



## TAREA1: INSTALACIONES

### Windows 11

[Creación de Máquina Virtual con Windows 11](#)  
[Instalación de SQL Server 2022 en Windows 11](#)  
[Configuración del Firewall en Windows 11](#)  
[Instalar GUI: SSMS - Azure Data Studio](#)  
[Explicar Instalación Desatendida](#)  
[Instalación bases de datos de ejemplo](#)

### Ubuntu

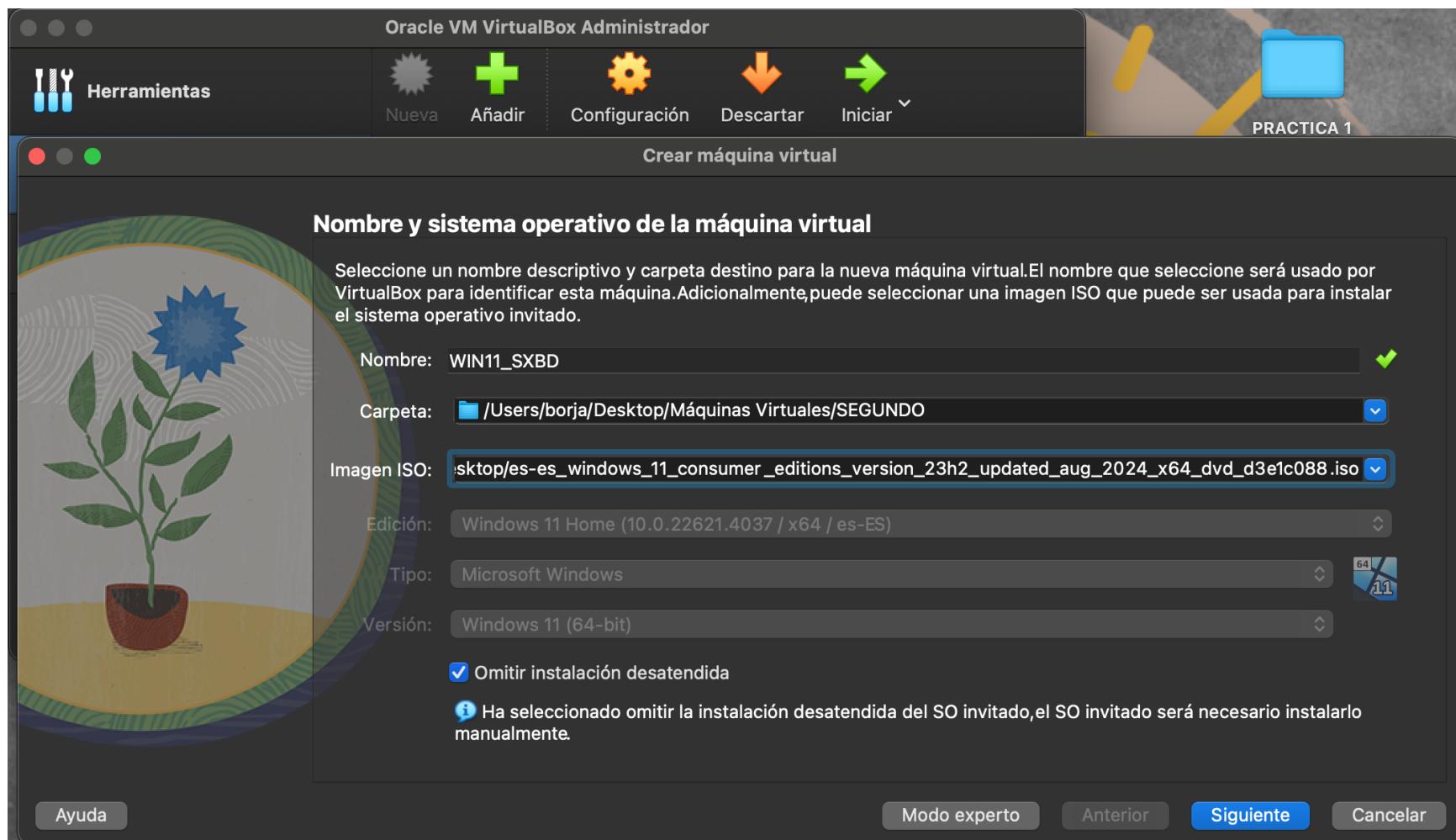
[Creación máquina virtual con Ubuntu](#)  
[Instalar SQL Server 2019](#)  
[Instalar GUI: Azure Data Studio](#)  
[Instalar Herramienta de Cliente : Sqlcmd](#)  
[Conectar desde VM W10 al Servidor UBUNTU.](#)

## Windows 11

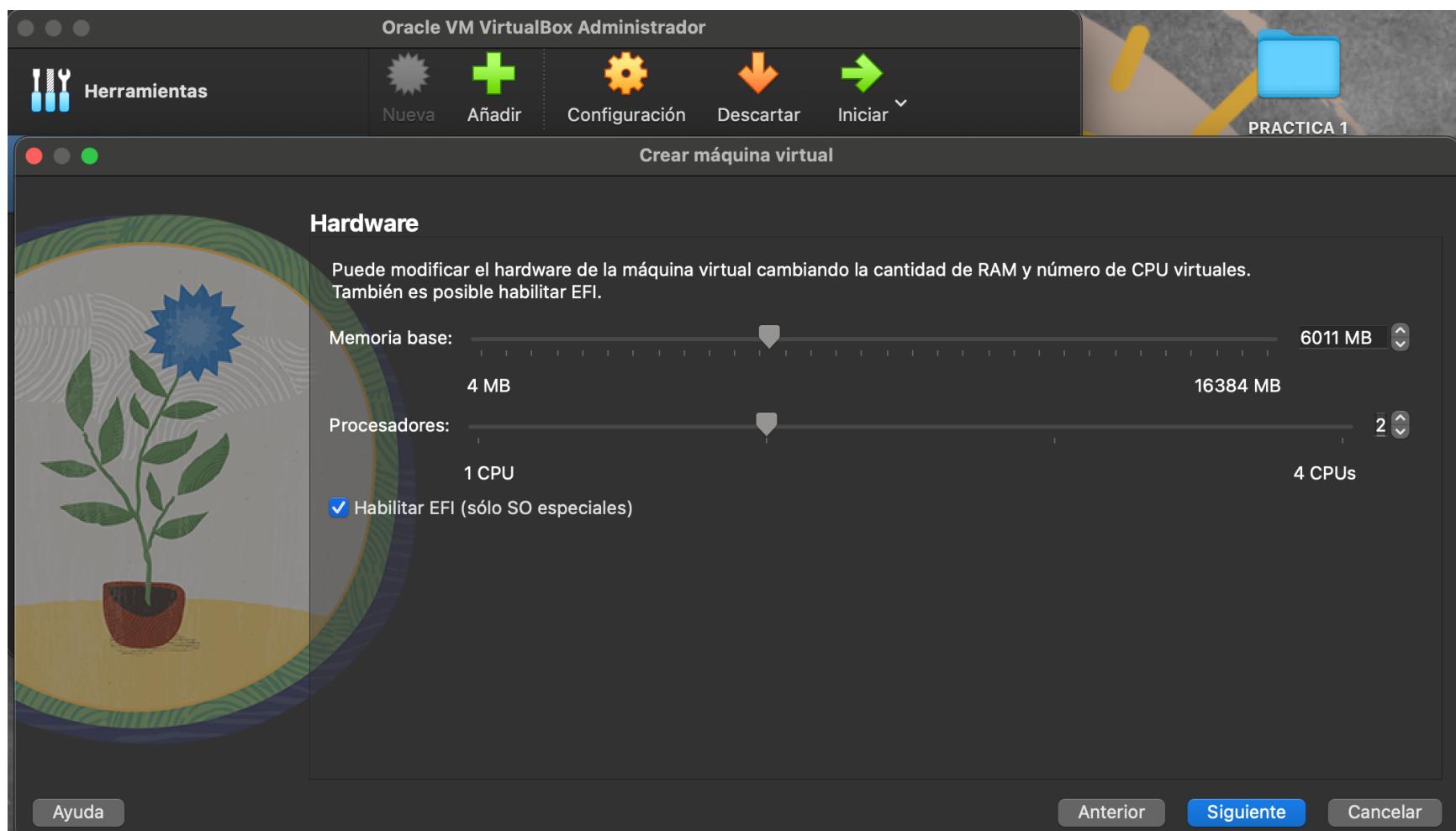
### **Creación de Máquina Virtual con Windows 11**

Lo primero que debemos hacer es instalar el software Virtual Box en nuestro ordenador, en este caso omitiremos la guía de la instalación por su simplicidad y comenzaremos directamente con la creación de la máquina virtual con Windows 11.

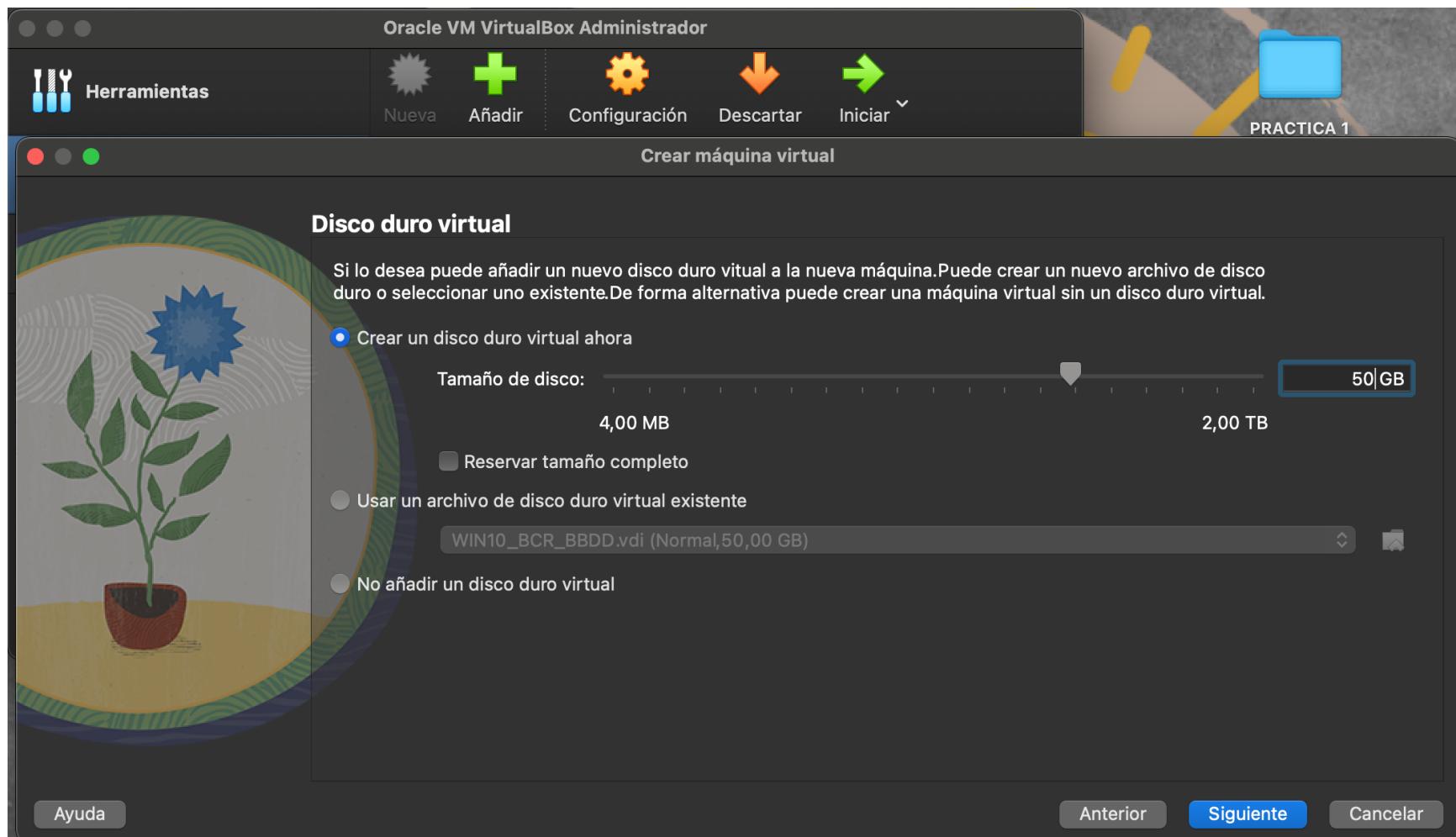
Una vez en Virtual Box, pincharemos en “Nueva”, asignaremos un nombre a la máquina virtual, escogeremos la carpeta donde se van a guardar los archivos, seleccionamos la iso del sistema operativo (en este caso Windows 11) y en caso de querer modificar algo durante la instalación debemos recordar marcar la opción de “Omitir instalación desatendida”.



En la siguiente pantalla, configuraremos los parámetros de hardware de la máquina virtual. Debemos tener en cuenta que debemos dejar recursos para que el sistema anfitrión (nuestro ordenador) funcione y al mismo tiempo darle a la máquina virtual suficientes recursos para poder trabajar con ella. En este caso le adjudicamos un tercio de la RAM y la mitad de procesadores.

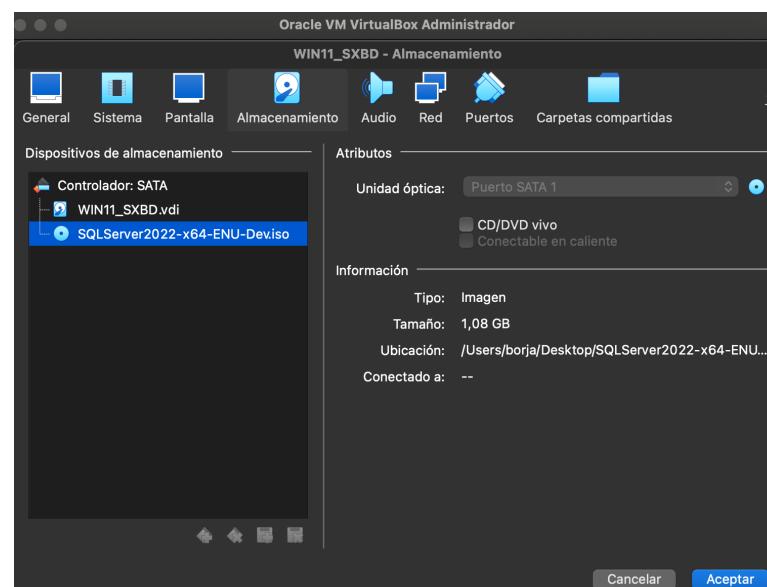


El siguiente paso es crear el disco duro de la máquina virtual. En este caso crearemos un disco de 50Gb pero sin marcar la opción "Reservar tamaño completo", de esta forma el disco duro irá creciendo de tamaño según lo vayamos llenando y nos ocupara menos espacio.



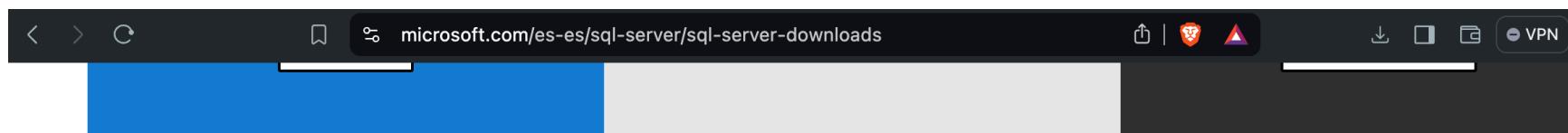
Para finalizar, se nos mostrará una pantalla resumen con todas las características, pulsamos en "Terminar" y ya tendremos nuestra máquina lista.

Una vez tengamos el sistema operativo funcionando, debemos instalar el Guest Additions que nos permitirá, entre otras cosas, poder arrastrar objetos de nuestro sistema anfitrión a nuestro sistema huésped (máquina virtual). Para ello en la configuración de la máquina virtual, accedemos al apartado "Almacenamiento" y en la unidad de cd montamos la imagen de Guest Addition que ya nos ofrece el propio sistema. Luego en la máquina virtual accedemos al cd y ejecutamos el instalador.



## Instalación de SQL Server 2022 en Windows 11

Para instalar el SQL Server, descargamos la imagen ISO y la montamos en la unidad de CD igual que hemos hecho con la imagen del Guest Addition. Luego arrancamos la máquina virtual y accedemos a la unidad de cd para ejecutar el instalador.



## O bien, descarga una edición especializada gratuita.



Developer

SQL Server 2022 Developer es una edición gratuita con todas las características que se puede usar como base de datos de desarrollo y pruebas en un entorno que no sea de producción.

