

Nombre: Tiffany Andrea Jordán Uquillas

Proyecto #1: Servidor Moodle 4.0 con Nginx

- La instalación de este servidor lo realizamos en una máquina virtual sin modificaciones, empezamos ejecutando `sudo apt-get update`, y luego `sudo apt-get upgrade` para tener actualizada los paquetes de nuestra máquina virtual.

```
tiffany@server01:~$ sudo apt-get update
[sudo] password for tiffany:
Obj:1 http://ec.archive.ubuntu.com/ubuntu jammy InRelease
Des:2 http://ec.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Des:3 http://ec.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Des:4 http://ec.archive.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Des:5 http://ec.archive.ubuntu.com/ubuntu jammy/main Translation-es [332 kB]
Des:6 http://ec.archive.ubuntu.com/ubuntu jammy/restricted Translation-es [964 B]
Des:7 http://ec.archive.ubuntu.com/ubuntu jammy/universe Translation-es [1.356 kB]
Des:8 http://ec.archive.ubuntu.com/ubuntu jammy/multiverse Translation-es [68,2 kB]
Des:9 http://ec.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1.162 kB]
Des:10 http://ec.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [248 kB]
Des:11 http://ec.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [16,1 kB]
Des:12 http://ec.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [1.114 kB]
0% [8 Translation-es store 0 B] [12 Packages 524 kB/1.114 kB 47%]
```

```
tiffany@server01:~$ sudo apt-get upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
Los siguientes paquetes se han retenido:
  linux-generic linux-headers-generic linux-image-generic ubuntu-advantage-tools
Se actualizarán los siguientes paquetes:
  apt apt-utils bind9-dnsutils bind9-host bind9-libs cloud-init curl distro-info-data file
  firmware-sof-signed gauk git git-man initramfs-tools initramfs-tools-bin initramfs-tools-core
  intel-microcode irqbalance kpartx libapt-pkg6.0 libc-bin libc6 libcurl3-gnutls libcurl4
  libgssapi-krb5-2 libk5crypto3 libkrb5-3 libkrb5support0 libldap-2.5-0 libldap-common
  libmagic-mgc libmagic1 libnetplan0 libnss-systemd libpam-systemd libprocps8 libsgutils2-2
  libssl3 libsystemd0 libudev1 libx11-6 libx11-data linux-firmware locales multipath-tools
  netplan.io open-vm-tools openssh-client openssh-server openssh-sftp-server openssl procps
  python3-urllib3 sg3-utils sg3-utils-udev sosreport systemd systemd-sysv systemd-timesyncd
  thermald ubuntu-drivers-common udev vim vim-common vim-runtime vim-tiny xxd
67 actualizados, 0 nuevos se instalarán, 0 para eliminar y 4 no actualizados.
Se necesita descargar 309 MB de archivos.
Se utilizarán 7.006 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] _
```

Instalación de Nginx, MariaDB y PHP

Antes de comenzar, se debe instalar Apache, MariaDB, PHP y otras bibliotecas PHP en su sistema. Primero, instalamos el servidor Apache y MariaDB usando el siguiente comando:

```
sudo apt-get install nginx mariadb-server -y
```

```
Des:37 http://ec.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libtiff5 amd64 4.3.0-6ubuntu0.6
[183 kB]
Des:38 http://ec.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libxpm4 amd64 1:3.5.12-1ubuntu0.
22.04.2 [36,7 kB]
Des:39 http://ec.archive.ubuntu.com/ubuntu jammy/main amd64 libgd3 amd64 2.3.0-2ubuntu2 [129 kB]
Des:40 http://ec.archive.ubuntu.com/ubuntu jammy/main amd64 libhtml-template-perl all 2.97-1.1 [59,1
kB]
Des:41 http://ec.archive.ubuntu.com/ubuntu jammy/main amd64 libtimedate-perl all 2.3300-2 [34,0 kB]
Des:42 http://ec.archive.ubuntu.com/ubuntu jammy/main amd64 libhttp-date-perl all 6.05-1 [9.920 B]
Des:43 http://ec.archive.ubuntu.com/ubuntu jammy/main amd64 libio-html-perl all 1.004-2 [15,4 kB]
Des:44 http://ec.archive.ubuntu.com/ubuntu jammy/main amd64 liblwp-mediatypes-perl all 6.04-1 [19,5
kB]
Des:45 http://ec.archive.ubuntu.com/ubuntu jammy/main amd64 libhttp-message-perl all 6.36-1 [76,8 kB]
```

Nombre: Tiffany Andrea Jordán Uquillas

De forma predeterminada, Ubuntu 22.04 se entrega con la versión PHP 8.1 y Moodle no es compatible con esta versión de PHP. Por lo tanto, necesitarás instalar PHP 7.4 en tu servidor.

Primero, instalamos todas las dependencias requeridas usando el siguiente comando:

```
sudo apt install software-properties-common ca-certificates lsb-release apt-transport-https -y
```

```
tiffany@server01:~$ sudo apt install software-properties-common ca-certificates lsb-release apt-transport-https -y
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
lsb-release ya está en su versión más reciente (11.1.0ubuntu4).
fijado lsb-release como instalado manualmente.
ca-certificates ya está en su versión más reciente (20230311ubuntu0.22.04.1).
fijado ca-certificates como instalado manualmente.
software-properties-common ya está en su versión más reciente (0.99.22.7).
fijado software-properties-common como instalado manualmente.
Se instalarán los siguientes paquetes NUEVOS:
  apt-transport-https
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 4 no actualizados.
Se necesita descargar 1.510 B de archivos.
Se utilizarán 170 kB de espacio de disco adicional después de esta operación.
Des:1 http://ec.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 apt-transport-https all 2.4.11-1 [1.510 B]
Descargados 1.510 B en 1s (2.183 B/s)
Seleccionando el paquete apt-transport-https previamente no seleccionado.
(Leyendo la base de datos ... 75524 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../apt-transport-https_2.4.11_all.deb ...
Desempaquetando apt-transport-https (2.4.11) ...
Configurando apt-transport-https (2.4.11) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
```

A continuación, agregamos el repositorio PHP a nuestro servidor con el siguiente comando:

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

Nombre: Tiffany Andrea Jordán Uquillas

```
tiffany@server01:~$ sudo add-apt-repository ppa:ondrej/php
PPA publishes dbgsym, you may need to include 'main/debug' component
Repository: 'deb https://ppa.launchpadcontent.net/ondrej/php/ubuntu/ jammy main'
Description:
Co-installable PHP versions: PHP 5.6, PHP 7.x, PHP 8.x and most requested extensions are included. Only Supported Versions of PHP (http://php.net/supported-versions.php) for Supported Ubuntu Releases (https://wiki.ubuntu.com/Releases) are provided. Don't ask for end-of-life PHP versions or Ubuntu releases, they won't be provided.

Debian oldstable and stable packages are provided as well: https://deb.sury.org/#debian-dpa

You can get more information about the packages at https://deb.sury.org

IMPORTANT: The <foo>-backports is now required on older Ubuntu releases.

BUGS&FEATURES: This PPA now has a issue tracker:
https://deb.sury.org/#bug-reporting

CAVEATS:
1. If you are using php-gearman, you need to add ppa:ondrej/pkg-gearman
2. If you are using apache2, you are advised to add ppa:ondrej/apache2
3. If you are using nginx, you are advised to add ppa:ondrej/nginx-mainline
   or ppa:ondrej/nginx

PLEASE READ: If you like my work and want to give me a little motivation, please consider donating regularly: https://donate.sury.org/

WARNING: add-apt-repository is broken with non-UTF-8 locales, see
https://github.com/oerdnj/deb.sury.org/issues/56 for workaround:

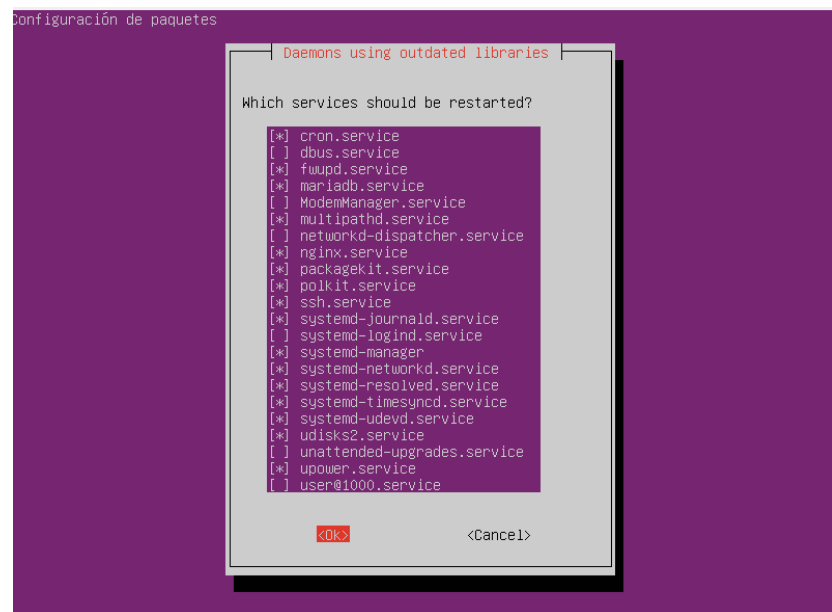
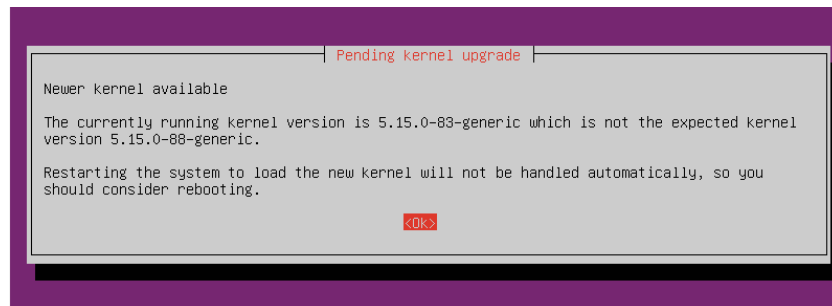
# LC_ALL=C.UTF-8 add-apt-repository ppa:ondrej/php
More info: https://launchpad.net/~ondrej/+archive/ubuntu/php
Adding repository.
Press [ENTER] to continue or Ctrl-c to cancel.
```

