



S.I.G.P.D.

Sistemas Operativos

CodeFlow

Rol	Apellido	Nombre	C.I	Email
Coordinador	Manzinalli	Felipe	5.714.301-9	felipe.manzinalli@estudiante.ceibal.edu.uy
Sub-Coordinador	Suarez	Pablo	5.656.677-9	psuarez@estudiante.ceibal.edu.uy
Integrante 1	Maldonado	Agustin	5.617.059-8	ezequiel.maldonado@estudiante.ceibal.edu.uy
Integrante 2	Diez	Daniel	5.547.065-6	danieldiez05@gmail.com

Docente: Borba, Gabriela

**Fecha de
culminación**

15/9/2025

SEGUNDA ENTREGA

ÍNDICE

Relevamiento de datos.....	2
Requisitos.....	2
Variantes descartadas.....	2
Justificación.....	2
Conclusión.....	2
Manual de instalación.....	3
Requisitos.....	3
Apache:.....	3
Mysql:.....	4
PHP:.....	5
Comandos útiles.....	5
Actualizar paquetes:.....	5
Respaldar base de datos.....	5
Instalacion y configuracion de red en ubuntu server.....	6
Objetivo:.....	6
Procedimiento:.....	6
Configuración de los puertos:.....	7
Gestion y administracion de paquetes del sistema operativo.....	9
Objetivo:.....	9
Servidor Web.....	9
Motor de Base de Datos.....	9
Lenguaje de servidor.....	9
Configuración del servicio SSH (Ubuntu Server).....	13
Instalamos el servicio SSH.....	13
Verificación de estado SSH.....	13
Reinicio del servicio SSH.....	14
Definición de medio de alta disponibilidad.....	15
Medios de respaldo.....	15
Alta disponibilidad.....	15
Aplicación al proyecto.....	15
Conclusión.....	16

Relevamiento de datos

Este informe tiene el objetivo de analizar y justificar la selección del sistema operativo mas adecuado para implementar en nuestro proyecto. Este mismo debera dar soporte a las aplicaciones definidas como mysql, phpmyadmin, apache con soporte php y openssh, garantizando nos estabilidad, seguridad y escalabilidad

Requisitos

- Compatibilidad con aplicaciones: debe soportar bases de datos (mysql), servidores web (Apache) y lenguaje como PHP
- Costo: este sistema operativo y las aplicaciones que se instalaran no tendrán un costo
- Estabilidad y seguridad: Actualizaciones periódicas además de un buen manejo de usuarios y permisos
- Soporte comunitario: muy amplia cantidad de usuarios utilizando el sistema pudiéndose dar soluciones a problemas concretos
- Escalabilidad: Capacidad de ampliar nuestros servicios o crecer en recursos sin afectar el rendimiento

Variantes descartadas

- Fedora Server: sistema moderno, seguro y con paquetes actualizados. Desventaja: ciclo de vida corto, requiere actualizar seguido.
- CentOS Stream: muy estable, derivado de Red Hat, usado en entornos empresariales. Desventaja: ahora recibe cambios frecuentes, lo que puede afectar la estabilidad.
- Linux Mint: fácil de usar y basado en Ubuntu, ideal para escritorio. Desventaja: no está pensado como sistema de servidor.

Justificación

Seleccionamos ubuntu server como sistema operativo ya que cuenta con ciertas ventajas las principales son:

- Costo cero en licencias, lo que reduce gastos del proyecto.
- Estabilidad comprobada en entornos de producción.
- Seguridad reforzada, con actualizaciones frecuentes y gran control de permisos.
- Flexibilidad para instalar software requerido sin limitaciones.
- Amplia documentación y foros de ayuda en línea.