[Descargar ahora](#)

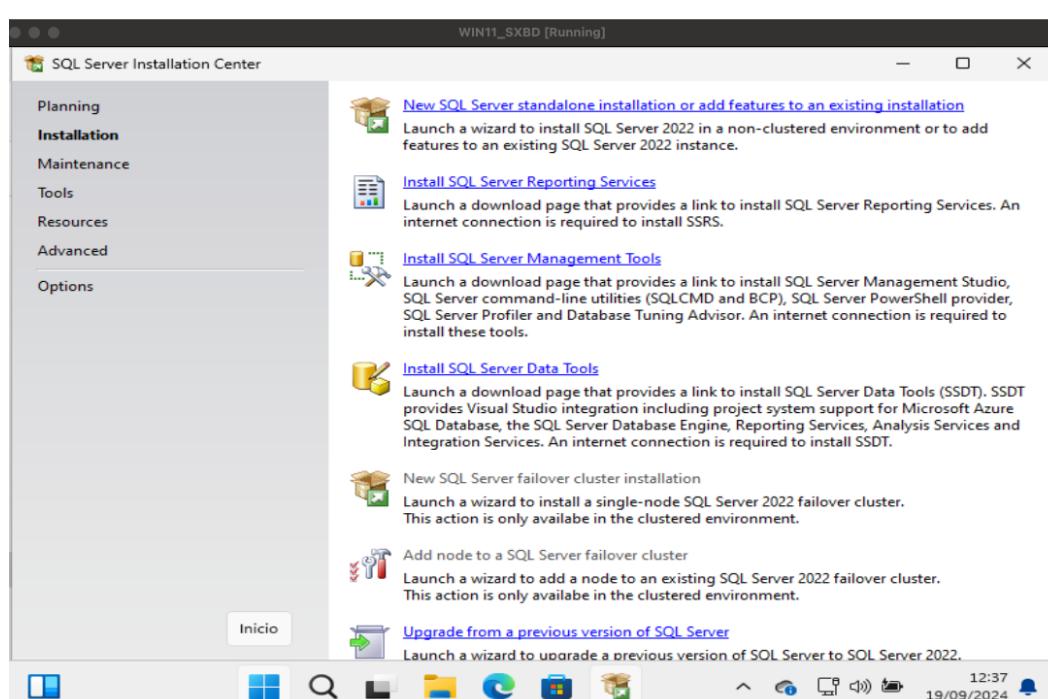


Express

SQL Server 2022 Express es una edición gratuita de SQL Server ideal para el desarrollo y la producción de aplicaciones de escritorio, aplicaciones web y pequeñas aplicaciones de servidor.

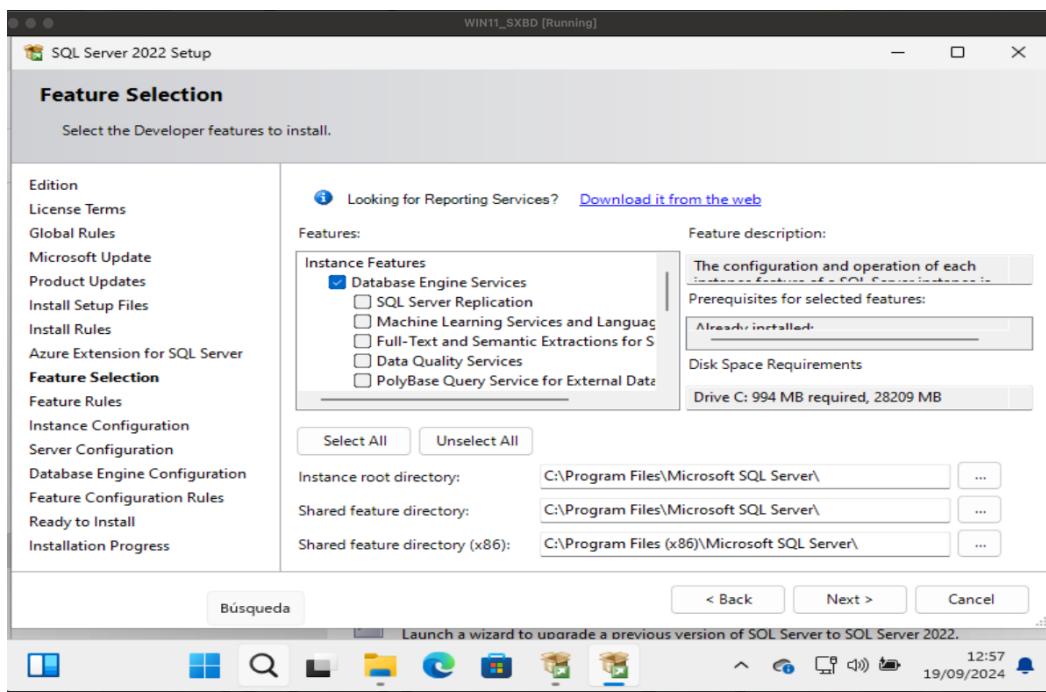
[Descargar ahora](#)

Una vez ejecutado el instalador, escogeremos "New SQL Server stand-alone installation..."

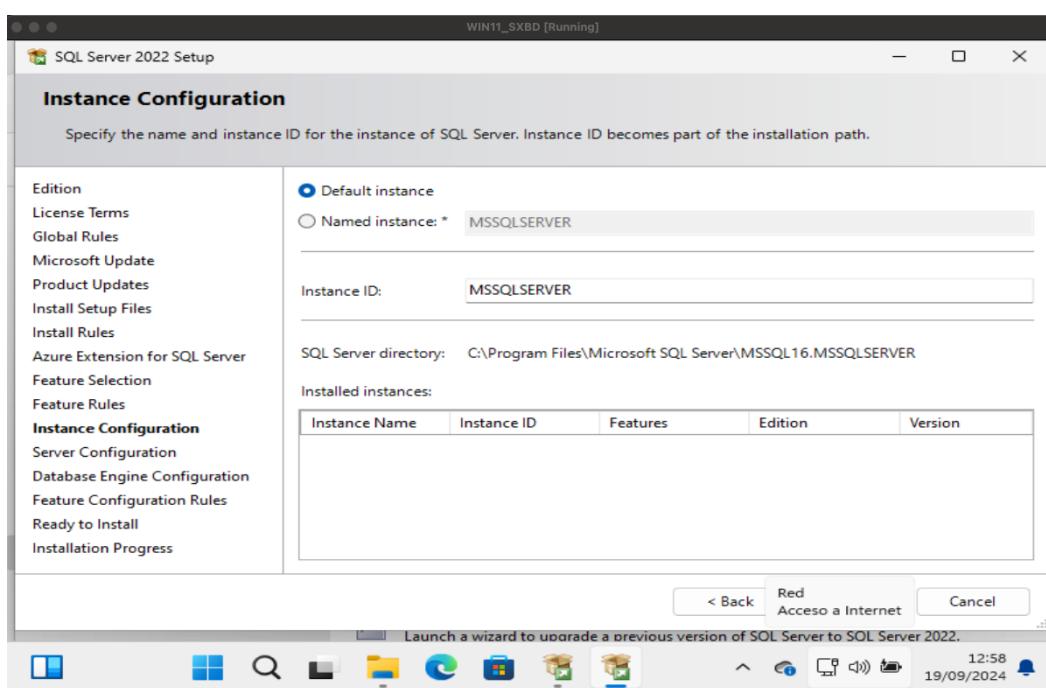


Escogemos la opción "Developer" que es la gratuita y presionamos en continuar. Nos dará un aviso de que tenemos activo el cortafuegos de Windows y para permitir conexiones entrantes debemos habilitar el puerto o deshabilitar el cortafuegos, ignoramos el warning y continuamos con la instalación (Lo abriremos mas adelante).

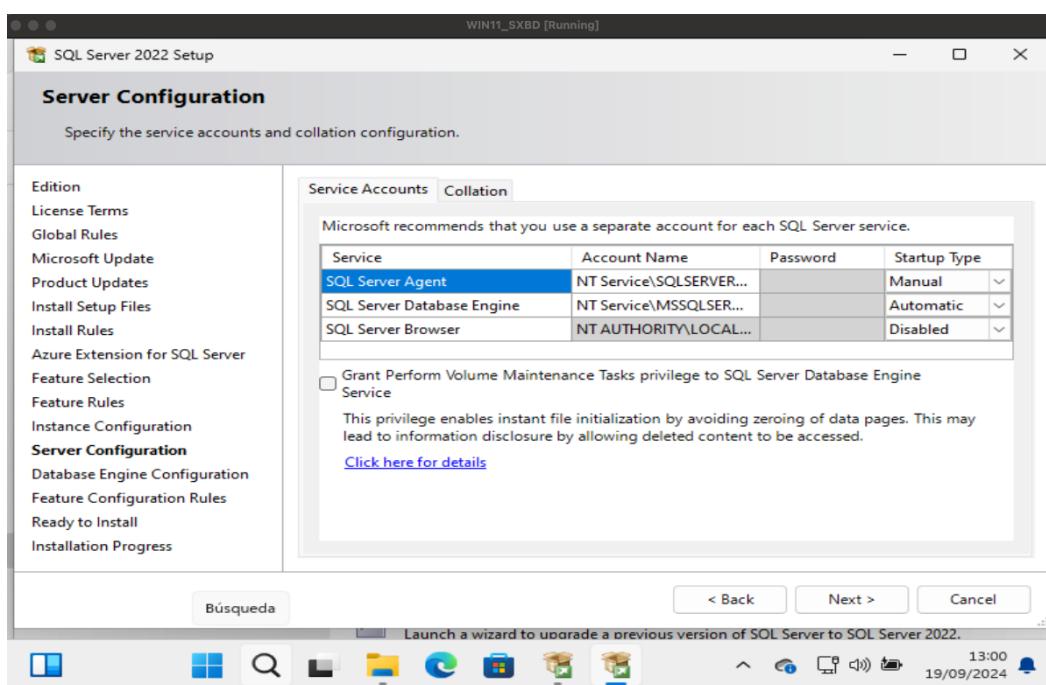
Llegaremos a una pantalla donde nos de a escoger las características que deseamos instalar, dependiendo de la versión tendremos mas o menos opciones, lo principal es seleccionar el motor de base de datos (Database Engine) y todo lo referente a clientes (las demás opciones no las instalaremos por el momento porque podrían generar errores y son características que no vamos a usar).



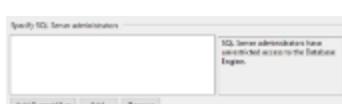
En el siguiente paso escogeremos la instancia por defecto, cada instancia es un servidor de base de datos, de forma que podremos tener varios en el mismo sistema.



En configuración de servidor lo dejaremos todo tal cual está, asegurándonos que Database Engine está en automático puesto que este es el motor de la propia base de datos.



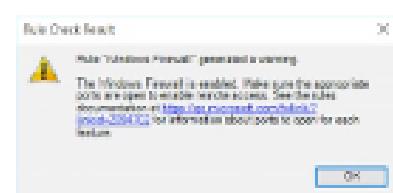
Por último llegamos a la pantalla de la autenticación, aquí dejaremos marcada la opción de autenticación con Windows y debemos añadir el usuario actual. En la siguiente pantalla también añadimos el usuario actual para darle permisos.



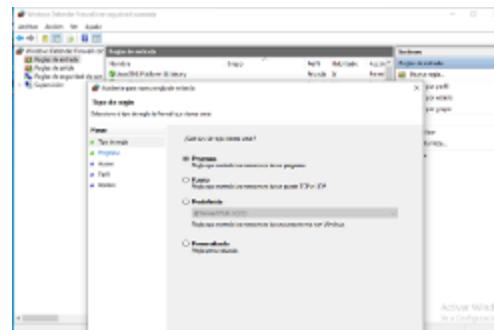
Y tras mostrarnos un resumen de la instalación, ya tendríamos listo nuestro SQL Server.

# Configuración del Firewall en Windows 11

Durante la instalación de SQL Server, se nos ha mostrado un warning que aunque nos ha permitido continuar con la instalación e incluso usar la aplicación, deberemos corregir para permitir que nos podamos conectar a nuestras bases de datos desde otros ordenadores o entornos.



Para ello iremos a la configuración del firewall de Windows, pinchamos en "Configuración avanzada" y aquí tendremos que abrir el puerto 1404 que es el que usa SQL Server.



Ahora abriremos el puerto de entrada 1404, pinchamos en "Regla de entrada" y luego en "Nueva regla". Se nos abrirá una nueva pantalla y en ella pinchamos en la opción "Puerto", seleccionamos "TCP" y abajo escribimos el puerto que queremos abrir, en este caso 1404.

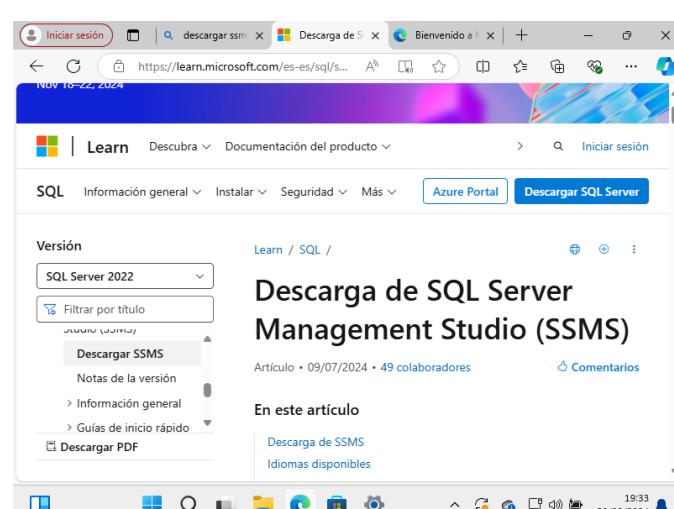


En la siguiente pantalla seleccionamos "Permitir la conexión" y marcamos todas las casillas para que la regla se aplique en todos los tipos de conexiones.

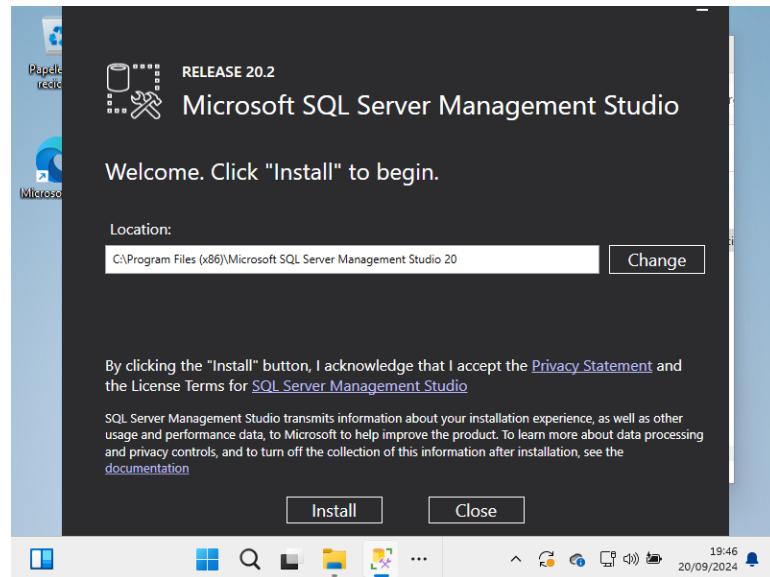
Por último ya solo nos queda dar un nombre a la regla y ya dejaríamos el puerto abierto para permitir las conexiones entrantes.