A continuación, actualizamos el repositorio usando el siguiente comando:

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

Nombre: Tiffany Andrea Jordán Uquillas

```
tiffany@server01:~$ sudo apt update
Obj:1 http://ec.archive.ubuntu.com/ubuntu jammy InRelease
Obj:2 https://ppa.launchpadcontent.net/ondrej/php/ubuntu jammy InRelease
Obj:3 http://ec.archive.ubuntu.com/ubuntu jammy-updates InRelease
Obj:4 http://ec.archive.ubuntu.com/ubuntu jammy-backports InRelease
Obj:5 http://ec.archive.ubuntu.com/ubuntu jammy-security InRelease
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se pueden actualizar 8 paquetes. Ejecute «apt list --upgradable» para verlos.
tiffany@server01:~$ sudo apt upgrade
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Calculando la actualización... Hecho
Se instalarán los siguientes paquetes NUEVOS:
  libabsl20210324 libaom3 libavif13 libdav1d5 libde265-0 libgav1-0 libheif1 libx265-199 libyuv0
  linux-headers-5.15.0-88 linux-headers-5.15.0-88-generic linux-image-5.15.0-88-generic
  linux-modules-5.15.0-88-generic linux-modules-extra-5.15.0-88-generic ubuntu-pro-client-110n
Se actualizarán los siguientes paquetes:
  libgd3 libpcre2-8-0 libpcre3 libxml2 linux-generic linux-headers-generic linux-image-generic
  ubuntu-advantage-tools
8 actualizados, 15 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
3 standard LTS security updates
Se necesita descargar 120 MB de archivos.
Se utilizarán 610 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] _
```

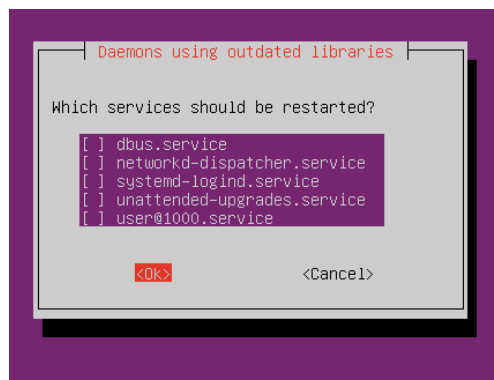
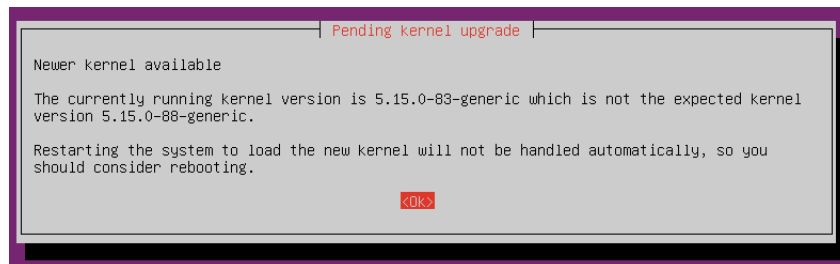


Una vez actualizado el repositorio, instalamos PHP con otras extensiones requeridas usando el siguiente comando:

Nombre: Tiffany Andrea Jordán Uquillas

```
sudo apt install php7.4 php7.4-fpm php7.4-common php7.4-mysql php7.4-gmp  
php7.4-curl php7.4-intl php7.4-mbstring php7.4-soap php7.4-xmlrpc php7.4-gd  
php7.4-xml php7.4-cli php7.4-zip unzip git curl -y
```

```
Descargados 5.740 kB en 3s (1.697 kB/s)  
Seleccionando el paquete libonig5:amd64 previamente no seleccionado.  
(Leyendo la base de datos ... 111217 ficheros o directorios instalados actualmente.)  
Preparando para desempaquetar .../00-libonig5_6.9.7.1-2build1_amd64.deb ...  
Desempaquetando libonig5:amd64 (6.9.7.1-2build1) ...  
Seleccionando el paquete libzip4:amd64 previamente no seleccionado.  
Preparando para desempaquetar .../01-libzip4_1.7.3-1ubuntu2_amd64.deb ...  
Desempaquetando libzip4:amd64 (1.7.3-1ubuntu2) ...  
Seleccionando el paquete php-common previamente no seleccionado.  
Preparando para desempaquetar .../02-php-common_2%3a93+ubuntu22.04.1+deb.sury.org+2_all.deb ...  
Desempaquetando php-common (2:93+ubuntu22.04.1+deb.sury.org+2) ...  
Seleccionando el paquete php7.4-common previamente no seleccionado.  
Preparando para desempaquetar .../03-php7.4-common_1%3a7.4.33-8+ubuntu22.04.1+deb.sury.org+1_amd64.deb ...  
Desempaquetando php7.4-common (1:7.4.33-8+ubuntu22.04.1+deb.sury.org+1) ...  
Seleccionando el paquete php7.4-json previamente no seleccionado.  
Preparando para desempaquetar .../04-php7.4-json_1%3a7.4.33-8+ubuntu22.04.1+deb.sury.org+1_amd64.deb ...  
Desempaquetando php7.4-json (1:7.4.33-8+ubuntu22.04.1+deb.sury.org+1) ...  
Seleccionando el paquete php7.4-opcache previamente no seleccionado.  
Preparando para desempaquetar .../05-php7.4-opcache_1%3a7.4.33-8+ubuntu22.04.1+deb.sury.org+1_amd64.deb ...  
Desempaquetando php7.4-opcache (1:7.4.33-8+ubuntu22.04.1+deb.sury.org+1) ...  
Seleccionando el paquete php7.4-readline previamente no seleccionado.  
Preparando para desempaquetar .../06-php7.4-readline_1%3a7.4.33-8+ubuntu22.04.1+deb.sury.org+1_amd64.deb ...  
Desempaquetando php7.4-readline (1:7.4.33-8+ubuntu22.04.1+deb.sury.org+1) ...  
Seleccionando el paquete php7.4-cli previamente no seleccionado.  
Preparando para desempaquetar .../07-php7.4-cli_1%3a7.4.33-8+ubuntu22.04.1+deb.sury.org+1_amd64.deb ...  
Desempaquetando php7.4-cli (1:7.4.33-8+ubuntu22.04.1+deb.sury.org+1) ...  
Seleccionando el paquete php7.4-fpm previamente no seleccionado.  
Preparando para desempaquetar .../08-php7.4-fpm_1%3a7.4.33-8+ubuntu22.04.1+deb.sury.org+1_amd64.deb ...  
Desempaquetando php7.4-fpm (1:7.4.33-8+ubuntu22.04.1+deb.sury.org+1) ...  
Progreso: [ 20% ] [#####.....]
```



Una vez que todos los paquetes estén instalados, editamos el archivo php.ini y cambiamos algunas configuraciones:

Nombre: Tiffany Andrea Jordán Uquillas

```
sudo nano /etc/php/7.4/fpm/php.ini
```

Cambiamos las siguientes líneas:

```
memory_limit = 256M

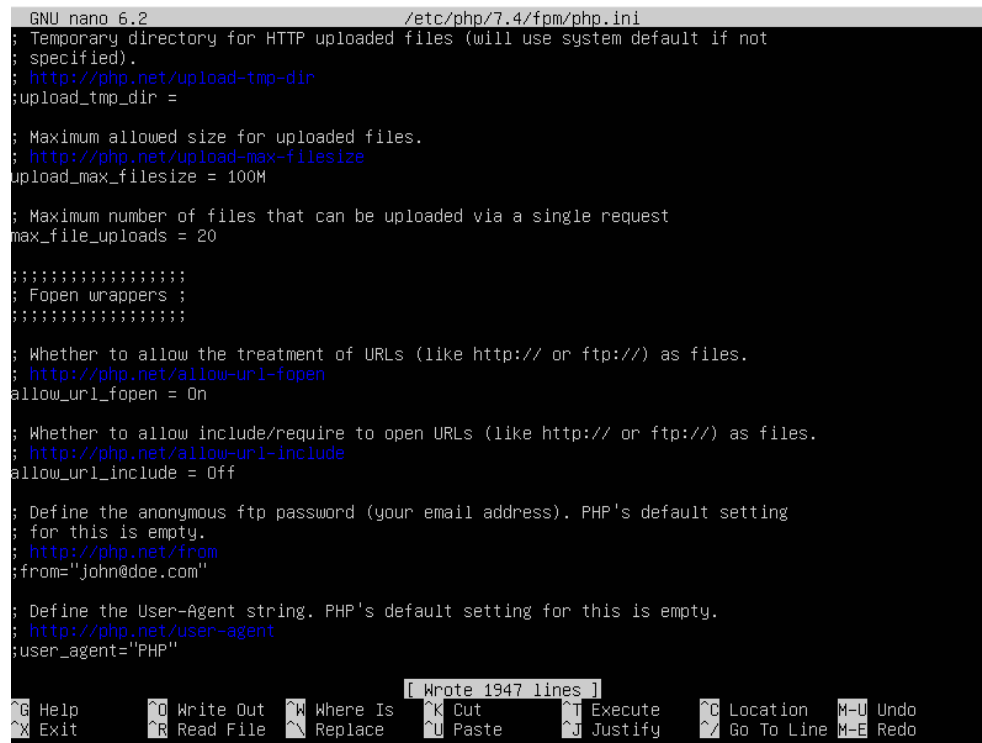
max_input_vars = 6000

cgi.fix_pathinfo = 0

upload_max_filesize = 100M

max_execution_time = 360

date.timezone = UTC
```



```
GNU nano 6.2 /etc/php/7.4/fpm/php.ini
; Temporary directory for HTTP uploaded files (will use system default if not
; specified).
; http://php.net/upload-tmp-dir
upload_tmp_dir =

; Maximum allowed size for uploaded files.
; http://php.net/upload-max-filesize
upload_max_filesize = 100M

; Maximum number of files that can be uploaded via a single request
max_file_uploads = 20

; Fopen wrappers
; Whether to allow the treatment of URLs (like http:// or ftp://) as files.
; http://php.net/allow-url-fopen
allow_url_fopen = On

; Whether to allow include/require to open URLs (like http:// or ftp://) as files.
; http://php.net/allow-url-include
allow_url_include = Off

; Define the anonymous ftp password (your email address). PHP's default setting
; for this is empty.
; http://php.net/from
from="john@doe.com"

; Define the User-Agent string. PHP's default setting for this is empty.
; http://php.net/user-agent
user_agent="PHP"

[ Wrote 1947 lines ]
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location  M-U Undo
^X Exit      ^R Read File ^M Replace   ^U Paste     ^J Justify   ^_ Go To Line M-E Redo
```

Guardamos y cerramos el archivo, luego reiniciamos el servicio PHP-FPM para aplicar los cambios:

```
sudo systemctl restart php7.4-fpm
```

Nombre: Tiffany Andrea Jordán Uquillas

Crear una base de datos para Moodle

Moodle usa MySQL o MariaDB como backend de base de datos, por lo que necesitarás crear una base de datos y un usuario para Moodle.

