

TEMA 2

INSTALACIÓN, CONFIGURACIÓN Y
DOCUMENTACIÓN DEL ENTORNO
DE DESARROLLO Y DEL ENTORNO
DE EXPLOTACIÓN

EJERCICIOS

REBECA SÁNCHEZ PÉREZ
IES LOS SAUCES

INDICE

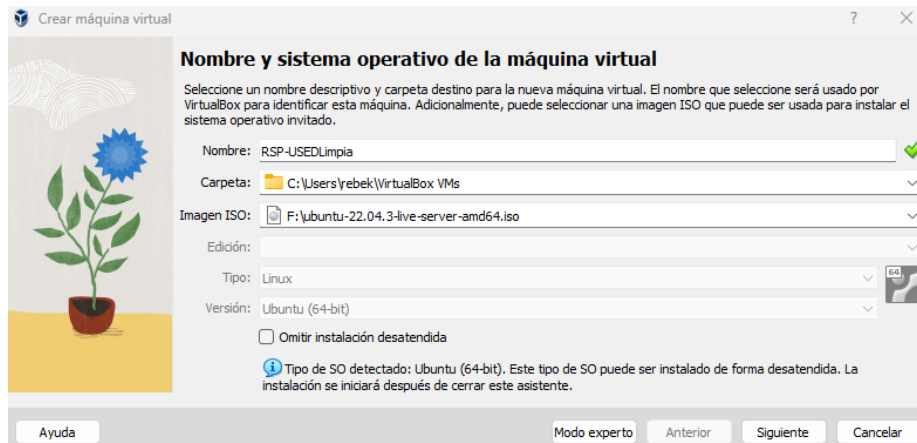
a. USED – UBUNTU SERVER	3
1. Configuración inicial.....	3
1.1 Características de la máquina virtual.....	3
1.2 Instalación	5
1.3 Nombre y configuración de red	12
1.4 Actualizar el sistema	14
1.5 Cortafuegos local	14
2. Cuentas de administración	16
3. Apache	17
4. PHP	20
5. MySQL	23
6. XDebug.....	27
6.1 Instalación	27
6.2 Ejecución desde NetBeans.....	28
7. Servidores DNS.....	28
8. Cuentas de desarrollo y hosting virtual	29
8.1 Creación de operadorweb	29
8.2 Creación del grupo ftpuser y enjaulamiento	29
d. WXED – WINDOWS X	32
1. Nombre y configuración de red	32
1.7 Conexión SSH desde Windows 10	32
2. Cuentas administradoras y cuenta de desarrollador	32
3. Navegadores	33
4. Filezilla.....	33
5. Notepad++	33
6. NetBeans.....	33
c. GITHUB – INTERNET	34

a. USED – UBUNTU SERVER

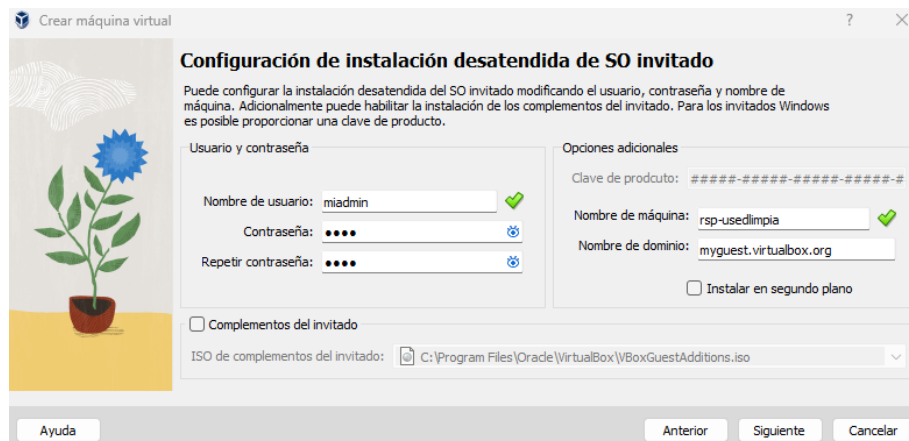
1. Configuración inicial

1.1 Características de la máquina virtual

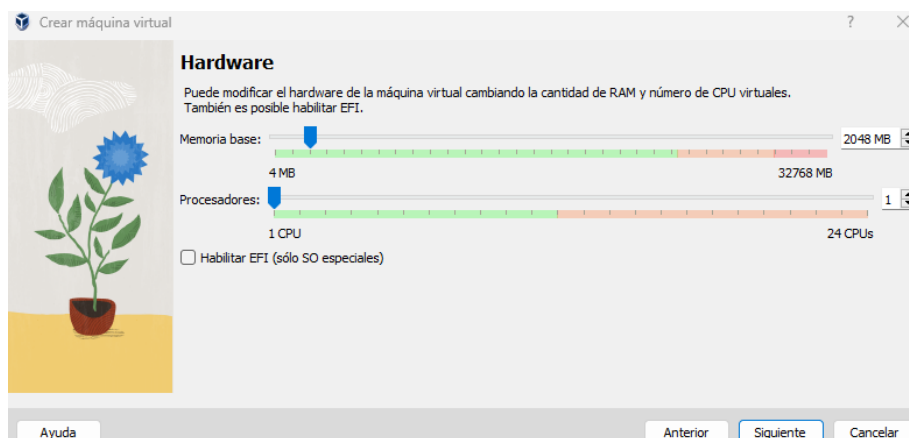
En primer lugar, abriremos el VirtualBox y crearemos una nueva máquina virtual que se llame RSP-USEDLimpia (Rebeca Sánchez Pérez - Ubuntu Server Entorno de Desarrollo Maquina limpia) con una ISO de un Ubuntu server 22.04.3



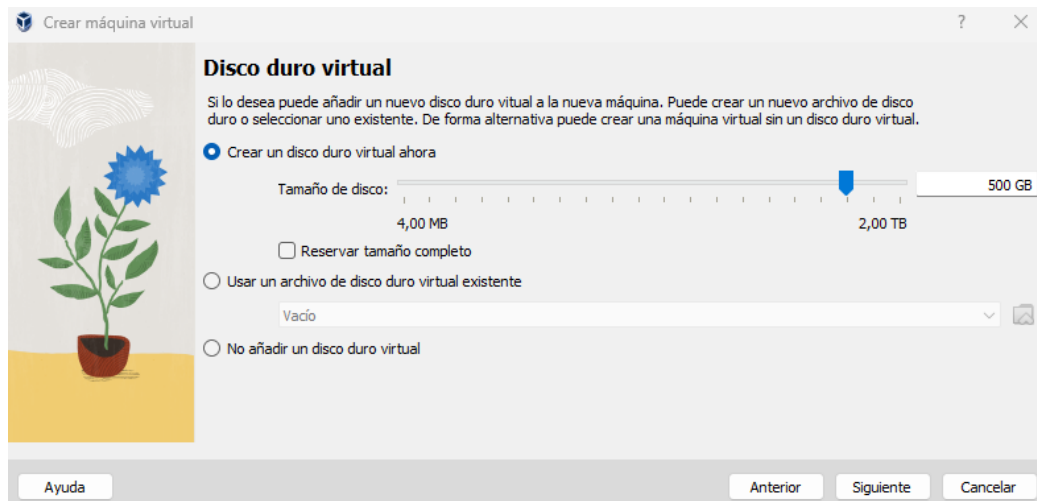
A continuación, elegiremos un nombre de usuario y una contraseña, en mi caso miadmin y paso.



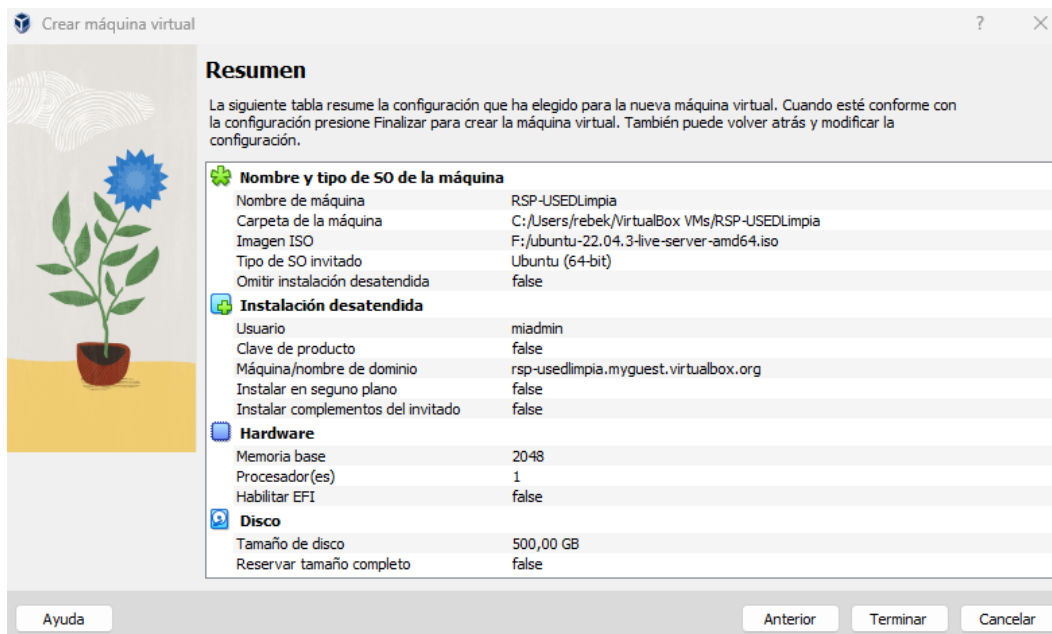
Seguimos con la configuración y reservamos 2048MB para la memoria RAM



Reservamos 500GB para la memoria del Disco Duro



Este será el resumen de la configuración de nuestra máquina de Ubuntu Server:



1.2 Instalación

Una vez creada la máquina, la iniciamos y comenzamos con la instalación seleccionando el idioma que queremos (español)

```
Willkommen! Bienvenue! Welcome! Добро пожаловать! Welkom! [ Help ]
Use UP, DOWN and ENTER keys to select your language.

[ Asturianu                ▾ ]
[ Bahasa Indonesia         ▾ ]
[ Català                   ▾ ]
[ Deutsch                   ▾ ]
[ English                   ▾ ]
[ English (UK)              ▾ ]
[ Español                   ▾ ]
[ Français                  ▾ ]
[ Galego                    ▾ ]
[ Deutsch                   ▾ ]
```