## Instalar GUI: SSMS - Azure Data Studio

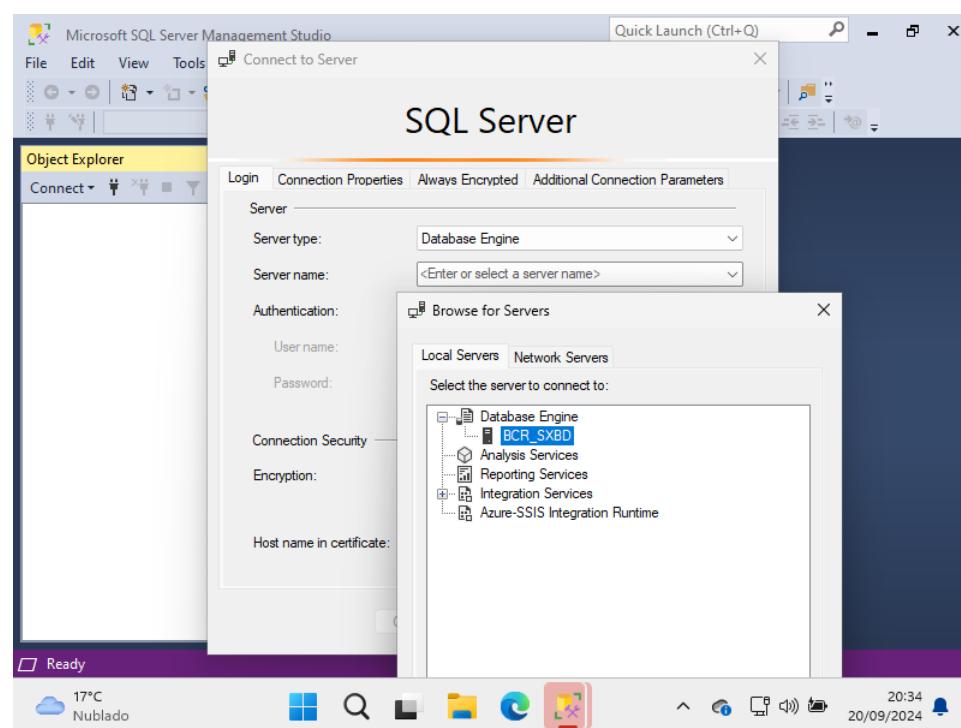
Tanto SSMS como de Azure Data Studio son entornos gráficos desde donde podemos gestionar nuestras bases de datos, sus instalaciones son muy similares. Comenzaremos con SSMS, podemos descargarlo gratis desde su web oficial.



Una vez descargado, ejecutamos el instalador y lo primero que nos pedirá será la ruta de la carpeta donde se instalará el software.



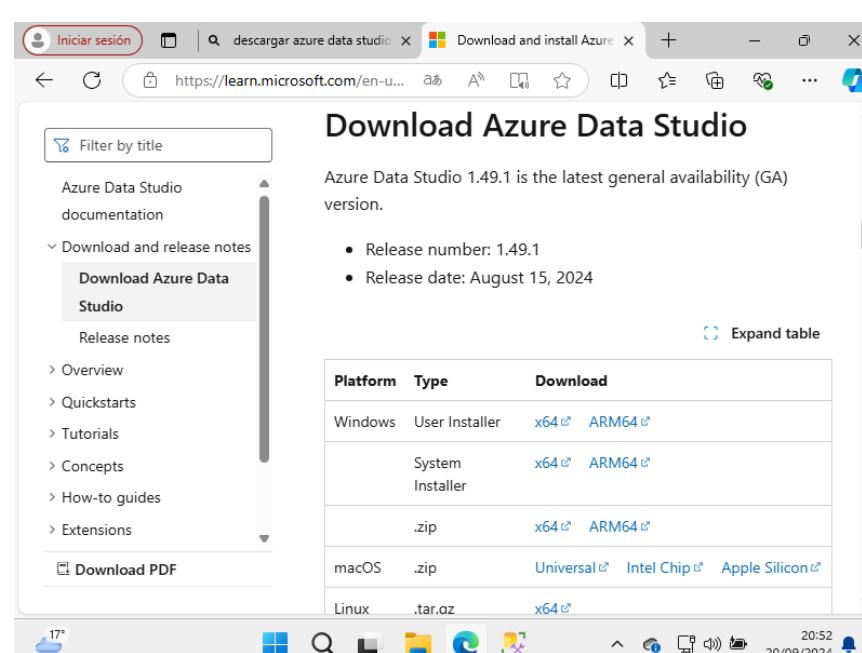
Luego dejaremos todas las opciones tal cual están y esperaremos a que finalice la instalación. Una vez finalizada la instalación, ejecutamos SSMS y al abrir tendremos que abrir el servidor y establecer una conexión. En "Server name" escogemos el que nos aparezca que será la conexión que se nos ha generado por defecto, en mi caso "BCR\_SXBD". El resto lo dejaremos todo tal cual está.



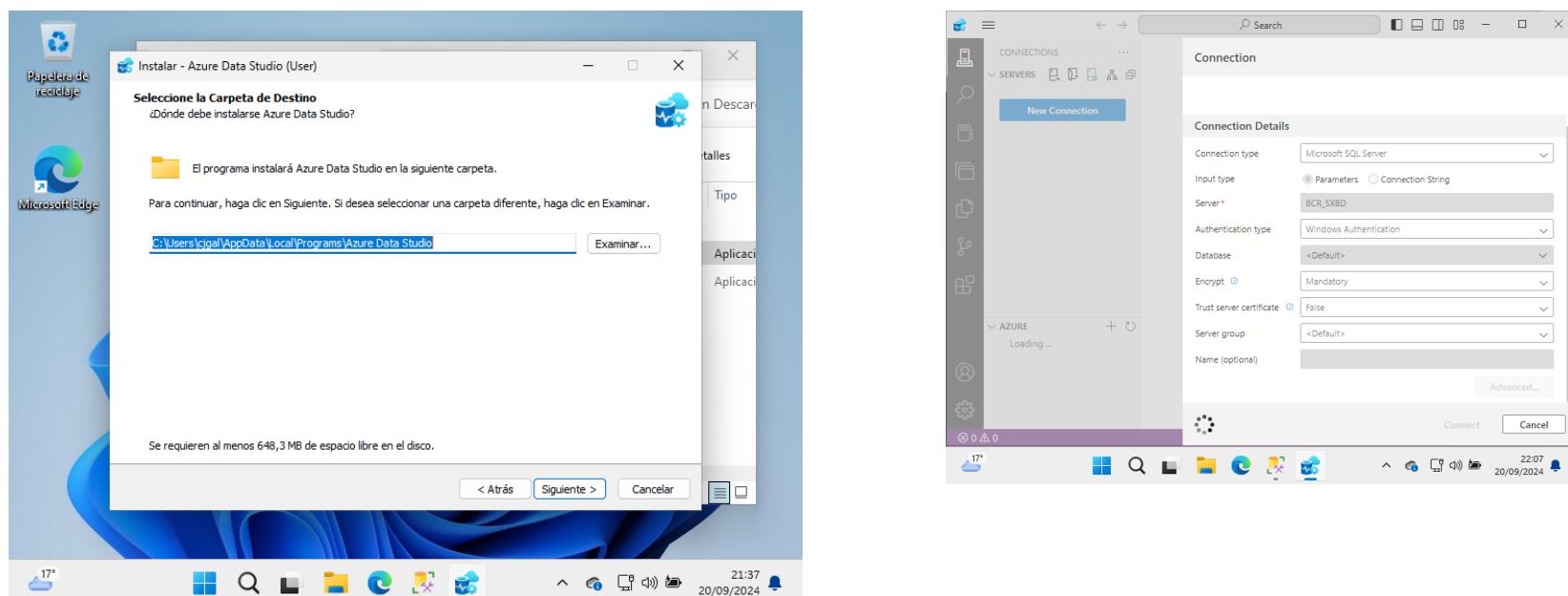
Al presionar en conectar, se conectará pero nos lanzará un warning que nos dice que aunque se ha establecido la conexión debemos añadir "Trust Server Certificate" a la conexión para desbloquearla, presionamos en "SI".



Ahora descargamos Azure Data Studio desde su web oficial.

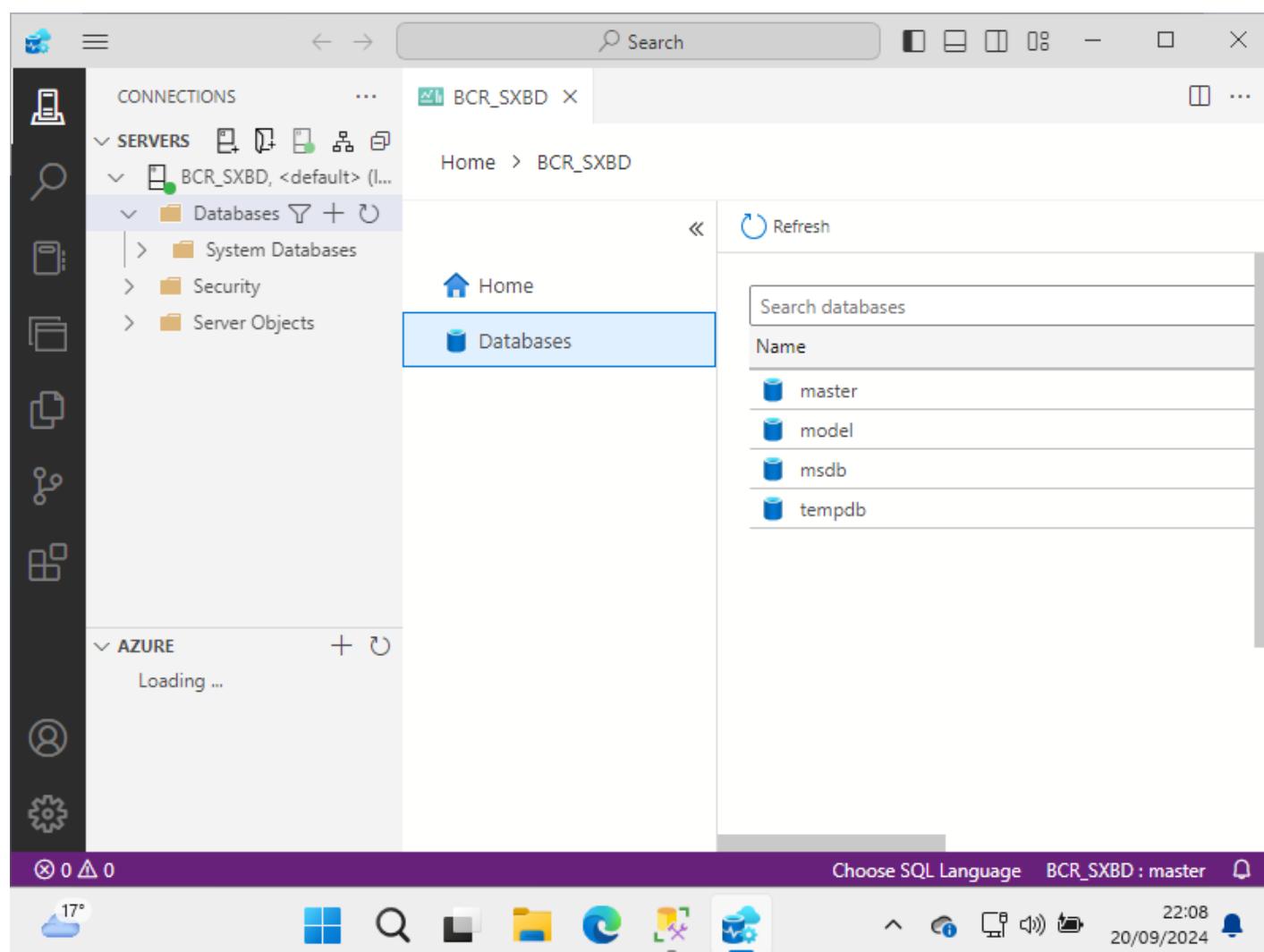


Y una vez descargado ejecutamos el archivo de instalación y tras aceptar términos y condiciones, se nos pedirá que confirmemos la carpeta de instalación igual que con SSMS.



Continuamos con la instalación, la cual no tiene mayor dificultad y al finalizar arrancaremos el software. Igual que en SSMS, se nos pedirá que creamos una nueva conexión, añadiremos el nombre del servidor (el mismo que hemos usado en SSMS) y en "Trust server certificate" escogeremos "True", de lo contrario nos lanzará el mismo warning que nos lanzó al arrancar el SSMS.

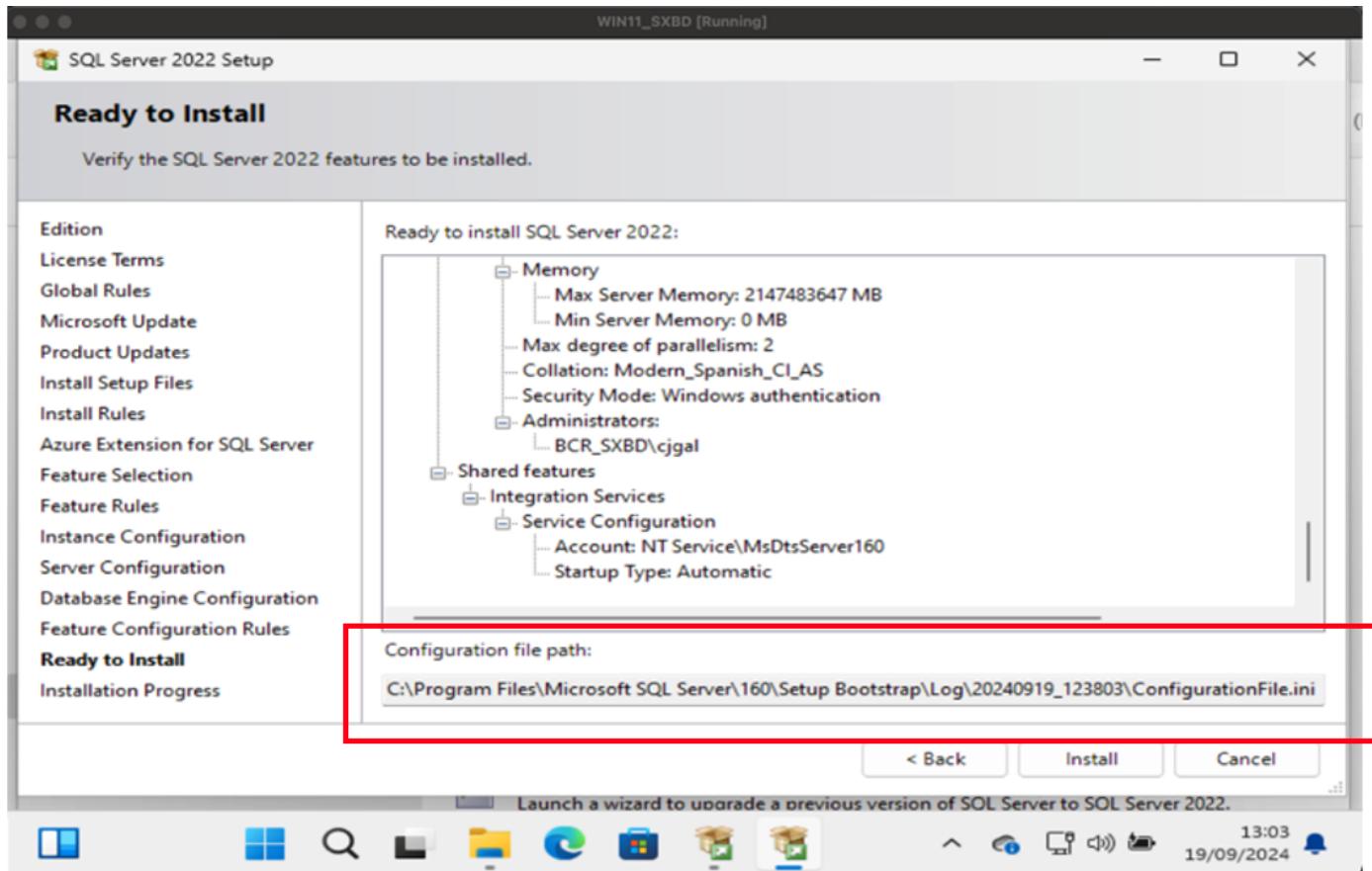
Ya podemos ver la conexión y nuestras bases de datos de prueba, igual que en SSMS.



## Explicar Instalación Desatendida

Una instalación desatendida es aquella que se ejecuta sin ir paso a paso en un entorno gráfico dándole los parámetros, de forma que nosotros ejecutamos la instalación y esta se desarrolla sin nuestra ayuda. De esta manera si tuviésemos que instalar SQL Server en varios ordenadores con la misma configuración, ejecutaríamos una línea de comandos dando unos parámetros o dándole un archivo de configuración y se instalaría automáticamente.

Durante la instalación, cuando se nos muestra el resumen de la misma, podemos ver el directorio donde se guarda el archivo con la configuración de la instalación que acabamos de realizar y cuyo archivo podremos utilizar para generar una instalación desatendida con estos mismos parámetros.