Primero, nos conectamos al shell MySQL con el siguiente comando:

```
mysql
```

Una vez que iniciemos sesión, pasamos a crear una base de datos y un usuario con el siguiente comando:

```
CREATE DATABASE moodledb;  
CREATE USER 'moodle'@'localhost' IDENTIFIED BY 'contraseña';
```

A continuación, otorgamos todos los privilegios a la base de datos de Moodle con el siguiente comando:

```
GRANT ALL ON moodledb.* TO 'moodle'@'localhost' WITH GRANT OPTION;
```

A continuación, borramos los privilegios y salimos de MySQL con el siguiente comando:

```
FLUSH PRIVILEGES;  
  
EXIT;
```

Nombre: Tiffany Andrea Jordán Uquillas

```
tiffany@server01:~$ sudo systemctl restart php7.4-fpm
tiffany@server01:~$ mysql
ERROR 1698 (28000): Access denied for user 'tiffany'@'localhost'
tiffany@server01:~$ sudo mysql
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 32
Server version: 10.6.12-MariaDB-0ubuntu0.22.04.1 Ubuntu 22.04

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> CREATE DATABASE moodledb;
Query OK, 1 row affected (0,001 sec)

MariaDB [(none)]> CREATE USER 'moodle'@'localhost' IDENTIFIED BY 'Killuahxh2004.';
Query OK, 0 rows affected (0,014 sec)

MariaDB [(none)]> GRANT ALL ON moodledb.* TO 'moodle'@'localhost' WITH GRANT OPTION;
Query OK, 0 rows affected (0,005 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0,001 sec)

MariaDB [(none)]> EXIT;_
```

A continuación, editamos el archivo de configuración predeterminado de MariaDB y definimos `innodb_file_format`:

```
sudo nano /etc/mysql/mariadb.conf.d/50-server.cnf
```

Agregamos las siguientes líneas dentro de la sección `[mysqld]`:

```
[mysqld]

innodb_file_format = Barracuda

innodb_file_per_table = 1

innodb_large_prefix = ON
```


Nombre: Tiffany Andrea Jordán Uquillas

```
GNU nano 6.2 /etc/mysql/mariadb.conf.d/50-server.cnf *
#
# These groups are read by MariaDB server.
# Use it for options that only the server (but not clients) should see
#
# this is read by the standalone daemon and embedded servers
[server]
#
# this is only for the mysqld standalone daemon
[mysqld]
innodb_file_format=Barracuda
innodb_file_per_table=1
innodb_large_prefix=ON
#
# * Basic Settings
#
#user                        = mysql
pid-file                    = /run/mysqld/mysqld.pid
basedir                    = /usr
#datadir                    = /var/lib/mysql
#tmpdir                     = /tmp
#
# Broken reverse DNS slows down connections considerably and name resolve is
# safe to skip if there are no "host by domain name" access grants
#skip-name-resolve
#
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address                = 127.0.0.1
#
# * Fine Tuning
#
File Name to Write: /etc/mysql/mariadb.conf.d/50-server.cnf
^G Help      M-D DOS Format  M-A Append    M-B Backup File
^O Cancel    M-M Mac Format  M-P Prepend   ^T Browse
```

Guardamos el archivo, luego reiniciamos el servicio MariaDB para aplicar los cambios:

```
sudo systemctl restart mariadb
```

```
tiffany@server01:~$ sudo systemctl restart mariadb
tiffany@server01:~$ _
```

Instalar Moodle en Ubuntu 22.04

Primero, cambiamos el directorio al directorio raíz de Apache y descargamos la última versión de Moodle con el siguiente comando:

```
cd /var/www/html
```

```
sudo git clone -b MOODLE_400_STABLE git://git.moodle.org/moodle.git moodle
```

Nombre: Tiffany Andrea Jordán Uquillas

```
tiffany@server01:~$ cd /
tiffany@server01:/$ cd /var/www/html/
tiffany@server01:/var/www/html$ sudo git clone -b MOODLE_400_STABLE git://git.moodle.org/moodle.git
moodle
Cloning into 'moodle'...
remote: Enumerating objects: 1425974, done.
remote: Counting objects: 100% (1350/1350), done.
remote: Compressing objects: 100% (501/501), done.
Receiving objects: 100% (1425974/1425974), 479.23 MiB | 4.27 MiB/s, done.
remote: Total 1425974 (delta 869), reused 1054 (delta 788), pack-reused 1424624
Resolving deltas: 100% (1071289/1071289), done.
Updating files: 100% (24345/24345), done.
tiffany@server01:/var/www/html$
```

A continuación, establecemos el permiso y la propiedad adecuados para Moodle:

```
mkdir -p /var/www/html/moodledata
chown -R www-data:www-data /var/www/html/moodle
chmod -R 755 /var/www/html/*
chown www-data:www-data /var/www/html/moodledata
```

```
tiffany@server01:/var/www/html$ mkdir -p /var/www/html/moodledata
mkdir: cannot create directory '/var/www/html/moodledata': Permission denied
tiffany@server01:/var/www/html$ sudo mkdir -p /var/www/html/moodledata
tiffany@server01:/var/www/html$ sudo chown -R www-data:www-data /var/www/html/moodle
tiffany@server01:/var/www/html$ sudo chmod -R 755 /var/www/html/*
tiffany@server01:/var/www/html$ sudo chown www-data:www-data /var/www/html/moodledata/
tiffany@server01:/var/www/html$
```

Configurar Nginx para Moodle

A continuación, debemos crear un archivo de configuración de host virtual Nginx para alojar Moodle:

```
sudo nano /etc/nginx/conf.d/moodle.conf
```

Agregamos las siguientes líneas:

```
server {
    listen 80;

    root /var/www/html/moodle;

    índex index.php index.html index.htm;
```

Nombre: Tiffany Andrea Jordán Uquillas

```
server_name moodle.ejemplo.com;

client_max_body_size 100M;

autoindex off;

location / {
    try_files $uri $uri/ =404;
}

location /dataroot/ {
    internal;
    alias /var/www/html/moodledata/;
}

location ~ [^/].php(/|$) {
    include snippets/fastcgi-php.conf;
    fastcgi_pass unix:/run/php/php7.4-fpm.sock;
    fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
    include fastcgi_params;
}
}
```

Nombre: Tiffany Andrea Jordán Uquillas

```
GNU nano 6.2 /etc/nginx/conf.d/moodle.conf *
server {
    listen 80;
    root /var/www/html/moodle;
    index index.php index.html index.htm;
    server_name moodle.example.com;

    client_max_body_size 100M;
    autoindex off;
    location / {
        try_files $uri $uri/ =404;
    }

    location /dataroot/ {
        internal;
        alias /var/www/html/moodledata/;
    }

    location ~ [^/]\.php(/|$) {
        include snippets/fastcgi-php.conf;
        fastcgi_pass unix:/run/php/php7.4-fpm.sock;
        fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
        include fastcgi_params;
    }
}
```

Guardamos y cerramos el archivo, luego verificamos Nginx para detectar cualquier error de sintaxis con el siguiente comando:

```
sudo nginx-t
```

Deberíamos obtener el siguiente resultado:

```
nginx: the configuration file /etc/nginx/nginx.conf is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
```

Finalmente, reiniciamos el servicio Nginx para aplicar los cambios:

```
sudo systemctl restart nginx
```

También podemos verificar el estado del servicio Nginx usando el siguiente comando y veremos que esta activo

```
sudo systemctl status nginx
```