Seguiremos con la instalación sin actualizar a la versión que nos recomiendan

```
Actualización del instalador disponible [ Help ]

Version 23.09.1 of the installer is now available (23.08.1 is currently
running).

You can read the release notes for each version at:

    https://github.com/canonical/subiquity/releases

If you choose to update, the update will be downloaded and the installation
will continue from here.

[ Actualizar al instalador nuevo ]
[ Continuar sin actualizar ]
[ Atrás ]
```

```
Configuración del teclado [ Help ]

Selecione a continuación la disposición del teclado o elija «Identificar
teclado» para detectarla automáticamente.

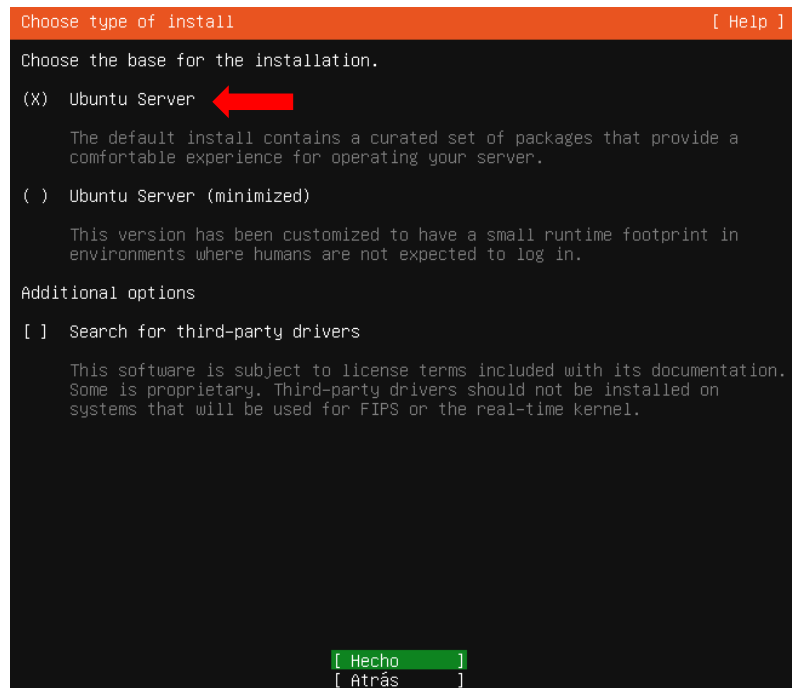
Disposición: [ Spanish ▾ ]

Variant: [ Spanish ▾ ]

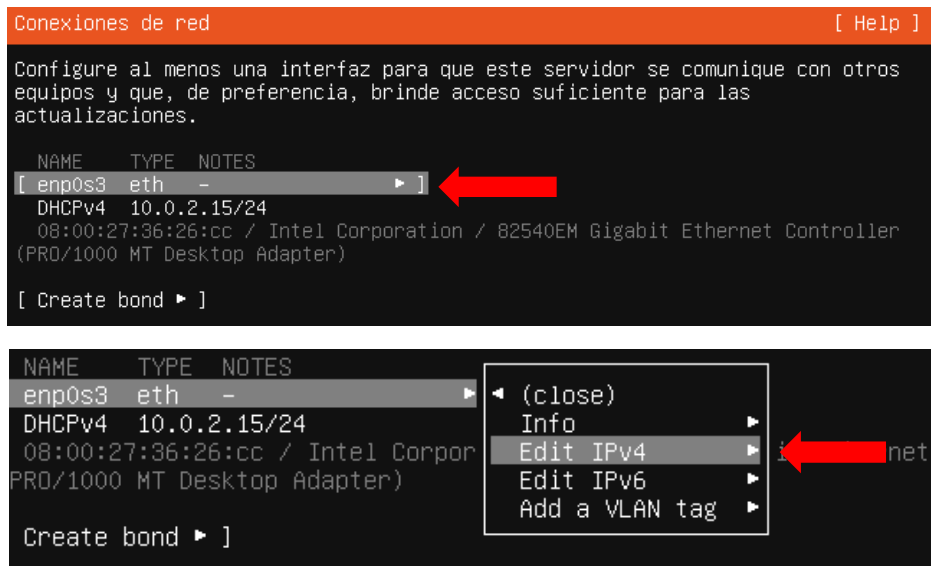
[ Identificar teclado ]

[ Hecho ]
[ Atrás ]
```



Seleccionamos el Ubuntu Server que aparece en primer lugar



A continuación, realizaremos la configuración de red. Esta configuración cambiará dependiendo de si la instalación la realizamos en casa o en clase ya que no coincidirán las direcciones IP de las máquinas anfitrionas (pertenecen a redes distintas). Configuramos la IPv4 de forma manual dándole la subred de 192.168.1.0/24 (en clase le daríamos 192.168.3.0/24), la dirección IP de 192.168.1.204 (en clase sería la 192.168.3.204) y la puerta de enlace de 192.168.1.1 (y en clase sería la 192.168.3.1). En ambos casos utilizaremos los servidores de DNS de 8.8.8.8 de Google.



Edit enp0s3 IPv4 configuration

Método de IPv4: Automático (DCHP)  

Manual
Desactivado

[Cancelar]

Edit enp0s3 IPv4 configuration

Método de IPv4: [Manual ]

Subred: 192.168.1.0/24

Dirección: 192.168.1.204

Puerta de enlace: 192.168.1.1

Servidores de nombres: 8.8.8.8
IP addresses, comma separated

Dominios de búsqueda:
Domains, comma separated

[Guardar]
[Cancelar]

Configure al menos una interfaz para que este servidor se comunique con otros equipos y que, de preferencia, brinde acceso suficiente para las actualizaciones.

NAME	TYPE	NOTES
enp0s3	eth	-
static	192.168.1.204/24	

08:00:27:38:28:CC / Intel Corporation / 82540EM Gigabit Ethernet Controller (PRD/1000 MT Desktop Adapter)

[Create bond ►]

En la siguiente pantalla nos pedirán una dirección proxy, simplemente tenemos que darle a “Hecho”

Configure proxy [Help]

If this system requires a proxy to connect to the internet, enter its details here.

Proxy address:

If you need to use a HTTP proxy to access the outside world, enter the proxy information here. Otherwise, leave this blank.

The proxy information should be given in the standard form of "http://[user][:pass@]host[:port]/".

[Hecho]
[Atrás]

Ahora esperamos a que se instalen las nuevas configuraciones y le damos a “Hecho”

```
Configure Ubuntu archive mirror [ Help ]

If you use an alternative mirror for Ubuntu, enter its details here.

Mirror address: http://es.archive.ubuntu.com/ubuntu
                You may provide an archive mirror that will be used instead of
                the default.

This mirror location does not seem to work. The output below may help explain
the problem. You can try again once the issue has been fixed (common problems
are network issues or the system clock being wrong).

[ Try again now ]

Ign:1 http://es.archive.ubuntu.com/ubuntu jammy InRelease
Ign:2 http://es.archive.ubuntu.com/ubuntu jammy-updates InRelease
Ign:3 http://es.archive.ubuntu.com/ubuntu jammy-backports InRelease
Ign:4 http://es.archive.ubuntu.com/ubuntu jammy-security InRelease
Ign:1 http://es.archive.ubuntu.com/ubuntu jammy InRelease
Ign:2 http://es.archive.ubuntu.com/ubuntu jammy-updates InRelease
Ign:3 http://es.archive.ubuntu.com/ubuntu jammy-backports InRelease
Ign:4 http://es.archive.ubuntu.com/ubuntu jammy-security InRelease
Ign:1 http://es.archive.ubuntu.com/ubuntu jammy InRelease
Ign:2 http://es.archive.ubuntu.com/ubuntu jammy-updates InRelease
Ign:3 http://es.archive.ubuntu.com/ubuntu jammy-backports InRelease
Ign:4 http://es.archive.ubuntu.com/ubuntu jammy-security InRelease
Err:1 http://es.archive.ubuntu.com/ubuntu jammy InRelease
      Fallo temporal al resolver «es.archive.ubuntu.com»
Err:2 http://es.archive.ubuntu.com/ubuntu jammy-updates InRelease
      Fallo temporal al resolver «es.archive.ubuntu.com»

[ Hecho ]
[ Atrás ]
```

```
Guided storage configuration [ Help ]

Configure a guided storage layout, or create a custom one:

(X) Use an entire disk

   [ VBOX_HARDDISK_VBc6d02768-4cd20ceb local disk 500.000G ▼ ]

[X] Set up this disk as an LVM group

   [ ] Encrypt the LVM group with LUKS

      Passphrase:

      Confirm passphrase:

( ) Custom storage layout

[ Hecho ]
[ Atrás ]
```

En la siguiente pantalla nos aparece un listado de las particiones que tiene nuestro disco duro. Tendremos que cambiarlas para que cumplan los requisitos de la máquina, para ello seleccionamos “Restablecer”


```
Storage configuration [ Help ]

RESUMEN DEL SISTEMA DE ARCHIVOS

PUNTO DE MONTAJE  TAMAÑO  TIPO  TIPO DE DISPOSITIVO
[ /               100.000G  new ext4  new LVM logical volume ► ]
[ /boot          2.000G  new ext4  new partition of disco local ► ]

DISPOSITIVOS DISPONIBLES

DISPOSITIVO  TIPO  TAMAÑO
[ ubuntu-vg (new)  LVM volume group  497.996G ► ]
espacio disponible  397.996G ►

[ Create software RAID (md) ► ]
[ Crear grupo de volúmenes (LVM) ► ]

DISPOSITIVOS UTILIZADOS

DISPOSITIVO  TIPO  TAMAÑO
[ ubuntu-vg (new)  LVM volume group  497.996G ► ]
ubuntu-lv      new, to be formatted as ext4, mounted at  100.000G ►
/

[ VBox_HARDDISK_VBc6d02768-4cd20ceb  disco local  500.000G ► ]
partition 1  new, BIOS grub spacer  1.000M ►
partition 2  new, to be formatted as ext4, mounted at  2.000G ►
/boot
partition 3  new, PV of LVM volume group ubuntu-vg  497.997G ►

[ Hecho ]
[ Restablecer ]
[ Atrás ]
```

Una vez pulsado el botón, aparece esta pantalla en la que tendremos que formatear las particiones del disco de forma que el Sistema (/) ocupe 150Gb y los datos (/var) el resto del disco, es decir, 349.997Gb.

```
Storage configuration [ Help ]

To continue you need to: Mount a filesystem at /
                        Select a boot disk

RESUMEN DEL SISTEMA DE ARCHIVOS

No se montó ningún disco o partición.

DISPOSITIVOS DISPONIBLES

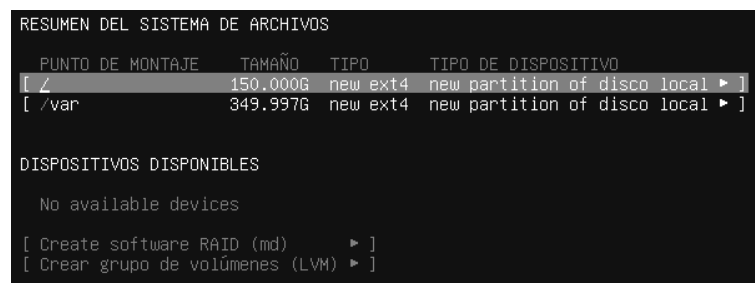
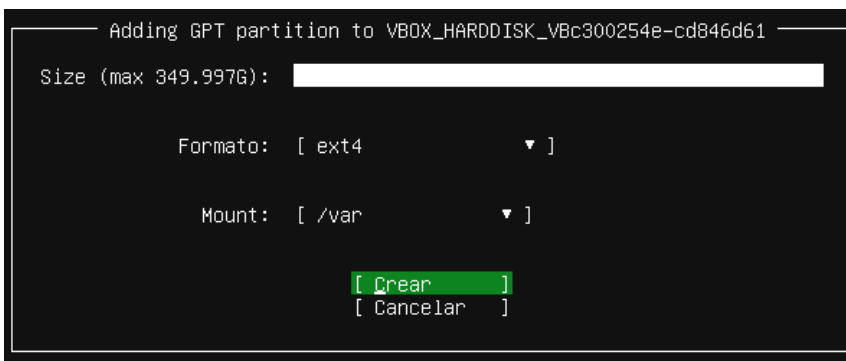
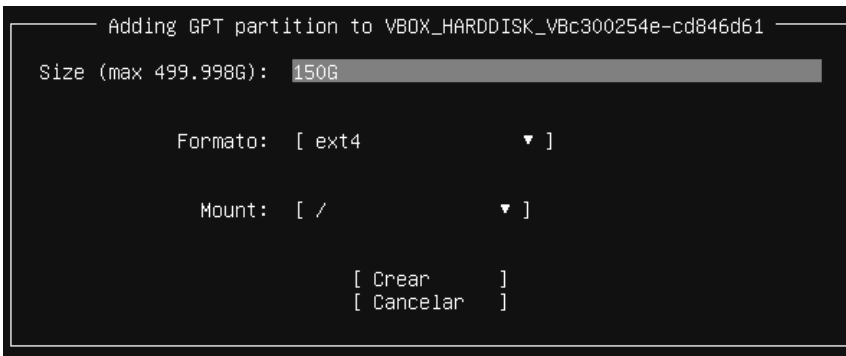
DISPOSITIVO  TIPO  TAMAÑO
[ VBox_HARDDISK_VBc6d02768-4cd20ceb  disco local  500.000G ► ]
espacio disponible  499.998G ►

[ Create software RAID (md) ► ]
[ Crear grupo de volúmenes (LVM) ► ]

DISPOSITIVOS UTILIZADOS

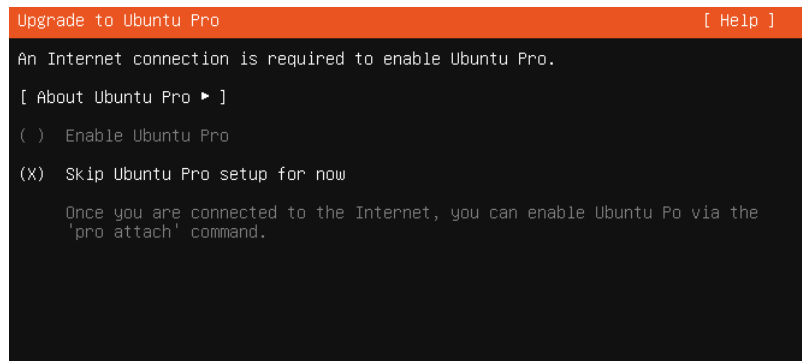
No used devices

[ Hecho ]
[ Restablecer ]
[ Atrás ]
```