Para ello abriremos una línea de comandos en Windows11, accedemos a la ubicación donde tengamos el instalador y escribiremos:

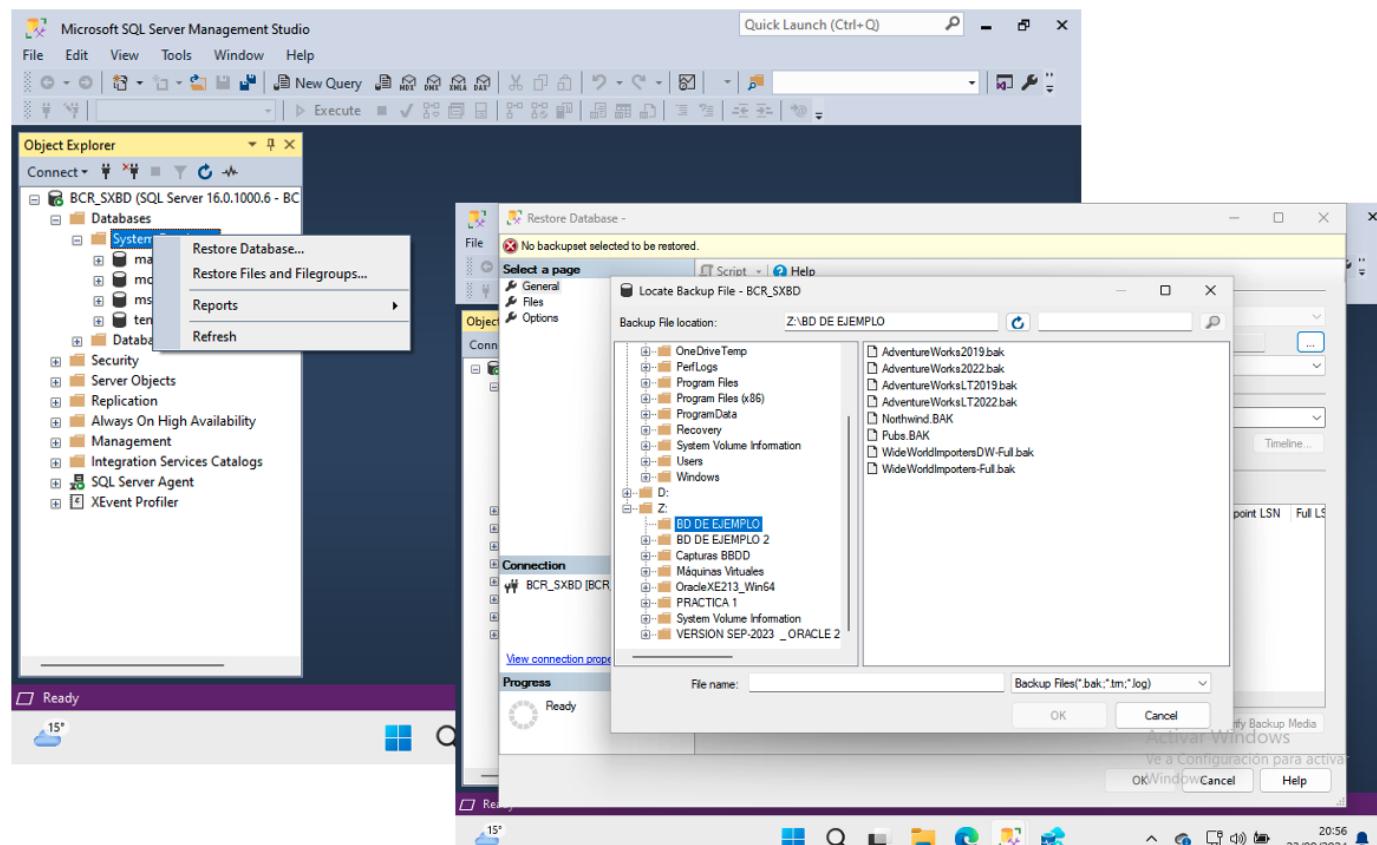
```
D:\>Setup.exe /ConfigurationFile= C:\Program Files\Microsoft SQL Server\160\Setup Bootstrap\Log\20240919_123803\ConfigurationFile.ini
```

## Instalación bases de datos de ejemplo

```
D:\>Setup.exe /ConfigurationFile= C:\Program Files\Microsoft SQL Server\160\Setup Bootstrap\Log\20240919_123803\ConfigurationFile.ini
```

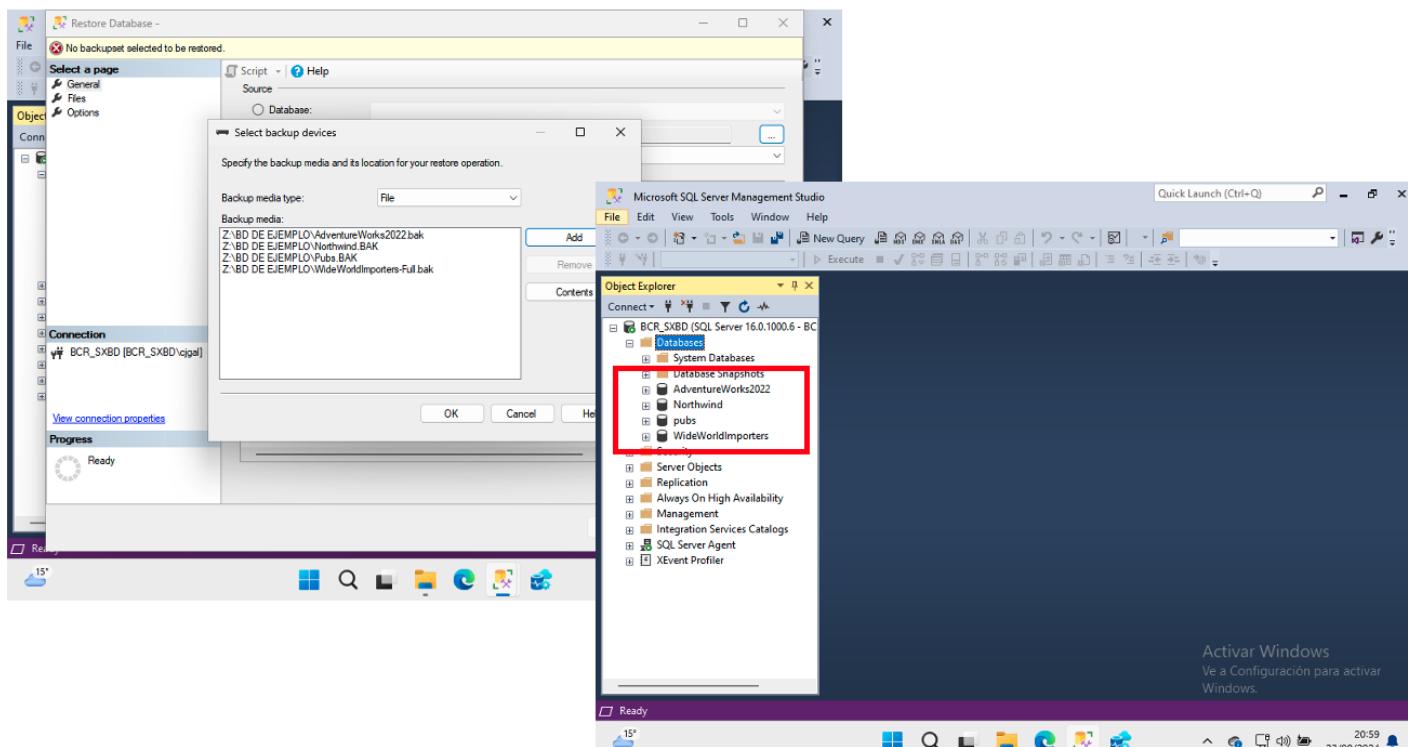
Microsoft tiene varias bases de datos de prueba que podemos usar para practicar y realizar pruebas con ellas. Para instalarlas, tan solo tendremos que descargar sus respectivos backup y restaurarlas en SQL server desde SSMS por ejemplo.

Nos situamos encima de Databases o de System database y presionamos en "Restore Database". Luego escogemos la base de datos que queremos restaurar, como por ejemplo adventureWorks.



Presionamos en "Ok" y luego de nuevo "Ok".

**Aunque nos deje añadir varios backup, solo se restaurará el primer backup seleccionado.**

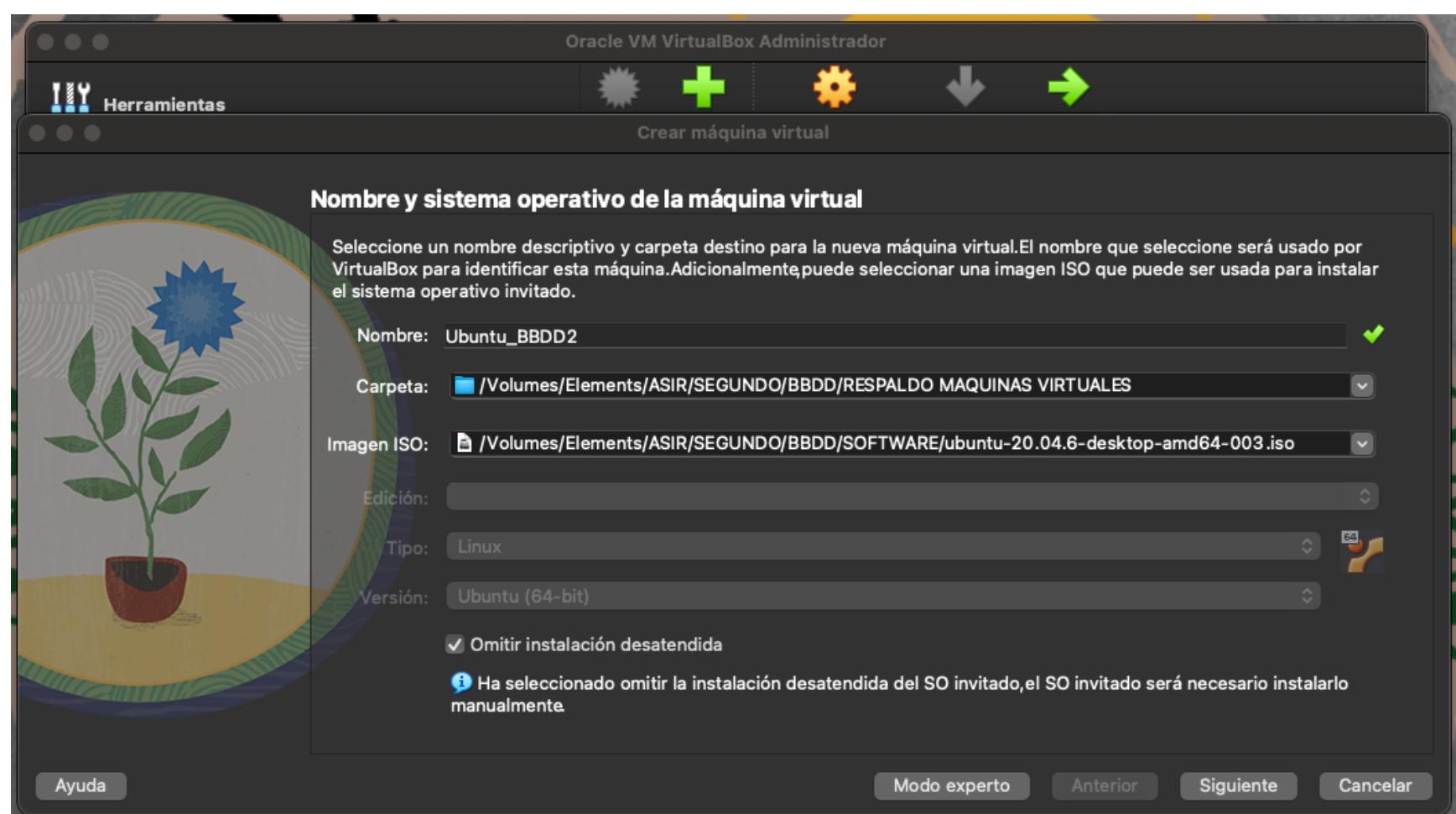


Repetimos la operación con todas las bases de datos de prueba y nos quedarán todas listas para usar.