Conclusión

En base al relevamiento realizado llegamos a la conclusión de que ubuntu server es la opción más adecuada para la implementación del servidor de nuestro proyecto ya que nos garantiza

Manual de instalación

El servidor LAMP (Linux, Apache, Mysql y PHP) permite crear un servidor web con soporte para aplicaciones dinámicas. En este manual se detalla como instalar el servidor en “Ubuntu server 22.04”

Requisitos

- Equipo con Ubuntu Server 22.04 instalado.
- Usuario con privilegios sudo.
- Conexión a internet estable.

Preparación del sistema

actualizamos los paquetes del sistema

```
root@codeflow:/home/administrador# apt install apache2
```

Instalación de los componentes

Apache:

instalación

```
root@codeflow:/home/administrador# apt install apache2_
```

verificación

```
root@codeflow:/home/administrador# systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
   Active: active (running) since Thu 2025-09-11 00:41:28 UTC; 5min ago
     Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 3580 (apache2)
    Tasks: 55 (limit: 4605)
   Memory: 5.2M (peak: 5.5M)
      CPU: 61ms
   CGroup: /system.slice/apache2.service
           └─3580 /usr/sbin/apache2 -k start
             └─3583 /usr/sbin/apache2 -k start
               └─3584 /usr/sbin/apache2 -k start
```

Mysql:

instalación:

```
root@codeflow:/home/administrador# apt install mysql-server
```

verificación:

```
root@codeflow:/home/administrador# systemctl status mysql
• mysql.service - MySQL Community Server
   Loaded: loaded (/usr/lib/systemd/system/mysql.service; enabled; preset: enabled)
   Active: active (running) since Thu 2025-09-11 13:48:18 UTC; 1min 1s ago
   Process: 5772 ExecStartPre=/usr/share/mysql/mysql-systemd-start pre (code=exited, status=0/SUCCESS)
   Main PID: 5791 (mysqld)
   Status: "Server is operational"
   Tasks: 37 (limit: 4605)
   Memory: 363.6M (peak: 378.0M)
   CPU: 1.384s
   CGroup: /system.slice/mysql.service
           └─5791 /usr/sbin/mysqld
```

configuración de seguridad

```
root@codeflow:/home/administrador# mysql_secure_installation_
```

después de este comando se deberá seleccionar la configuración deseada a través de un Y(si) o N(no) dependiendo de lo que se quiera realizar se configura de una manera u otra

```
secure enough. Would you like to setup VALIDATE PASSWORD component?

Press y|Y for Yes, any other key for No:

Skipping password set for root as authentication with auth_socket is used by default.
If you would like to use password authentication instead, this can be done with the "ALTER_USER" command.
See https://dev.mysql.com/doc/refman/8.0/en/alter-user.html#alter-user-password-management for more information.

By default, a MySQL installation has an anonymous user,
allowing anyone to log into MySQL without having to have
a user account created for them. This is intended only for
testing, and to make the installation go a bit smoother.
You should remove them before moving into a production
environment.

Remove anonymous users? (Press y|Y for Yes, any other key for No) : n

... skipping.

Normally, root should only be allowed to connect from
'localhost'. This ensures that someone cannot guess at
the root password from the network.

Disallow root login remotely? (Press y|Y for Yes, any other key for No) : n

... skipping.

By default, MySQL comes with a database named 'test' that
anyone can access. This is also intended only for testing,
and should be removed before moving into a production
environment.

Remove test database and access to it? (Press y|Y for Yes, any other key for No) : y
- Dropping test database...
Success.

- Removing privileges on test database...
Success.

Reloading the privilege tables will ensure that all changes
made so far will take effect immediately.

Reload privilege tables now? (Press y|Y for Yes, any other key for No) : y
Success.

All done!
root@codeflow:/home/administrador# _
```

PHP:

Instalar PHP y los módulos:

```
root@codeflow:/home/administrador# apt install php libapache2-mod-php php-mysql -y_
```