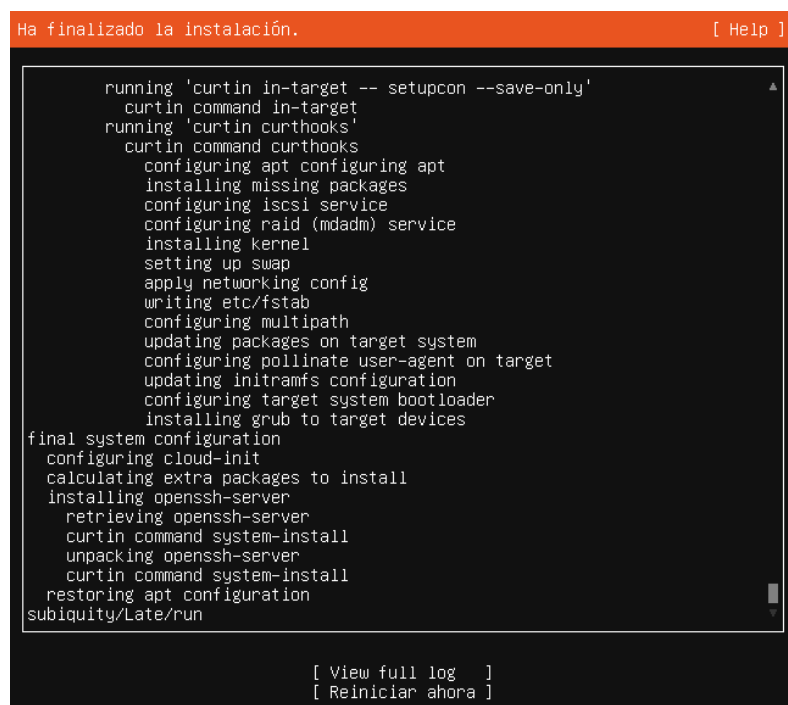
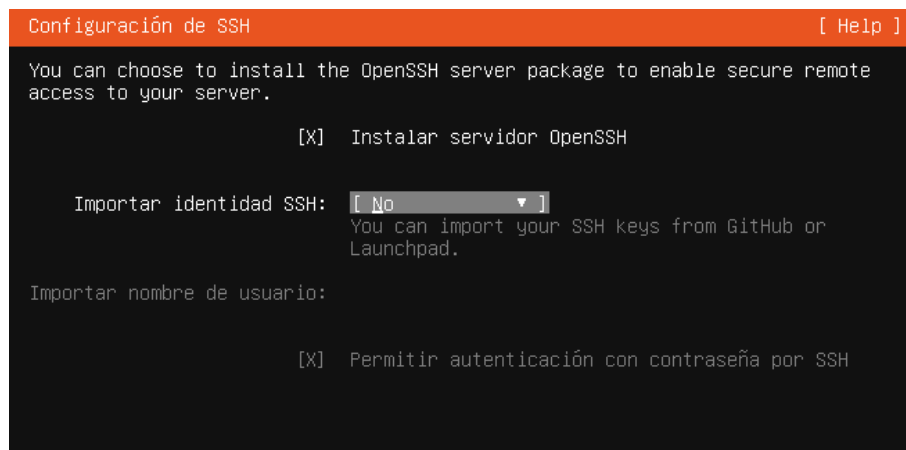


A continuación, elegiremos el nombre de la máquina, el usuario y la contraseña





Instalamos el servidor OpenSSH, esperamos a que se complete la instalación y reiniciamos



1.3 Nombre y configuración de red

Para cambiar el nombre del equipo, abrimos el nuevo Ubuntu Server, usamos el comando “`sudo hostnamectl set-hostname rsp-used`” y modificamos el fichero `host` con “`sudo nano /etc/hosts`” sustituyendo “`rsp-usedlimpia`” por “`rsp-used`”

```
miadmin@rsp-usedlimpia:~$ sudo hostnamectl set-hostname rsp-used
[sudo] password for miadmin:
miadmin@rsp-usedlimpia:~$ sudo nano /etc/hosts_
```

```
127.0.0.1 localhost
127.0.1.1 rsp-usedlimpia_
# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

Una vez guardados los cambios, reiniciamos con “`reboot`” y escribimos “`hostname`” para confirmar que se han realizado los cambios

```
miadmin@rsp-usedlimpia:~$ reboot_
```

```
miadmin@rsp-used:~$ hostname
rsp-used
```

Si queremos acceder a la configuración de la red, nos dirigimos a la ruta de los archivos de configuración escribiendo “`cd /etc/netplan`”. Para modificar el archivo de red usaremos en comando “`sudo nano 00-installer-config.yaml`”

```
miadmin@rsp-usedlimpia:~$ cd /etc/netplan/
miadmin@rsp-usedlimpia:/etc/netplan$ sudo nano 00-installer-config.yaml _
```

A continuación, aparecerá esta pantalla con el contenido de ese archivo. Si queremos cambiar la IP de nuestra maquina basta con editar la que se muestra en el archivo (es muy importante respetar la tabulación y el espaciado de este archivo). Para aplicar los cambios usamos el comando “`sudo netplan apply`” y se cambiará la dirección IP por la nueva.

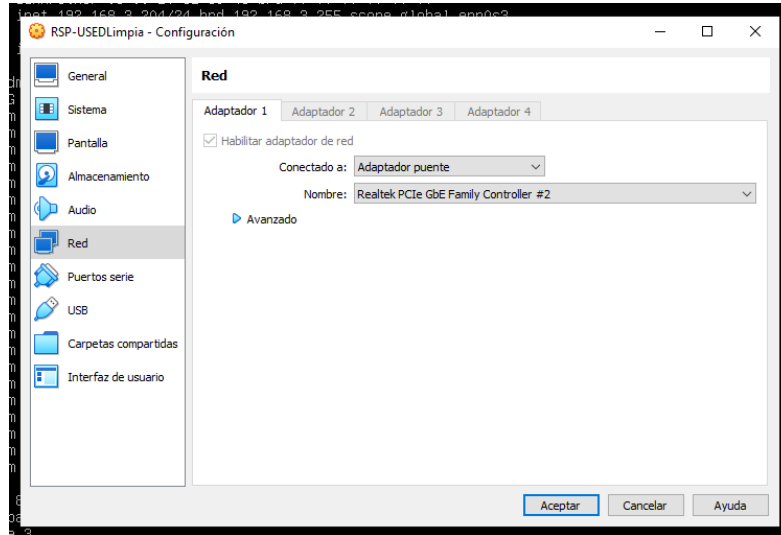
```
# This is the network config written by 'subiquity'
network:
  ethernets:
    enp0s3:
      addresses:
        - 192.168.1.204/24
      nameservers:
        addresses:
          - 8.8.8.8
          - 8.8.8.8
        search: []
      routes:
        - to: default
          via: 192.168.1.1
      version: 2
```

```
miadmin@rsp-usedlimpia:/etc/netplan$ sudo netplan apply
```

Iniciamos sesión con el usuario “`miadmin`” y comprobamos que tenemos conexión a internet con el comando “`ping google.es`”. En el caso de que no se conecte tendremos que cambiar

la configuración de red de la maquina desde VirtualBox a adaptador puente y reiniciar la maquina con el comando “reboot”

```
miadmin@rsp-usedlimpia:~$ ping google.es  
ping: google.es: Temporary failure in name resolution
```



1.4 Actualizar el sistema

Cuando tengamos conexión a internet escribimos “sudo apt update” y “sudo apt upgrade” para actualizar los repositorios y librerías.

Para configurar la fecha y la hora antes debemos conocer cuál es la fecha y hora activa en nuestro server, para ello escribimos “date” y para cambiar a la franja horaria de Madrid usaremos el comando “sudo timedatectl set-timezone Europe/Madrid”

```
miadmin@rsp-usedlimpia:~$ date
mar 03 oct 2023 09:01:54 UTC
```

```
miadmin@rsp-usedlimpia:~$ sudo timedatectl set-timezone Europe/Madrid
[sudo] password for miadmin:
miadmin@rsp-usedlimpia:~$ date
mar 03 oct 2023 11:07:55 CEST
miadmin@rsp-usedlimpia:~$ _
```

1.5 Cortafuegos local

En primer lugar, tenemos que saber cuál es el estado del cortafuegos con el comando “sudo ufw status”, si esta inactivo lo activamos con “systemctl status ufw” y “sudo ufw enable”

```
miadmin@rsp-usedlimpia:~$ sudo ufw status
Status: inactive
```

```
miadmin@rsp-usedlimpia:~$ systemctl status ufw
• ufw.service - Uncomplicated firewall
   Loaded: loaded (/lib/systemd/system/ufw.service; enabled; vendor preset: enabled)
   Active: active (exited) since Tue 2023-10-03 08:17:01 UTC; 17min ago
     Docs: man:ufw(8)
   Main PID: 555 (code=exited, status=0/SUCCESS)
    CPU: 1ms

oct 03 08:17:01 rsp-usedlimpia systemd[1]: Starting Uncomplicated firewall...
oct 03 08:17:01 rsp-usedlimpia systemd[1]: Finished Uncomplicated firewall.
miadmin@rsp-usedlimpia:~$ _
```

```
miadmin@rsp-usedlimpia:~$ sudo ufw enable
Firewall is active and enabled on system startup
miadmin@rsp-usedlimpia:~$
```

Cuando tenemos el cortafuegos activo abrimos el puerto 22 con “sudo ufw allow 22” y comprobamos que está abierto con “sudo ufw status”

```
miadmin@rsp-usedlimpia:~$ sudo ufw allow 22
Rule added
Rule added (v6)
miadmin@rsp-usedlimpia:~$ sudo ufw status
Status: active

To Action From
--
22 ALLOW Anywhere
22 (v6) ALLOW Anywhere (v6)

miadmin@rsp-usedlimpia:~$ _
```


2. Cuentas de administración

Para la creación de un usuario nuevo usamos el comando “sudo adduser miadmin2” y rellenamos los campos que queramos por ejemplo la contraseña