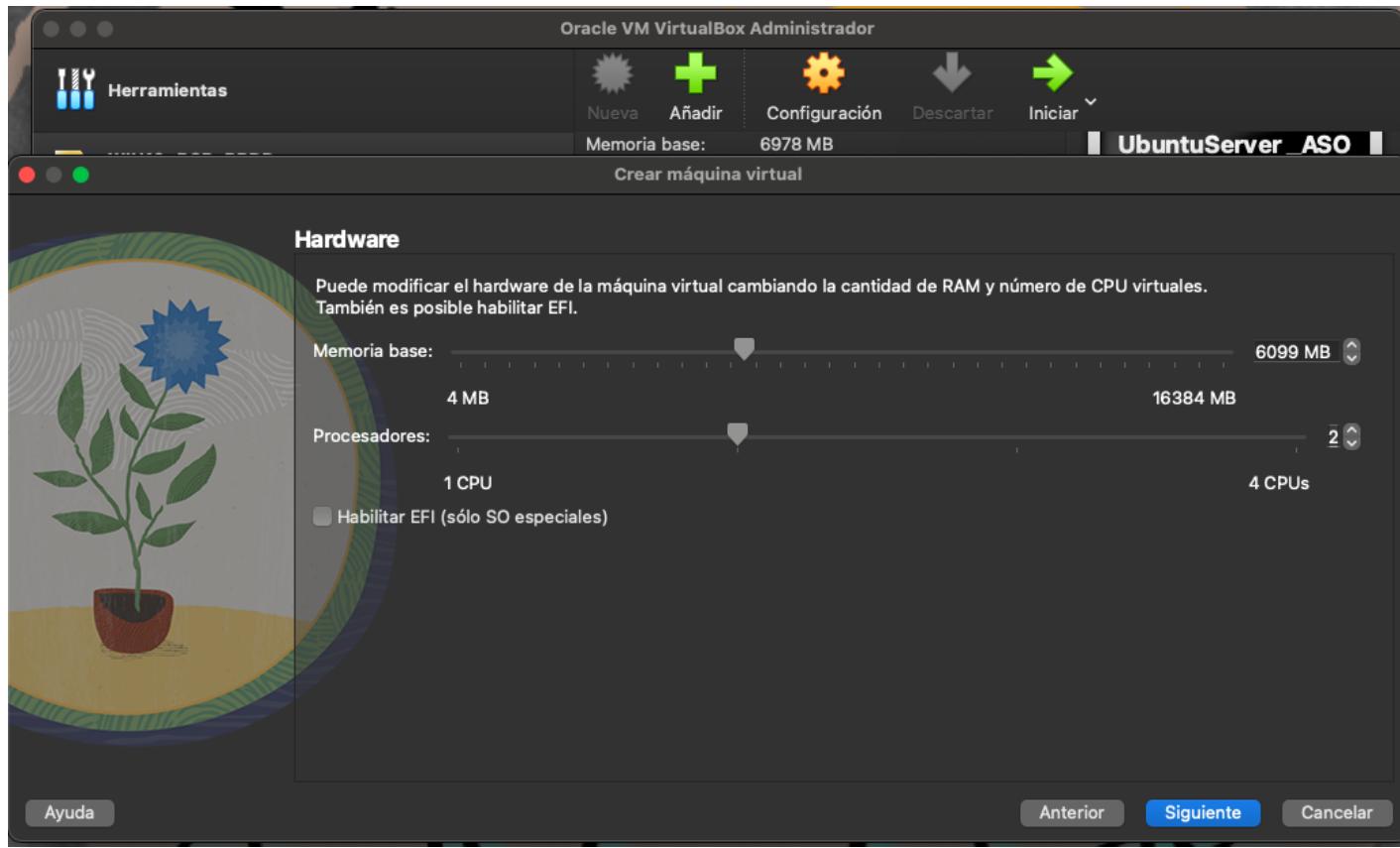
## Ubuntu

### Creación máquina virtual con Ubuntu

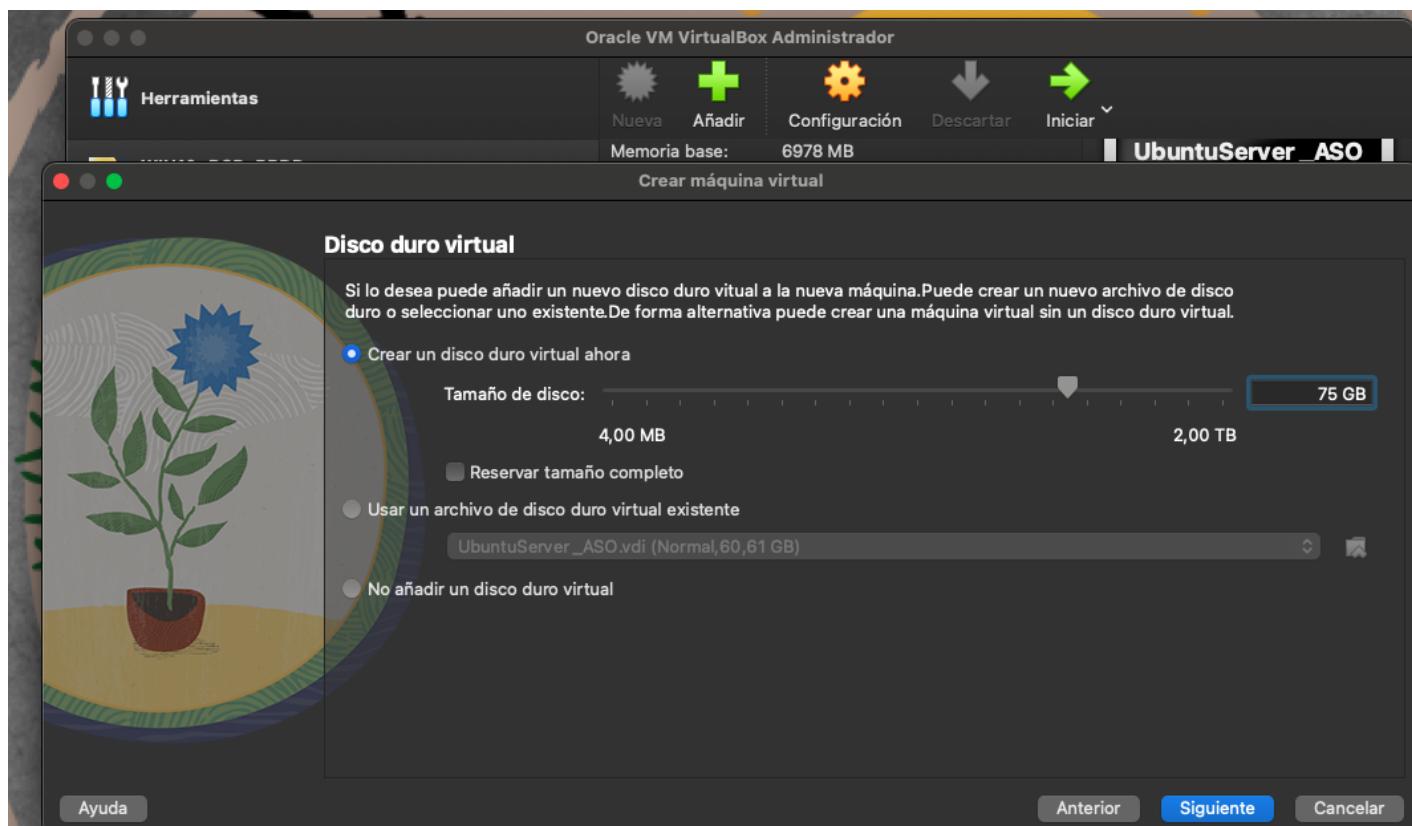
Lo primero que haremos, será crear la máquina virtual del mismo modo que hemos hecho para Windows11. Una vez en VirtualBox pulsamos en “Nueva” y configuramos la nueva máquina virtual, le damos un nombre, escogemos la carpeta donde se van a guardar los archivos y seleccionamos la imagen ISO. Debemos acordarnos de marcar la casilla de “Omitir instalación desatendida”.



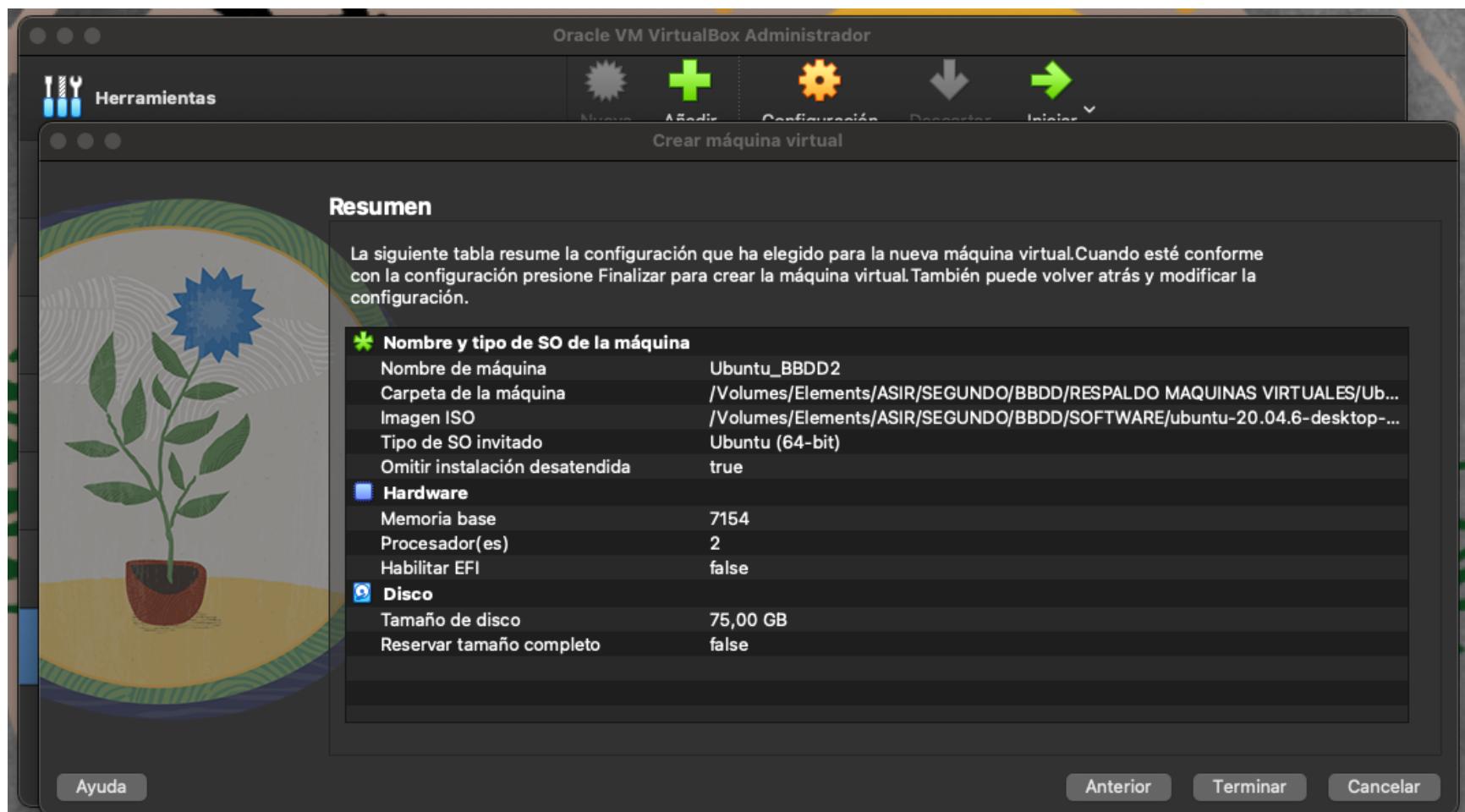
Asignamos la cantidad de memoria RAM y los procesadores. No combiene asignar recursos en exceso puesto que sino dejaremos sin recursos a nuestro ordenador y además nos dará problemas en caso de necesitar tener varias máquinas virtuales funcionando a la vez.



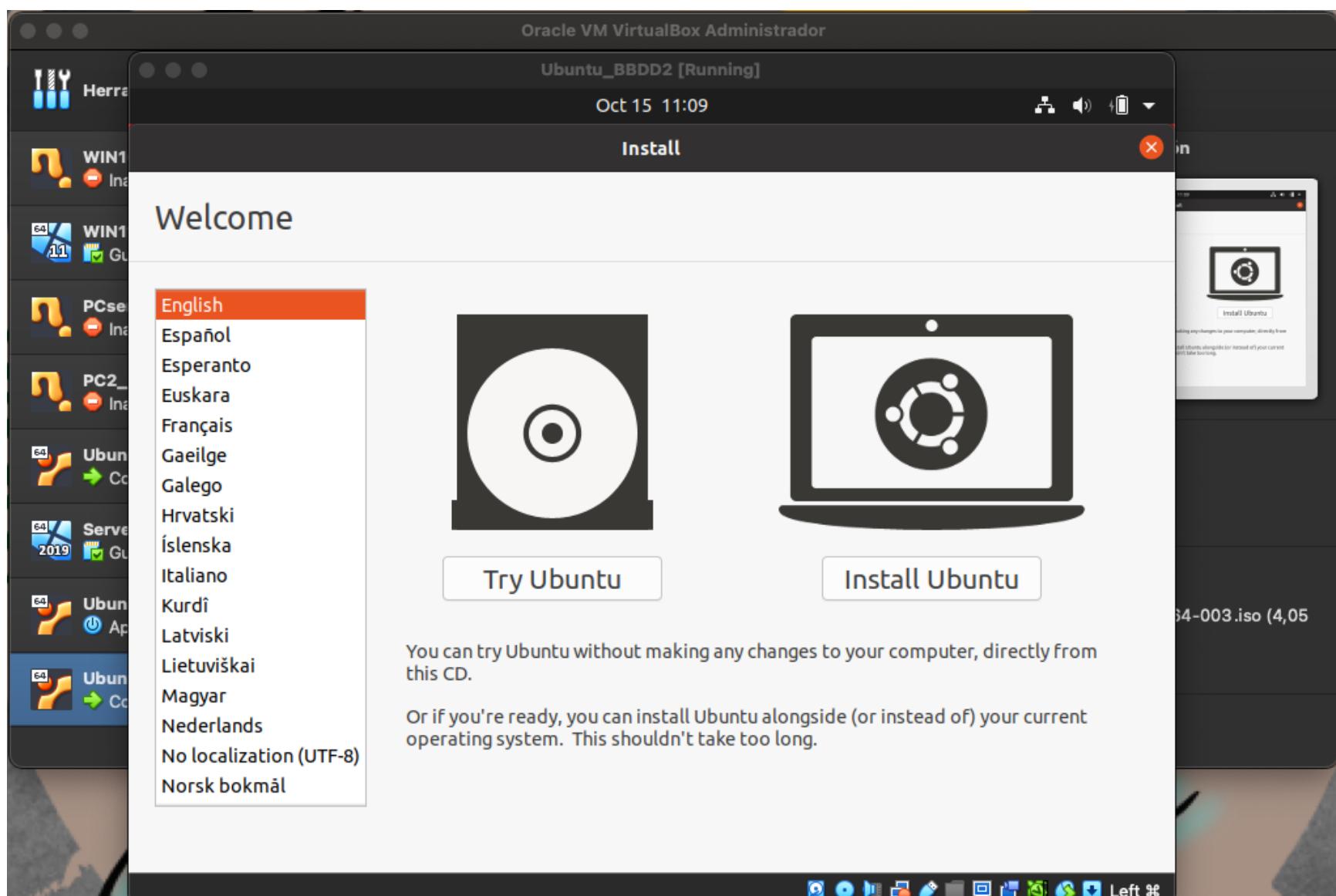
Asignamos un tamaño al disco duro virtual y nos fijamos que no esté marcada la opción “Reservar tamaño completo”, de esta manera el disco duro irá creciendo de tamaño conforme lo vaya necesitando y no nos ocupará tanto espacio.



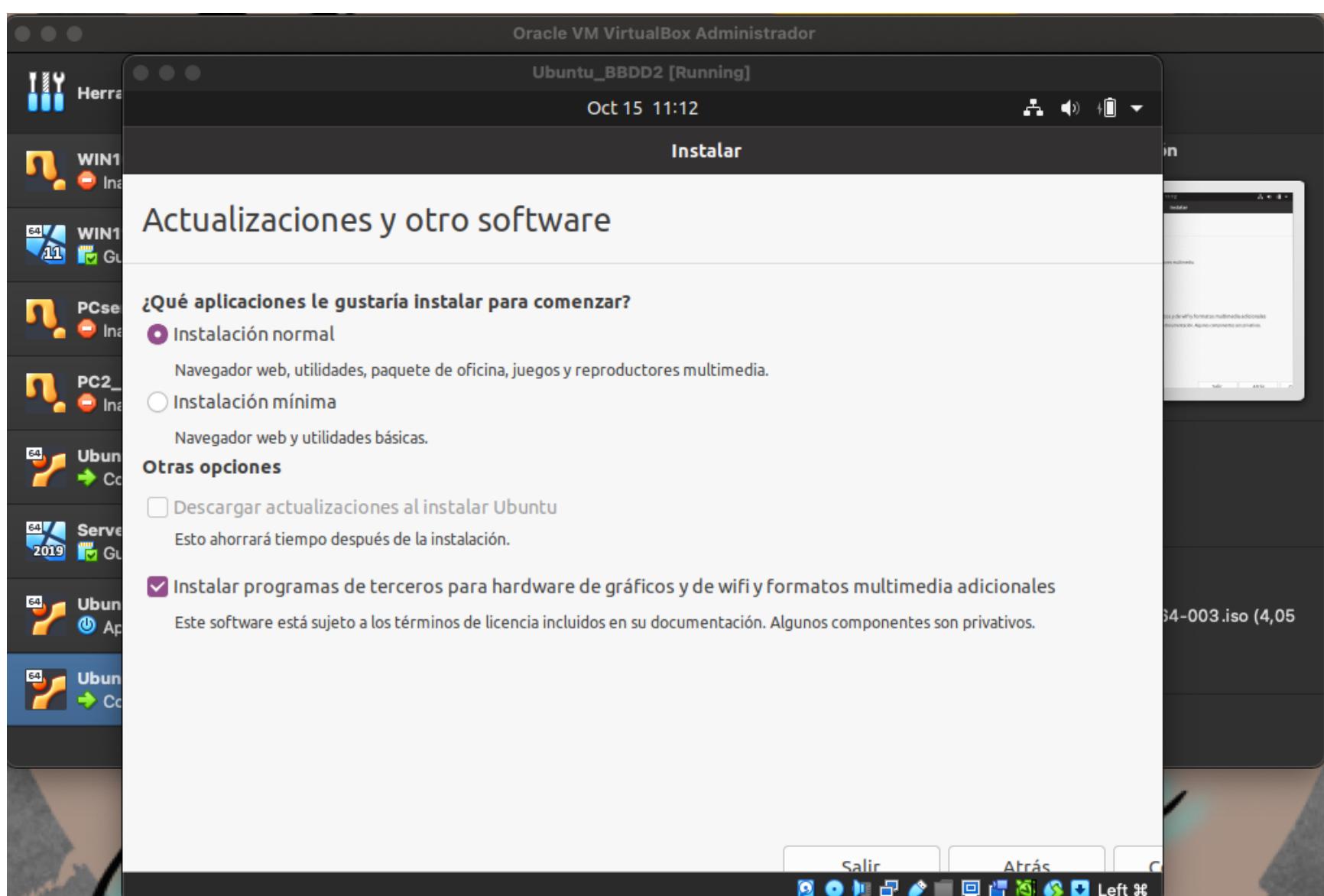
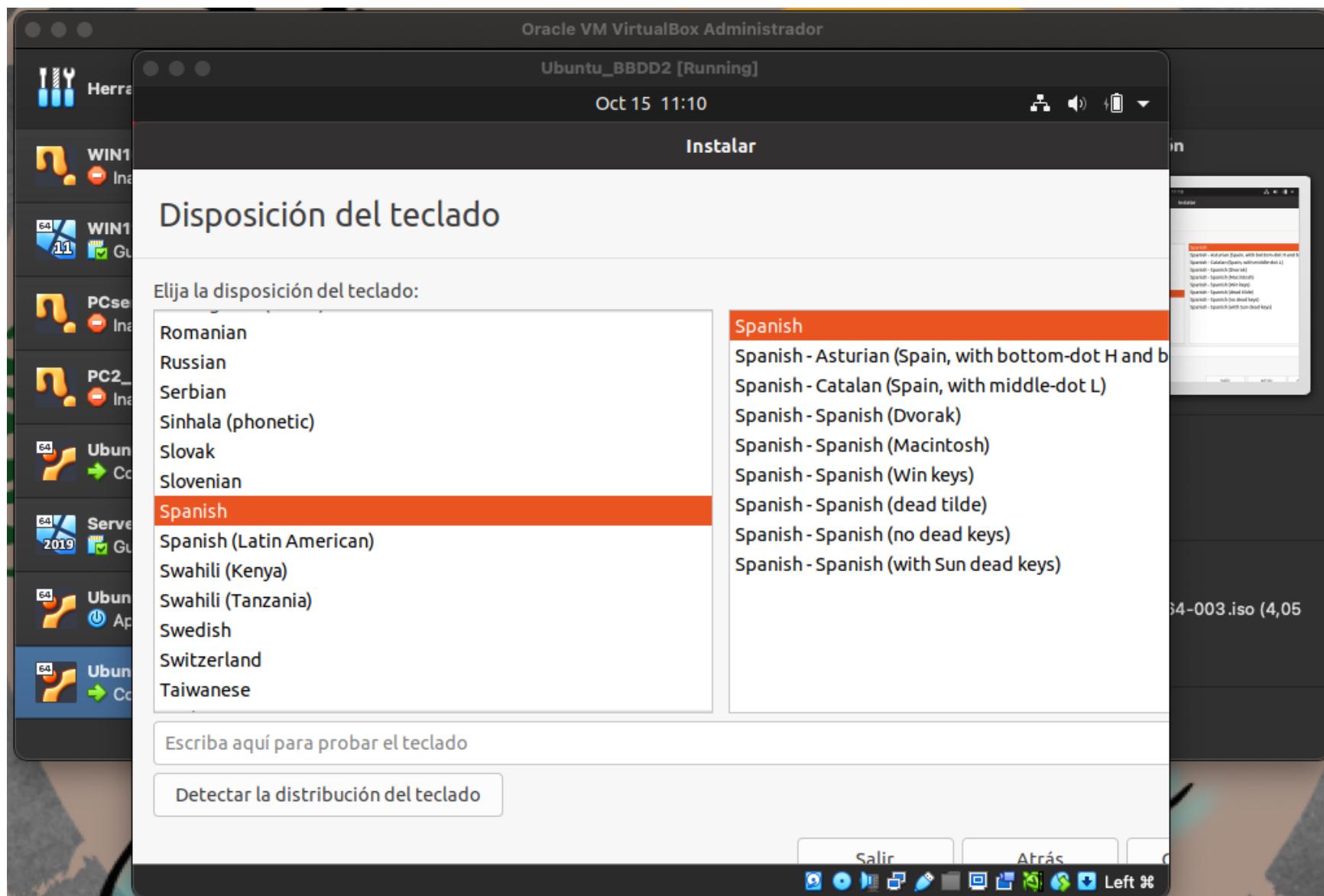
Por último se nos mostrará un cuadro resumen.