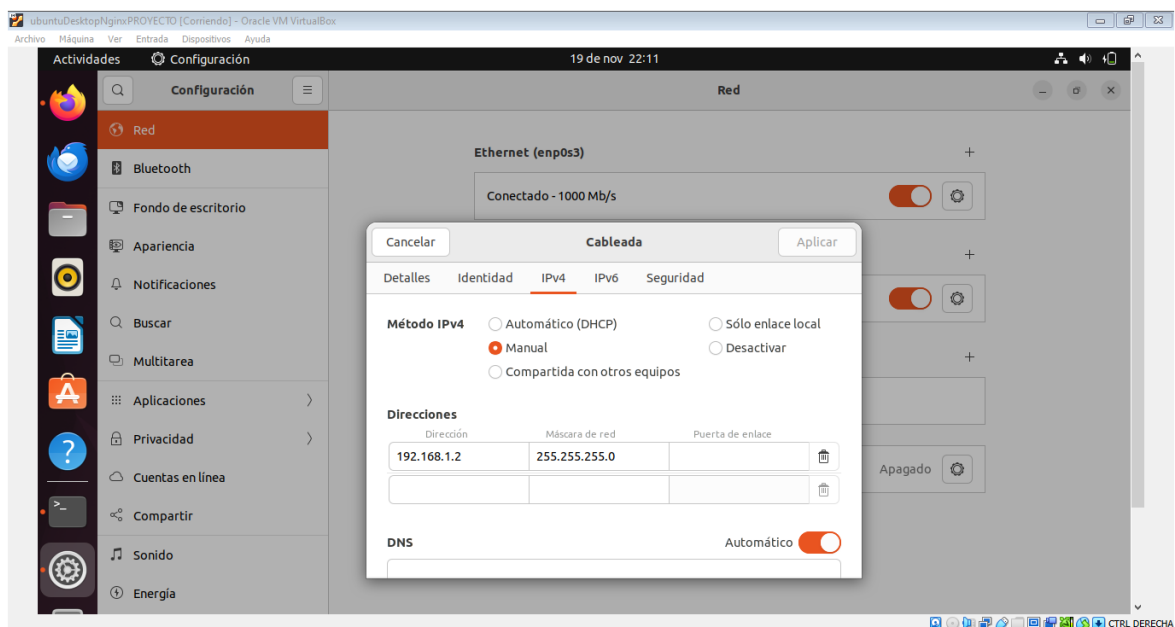
Nombre: Tiffany Andrea Jordán Uquillas

```
tiffany@server01:/var/www/html$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
tiffany@server01:/var/www/html$ sudo systemctl restart nginx
tiffany@server01:/var/www/html$ sudo systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
   Active: active (running) since Mon 2023-11-20 01:30:07 UTC; 6s ago
     Docs: man:nginx(8)
   Process: 27955 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0)
   Process: 27957 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0)
   Main PID: 27958 (nginx)
    Tasks: 2 (limit: 1013)
   Memory: 3.2M
      CPU: 95ms
   CGroup: /system.slice/nginx.service
           └─27958 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             └─27959 "nginx: worker process"

nov 20 01:30:07 server01 systemd[1]: Stopped A high performance web server and a reverse proxy serv
nov 20 01:30:07 server01 systemd[1]: Starting A high performance web server and a reverse proxy serv
nov 20 01:30:07 server01 systemd[1]: Started A high performance web server and a reverse proxy serv
lines 1-17/17 (END)
```

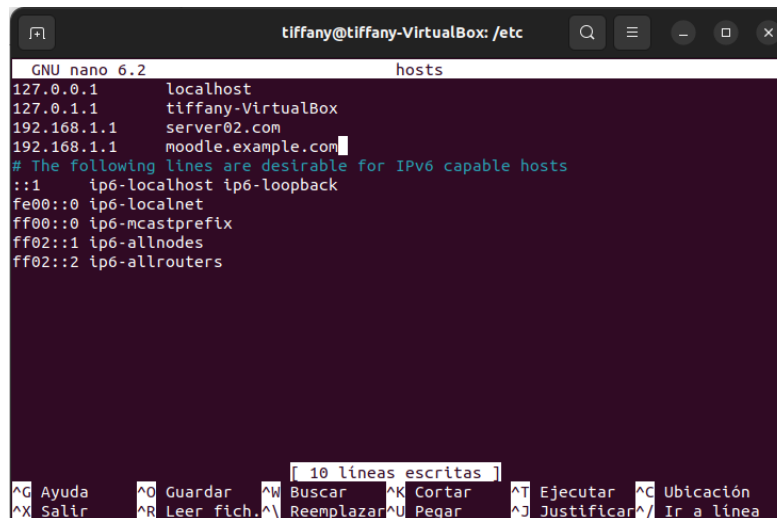
Acceder a la interfaz web de Moodle en Ubuntu Desktop

Dejamos configurado el enp0s8 de la siguiente manera:



Configuramos y añadimos con `sudo nano /etc/hosts` en el desktop la dirección de nuestro server junto con la web que va a proporcionar.