```
miadmin@rsp-used:~$ sudo adduser miadmin2
[sudo] password for miadmin:
Adding user `miadmin2' ...
Adding new group `miadmin2' (1001) ...
Adding new user `miadmin2' (1001) with group `miadmin2' ...
Creating home directory `/home/miadmin2' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for miadmin2
Enter the new value, or press ENTER for the default
  Full Name []:
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n]
miadmin@rsp-used:~$ _
```

Si queremos añadir el nuevo usuario al grupo de super usuarios (es decir, el grupo de los admins o root) escribiremos el comando “sudo usermod -aG sudo miadmin2”

```
miadmin@rsp-used:~$ sudo usermod -aG sudo miadmin2
miadmin@rsp-used:~$ _
```

FALTA REVISION

3. Apache

En la maquina RSP-USED instalaremos el servicio Apache2, pero antes de eso tenemos que actualizar los repositorios de Linux con el comando “sudo apt-get update”.

Para instalar Apache escribiremos “sudo apt-get install apache2”:

```
miadmin@rsp-used:~$ sudo apt-get install apache2
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
apache2 ya está en su versión más reciente (2.4.52-1ubuntu4.6).
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 5 no actualizados.
```

Una vez instalado Apache, tenemos que permitir el acceso a Apache al puerto 80 del cortafuegos de nuestro server mediante el comando “sudo ufw allow apache” y para comprobarlo escribimos “sudo ufw status”

```
miadmin@rsp-used:~$ sudo ufw allow apache
Rule added
Rule added (v6)
```

```
miadmin@rsp-used:~$ sudo ufw status
Status: active

To Action From
--
22 ALLOW Anywhere
Apache ← ALLOW Anywhere
22 (v6) ALLOW Anywhere (v6)
Apache (v6) ← ALLOW Anywhere (v6)
```

Modificamos con “sudo nano /etc/apache2/sites-available/000-default.conf” el fichero de configuración para grabar los registros logs en un directorio específico. Para ello escribimos estas 2 líneas:

```
# It is also possible to configure the loglevel for particular
# modules, e.g.
#LogLevel info ssl:warn

ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined
ErrorLog /var/www/html/log/error.log
CustomLog /var/www/html/log/access.log combined

# For most configuration files from conf-available/, which
```

- La línea ErrorLog /var/www/html/log/error.log sirve para generar un archivo (error.log) con registros logs de los errores en la ruta que se especifica (en /var/www/html/log).
- La línea CustomLog /var/www/html/log/access.log combined sirve para generar una serie de registros de solicitudes o acceso a archivos en un servidor web.

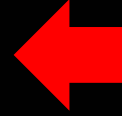
Después de esto creamos el directorio para almacenar los logs que hemos especificado, para ello usamos “sudo mkdir /var/www/html/log” y reiniciamos el servicio de apache.

A continuación, configuraremos el archivo de configuración de apache con “sudo nano /etc/apache2/apache2.conf” y escribimos lo siguiente:

```
<Directory /usr/share>
    AllowOverride None
    Require all granted
</Directory>

<Directory /var/www/>
    Options Indexes FollowSymLinks
    AllowOverride All
    Require all granted
</Directory>

#<Directory /srv/>
#     Options Indexes FollowSymLinks
```



Ahora creamos el archivo .htaccess escribiendo con “sudo nano /var/www/html/.htaccess” y escribimos:

```
GNU nano 6.2
DirectoryIndex index.php index.html
```

Para validar los archivos de configuración de Apache escribimos el comando “sudo apache2ctl configtest” y en nuestro caso aparece un warning que tenemos que subsanar modificando el fichero de configuración del servicio

```
miadmin@rsp-used:~$ sudo apache2ctl configtest
[sudo] password for miadmin:
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.0.1.1. Set the 'ServerName' directive globally to suppress this message
Syntax OK
```

Usamos “sudo nano /etc/apache2/apache2.conf” y al final del archivo escribimos “ServerName rsp-used” y reiniciamos el servicio

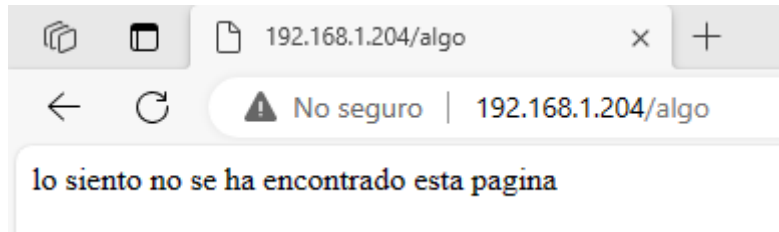
```
# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
ServerName rsp-used
```

Para instalar un navegador de texto usamos el comando “sudo apt install lynx” y para activar el módulo de apache *status* usamos “sudo a2enmod status” y reiniciamos apache2

```
miadmin@rsp-used:~$ sudo a2enmod status
Module status already enabled
miadmin@rsp-used:~$
```

Para ver los informes escribimos en la terminal “apache2ctl fullstatus” y para la personalización de errores podemos editar el fichero .htaccess (con “sudo nano /var/www/html/.htaccess”) y escribir el mensaje que queremos

```
DirectoryIndex index.php index.html
ErrorDocument 404 "lo siento no se ha encontrado esta pagina"
```



A continuación, instalaremos un módulo de apache llamado Alias que nos permite el acceso a directorios superiores a los que puede acceder un usuario. En primer lugar, modificamos el fichero de configuración de apache (“sudo nano /etc/apache2/apache2.conf”) cambiando la siguiente línea:

```
<Directory /var/www/>
    Options -Indexes +FollowSymLinks
    AllowOverride All
    Require all granted
</Directory>
```

Para hacer una prueba creamos una carpeta *doc* en */home/miadmin* y dentro de esta creamos 2 archivos de texto *file1.txt* y *file2.txt*. Usamos el comando “sudo nano /etc/apache2/mods-enabled/alias.conf” para modificar el archivo de configuración de alias y escribimos:

```
</Directory>
Alias /documentos /home/miadmin/doc
<Directory /home/miadmin/doc>
    Options FollowSymLinks Indexes
    AllowOverride None
    Require all granted
</Directory>
```

Usamos “sudo service apache2 restart”. Con esto lo que hacemos es vincular un directorio fantasma */documentos* a */doc* para poder ver lo que hay dentro de esa carpeta

4. PHP

Antes de instalar php tendremos que actualizar el sistema con “sudo apt update” y para instalarlo escribimos “sudo apt -y install php8.1” (instalamos la versión 8.1 porque es la más nueva que es estable) y para comprobar la versión podemos utilizar “sudo php -v”

```
miadmin@rsp-used:~$ sudo php -v
PHP 8.1.2-1ubuntu2.14 (cli) (built: Aug 18 2023 11:41:11) (NTS)
Copyright (c) The PHP Group
Zend Engine v4.1.2, Copyright (c) Zend Technologies
with Zend OPcache v8.1.2-1ubuntu2.14, Copyright (c), by Zend Technologies
```

Instalamos la librería de módulos de apache con “sudo apt install libapache2-mod-php” y para crear un archivo info.php usamos “sudo nano /var/www/html/info.php” y escribimos lo siguiente:

```
<?php
phpinfo();
?>
```

A continuación, reiniciamos el servicio de apache con “sudo service apache2 restart” y para comprobar que todo está bien escribimos en nuestro navegador 192.168.3.204/info.php y si aparece esta pantalla significa que se ha creado correctamente:

PHP Version 8.1.2-1ubuntu2.14	
	
System	Linux rsp-used 5.15.0-86-generic #96-Ubuntu SMP Wed Sep 20 08:23:49 UTC 2023 x86_64
Build Date	Aug 18 2023 11:41:11
Build System	Linux
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/8.1/apache2
Loaded Configuration File	/etc/php/8.1/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/8.1/apache2/conf.d
Additional .ini files parsed	/etc/php/8.1/apache2/conf.d/10-opcache.ini, /etc/php/8.1/apache2/conf.d/10-pdo.ini, /etc/php/8.1/apache2/conf.d/20-calendar.ini, /etc/php/8.1/apache2/conf.d/20-ctype.ini, /etc/php/8.1/apache2/conf.d/20-exif.ini, /etc/php/8.1/apache2/conf.d/20-ffi.ini, /etc/php/8.1/apache2/conf.d/20-fileinfo.ini, /etc/php/8.1/apache2/conf.d/20-ftp.ini, /etc/php/8.1/apache2/conf.d/20-gettext.ini, /etc/php/8.1/apache2/conf.d/20-gd.ini, /etc/php/8.1/apache2/conf.d/20-iconv.ini, /etc/php/8.1/apache2/conf.d/20-phar.ini, /etc/php/8.1/apache2/conf.d/20-posix.ini, /etc/php/8.1/apache2/conf.d/20-readline.ini, /etc/php/8.1/apache2/conf.d/20-shmop.ini, /etc/php/8.1/apache2/conf.d/20-sockets.ini, /etc/php/8.1/apache2/conf.d/20-sysvmsg.ini

Instalamos los módulos de SOAP y de xml con el comando “sudo apt install -y php8.1-xml php8.1-soap” y reiniciamos el servicio de apache2. En 192.168.3.204/info.php podemos comprobar que se han instalado correctamente

SNMP	Rasmus Lerdorf, Haimi Hazewinkel, Mike Jackson, Steven Lawrence, Jonatan M.
SOAP	Brad Lafountain, Shane Caraveo, Dmitry Stogov
Sockets	Chris Vandomelen, Sterling Hughes, Daniel Beulshausen, Jason Greene
Sodium	Frank Denis
SPL	Marcus Boerger, Etienne Kneuss
SQLite 3.x driver for PDO	Wez Furlong
SQLite3	Scott MacVicar, Ilia Alshanetsky, Brad Dewar
System V Message based IPC	Wez Furlong
System V Semaphores	Tom May
System V Shared Memory	Christian Cartus
tidy	John Coggeshall, Ilia Alshanetsky
tokenizer	Andrei Zmievski, Johannes Schlueter
XML	Stig Bakken, Thies C. Arntzen, Sterling Hughes
XMLReader	Rob Richards
XMLWriter	Rob Richards, Pierre-Alain Joye
XSL	Christian Stroh, Rob Richards

Ahora cambiaremos el fichero de configuración php.ini pero antes de esto haremos una copia de seguridad. Cambiamos de directorio a `"cd /etc/php/8.1/apache2"` y usamos el comando `"sudo cp php.ini php.ini.backup"` para hacer la copia.

