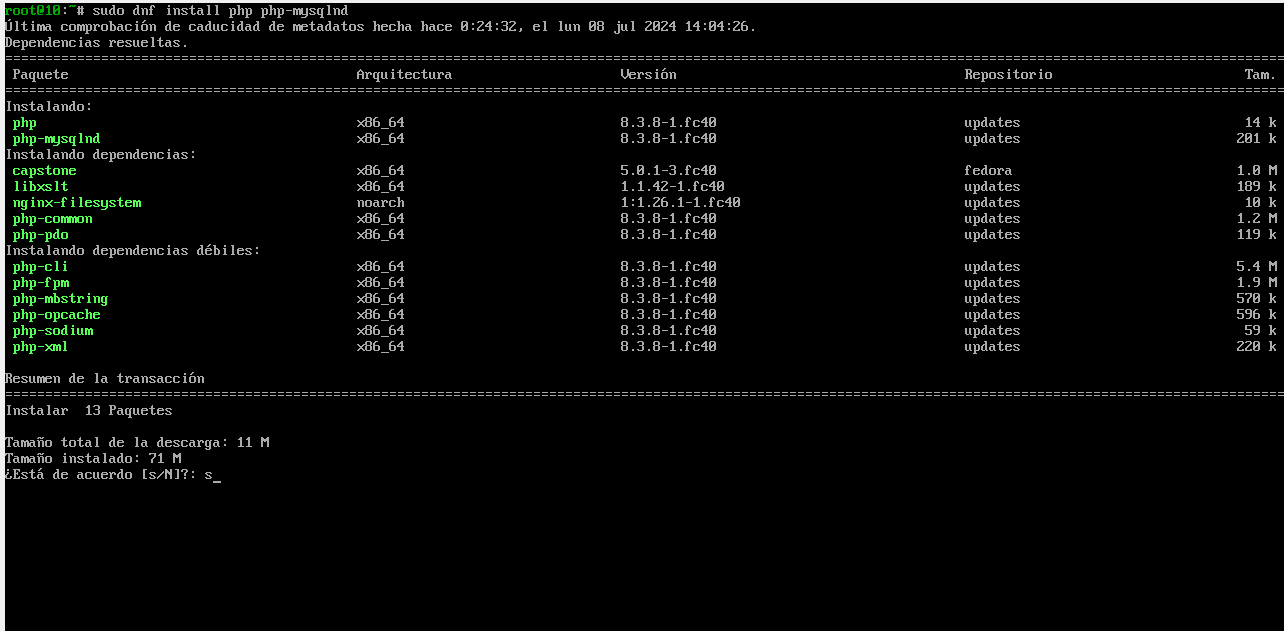
5. Iniciar y habilitar MySQL: inicie el servicio MySQL y habilítelo para que se inicie en el arranque usando los siguientes comandos:

sudo systemctl start mariadb.service

sudo systemctl enable mariadb.service

6. Instale PHP: instale PHP y las extensiones necesarias para conectarse con Apache y MySQL usando el siguiente comando:

sudo dnf install php php-mysqlnd



7. Reinicie Apache: después de instalar PHP, reinicie el servicio Apache para habilitar la compatibilidad con PHP:

sudo systemctl restart httpd

## Configuración servicios ssh

conexión mediante terminal

ssh root@192.168.170.3:22

comando usuario @ ip : puerto

apt -get install openssh-server -y

comando para instalar

systemctl start sshd

ejecutar la instalación

systemctl power off

apaga la máquina

servicio-sshd=$(systemctl is-active sshd)

echo 1) “Agregar Usuario”

echo 2)”Eliminar Ususuario”

if[[$servicio-sshd == active]];then

echo “3) Desactivar SSH”

else

echo “3) Activar SSH”0

echo “ 4) Salir”

case $op in

3) if [[$servicio-sshd == active]]; then

systemctl stop sshd

else

systemctl start sshd

fi;;

# Anexos

# Bibliografia

AWS. *¿Qué es DOCKER?*. Recuperado el 20 de Junio de 2024, de <https://aws.amazon.com/es/docker/>

Docker.docs. *Instalar Docker Engine en Fedora*. Recuperado el 20 de Junio de 2024, de <https://docs.docker.com/engine/install/fedora/>

Fedora. (2024). *Descargar Fedora Server 40.* Recuperado el 20 de Junio de 2024, de <https://fedoraproject.org/es/server/download>

Informatecdigital. (2023). *Fedora: Características y Funciones.* Recuperado el 20 de Junio de 2024, de

<https://informatecdigital.com/sistemas/fedora-caracteristicas-y-funciones/>

Nothan, J. (2024). *Configuración de LAMP en Fedora Server.* Recuperado el 21 de Junio de 2024, de

<https://medium.com/@nothanjack/lamp-stack-setup-on-fedora-39-command-line-essentials-fda3be79edd8>

Orio, G. (2024). *SSH y configuración de tuneles.* Recuperado el 2 de Julio de 2024, de

<https://www.strsistemas.com/blog/ssh-y-configuracion-de-tuneles#:~:text=%C2%BFQu%C3%A9%20es%20SSH%3F,especial%20%C3%A9nfasis%20en%20la%20seguridad>