Verificamos la versión (8.3.6) :

```
root@codeflow:/home/administrador# php -v
PHP 8.3.6 (cli) (built: Jul 14 2025 18:30:55) (NTS)
Copyright (c) The PHP Group
Zend Engine v4.3.6, Copyright (c) Zend Technologies
    with Zend OPcache v8.3.6, Copyright (c), by Zend Technologies
root@codeflow:/home/administrador#
```

Comandos útiles

Iniciar servicio

- `sudo systemctl start servicio`

Detener un servicio

- `sudo systemctl stop servicio`

Reiniciar servicio

- `sudo systemctl restart servicio`

Servicios: apache2, mysql

Mantenimiento**Actualizar paquetes:**

```
sudo apt update && sudo apt upgrade -y
```

Respaldar base de datos

```
mysqldump -u usuario -p prueba > respaldo.sql
```

Instalacion y configuracion de red en ubuntu server

Objetivo:

Configurar una la red un servidor de ubuntu con el fin de permitir que tanto local como a través de internet exista una conexión entre equipos

Procedimiento:

Instalamos una utilidad de red

```
root@codeflow:/home/administrador# apt install net-tools_
```

Entramos a la carpeta /etc/netplan y editamos el archivo .yaml que se aloja ahí

Esta es la configuración predeterminada

```
GNU nano 7.2
network:
  version: 2
  ethernets:
    enp0s3:
      dhcp4: true
```

y esta la aplicada a nuestra red

```
GNU nano 7.2
network:
  ethernets:
    enp0s3:
      dhcp4: no
      addresses:
        - 192.168.0.50/24
      routes:
        - to: default
          via: 192.168.0.1
      nameservers:
        addresses:
          - 8.8.8.8
          - 8.8.4.4
```

aplicamos la nueva configuración de red

```
root@codeflow:/etc/netplan# netplan apply
```

Verificamos

```
root@codeflow:/etc/netplan# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:e1:7a:41 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 metric 100 brd 10.0.2.255 scope global dynamic enp0s3
        valid_lft 85145sec preferred_lft 85145sec
    inet6 fe80::a00:27ff:fe01:7a41/64 scope link
        valid_lft forever preferred_lft forever
root@codeflow:/etc/netplan# _
```

Configuración de los puertos:

Verificamos el estado del firewall y en este caso esta desactivado así que lo activamos

```
root@codeflow:/etc/netplan# ufw status
Status: inactive
root@codeflow:/etc/netplan# ufw enable
Firewall is active and enabled on system startup
root@codeflow:/etc/netplan#
```

Añadimos los puertos 22445(SSH), 80(página web pero sin cifrado) y 433(página web segura)

```
root@codeflow:/# ufw allow 22/tcp
Rule added
Rule added (v6)
root@codeflow:/# ufw allow 80/tcp
Rule added
Rule added (v6)
root@codeflow:/# ufw allow 433/tcp
Rule added
Rule added (v6)
root@codeflow:/# _
```

Verificamos que los puertos se hayan abierto


```
root@codeflow:/# ufw allow 22445
Rule added
Rule added (v6)
root@codeflow:/# ufw status
Status: active

To                Action            From
--                -
22/tcp            DENY              Anywhere
80/tcp            ALLOW             Anywhere
433/tcp           ALLOW             Anywhere
22445             ALLOW             Anywhere
22/tcp (v6)       DENY              Anywhere (v6)
80/tcp (v6)       ALLOW             Anywhere (v6)
433/tcp (v6)      ALLOW             Anywhere (v6)
22445 (v6)        ALLOW             Anywhere (v6)

root@codeflow:/#
```

Gestion y administracion de paquetes del sistema operativo

Objetivo:

Instalar, actualizar y eliminar paquetes de software utilizando el gestor de paquetes “APT”, asegurándonos un servidor funcional y actualizado

Servidor Web

- **apache2**

Es el servicio que recibe las peticiones HTTP/HTTPS y entrega páginas al navegador. Popular, estable y con buena documentación.

Motor de Base de Datos

- **mysql-server**

Gestiona bases de datos relacionales, necesarias para almacenar información de usuarios, productos o contenidos.