```
miadmin@rsp-used:/etc/php/8.1/apache2$ sudo cp php.ini php.ini.backup
miadmin@rsp-used:/etc/php/8.1/apache2$ ls
conf.d  php.ini  php.ini.backup
```

Usamos el comando “sudo nano php.ini” para modificar el fichero de configuración de php. Buscamos en el archivo la línea “display errors” y cambiamos la línea de display_errors=Off y la ponemos en On

```
; Development Value: On
; Production Value: Off
; https://php.net/display-errors
display_errors = On
```

En ese mismo archivo buscamos la línea “display_startup_errors” y lo activamos igual que el anterior

```
; Development Value: On
; Production Value: Off
; https://php.net/display-startup-errors
display_startup_errors = On_
```

Guardamos el archivo, reiniciamos el servicio de apache y para comprobarlo volvemos al navegador escribiendo 192.168.3.204/info.php y buscamos las configuraciones que acabamos de activar para comprobar que se han cambiado correctamente

disable_functions	no value	no value
display_errors	On	On
display_startup_errors	On	On
doc root	no value	no value

Por último, volvemos a modificar el php.ini y cambiamos los valores de “memory_limit” para que tenga una memoria de 256M

```
; Maximum amount of memory a script can consume (128MB -> 256MB)
; https://php.net/memory-limit
memory_limit = 256M
```

Como hemos hecho anteriormente hacemos `“sudo service apache2 restart”` para reiniciar apache y comprobamos que se haya cambiado la memoria en nuestro navegador

max_multipart_body_parts	-1
memory_limit	256M

5. MySQL

Primero actualizamos las librerías de Linux con “sudo apt update” y “sudo apt upgrade” y después instalamos el servicio de MySQL con “sudo apt install mysql-server”. Para visualizar la versión que se ha instalado, usamos el comando “mysql --version”

```
miadmin@rsp-used:~$ mysql --version
mysql Ver 8.0.34-0ubuntu0.22.04.1 for Linux on x86_64 ((Ubuntu))
miadmin@rsp-used:~$ _
```

Para permitir la conexión desde cualquier IP necesitamos entrar al fichero de configuración de MySQL (“sudo nano /etc/mysql/mysql.conf.d/mysqld.cnf”) y comentar las siguientes líneas: “bind-address = 127.0.0.1” y “mysqlx-bind-address = 127.0.0.1”

```
#
# Instead of skip-networking the default is now to listen on
# localhost which is more compatible and is not less secure
#bind-address            = 127.0.0.1
#mysqlx-bind-address     = 127.0.0.1
#
# * Fine Tuning
#
```

Guardamos el archivo y reiniciamos mysql con “sudo service mysql restart” y después abrimos el puerto 3306 con “sudo ufw allow 3306”

```
miadmin@rsp-used:~$ sudo ufw status
Status: active

To Action From
--
22 ALLOW Anywhere
Apache ALLOW Anywhere
9003 ALLOW Anywhere
3306 ALLOW Anywhere
22 (v6) ALLOW Anywhere (v6)
Apache (v6) ALLOW Anywhere (v6)
9003 (v6) ALLOW Anywhere (v6)
3306 (v6) ALLOW Anywhere (v6)
```

A continuación, ejecutaremos un script para realizar una configuración inicial de mysql con el comando “sudo mysql_secure_installation” y responderemos “yes” a todos los pasos excepto el de nivel de seguridad de contraseña que elegiremos el “LOW”

```
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) : y
Success.

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!
```


Ahora volvemos a reiniciar MySQL e instalamos las librerías de php que son necesarias para la conexión de mysql con php con el comando “`sudo apt install libapache2-mod-php8.1 php8.1-mysql`” y reiniciamos apache. Ejecutamos mysql con “`sudo mysql`” y una vez dentro creamos el usuario administrador de las bases de datos de nuestro servidor. Para ello escribimos:

- Create user ‘adminsqli’@’%’ identified by ‘pass’;
- Grant all privileges on *.* to ‘adminsqli’@’%’ with grant option;

```
mysql> select user,plugin,host from user where user like 'adminsqli';
+-----+-----+-----+
| user      | plugin                | host |
+-----+-----+-----+
| adminsqli | caching_sha2_password | %    |
+-----+-----+-----+
1 row in set (0,00 sec)
```

En el caso de que no nos permita crear el usuario saltando el error de que la contraseña no cumple con los requisitos, tendremos que cambiar las variables globales de mysql para admitir contraseñas con 4 caracteres como mínimo. Para ello dentro de la consola de mysql escribiremos “`show variables like ‘validate_password%’`” y se mostrara la tabla de las variables que controlan los requisitos de las contraseñas de los usuarios.

```
mysql> show variables like 'validate_pass%';
+-----+-----+-----+
| Variable_name | Value |
+-----+-----+-----+
| validate_password.changed_characters_percentage | 0 |
| validate_password.check_user_name | ON |
| validate_password.dictionary_file |  |
| validate_password.length | 8 |
| validate_password.mixed_case_count | 1 |
| validate_password.number_count | 1 |
| validate_password.policy | MEDIUM |
| validate_password.special_char_count | 1 |
+-----+-----+-----+
8 rows in set (0,00 sec)
```

Una vez aquí, cambiaremos el valor de “length” y “policy” con “`set global validate_password.length=4;`” y “`set global validate_password.policy=LOW;`”

```
mysql> set global validate_password.length=4;
Query OK, 0 rows affected (0,00 sec)

mysql> set global validate_password.policy=LOW;
Query OK, 0 rows affected (0,00 sec)

mysql> show variables like 'validate_password%';
+-----+-----+-----+
| Variable_name | Value |
+-----+-----+-----+
| validate_password.changed_characters_percentage | 0 |
| validate_password.check_user_name | ON |
| validate_password.dictionary_file |  |
| validate_password.length | 4 |
| validate_password.mixed_case_count | 1 |
| validate_password.number_count | 1 |
| validate_password.policy | LOW |
| validate_password.special_char_count | 1 |
+-----+-----+-----+
8 rows in set (0,00 sec)
```

FALTA PHPMYADMIN

6. XDebug

6.1 Instalación

Antes de instalar XDebug comprobaremos que no esté instalado anteriormente con “`php -m | grep xdebug`”, si la terminal no devuelve nada significa que ese modulo no está instalado. Para instalarlo escribimos “`sudo apt install php8.1-xdebug`”

```
miadmin@rsp-used:~$ sudo apt install php8.1-xdebug
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
php8.1-xdebug ya está en su versión más reciente (3.1.2)
0 actualizados, 0 nuevos se instalarán, 0 para eliminar
```



A continuación, cambiaremos de directorio a “`cd /etc/php/8.1/apache2/conf.d/`” **cd /etc/php/8.1/mods-available/** y modificamos el archivo 20-xdebug.ini con “`sudo nano 20-xdebug.ini`” y escribimos las siguientes líneas: ***AQUI HAY QUE CAMBIAR A LAS NUEVAS DIRECTIVAS***

```
zend_extension=xdebug.so
xdebug.mode=debug
xdebug.client_host=localhost
xdebug.client_port=9003
xdebug.idekey="netbeans-xdebug"
```

```
zend_extension=xdebug.so
xdebug.extended_info=on
xdebug.remote_host=192.168.3.204
xdebug.remote_port=9003
xdebug.remote_handler=dbgp
xdebug.remote_autostart=on
xdebug.remote_enable = on
xdebug.mode=debug
xdebug.client_host=localhost
xdebug.client_port=9003
xdebug.idekey="netbeans-xdebug"
```

Para que se pueda conectar por el puerto 9003 tendremos que abrirlo mediante “`sudo ufw allow 9003`” y con “`sudo ufw status`” vemos los puertos que están abiertos

```
miadmin@rsp-used:/etc/php/8.1/apache2/conf.d$ sudo ufw allow 9003
Rule added
Rule added (v6)
miadmin@rsp-used:/etc/php/8.1/apache2/conf.d$ sudo ufw status
Status: active

To Action From
--
22 ALLOW Anywhere
Apache ALLOW Anywhere
9003  ALLOW Anywhere
22 (v6) ALLOW Anywhere (v6)
Apache (v6) ALLOW Anywhere (v6)
9003 (v6)  ALLOW Anywhere (v6)
```

Para terminar, reiniciamos el servicio con “`sudo service apache2 restart`”.

6.2 Ejecución desde NetBeans

Para comenzar con la ejecución, pondremos un punto de ruptura o breakpoint en una de nuestras líneas de código, al ser posible donde una variable cambie de valor.

REVISAR

7. Servidores DNS

8. Cuentas de desarrollo y hosting virtual

8.1 Creación de operadorweb

Tenemos que crear un usuario que se encargue de conectarse al servidor mediante http/s desde la maquina anfitriona que usaremos de cliente. Para eso usamos el comando “sudo adduser --home /var/www/html --no-create-home --ingroup www-data operadorweb” que creara el usuario operadorweb en el grupo www-data y con el home en el directorio html.

```
miadmin@rsp-used:~$ sudo adduser --home /var/www/html --no-create-home --ingroup www-data operadorweb
Adding user `operadorweb' ...
Adding new user `operadorweb' (1002) with group `www-data' ...
Not creating home directory `/var/www/html'.
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for operadorweb
Enter the new value, or press ENTER for the default
  Full Name []:
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n]
miadmin@rsp-used:~$
```

Para confirmar que se ha creado usamos el comando “id operadorweb” y se mostrará la información sobre ese usuario.

```
miadmin@rsp-used:~$ id operadorweb
uid=1002(operadorweb) gid=33(www-data) groups=33(www-data)
```

A continuación, cambiamos el propietario del directorio /var/www/html para que sea operadorweb con “sudo chown -R operadorweb:www-data /var/www/html” y seguido de esto cambiamos los permisos del archivo que se encuentra en ese directorio para que pueda ser legible, modificado y ejecutado por ese usuario con “sudo chmod -R 2775 /var/www/html”

```
miadmin@rsp-used:~$ sudo chown -R operadorweb:www-data /var/www/html
miadmin@rsp-used:~$ sudo chmod -R 2775 /var/www/html
```

```
miadmin@rsp-used:~$ ll /var/www/html/
total 20
drwxrwsr-x 2 operadorweb www-data 4096 oct 4 00:25 ./
drwxr-xr-x 3 root         root    4096 oct 4 00:25 ../
-rwxrwsr-x 1 operadorweb www-data 10671 oct 4 00:25 index.html*
```

8.2 Creación del grupo ftpuser y enjaulamiento

Creamos el grupo ftpuser al que pertenecerá el operadorweb con el comando “sudo groupadd ftpuser”

```
miadmin@rsp-used:~$ sudo groupadd ftpuser
[sudo] password for miadmin:
```

Después de esto, cambiaremos de directorio a “cd /etc/ssh” y crearemos una copia de seguridad del fichero de configuracion sshd_config.d con el comando “sudo cp -r ssh_config.d sshd_config.d.backup”. Una vez creada la copia, editaremos el archivo original (con “sudo nano sshd_config”) de la siguiente manera:

```
# override default of no subsystems
#Subsystem      sftp      /usr/lib/openssh/sftp-server
Subsystem sftp internal-sftp

# Example of overriding settings on a per-user basis
#Match User anoncvs
#      X11Forwarding no
#      AllowTcpForwarding no
#      PermitTTY no
#      ForceCommand cvs server
Match Group ftpuser
ChrootDirectory %h
ForceCommand internal-sftp -u 2
AllowTcpForwarding yes
PermitTunnel no
X11Forwarding no_
```

- Comentamos la línea “Subsystem sftp /usr/lib/openssh/sftp-server” que viene por defecto y escribimos la que viene a continuación “Subsystem sftp internal-sftp”
- Al final del fichero agregamos las líneas de texto que vienen indicadas por la segunda flecha

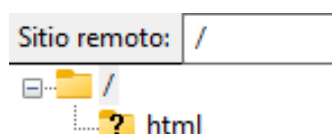
Una vez guardada la configuración, reiniciaremos ssh con “sudo service ssh restart”. Esta configuración nos permite controlar el servicio SFTP con el grupo de usuarios “ftpuser”. Ahora procedemos a enjaular a nuestro usuario operadorweb en el directorio “/var/www”, esto significa que el usuario operadorweb nunca va a tener acceso a los archivos y directorios que estén por encima del directorio en el que se encuentra enjaulado. Para ello debemos parar el servicio de apache con “sudo service apache2 stop” y después usaremos el comando “sudo usermod operadorweb -d /var/www” que modifica el usuario para que su home sea /var/www

```
miadmin@rsp-used:/etc/ssh$ sudo usermod operadorweb -d /var/www
miadmin@rsp-used:/etc/ssh$ _
```

A continuación, usamos el comando “sudo chown root:root /var/www” para cambiar el propietario de ese directorio y seguido de esto con “sudo chmod 555 /var/www” y “sudo usermod -G ftpuser operadorweb” le daremos permisos de lectura y ejecución y agregaremos el usuario operadorweb al grupo ftpuser.