Nombre: Tiffany Andrea Jordán Uquillas

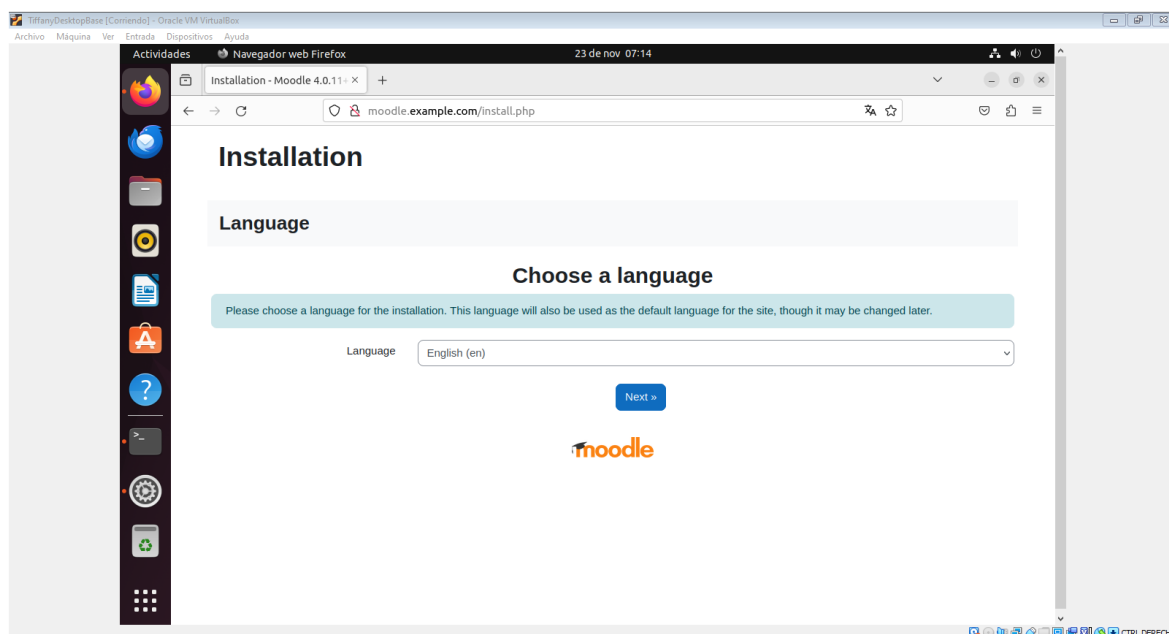


```
tiffany@tiffany-VirtualBox: /etc
GNU nano 6.2 hosts
127.0.0.1 localhost
127.0.1.1 tiffany-VirtualBox
192.168.1.1 server02.com
192.168.1.1 moodle.example.com
# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

10 líneas escritas

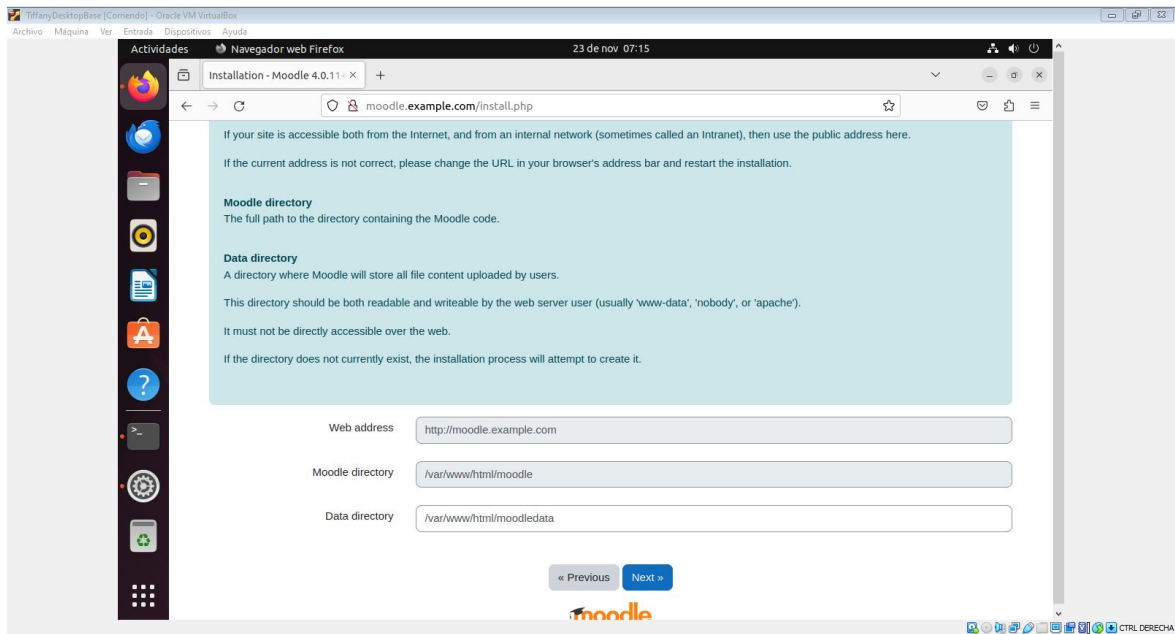
Ayuda Guardar Buscar Cortar Ejecutar Ubicación
Salir Leer fich. Reemplazar Pegar Justificar Ir a línea

Ahora, abrimos nuestro navegador web en el Ubuntu desktop y accedemos a la interfaz web de Moodle usando la URL **http://moodle.example.com** . Deberías ver la página de instalación de Moodle y seleccionamos el idioma:

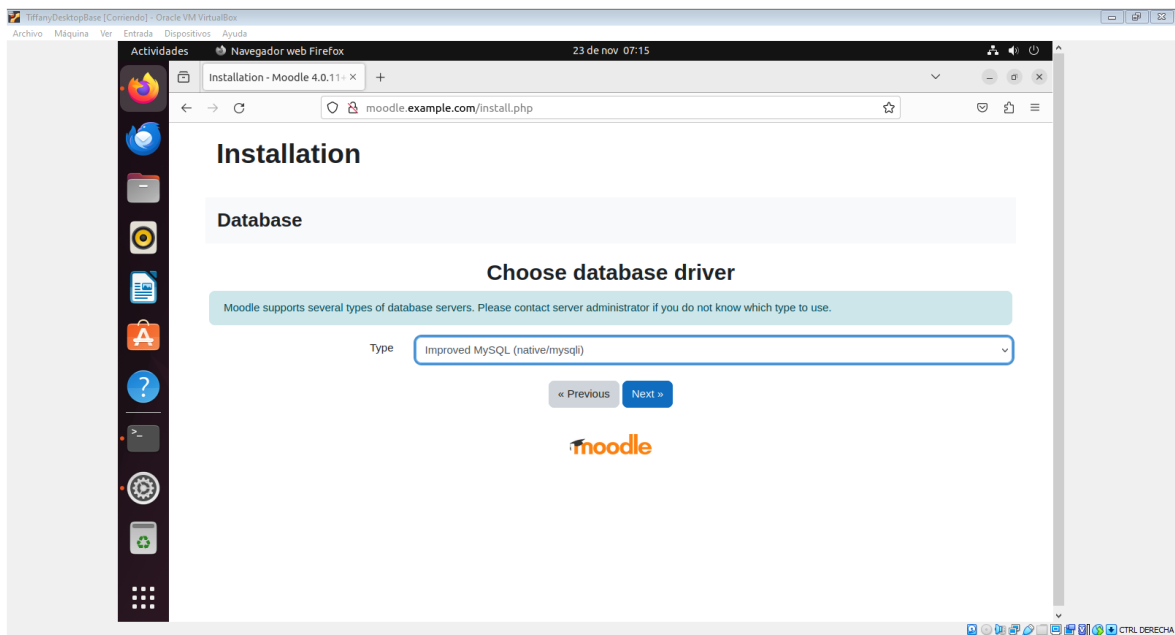


Proporcione su dirección web de Moodle, ruta del directorio, ruta del directorio de datos y haga clic en **Siguiente**. Deberías ver la siguiente página:

Nombre: Tiffany Andrea Jordán Uquillas

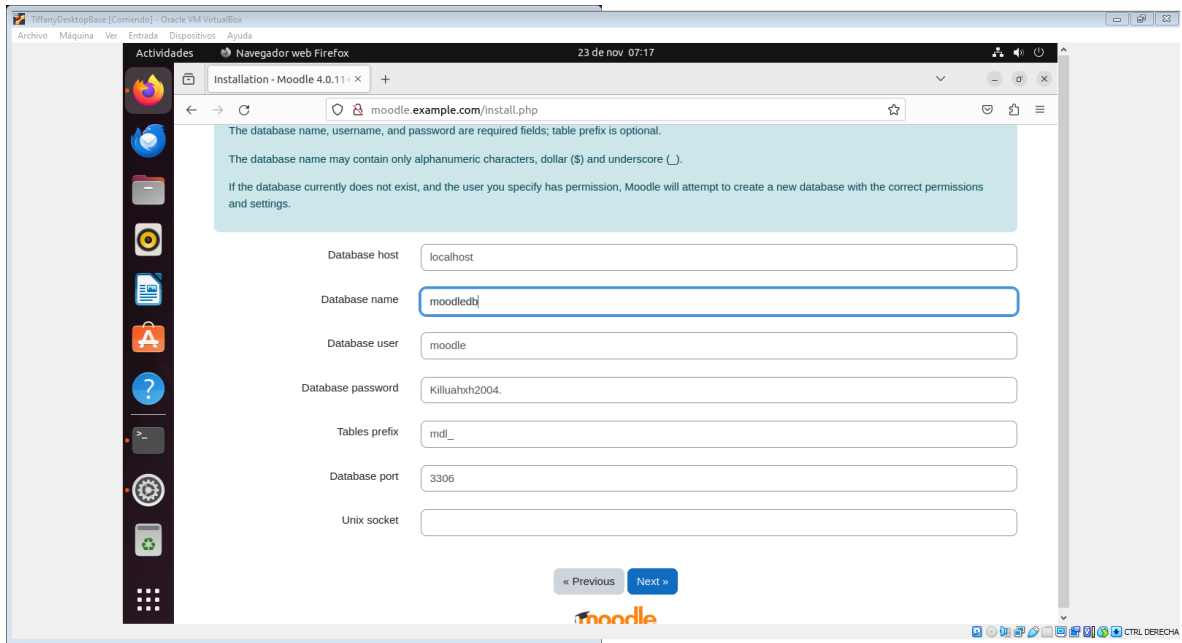


Seleccione el tipo de controlador de su base de datos y haga clic en Siguiente. Deberías ver la siguiente página:

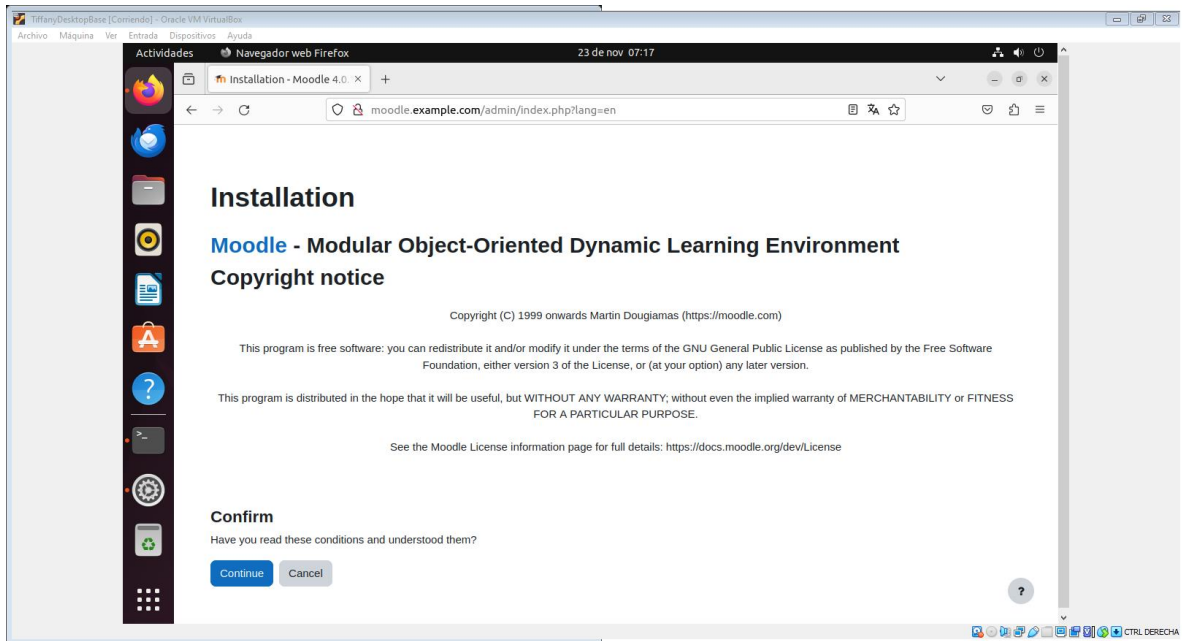


Proporcione el host de su base de datos, el nombre de la base de datos, el nombre de usuario, la contraseña y haga clic en **Siguiente**. Deberías ver la siguiente página:

Nombre: Tiffany Andrea Jordán Uquillas

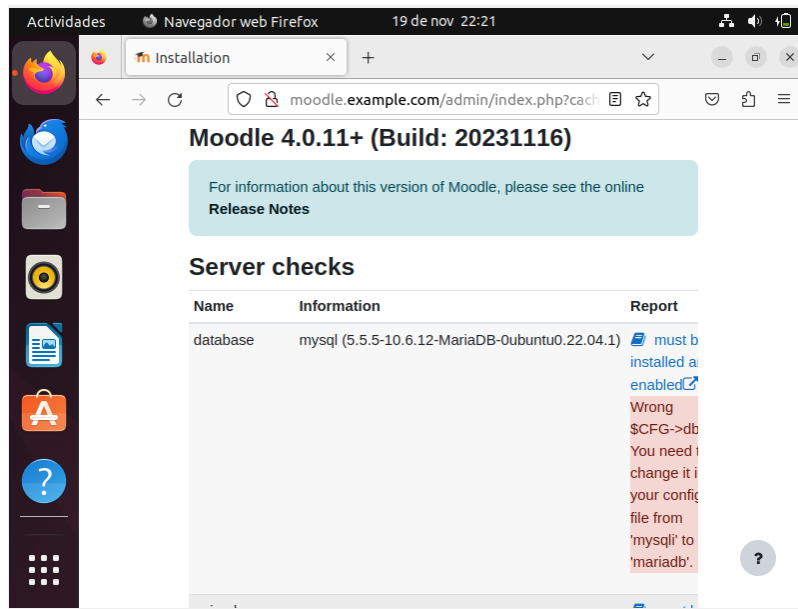


Haga clic en **Continuar** para confirmar todas las condiciones. Deberías ver la siguiente página:



Asegúrese de que todas las extensiones PHP requeridas estén instaladas, luego haga clic en **Continuar**. Deberías ver la siguiente página:

Nombre: Tiffany Andrea Jordán Uquillas



- Si te sale un error como en la imagen anterior simplemente deberíamos editar el archivo con `sudo nano /moodle/config.php`, allí cambiamos la parte dbtype tal como se muestra en la imagen:

Nombre: Tiffany Andrea Jordán Uquillas

```
GNU nano 6.2 moodie/config.php *
<?php // Moodle configuration file

unset($CFG);
global $CFG;
$CFG = new stdClass();

$CFG->dbtype      = 'mariadb';
$CFG->dblibrary   = 'native';
$CFG->dbhost      = 'localhost';
$CFG->dbname      = 'moodledb';
$CFG->dbuser      = 'moodle';
$CFG->dbpass      = 'Killuahxh2004.';
$CFG->prefix      = 'mdl_';
$CFG->dboptions   = array (
  'dbpersist' => 0,
  'dbport'    => 3306,
  'dbsocket'  => '',
  'dbcollation' => 'utf8mb4_general_ci',
);