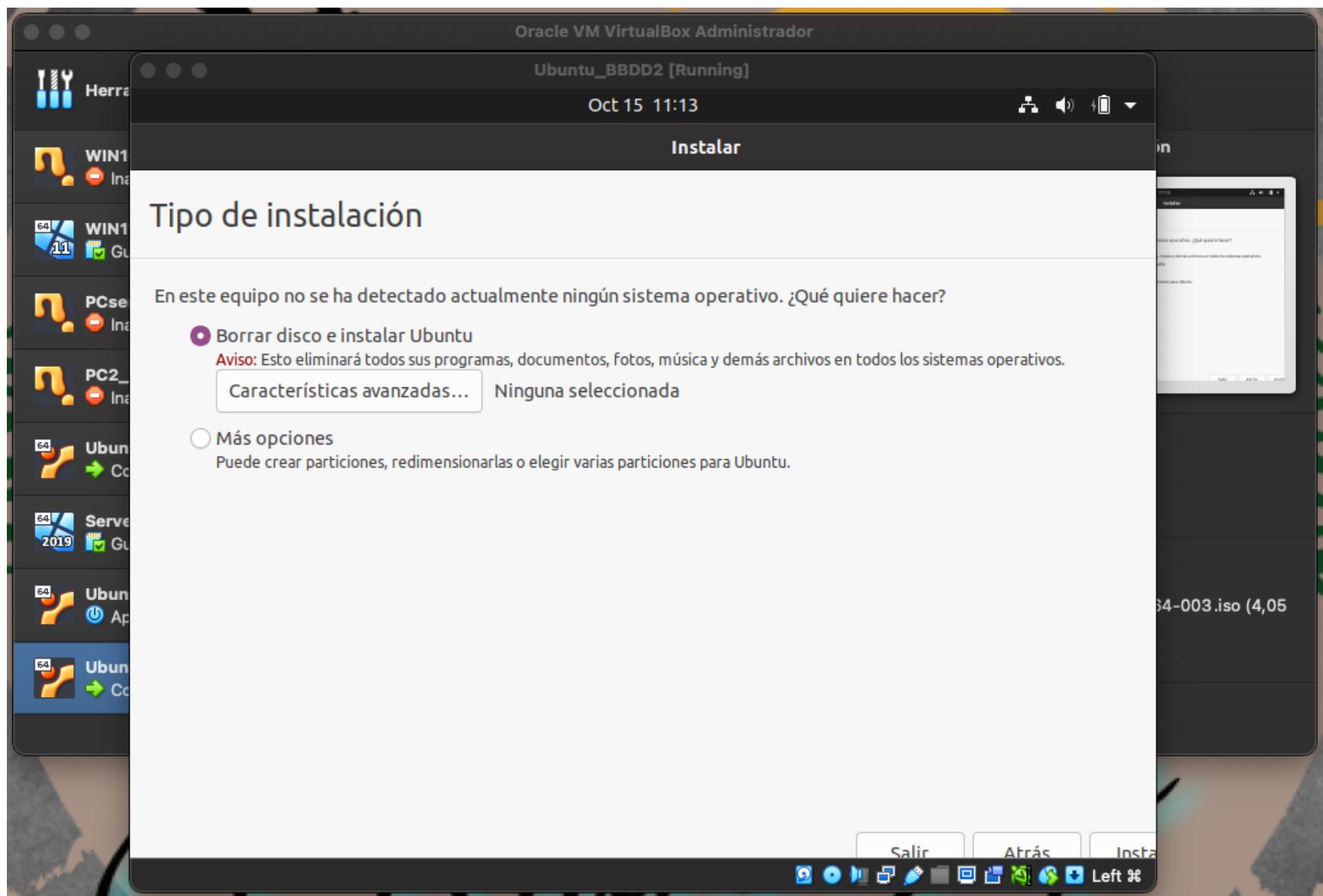


Ahora arrancamos la máquina virtual para comenzar con la instalación del sistema operativo. Seleccionaremos el idioma y "Install Ubuntu".

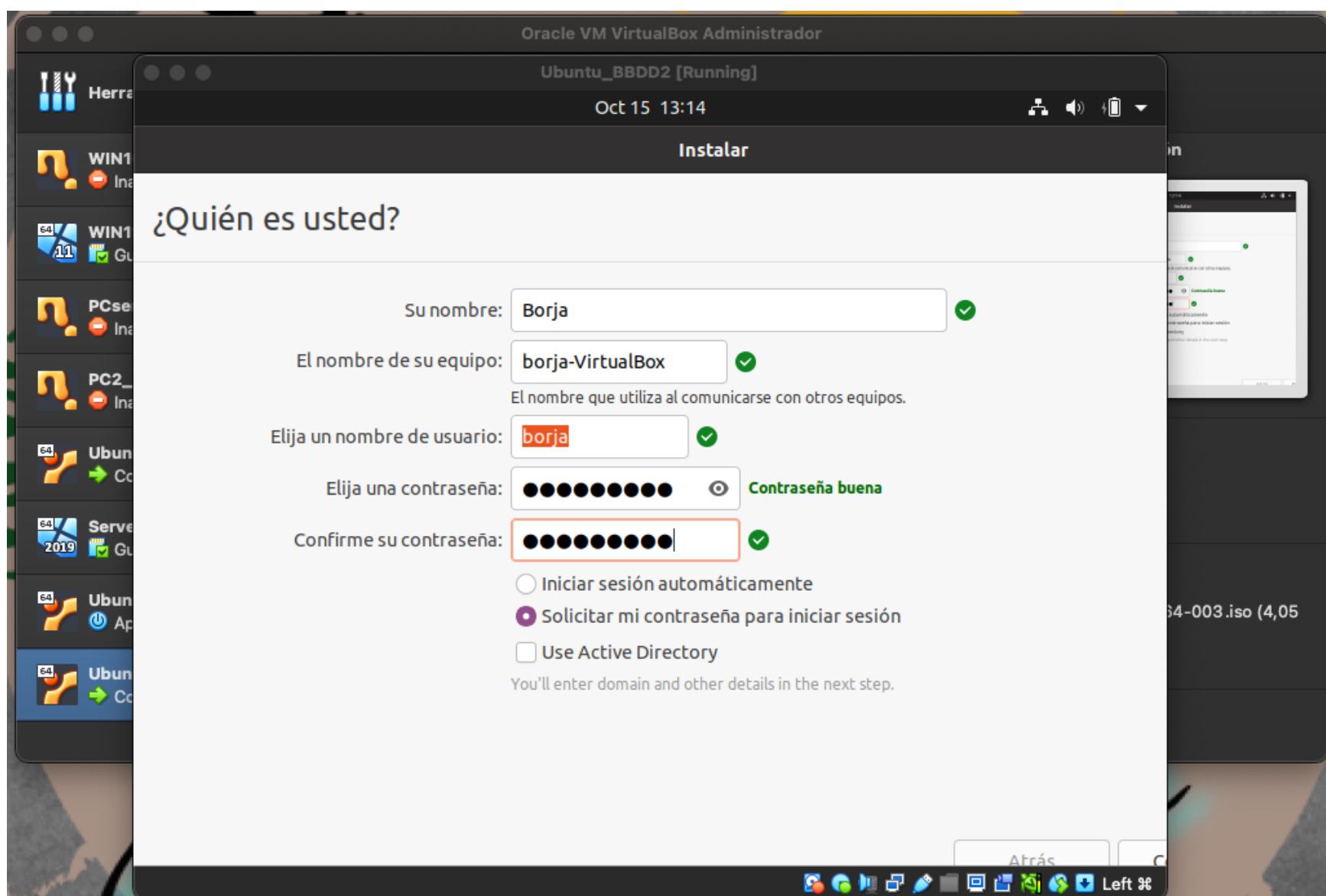


Seleccionaremos el tipo de teclado, instalación normal, "Borrar disco e instalar Ubuntu" y presionamos "Instalar".





Creamos nuestra cuenta de usuario y añadimos la contraseña.

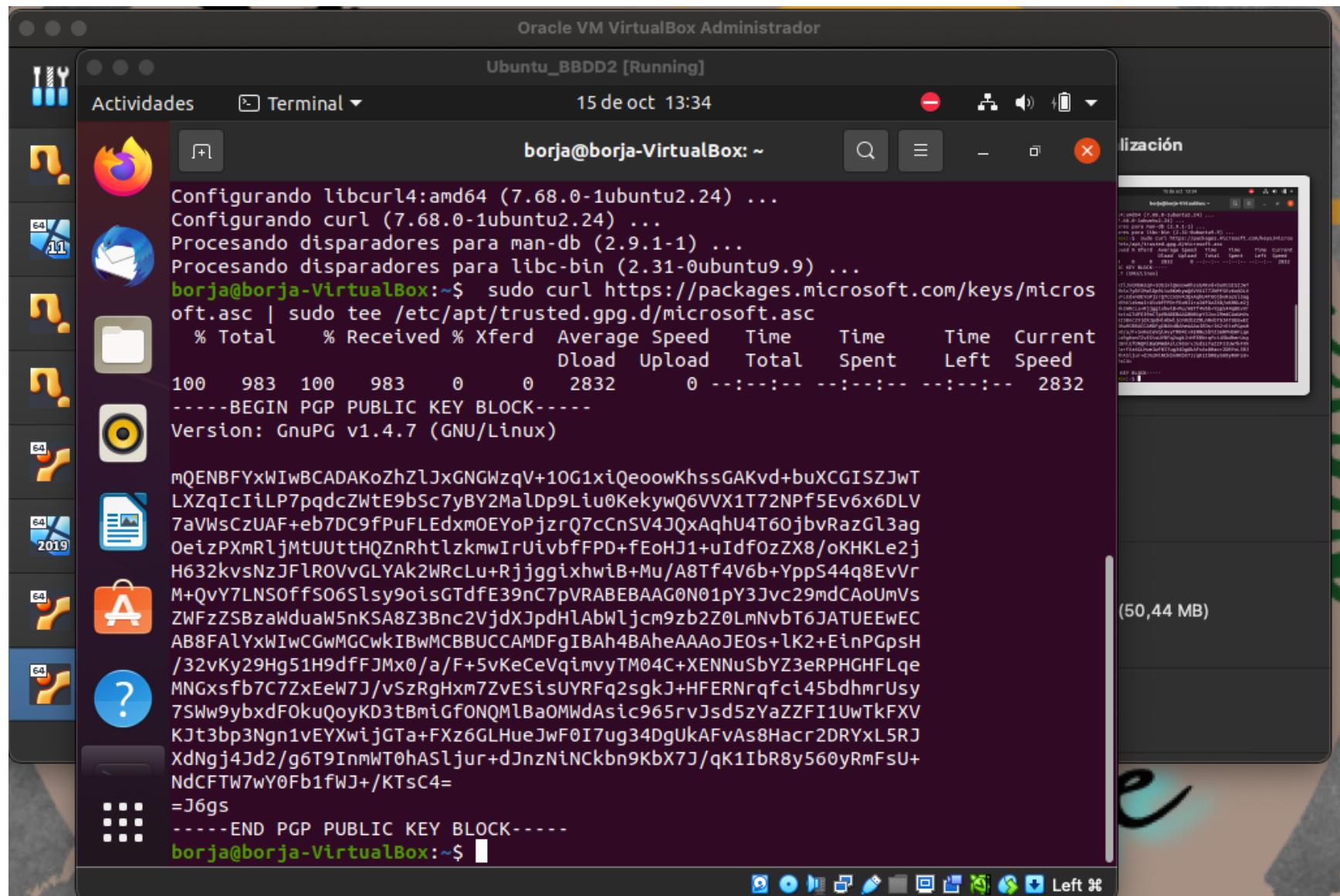


Por último dejamos marcada la opción de "Borrar disco e instalar Ubuntu" y dejamos que finalice la instalación.

## Instalar SQL Server 2019

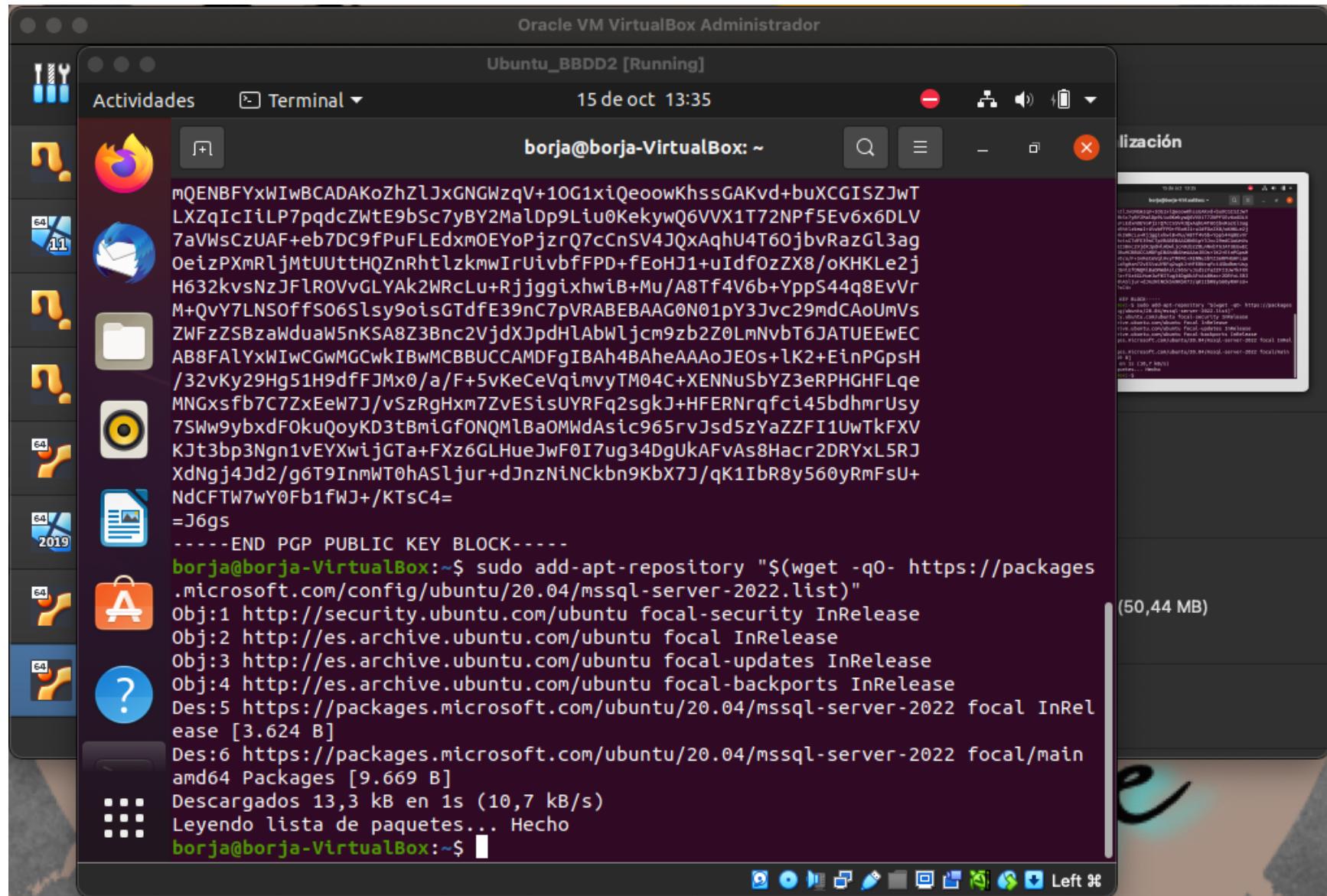
Una vez instalado el sistema operativo, iniciamos el sistema y abrimos la terminal. Aquí escribiremos

```
sudo curl https://packages.microsoft.com/keys/microsoft.asc | sudo tee /etc/apt/trusted
```



con esto importaremos las claves GPG del repositorio. Luego escribiremos

```
sudo add-apt-repository "$(wget -qO- https://packages.microsoft.com/config/ubuntu/20.04/r
```