```
drwxrwxrwt  6 root root    4096 oct 30 18:48 tmp/
dr-xr-xr-x  3 root root    4096 oct  4 00:25 www/
miadmin@rsp-used:/var$
```

Si queremos comprobar que el enjaulamiento está correcto, nos conectamos desde FileZilla el servidor con el usuario operadorweb y si el directorio raíz contiene /www significa que ha ido correctamente.



b. WXED – WINDOWS X

1. Nombre y configuración de red

MODIFICAR DOCUMENTO, AGREGAR APARTADOS DE LOS PROGRAMAS

Ahora pasamos a trabajar desde el Windows 10 de nuestra maquina anfitriona. Abrimos el cmd y escribimos “ssh miadmin@192.168.3.204” para conectarnos desde nuestro windows al terminal de nuestro entorno de desarrollo. Si al intentar conectarte te da problemas porque ya has usado la IP de la maquina nueva para otra máquina, tendremos que entrar en C: > Usuarios > TuUsuario > .ssh y eli

1.7 Conexión SSH desde Windows 10

Ahora pasamos a trabajar desde el Windows 10 de nuestra maquina anfitriona. Abrimos el cmd y escribimos “ssh miadmin@192.168.3.204” para conectarnos desde nuestro windows al terminal de nuestro entorno de desarrollo. Si al intentar conectarte te da problemas porque ya has usado la IP de la maquina nueva para otra máquina, tendremos que entrar en C: > Usuarios > TuUsuario > .ssh y eliminamos un archivo llamado “known_hosts” y volver a ejecutar el comando “ssh miadmin@192.168.3.204”.

```
C:\Users\daw2>ssh miadmin@192.168.3.204
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@   WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!   @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!
Someone could be eavesdropping on you right now (man-in-the-middle attack)!
It is also possible that a host key has just been changed.
The fingerprint for the ECDSA key sent by the remote host is
SHA256:P2PA+gsJ4IG2G9BLN0VggONunE0/1ENT5JRi6htw/8M.
Please contact your system administrator.
Add correct host key in C:\Users\daw2\.ssh\known_hosts to get rid of this message.
Offending ECDSA key in C:\Users\daw2\.ssh\known_hosts:1
ECDSA host key for 192.168.3.204 has changed and you have requested strict checking.
Host key verification failed.
```

equipo > SISTEMA (C:) > Usuarios > daw2 > .ssh

Nombre	Fecha de modificación	Tipo	Tamaño
known_hosts	03/10/2023 10:50	Archivo	1 KB

```
Active ESM Apps para recibir futuras actualizaciones de
Vea https://ubuntu.com/esm o ejecute «sudo pro status»