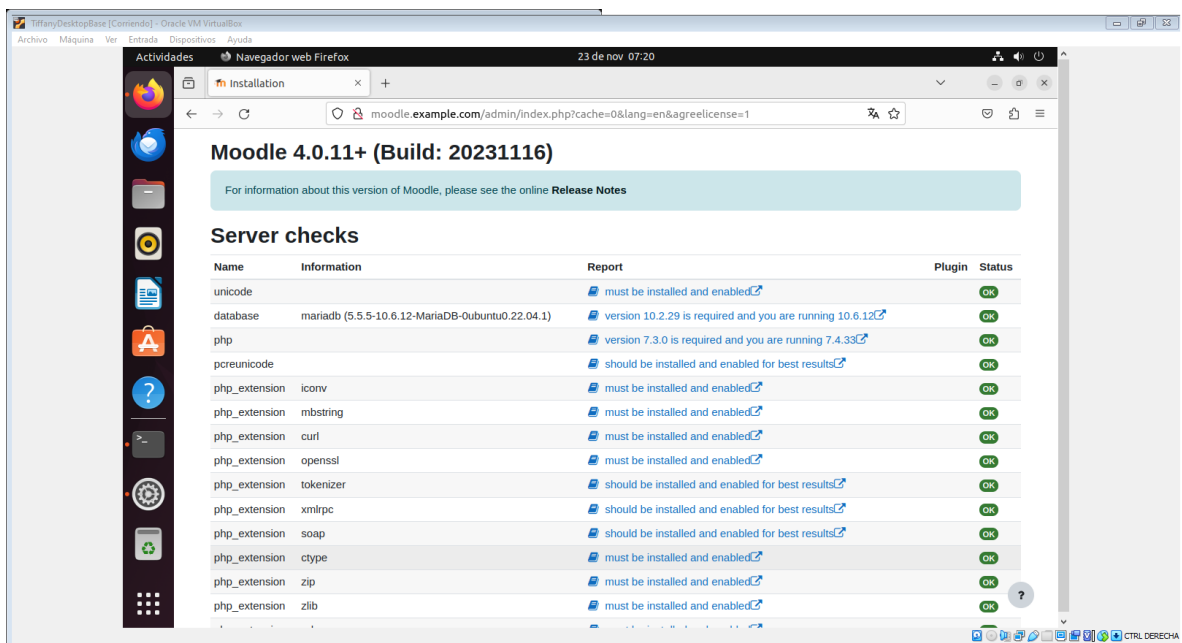
$CFG->wwwroot     = 'http://moodle.example.com';
$CFG->dataroot    = '/var/www/html/moodledata';
$CFG->admin       = 'admin';

$CFG->directorypermissions = 0777;

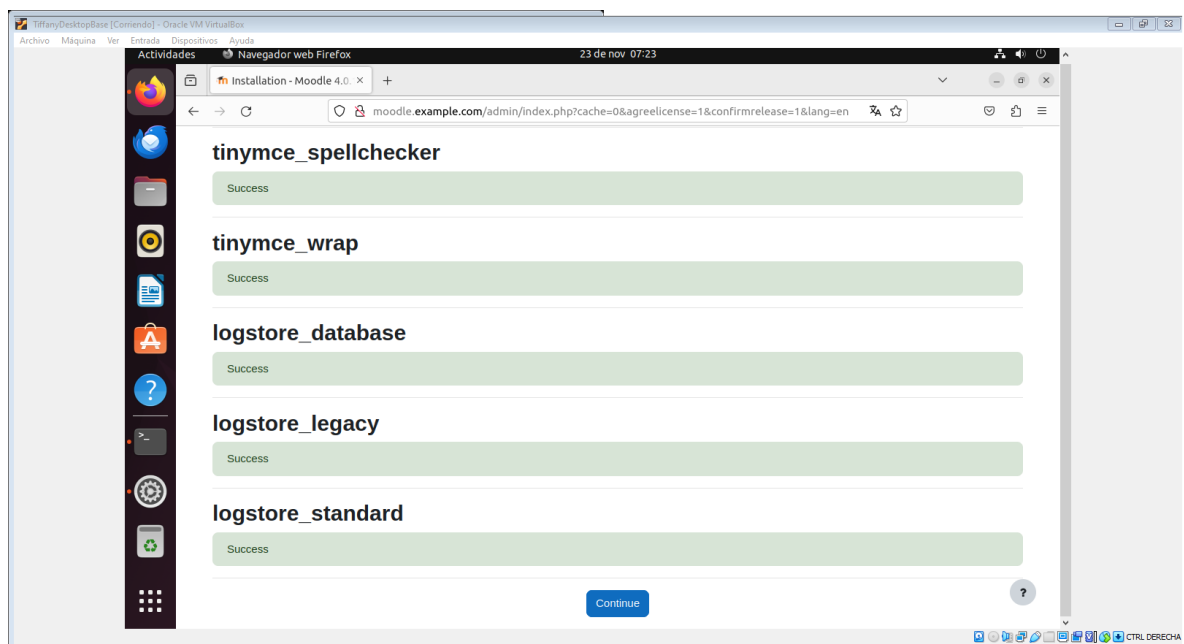
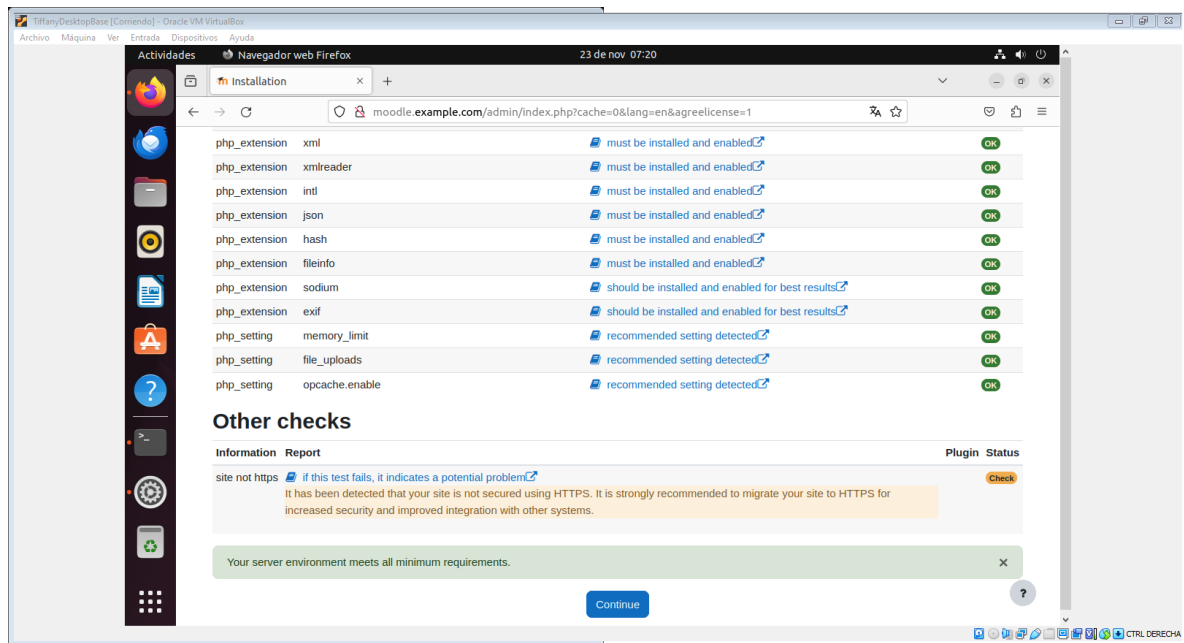
require_once(__DIR__ . '/lib/setup.php');

// There is no php closing tag in this file,
// it is intentional because it prevents trailing whitespace problems!