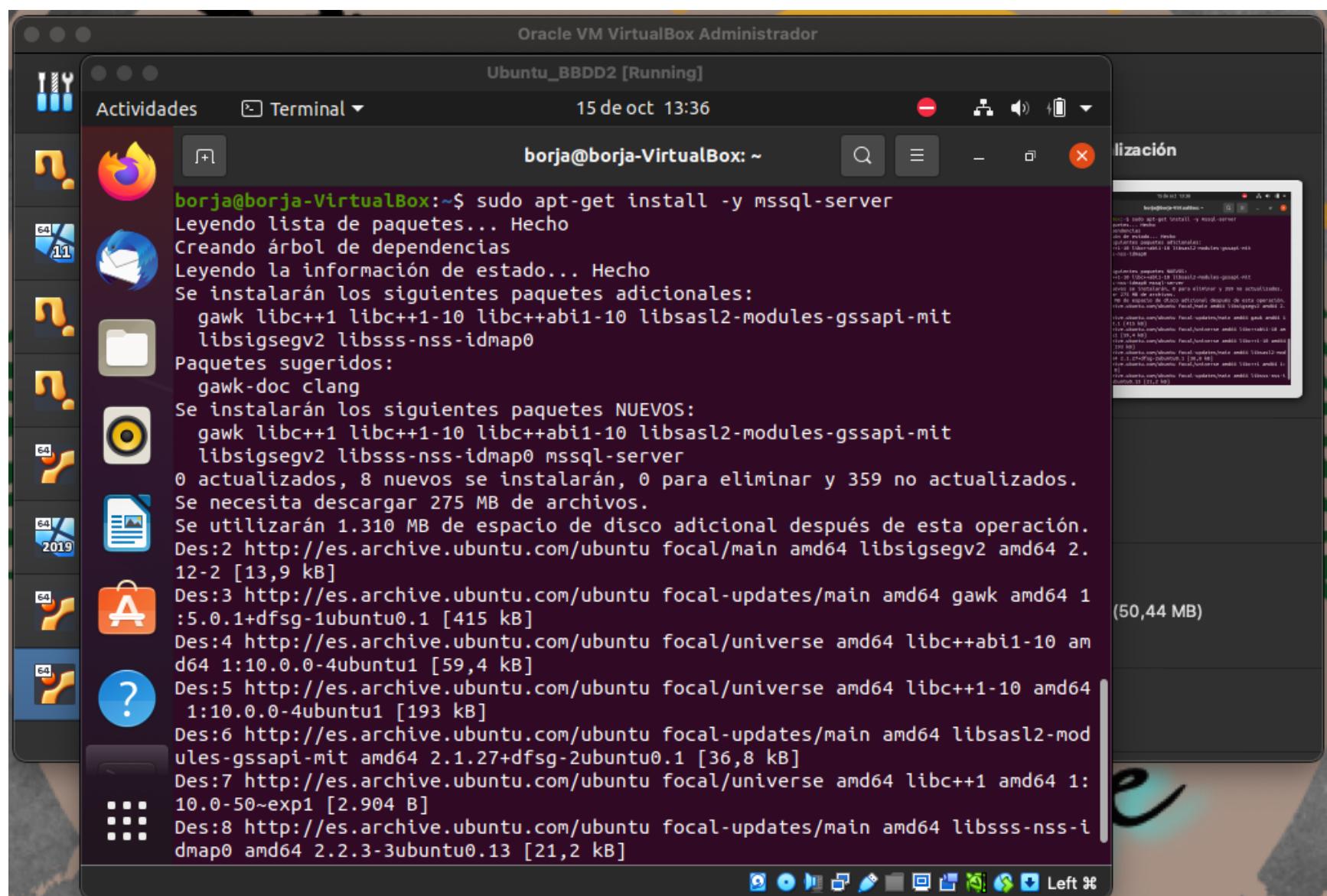


con esto registramos el repositorio de Ubuntu en SQL Server. Por último, instalamos SQL Server con

```

sudo apt-get update
sudo apt-get install -y mssql-server

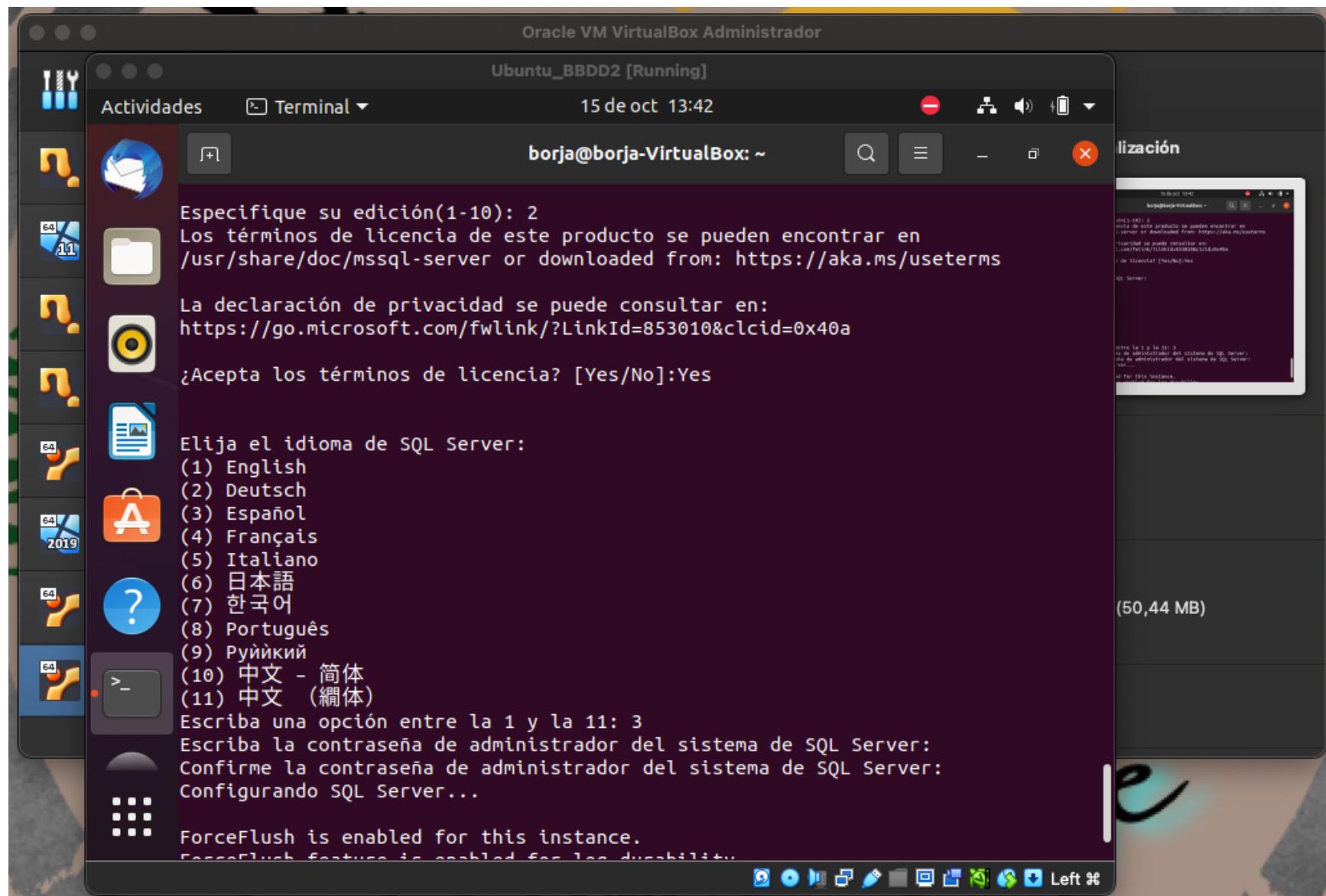
```



Ahora configuraremos SQL Server con:

```
sudo /opt/mssql/bin/mssql-conf setup
```

Nos pedirá que seleccionemos la edición, pulsamos "2" para escoger la enterprise que es la gratuita, luego aceptamos los términos, seleccionamos idioma español con el "3" y por último escribimos dos veces la contraseña del usuario administrador.



```
sudo ufw allow 22
sudo ufw allow 1433
sudo ufw enable
```

Ubuntu\_BBDD2 [Running]

Actividades Terminal 15 de oct 13:44 borja@borja-VirtualBox: ~

```
(6) 日本語
(7) 한국어
(8) Português
(9) Ру́сский
(10) 中文 - 简体
(11) 中文 (縹體)
Escriba una opción entre la 1 y la 11: 3
Escriba la contraseña de administrador del sistema de SQL Server:
Confirme la contraseña de administrador del sistema de SQL Server:
Configurando SQL Server...

ForceFlush is enabled for this instance.
ForceFlush feature is enabled for log durability.
Created symlink /etc/systemd/system/multi-user.target.wants/mssql-server.service → /lib/systemd/system/mssql-server.service.
La configuración se ha completado correctamente. SQL Server se está iniciando.
borja@borja-VirtualBox:~$ sudo ufw allow 22
sudo: ufw: orden no encontrada
borja@borja-VirtualBox:~$ sudo ufw allow 1433
sudo: ufw: orden no encontrada
borja@borja-VirtualBox:~$ sudo ufw allow 22
Reglas actualizadas
Reglas actualizadas (v6)
borja@borja-VirtualBox:~$ sudo ufw allow 1433
Reglas actualizadas
Reglas actualizadas (v6)
borja@borja-VirtualBox:~$ sudo ufw enable
El cortafuegos está activo y habilitado en el arranque del sistema
borja@borja-VirtualBox:~$
```

(50,44 MB)

Ubuntu\_BBDD2 [Running]

Actividades Terminal 15 de oct 13:45 borja@borja-VirtualBox: ~

```
La configuración se ha completado correctamente. SQL Server se está iniciando.
borja@borja-VirtualBox:~$ sudo ufw allow 22
sudo: ufw: orden no encontrada
borja@borja-VirtualBox:~$ sudo ufw allow 1433
sudo: ufw: orden no encontrada
borja@borja-VirtualBox:~$ sudo ufw allow 22
Reglas actualizadas
Reglas actualizadas (v6)
borja@borja-VirtualBox:~$ sudo ufw allow 1433
Reglas actualizadas
Reglas actualizadas (v6)
borja@borja-VirtualBox:~$ sudo ufw enable
El cortafuegos está activo y habilitado en el arranque del sistema
borja@borja-VirtualBox:~$ sudo systemctl restart mssql-server
borja@borja-VirtualBox:~$ sudo systemctl status mssql-server
● mssql-server.service - Microsoft SQL Server Database Engine
   Loaded: loaded (/lib/systemd/system/mssql-server.service; enabled; vendor>
   Active: active (running) since Tue 2024-10-15 13:45:05 CEST; 17ms ago
     Docs: https://docs.microsoft.com/en-us/sql/linux
      Main PID: 6188 (sqlservr)
        Tasks: 1
       Memory: 2.4M
      CGroup: /system.slice/mssql-server.service
              └─6188 /opt/mssql/bin/sqlservr

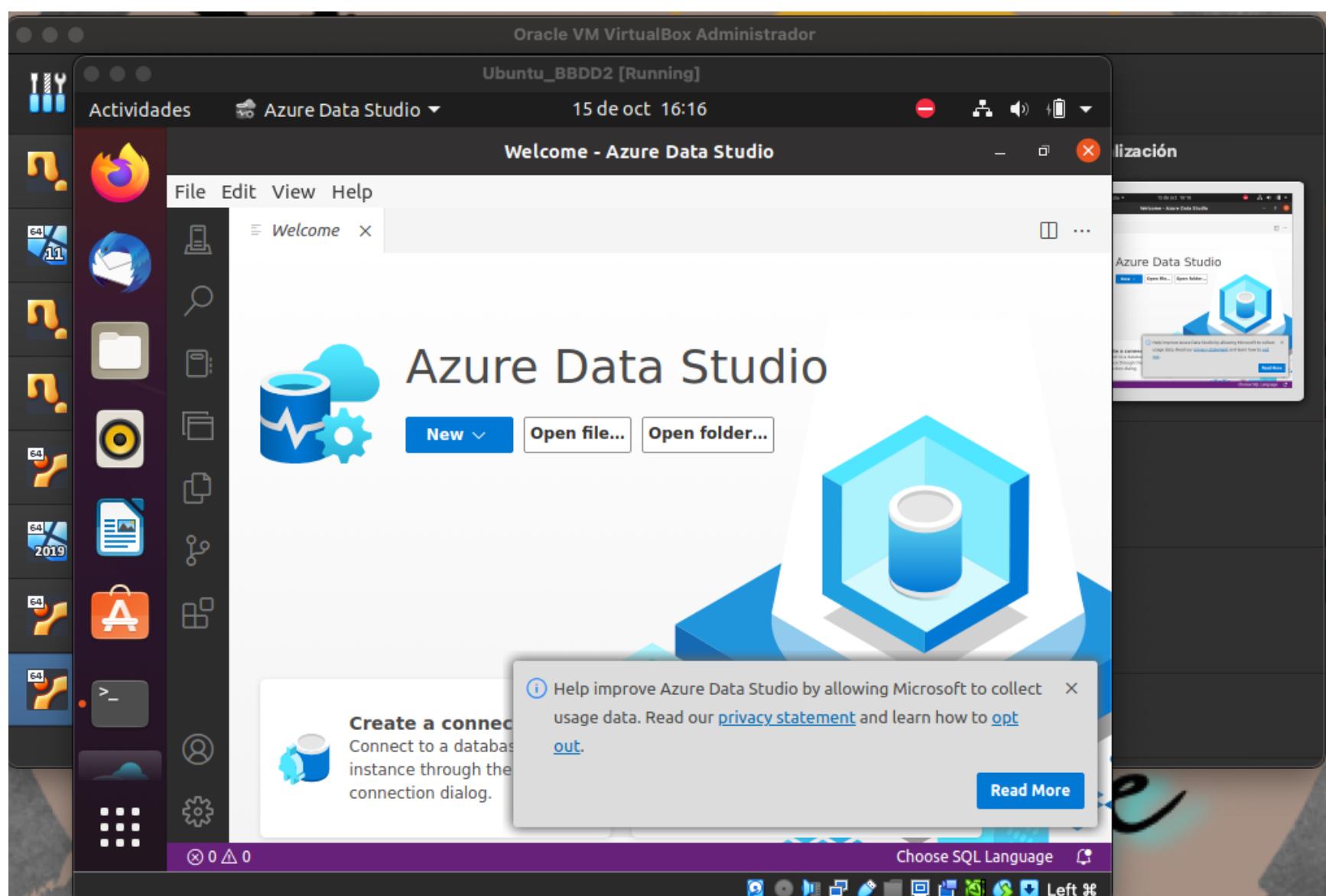
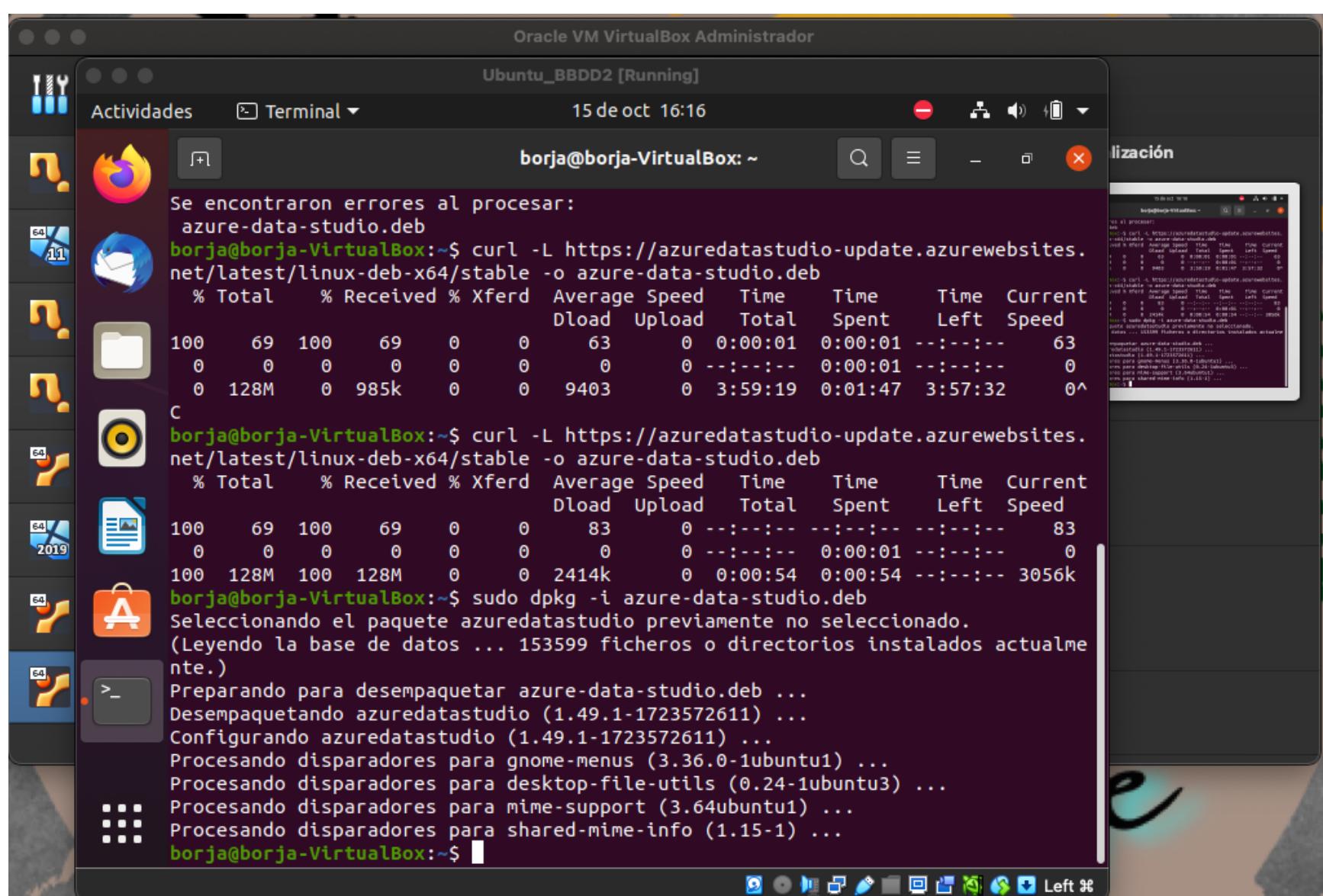
oct 15 13:45:05 borja-VirtualBox systemd[1]: Started Microsoft SQL Server Data>
lines 1-11/11 (END)
^C
borja@borja-VirtualBox:~$
```