Failed to connect to https://changelogs.ubuntu.com/met

*** System restart required ***
Last login: Tue Oct 3 08:17:46 2023
miadmin@rsp-usedlimpia:~$
```

2. Cuentas administradoras y cuenta de desarrollador

3. Navegadores

COMPLETAR

4. Filezilla

COMPLETAR

5. Notepad++

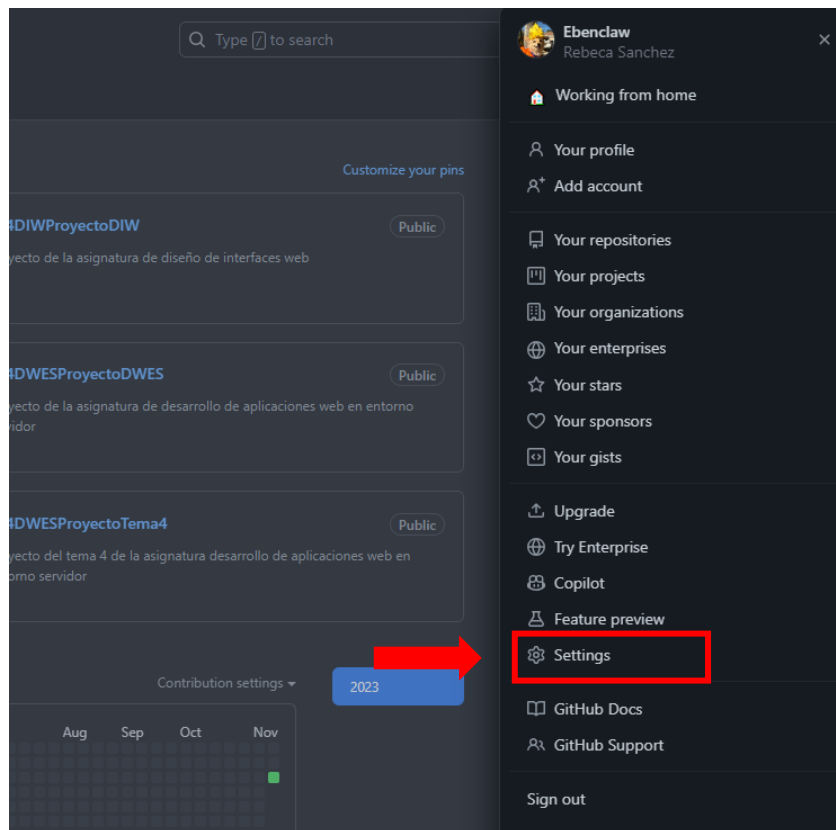
6. NetBeans

COMPLETAR

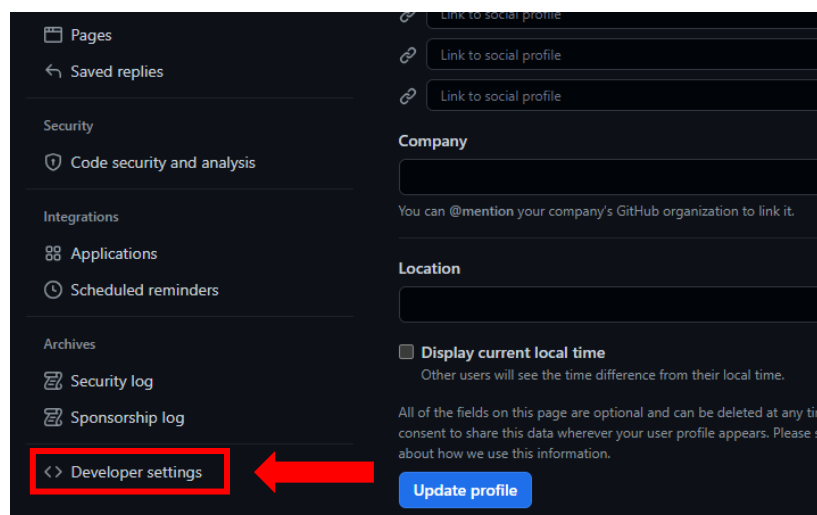
c. GITHUB – INTERNET

1. Creación de token desde GitHub

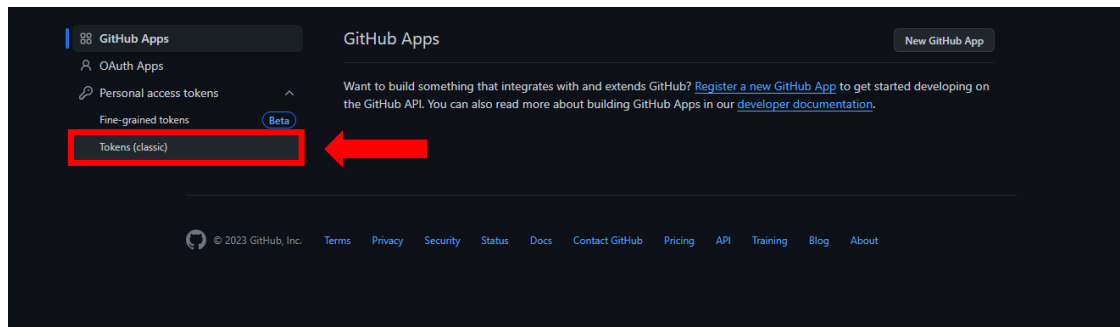
Para comenzar a trabajar en GitHub, necesitamos tener una cuenta registrada y una vez la tengamos, comenzaremos creando un token. Un token es una contraseña de autenticación usada para manejar la API de GitHub, para crearlo nos dirigiremos a la derecha de la parte superior de la página donde aparece nuestro perfil y cuando le hagamos clic, se desplegará un menú con distintas opciones. Nos dirigiremos a la opción de “settings”



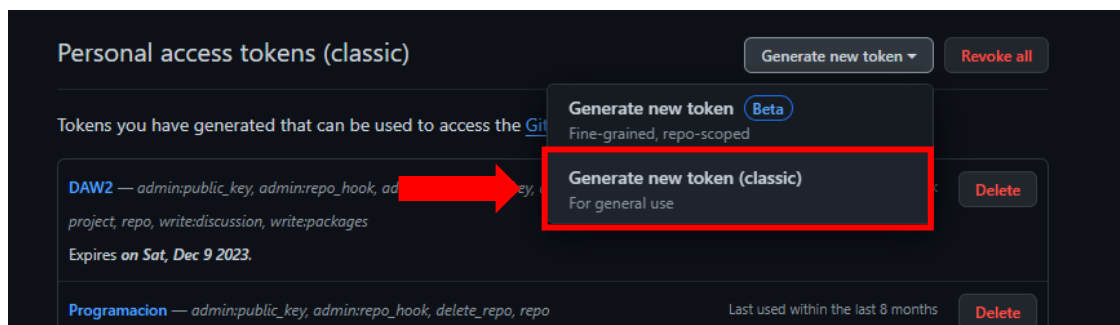
En la siguiente ventana, nos desplazaremos hasta la parte inferior y seleccionaremos la opción “developer settings” del menu de la izquierda



Nos dirigimos a “Personal access tokens”>”Tokens (clasic)”



A continuación, nos aparecerá la lista de nuestros tokens. Para crear uno nuevo, desplegamos el botón “Generate new token” y seleccionamos la opción de “Generate new token (clasic)”



COMPLETAR

2. Creación de un repositorio

1. Creacion de token

2. Creacion de un repositorio

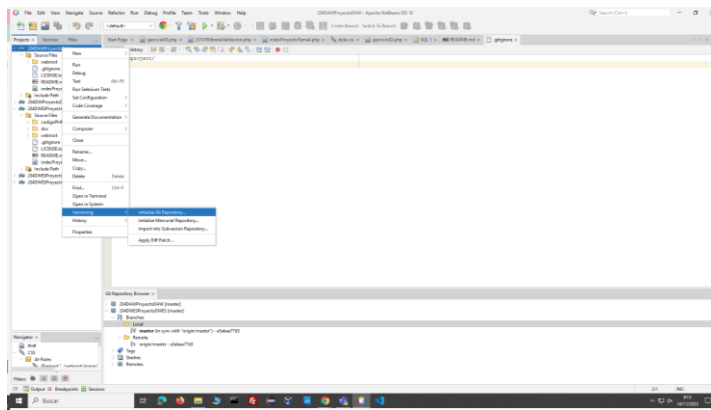
2.1. Con entorno grafico desde netbeans

2.2. Con modo comando

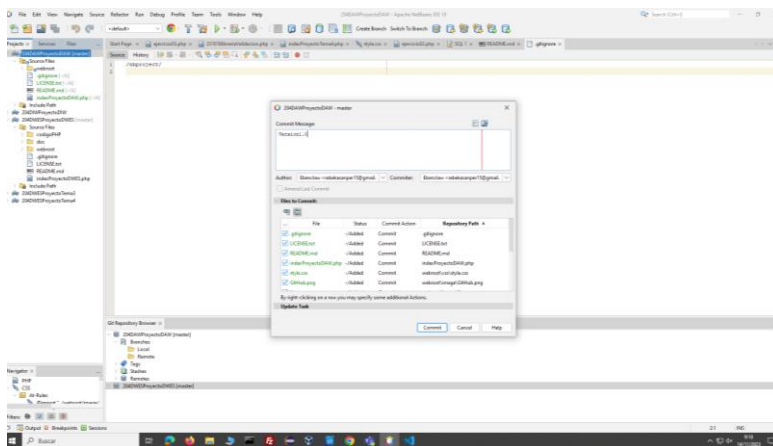
-Se crea el repositorio en github vacio (desde el navegador)

-En netbeans o en la carpeta de local, añadimos un README.md y una LICENCE.txt en cada proyecto y un .gitignore con el contenido /nbproject/ para que no suba esa carpeta al repositorio remoto

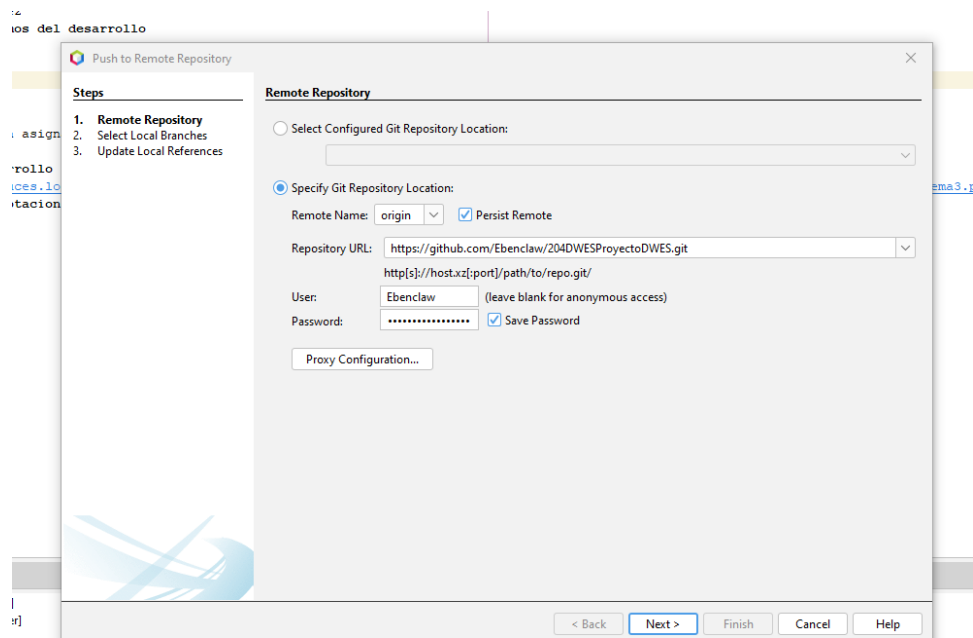
-inicializamos el versioning git desde netbeans



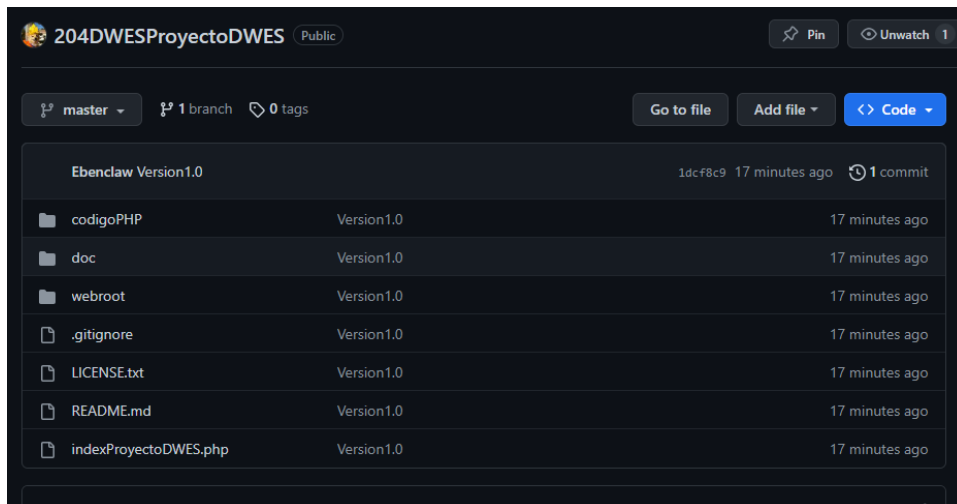
-hacemos el primer comit en local en master con la Version1.0



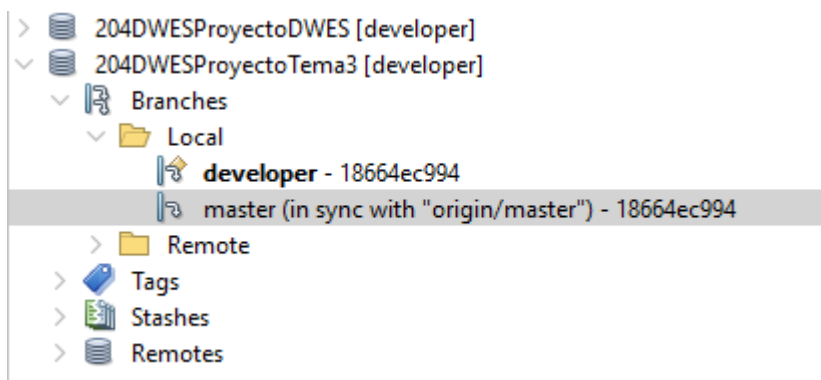
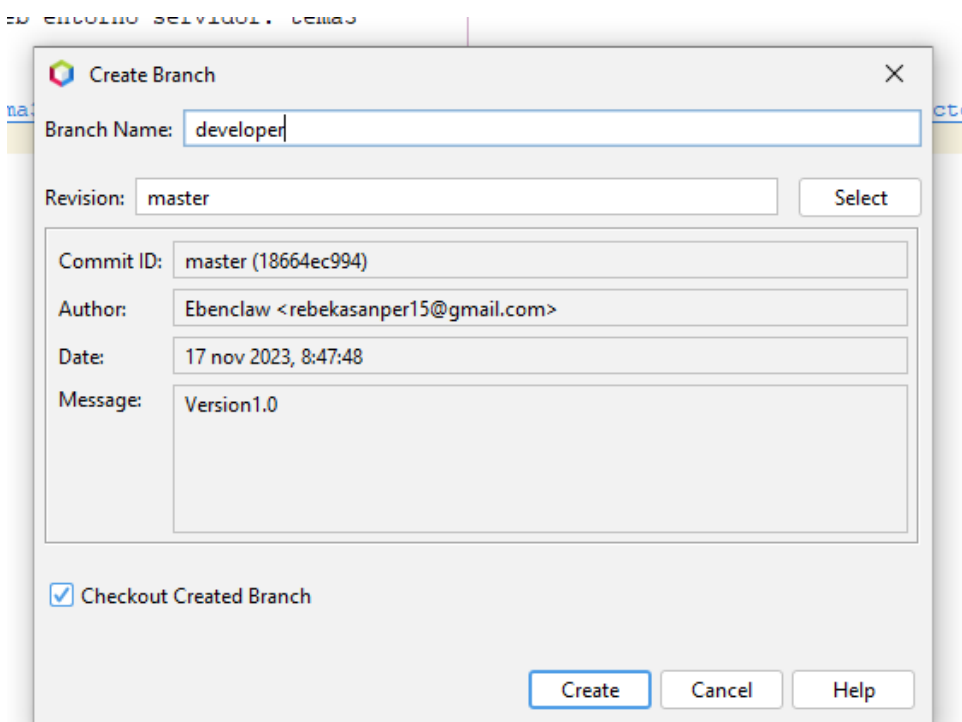
-hacemos un push desde netbeans al repositorio de github



-comprobamos que tenemos el proyecto subido en github



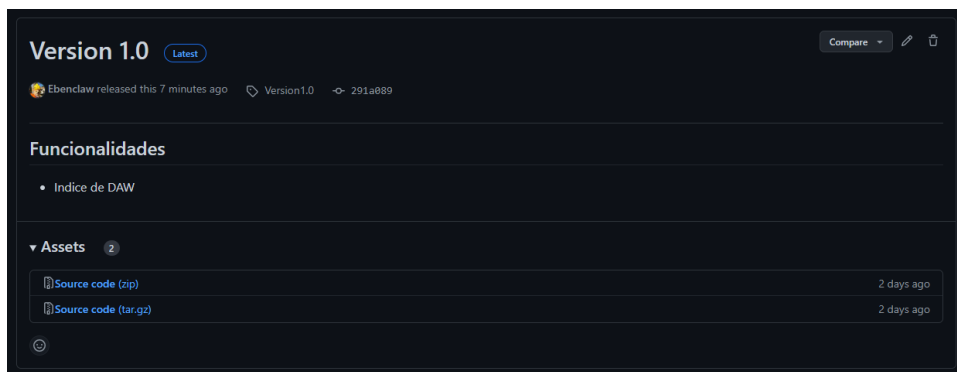
-se crea una rama a partir de la version 1.0 de la master en local llamada “developer” con
git>branch/tag>create branch



3. CREAR RELEASE

-en el proyecto de github a la derecha pone create a new release

The screenshot shows the 'Create a new release' form in GitHub. At the top, there are tabs for 'Releases' and 'Tags'. Below them, a dropdown menu shows 'Version1.0' and a 'Target: master' button. A message states: 'Excellent! This tag will be created from the target when you publish this release.' There is a text input field containing 'Version 1.0'. Below this are 'Write' and 'Preview' tabs. A rich text editor follows, with a toolbar containing icons for bold, italic, link, list, quote, code, and other formatting options. The text area contains '## Funcionalidades' and '- Indice de DAW'. Below the text area is a section for attaching files, with a message: 'Attach files by dragging & dropping, selecting or pasting them.' and a button to 'Generate release notes'. At the bottom, there is a checkbox for 'Set as a pre-release' and two buttons: 'Publish release' and 'Save draft'.