Lenguaje de servidor

- **php**
- **libapache2-mod-php**
- **php-mysql**

Permite generar páginas dinámicas y conectar el servidor web con la base de datos.

Apache:

Instalación

```
root@codeflow:/home/administrador# apt install apache2_
```

Verificación

```
root@codeflow:/home/administrador# systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
   Active: active (running) since Thu 2025-09-11 00:41:28 UTC; 5min ago
     Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 3580 (apache2)
    Tasks: 55 (limit: 4605)
   Memory: 5.2M (peak: 5.5M)
      CPU: 61ms
   CGroup: /system.slice/apache2.service
           └─3580 /usr/sbin/apache2 -k start
             └─3583 /usr/sbin/apache2 -k start
               └─3584 /usr/sbin/apache2 -k start
```

Mysql:

Instalación:

```
root@codeflow:/home/administrador# apt install mysql-server
```

Verificación:

```
root@codeflow:/home/administrador# systemctl status mysql
● mysql.service - MySQL Community Server
   Loaded: loaded (/usr/lib/systemd/system/mysql.service; enabled; preset: enabled)
   Active: active (running) since Thu 2025-09-11 13:48:18 UTC; 1min 1s ago
   Process: 5772 ExecStartPre=/usr/share/mysql/mysql-systemd-start pre (code=exited, status=0/SUCCESS)
   Main PID: 5791 (mysqld)
   Status: "Server is operational"
   Tasks: 37 (limit: 4605)
   Memory: 363.6M (peak: 378.0M)
   CPU: 1.384s
   CGroup: /system.slice/mysql.service
           └─5791 /usr/sbin/mysqld
```

Configuración de seguridad

```
root@codeflow:/home/administrador# mysql_secure_installation_
```

después de este comando se deberá seleccionar la configuración deseada a través de un Y(si) o N(no) dependiendo de lo que se quiera realizar se configura de una manera u otra

```
secure enough. Would you like to setup VALIDATE PASSWORD component?

Press y|Y for Yes, any other key for No:

Skipping password set for root as authentication with auth_socket is used by default.
If you would like to use password authentication instead, this can be done with the "ALTER_USER" command.
See https://dev.mysql.com/doc/refman/8.0/en/alter-user.html#alter-user-password-management for more information.

By default, a MySQL installation has an anonymous user,
allowing anyone to log into MySQL without having to have
a user account created for them. This is intended only for
testing, and to make the installation go a bit smoother.
You should remove them before moving into a production
environment.

Remove anonymous users? (Press y|Y for Yes, any other key for No) : n

... skipping.

Normally, root should only be allowed to connect from
'localhost'. This ensures that someone cannot guess at
the root password from the network.

Disallow root login remotely? (Press y|Y for Yes, any other key for No) : n

... skipping.
By default, MySQL comes with a database named 'test' that
anyone can access. This is also intended only for testing,
and should be removed before moving into a production
environment.

Remove test database and access to it? (Press y|Y for Yes, any other key for No) : y
- Dropping test database...
Success.

- Removing privileges on test database...
Success.

Reloading the privilege tables will ensure that all changes
made so far will take effect immediately.

Reload privilege tables now? (Press y|Y for Yes, any other key for No) : y
Success.

All done!
root@codeflow:/home/administrador# _
```

PHP:

Instalar PHP y los módulos:

```
root@codeflow:/home/administrador# apt install php libapache2-mod-php php-mysql -y_
```

Verificamos la versión (8.3.6) :

```
root@codeflow:/home/administrador# php -v
PHP 8.3.6 (cli) (built: Jul 14 2025 18:30:55) (NTS)
Copyright (c) The PHP Group
Zend Engine v4.3.6, Copyright (c) Zend Technologies
    with Zend OPcache v8.3.6, Copyright (c), by Zend Technologies
root@codeflow:/home/administrador#
```