File Name to Write: moodie/config.php
M-A Append M-B Backup File
M-D DOS Format M-M Mac Format M-P Prepend M-T Browse
```



Nombre: Tiffany Andrea Jordán Uquillas



Configuramos los datos para nuestro Moodle y damos clic en upgrade profile.

Nombre: Tiffany Andrea Jordán Uquillas

TiffanyDesktopBase [Comando] - Oracle VM VirtualBox

Archivo Máquina Ver Entrada Dispositivos Ayuda

Actividades

Navegador web Firefox 23 de nov 07:28

Installation x +

moodle.example.com/user/editadvanced.php

General

Username

Choose an authentication method ☐ Manual accounts

The password must have at least 8 characters, at least 1 digit(s), at least 1 lower case letter(s), at least 1 upper case letter(s), at least 1 special character(s) such as *, -, or #.

New password

First name

Last name

Email address

Email visibility

City/town

Select a country

Force password change ☐

TiffanyDesktopBase [Comando] - Oracle VM VirtualBox

Archivo Máquina Ver Entrada Dispositivos Ayuda

Actividades

Navegador web Firefox 23 de nov 07:28

Installation x +

moodle.example.com/user/editadvanced.php

Last name

Email address

Email visibility

City/town

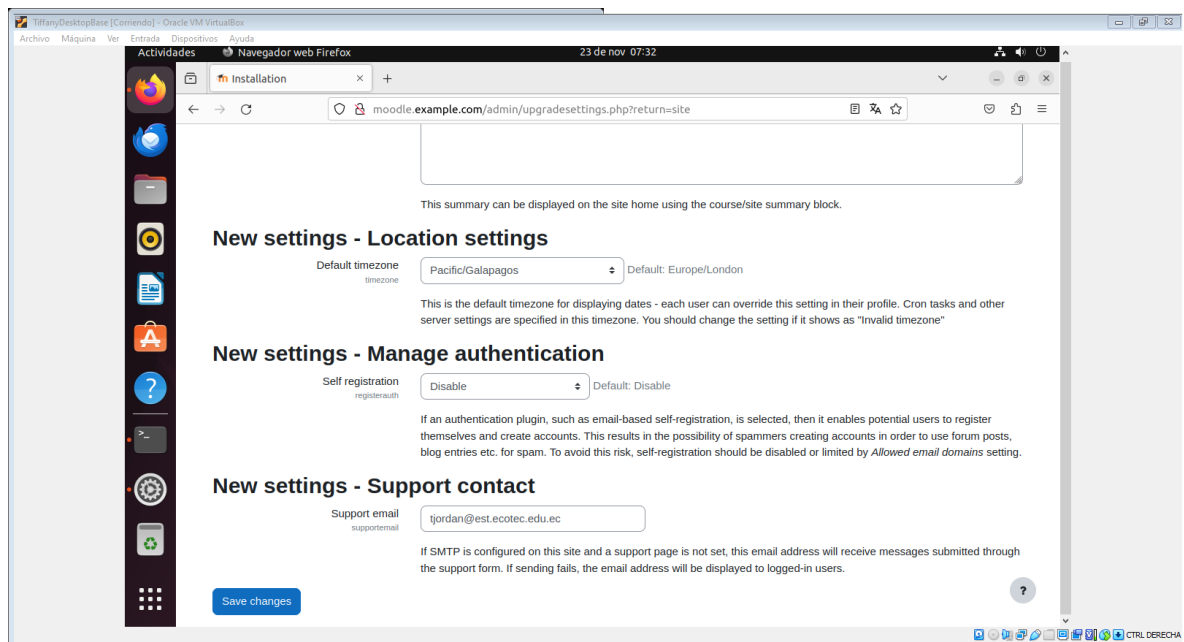
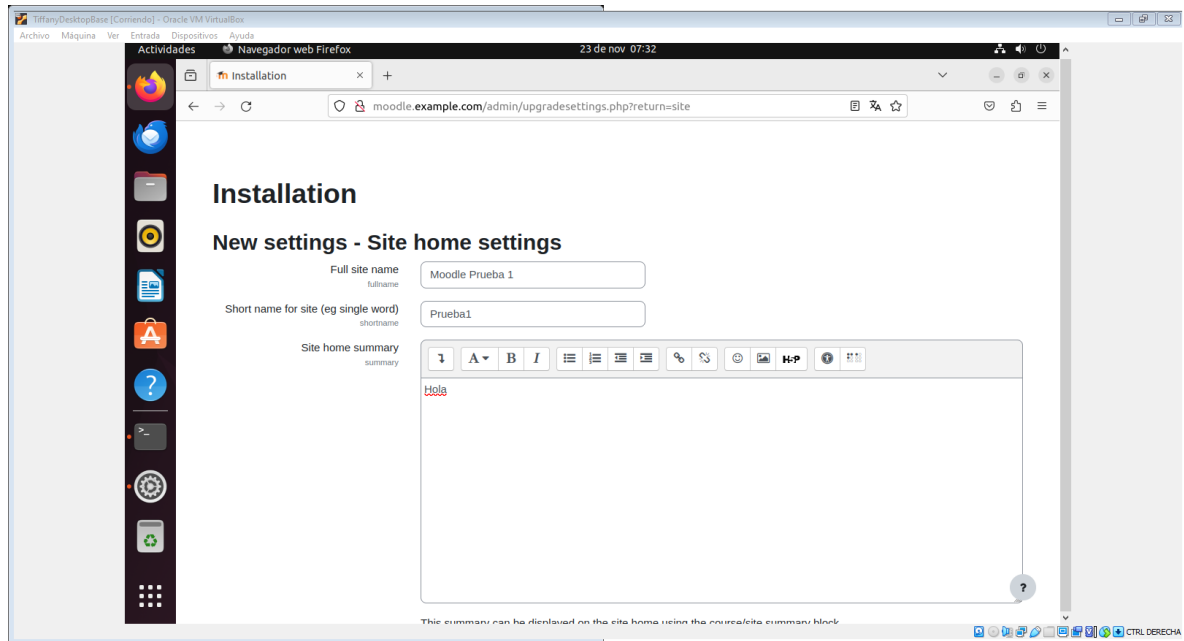
Select a country

Timezone

Description

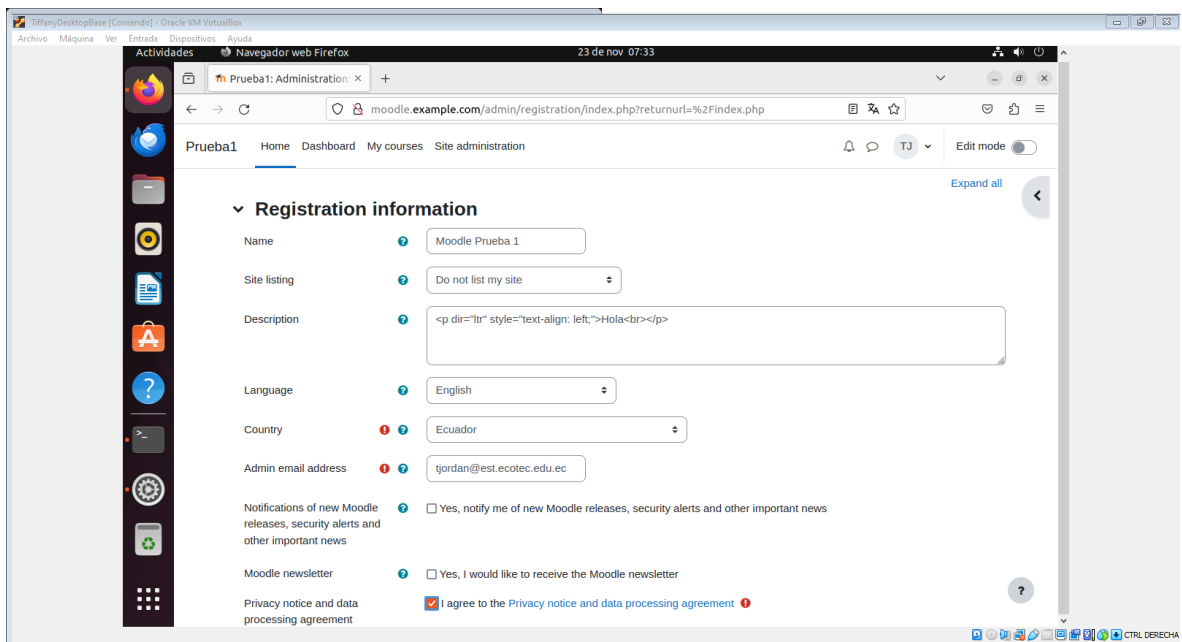
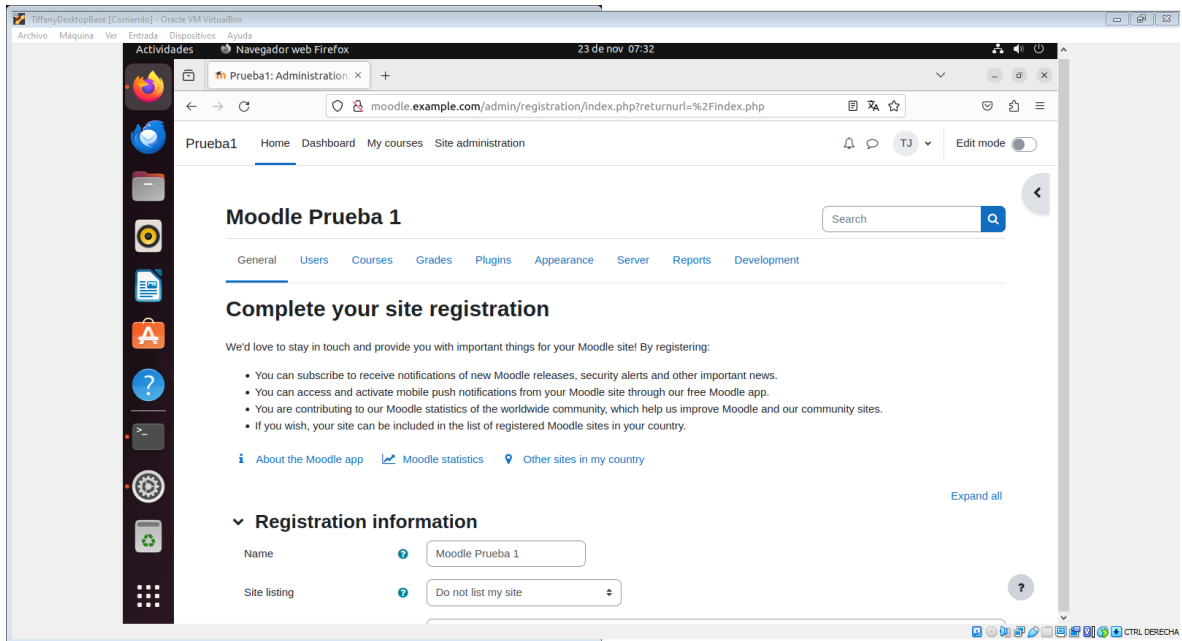
HTML format

Nombre: Tiffany Andrea Jordán Uquillas