```
miadmin@rsp-used:~/204DAWProyectoDAW$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
miadmin@rsp-used:~/204DAWProyectoDAW$ git log
commit 291a08901ab2563467093777e86298c798653e2f (HEAD -> master, tag: Version1.0, origin/master, origin/HEAD)
Author: Ebenclaw <rebekasanper15@gmail.com>
Date: Tue Nov 14 09:22:00 2023 +0100

    Version1.0
miadmin@rsp-used:~/204DAWProyectoDAW$ git branch developer
miadmin@rsp-used:~/204DAWProyectoDAW$ git checkout developer
Switched to branch 'developer'
miadmin@rsp-used:~/204DAWProyectoDAW$ git branch
* developer
master
```

```

miadmin@rsp-used:~/204DAWProyectoDAW/dns$ cp /etc/bind/db.rebeca.local .
miadmin@rsp-used:~/204DAWProyectoDAW/dns$ cp /etc/bind/db.3.168.192.in-addr.arpa .
miadmin@rsp-used:~/204DAWProyectoDAW/dns$ cp /etc/bind/named.conf.local .
miadmin@rsp-used:~/204DAWProyectoDAW/dns$ ls
db.3.168.192.in-addr.arpa  db.rebeca.local  named.conf.local
miadmin@rsp-used:~/204DAWProyectoDAW/dns$ git status
On branch developer
Untracked files:
  (use "git add <file>..." to include in what will be committed)
  ./

nothing added to commit but untracked files present (use "git add" to track)
miadmin@rsp-used:~/204DAWProyectoDAW/dns$ git add .
miadmin@rsp-used:~/204DAWProyectoDAW/dns$ git status
On branch developer
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   db.3.168.192.in-addr.arpa
    new file:   db.rebeca.local
    new file:   named.conf.local

miadmin@rsp-used:~/204DAWProyectoDAW/dns$ git commit -m "ficheros de configuracion DNS"
Author identity unknown

*** Please tell me who you are.

Run

  git config --global user.email "you@example.com"
  git config --global user.name "Your Name"

to set your account's default identity.
Omit --global to set the identity only in this repository.

fatal: unable to auto-detect email address (got 'miadmin@rsp-used.(none)')
miadmin@rsp-used:~/204DAWProyectoDAW/dns$ git config --global user.email "rebekasanper15@gmail.com"
miadmin@rsp-used:~/204DAWProyectoDAW/dns$ git config --global user.name "Ebenclaw"
miadmin@rsp-used:~/204DAWProyectoDAW/dns$ git commit -m "ficheros de configuracion DNS"
[developer a49ee92] ficheros de configuracion DNS
 3 files changed, 50 insertions(+)
 create mode 100644 dns/db.3.168.192.in-addr.arpa
 create mode 100644 dns/db.rebeca.local
 create mode 100644 dns/named.conf.local
miadmin@rsp-used:~/204DAWProyectoDAW/dns$

miadmin@rsp-used:~/204DAWProyectoDAW$ mkdir apache
miadmin@rsp-used:~/204DAWProyectoDAW$ ls /etc/apache2
apache2.conf  conf-enabled  magic  mods-enabled  sites-available
conf-available  envvars  mods-available  ports.conf  sites-enabled
miadmin@rsp-used:~/204DAWProyectoDAW$ cd apa
-bash: cd: apa: No such file or directory
miadmin@rsp-used:~/204DAWProyectoDAW$ cd apache/
miadmin@rsp-used:~/204DAWProyectoDAW/apache$ cp /etc/apache2/apache2.conf .
miadmin@rsp-used:~/204DAWProyectoDAW/apache$ cp /etc/apache2/sites-available/000-default.conf .
miadmin@rsp-used:~/204DAWProyectoDAW/apache$ ls
000-default.conf  apache2.conf
miadmin@rsp-used:~/204DAWProyectoDAW/apache$ git add .
miadmin@rsp-used:~/204DAWProyectoDAW/apache$ git commit -m "ficheros de configuracion Apache"
[developer 67e64af] ficheros de configuracion Apache
 2 files changed, 261 insertions(+)
 create mode 100644 apache/000-default.conf
 create mode 100644 apache/apache2.conf
miadmin@rsp-used:~/204DAWProyectoDAW/apache$

miadmin@rsp-used:~/204DAWProyectoDAW$ mkdir ssh
miadmin@rsp-used:~/204DAWProyectoDAW$ ls /etc/ssh/ssh
ssh_config  sshd_config.d/  ssh_host_ecdsa_key.pub  ssh_host_rsa_key.pub
ssh_config.d/  ssh_host_dsa_key  ssh_host_ed25519_key  ssh_import_id
ssh_config.d.backup/  ssh_host_dsa_key.pub  ssh_host_ed25519_key.pub  ssh_host_rsa_key.pub
sshd_config  ssh_host_ecdsa_key  ssh_host_rsa_key
miadmin@rsp-used:~/204DAWProyectoDAW$ ls /etc/ssh/sshd_config
/etc/ssh/sshd_config
miadmin@rsp-used:~/204DAWProyectoDAW$ cd ssh
miadmin@rsp-used:~/204DAWProyectoDAW/ssh$ cp /etc/ssh/sshd_config .
miadmin@rsp-used:~/204DAWProyectoDAW/ssh$ ls
sshd_config
miadmin@rsp-used:~/204DAWProyectoDAW/ssh$ git add .
miadmin@rsp-used:~/204DAWProyectoDAW/ssh$ git commit -m "ficheros de configuracion SSH"
[developer b157467] ficheros de configuracion SSH
 1 file changed, 129 insertions(+)
 create mode 100644 ssh/sshd_config
miadmin@rsp-used:~/204DAWProyectoDAW/ssh$

```

```

miadmin@rsp-used:~/204DAWProyectoDAW/ssh$ git log
commit b1574678034460447154b6b8e1598815199f4612 (HEAD -> developer)
Author: Ebenclaw <rebekasanper15@gmail.com>
Date: Thu Nov 16 09:29:18 2023 +0100

    ficheros de configuracion SSH

commit 67e64affa4c643a65e612c77058dc52c5045cc21
Author: Ebenclaw <rebekasanper15@gmail.com>
Date: Thu Nov 16 09:26:31 2023 +0100

    ficheros de configuracion Apache

commit a49ee92181d52fe19ed3a96b49ceda575f1f2ca6
Author: Ebenclaw <rebekasanper15@gmail.com>
Date: Thu Nov 16 09:17:52 2023 +0100

    ficheros de configuracion DNS

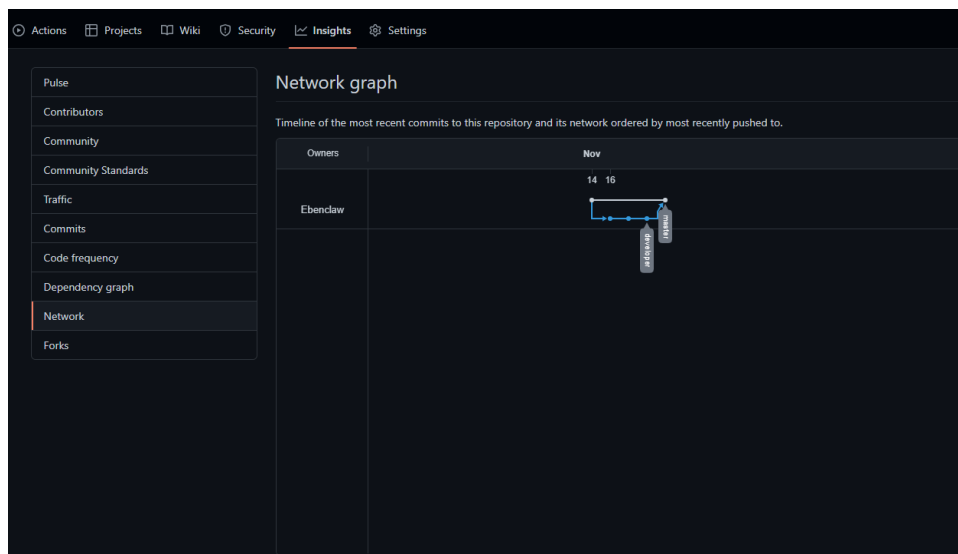
commit 291a08901ab2563467093777e86298c798652f (tag: Version1.0, origin/master, origin/HEAD, master)
Author: Ebenclaw <rebekasanper15@gmail.com>
Date: Tue Nov 14 09:22:00 2023 +0100

    Version1.0

miadmin@rsp-used:~/204DAWProyectoDAW/ssh$ git push origin developer
Username for 'https://github.com': Ebenclaw
Password for 'https://Ebenclaw@github.com':
Enumerating objects: 16, done.
Counting objects: 100% (16/16), done.
Compressing objects: 100% (14/14), done.
Writing objects: 100% (15/15), 6.85 KiB | 876.00 KiB/s, done.
Total 15 (delta 3), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (3/3), done.
remote:
remote: Create a pull request for 'developer' on GitHub by visiting:
remote: https://github.com/Ebenclaw/204DAWProyectoDAW/pull/new/developer
remote:
To https://github.com/Ebenclaw/204DAWProyectoDAW.git
 * [new branch]      developer -> developer
miadmin@rsp-used:~/204DAWProyectoDAW/ssh$

miadmin@rsp-used:~/204DAWProyectoDAW$ git checkout master
Already on 'master'
Your branch is up to date with 'origin/master'.
miadmin@rsp-used:~/204DAWProyectoDAW$ git branch
  developer
* master
miadmin@rsp-used:~/204DAWProyectoDAW$ git merge --no-ff developer
Merge made by the 'ort' strategy.
 apache/000-default.conf | 33 ++++++++
 apache/apache2.conf     | 228 ++++++++
 dns/db.3.168.192.in-addr.arpa | 15 +++++
 dns/db.rebeca.local     | 18 ++++++
 dns/named.conf.local    | 17 ++++++
 ssh/sshd_config         | 129 ++++++++
 6 files changed, 440 insertions(+)
 create mode 100644 apache/000-default.conf
 create mode 100644 apache/apache2.conf
 create mode 100644 dns/db.3.168.192.in-addr.arpa
 create mode 100644 dns/db.rebeca.local
 create mode 100644 dns/named.conf.local
 create mode 100644 ssh/sshd_config
miadmin@rsp-used:~/204DAWProyectoDAW$

```




```

dns/named.conf.local | 17 ++++++
ssh/sshd_config | 129 ++++++
6 files changed, 440 insertions(+)
create mode 100644 apache/000-default.conf
create mode 100644 apache/apache2.conf
create mode 100644 dns/db.3.168.192.in-addr.arpa
create mode 100644 dns/db.rebeca.local
create mode 100644 dns/named.conf.local
create mode 100644 ssh/sshd_config
miadmin@rsp-used:~/204DAWProyectoDAW$ git push origin
Username for 'https://github.com': Ebenclaw
Password for 'https://Ebenclaw@github.com':
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 228 bytes | 228.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Ebenclaw/204DAWProyectoDAW.git
291a089..0b1810c master -> master
miadmin@rsp-used:~/204DAWProyectoDAW$ git branch
  developer
* master
miadmin@rsp-used:~/204DAWProyectoDAW$ git checkout developer
Switched to branch 'developer'

miadmin@rsp-used:~/204DAWProyectoDAW$ git merge --ff master
Updating b157467..0b1810c
Fast-forward
miadmin@rsp-used:~/204DAWProyectoDAW$ git push origin
fatal: The current branch developer has no upstream branch.
To push the current branch and set the remote as upstream, use

    git push --set-upstream origin developer

miadmin@rsp-used:~/204DAWProyectoDAW$ git push origin developer
Username for 'https://github.com': Ebenclaw
Password for 'https://Ebenclaw@github.com':
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Ebenclaw/204DAWProyectoDAW.git
b157467..0b1810c developer -> developer
miadmin@rsp-used:~/204DAWProyectoDAW$

miadmin@rsp-used:~/204DAWProyectoDAW$ cd 204DAWProyectoDAW/
miadmin@rsp-used:~/204DAWProyectoDAW$ git log --graph --oneline --all
* 0b1810c (HEAD -> developer, origin/master, origin/developer, origin/HEAD, master) Merge branch 'developer'
|
| * b157467 ficheros de configuracion SSH
| * 67e64af ficheros de configuracion Apache
| * a49ee92 ficheros de configuracion DNS
|/
* 291a089 (tag: Version1.0) Version1.0
miadmin@rsp-used:~/204DAWProyectoDAW$ git fetch --all
Fetching origin
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 647 bytes | 71.00 KiB/s, done.
From https://github.com/Ebenclaw/204DAWProyectoDAW
0b1810c..5c1f32e developer -> origin/developer
miadmin@rsp-used:~/204DAWProyectoDAW$ git log --graph --oneline --all
* 5c1f32e (origin/developer) Create file.txt
* 0b1810c (HEAD -> developer, origin/master, origin/HEAD, master) Merge branch 'developer'
|
| * b157467 ficheros de configuracion SSH
| * 67e64af ficheros de configuracion Apache
| * a49ee92 ficheros de configuracion DNS
|/
* 291a089 (tag: Version1.0) Version1.0
miadmin@rsp-used:~/204DAWProyectoDAW$ git diff 5c1f32e 0b1810c
diff --git a/file.txt b/file.txt
deleted file mode 100644
index 8b13789..0000000
--- a/file.txt
+++ /dev/null
@@ -1,0,0 @@
-

```