Configuración del servicio SSH (Ubuntu Server)

El servicio SSH (Secure Shell) permite la administración remota de un servidor a través de un canal seguro cifrado. El objetivo es instalar y configurar el SSH en una máquina virtual

Instalamos el servicio SSH

-apt install openssh-server

Verificación de estado SSH

-systemctl status ssh

```
root@codeflow:/# systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; disabled; preset: enabled)
   Active: active (running) since Mon 2025-09-15 04:21:51 UTC; 12h ago
 TriggeredBy: ● ssh.socket
     Docs: man:sshd(8)
           man:sshd_config(5)
    Main PID: 1321 (sshd)
      Tasks: 1 (limit: 4605)
     Memory: 1.3M (peak: 3.8M)
        CPU: 54ms
    CGroup: /system.slice/ssh.service
            └─1321 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"
```

entramos a la carpeta /etc/ssh y entramos al archivo sshd_config y configuramos

```
GNU nano 7.2 sshd_config
Include /etc/ssh/sshd_config.d/*.conf

# When systemd socket activation is used (the default), the socket
# configuration must be re-generated after changing Port, AddressFamily, or
# ListenAddress.
#
# For changes to take effect, run:
#
#   systemctl daemon-reload
#   systemctl restart ssh.socket
#
#Port 22445
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
#PermitRootLogin no
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

# Expect .ssh/authorized_keys2 to be disregarded by default in future.
#AuthorizedKeysFile .ssh/authorized_keys .ssh/authorized_keys2

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
```

-Port 22445

-permitrootlogin no

-pubkeyauthentication yes

-passwordauthentication no

Reinicio del servicio SSH

```
root@codeflow:/# systemctl restart ssh
```

Verificamos el funcionamiento con -p accedemos a través del puerto 22445

```
root@codeflow:/# ssh -p 22445 administrador@192.168.1.50
```

Definición de medio de alta disponibilidad

Es de vital importancia que la información y los servicios de un servidor se implementan métodos para respaldar y asegurar los datos a largo plazo contando con mecanismos de alta disponibilidad que permiten proteger estos datos

Medios de respaldo

El respaldo a largo plazo consiste en realizar copias de seguridad que se conservan durante meses o años, garantizando que los datos estén accesibles incluso si ocurren fallos graves.

- **Soportes físicos:** es adecuado usar discos externos para respaldar datos ya que podemos almacenar copias completas del sistema o fallos
- **Almacenamiento en la nube:** utilizando google cloud es un medio confiable y seguro además de tener un precio accesible
- **Política de retención:** por ejemplo, respaldos semanales conservados 6 meses y respaldos mensuales guardados durante 3 años.
- **Seguridad:** todas las copias deben cifrarse para evitar accesos no autorizados.

Alta disponibilidad

La alta disponibilidad busca que los servicios se mantengan activos incluso si un componente falla.

- **Copias en dos discos (RAID 1):** si un disco se rompe, el otro sigue con los datos.
- **Servidor de respaldo:** otro equipo puede tomar el lugar si el principal se cae.
- **Balanceo de carga:** repartir el trabajo entre varios servidores para que no se sature uno solo.
- **Monitoreo:** usar programas que avisen si algo falla para arreglarlo rápido.

Aplicación al proyecto

Para el servidor implementado en Ubuntu se propone:

- Configurar respaldos automáticos semanales en un disco externo y respaldos mensuales en la nube para asegurar la persistencia a largo plazo.
- Instalar un sistema de monitoreo básico con **systemd** y alertas por correo electrónico.

- Documentar procedimientos de restauración para que, en caso de error, se pueda recuperar la información de manera rápida y confiable.

Conclusión

Los medios de respaldo que utilizamos a largo plazo son pilares fundamentales en una buena administración de servidores. La aplicación de la misma en ubuntu server nos asegura la integridad de los datos y la continuidad del servicio, previniendo fallos técnicos o humanos