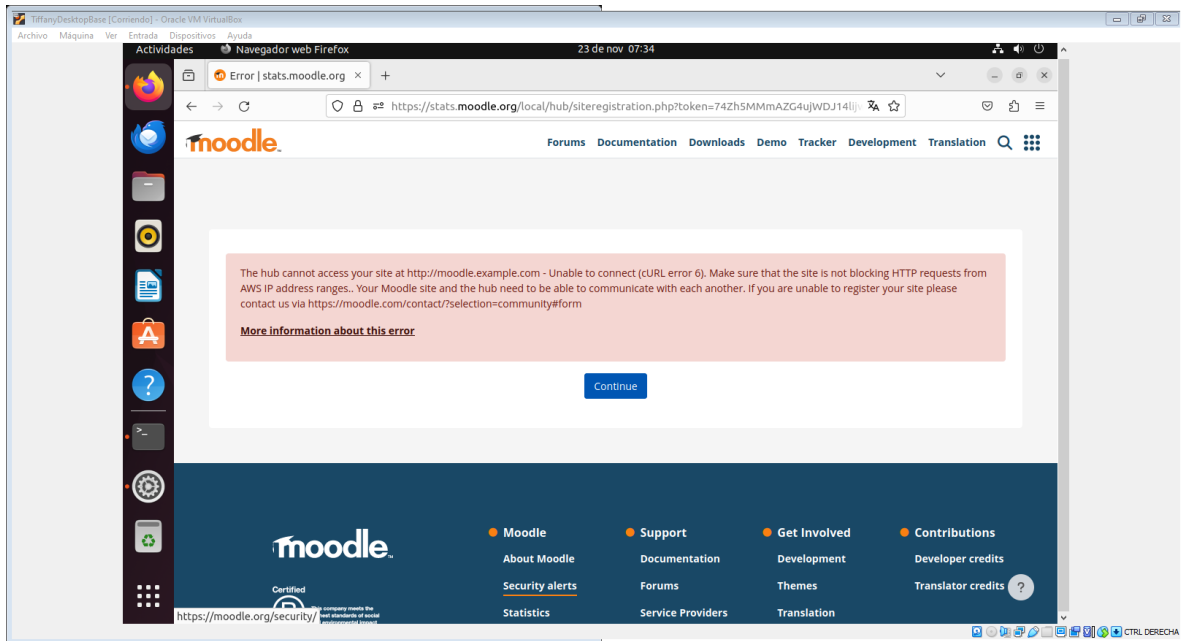
Completamos el registro, dado que no tenemos certificados SSL, podemos darle a skip a continuación y en cambio sí llenamos los datos (no olvidar darle a aceptar términos y condiciones) y le damos a continuar el registrar el sitio nos va a salir el error que se muestra:

Nombre: Tiffany Andrea Jordán Uquillas

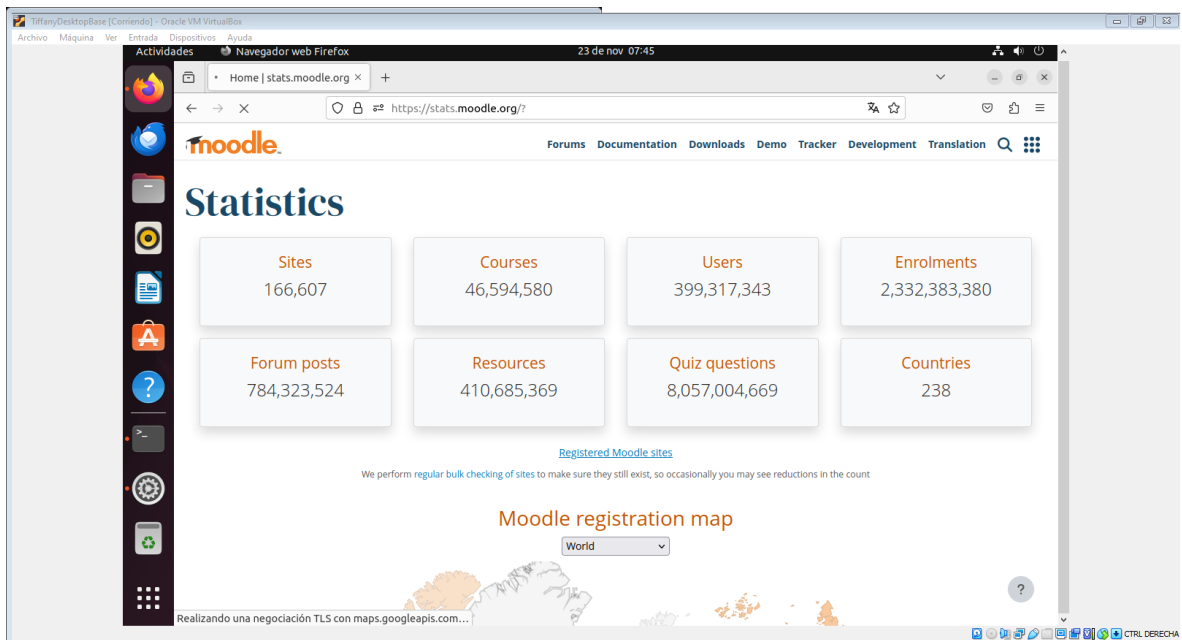


Lo que se muestra cuando no se tiene certificados SSL:

Nombre: Tiffany Andrea Jordán Uquillas

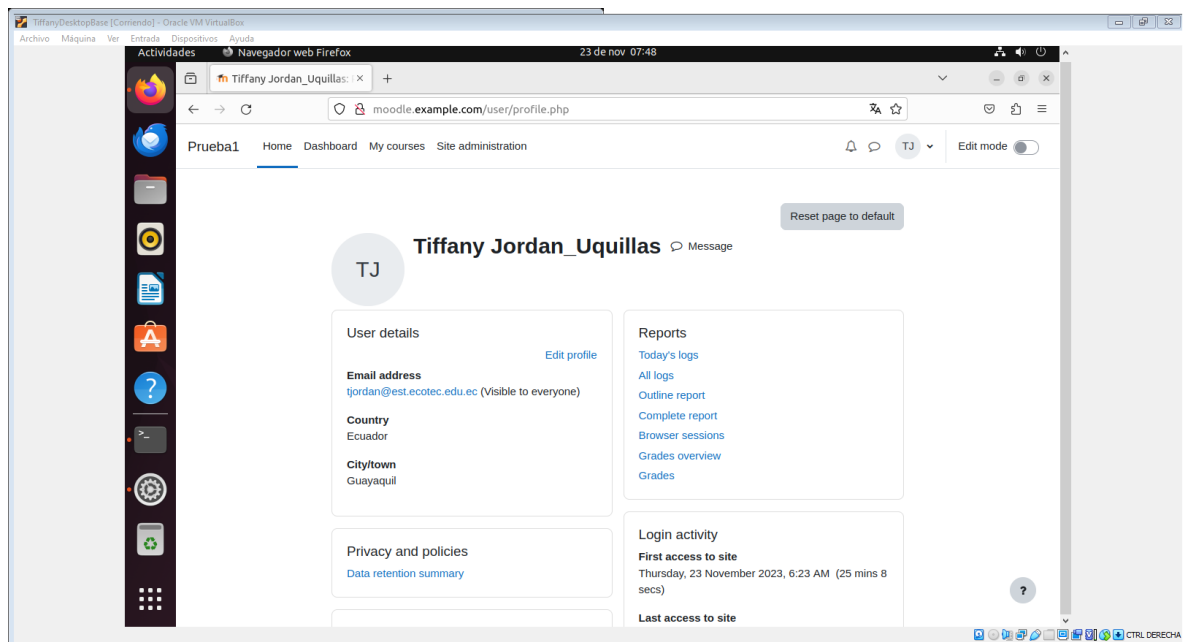
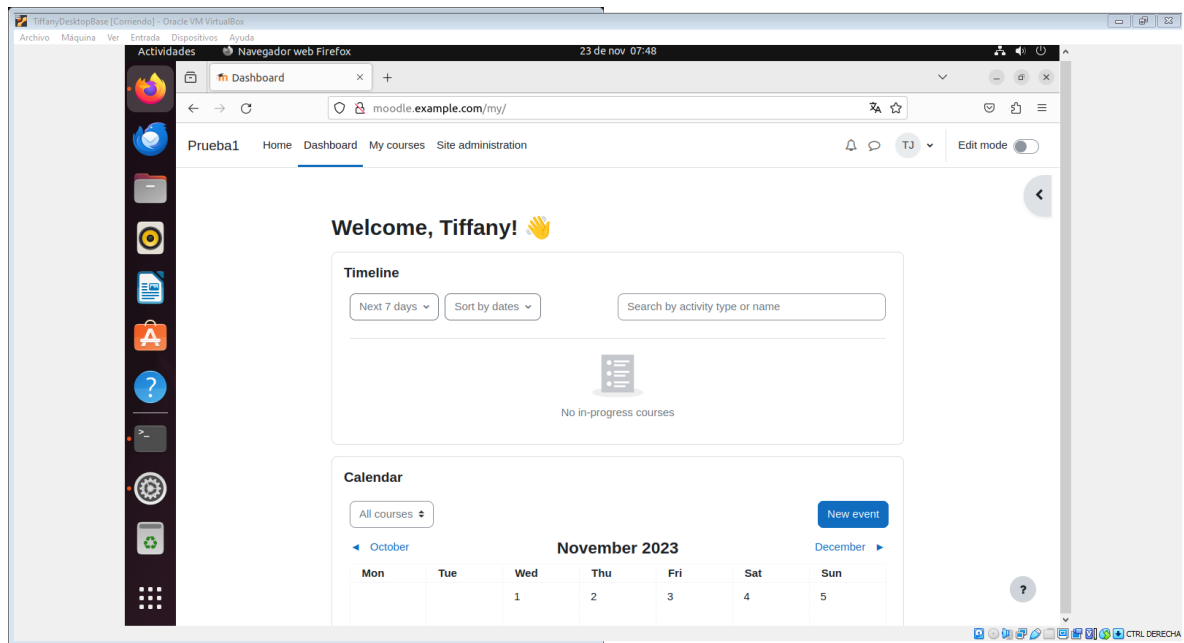


El sitio nos redirigirá a la página oficial de Moodle:



Para poder observar nuestra plataforma volvemos a acceder a la ruta que habíamos designado, que en este caso es “moodle.example.com”.

Nombre: Tiffany Andrea Jordán Uquillas



Certificados SSL:

No podemos crear un certificado SSL para un dominio que no existe o que no nos pertenece. Los certificados SSL son emitidos por autoridades de certificación confiables después de verificar que el solicitante tiene control sobre el dominio para el cual se está emitiendo el certificado. Esto se hace para garantizar la seguridad y la autenticidad de las conexiones seguras.

Nombre: Tiffany Andrea Jordán Uquillas

Si se desea un certificado SSL para un dominio, primero debes registrar ese dominio y asegurarte de tener el control sobre él antes de solicitar el certificado.