(50,44 MB)

## Instalar GUI: Azure Data Studio

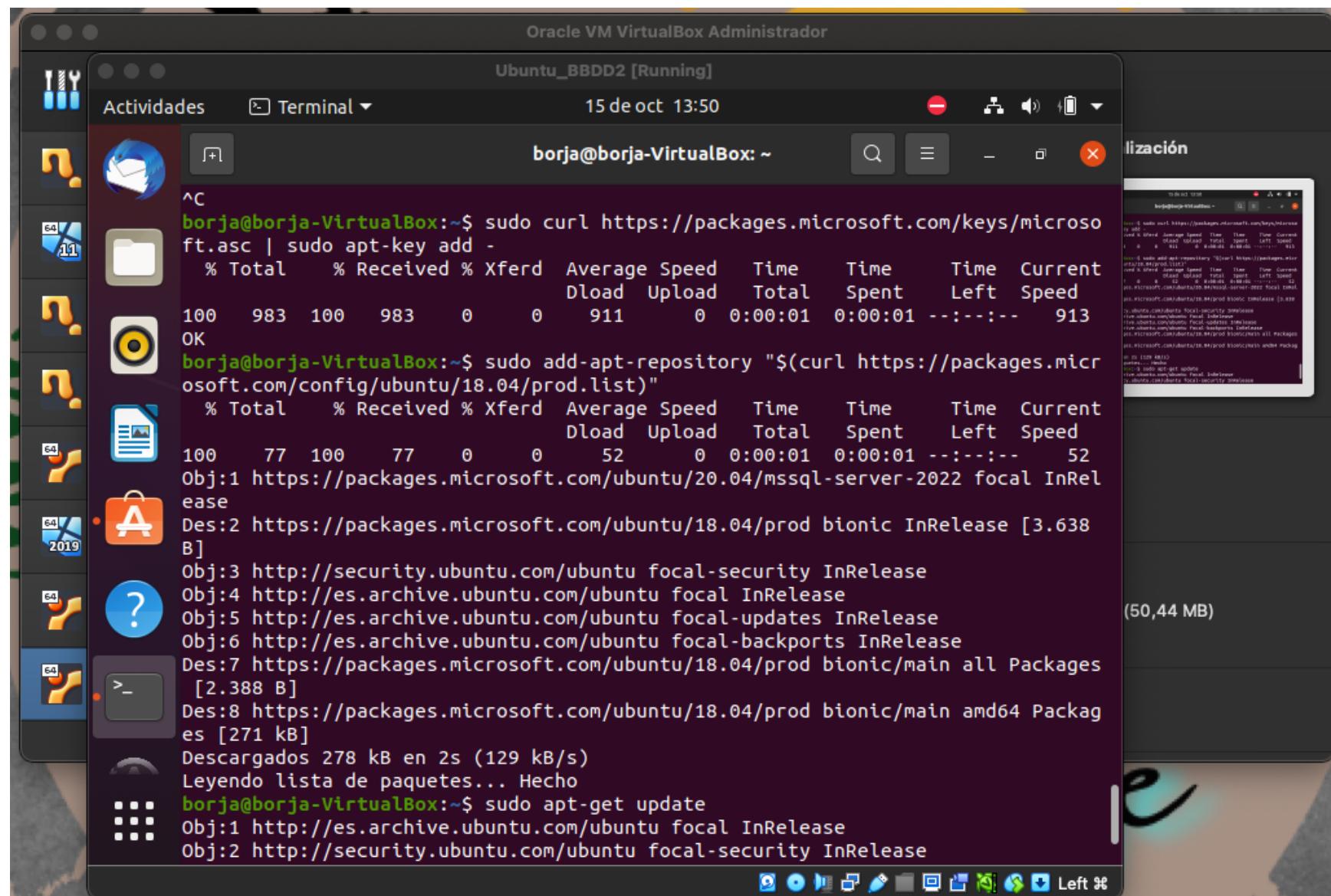
Tenemos varias opciones para instalar Azure Data Studio, hemos escogido hacerlo desde la terminal. Primero descargamos el paquete mediante curl y luego lo instalamos con dpkg:

```
curl -L https://azuredatastudio-update.azurewebsites.net/latest/linux-deb-x64/stable -o azure-data-studio.deb  
sudo dpkg -i azure-data-studio.deb
```



## Instalar Herramienta de Cliente : Sqlcmd

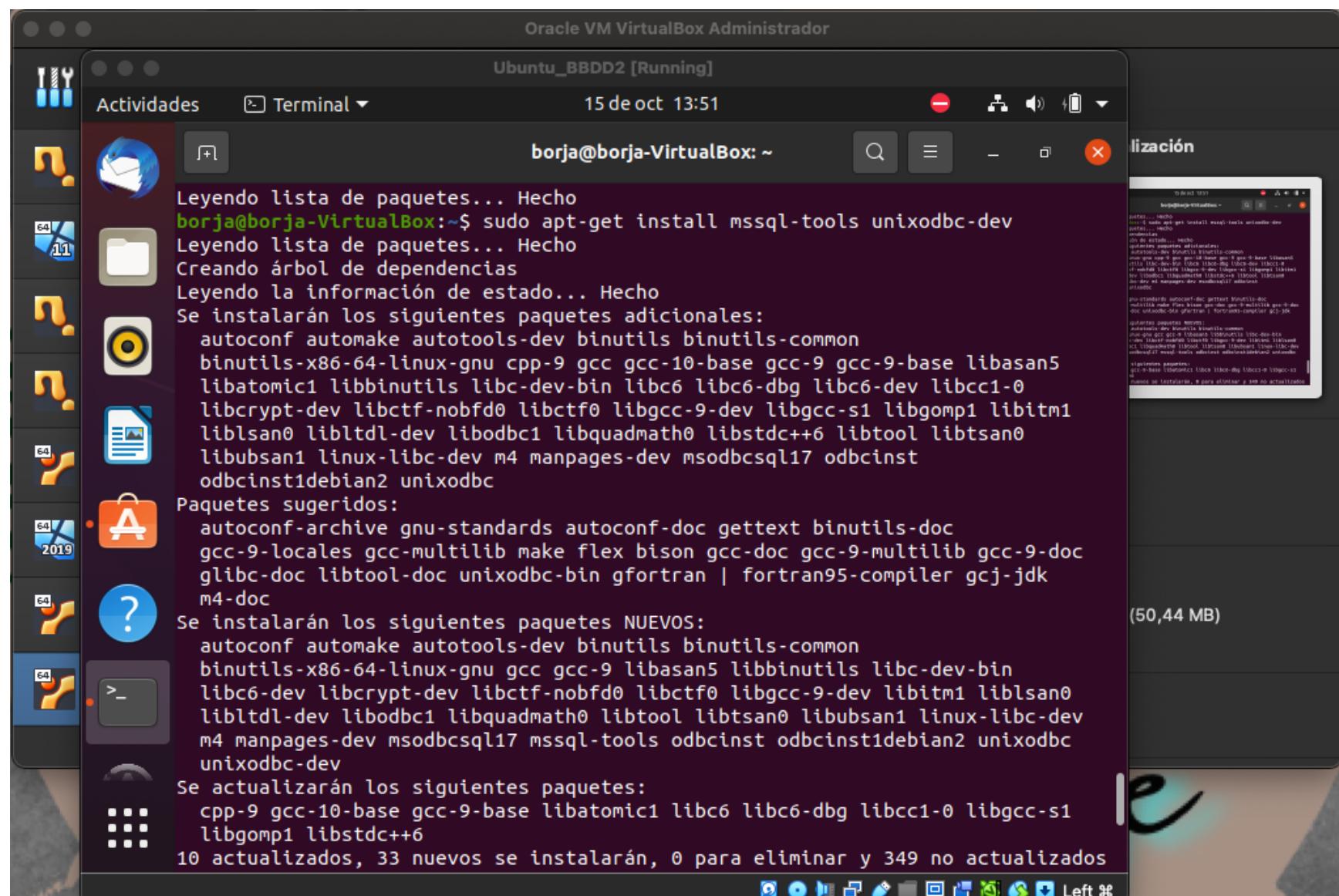
Para instalar Sqlcmd primero importamos las claves y registramos el repositorio con:



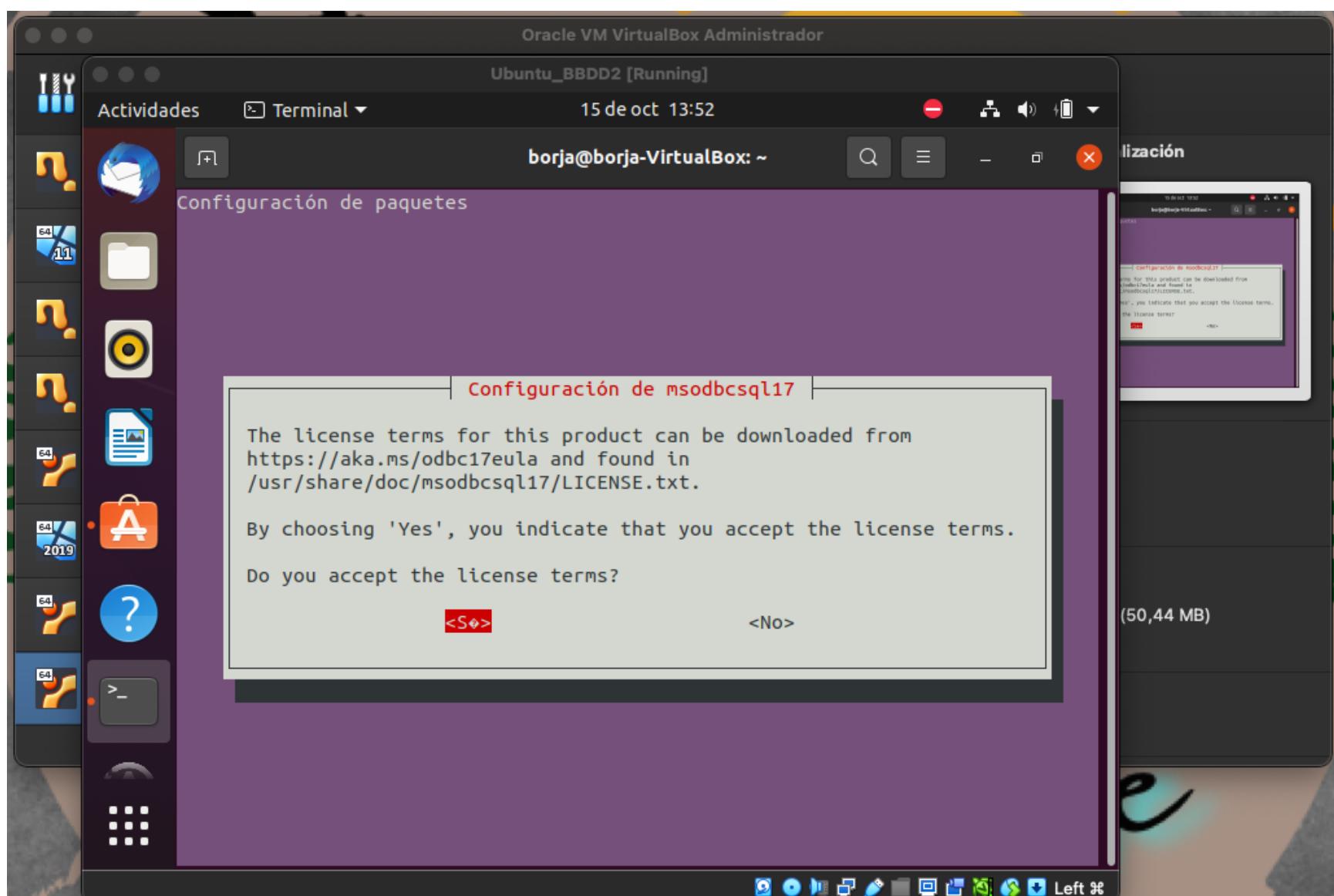
```
borja@borja-VirtualBox:~$ sudo curl https://packages.microsoft.com/keys/microsoft.asc | sudo apt-key add -
% Total    % Received % Xferd  Average Speed   Time   Time     Current
          Dload  Upload Total   Spent   Left Speed
100    983  100    983    0      0  911      0  0:00:01  0:00:01  -----  913
OK
borja@borja-VirtualBox:~$ sudo add-apt-repository "$(curl https://packages.microsoft.com/config/ubuntu/18.04/prod.list)"
% Total    % Received % Xferd  Average Speed   Time   Time     Current
          Dload  Upload Total   Spent   Left Speed
100     77  100     77    0      0   52      0  0:00:01  0:00:01  -----   52
Obj:1 https://packages.microsoft.com/ubuntu/20.04/mssql-server-2022 focal InRelease
Des:2 https://packages.microsoft.com/ubuntu/18.04/prod bionic InRelease [3.638 B]
Obj:3 http://security.ubuntu.com/ubuntu focal-security InRelease
Obj:4 http://es.archive.ubuntu.com/ubuntu focal InRelease
Obj:5 http://es.archive.ubuntu.com/ubuntu focal-updates InRelease
Obj:6 http://es.archive.ubuntu.com/ubuntu focal-backports InRelease
Des:7 https://packages.microsoft.com/ubuntu/18.04/prod bionic/main all Packages [2.388 B]
Des:8 https://packages.microsoft.com/ubuntu/18.04/prod bionic/main amd64 Packages [271 kB]
Descargados 278 kB en 2s (129 kB/s)
Leyendo lista de paquetes... Hecho
borja@borja-VirtualBox:~$ sudo apt-get update
Obj:1 http://es.archive.ubuntu.com/ubuntu focal InRelease
Obj:2 http://security.ubuntu.com/ubuntu focal-security InRelease
```

Luego instalamos sqlcmd con:

```
sudo apt-get install mssql-tools unixodbc-dev
```



```
Leyendo lista de paquetes... Hecho
borja@borja-VirtualBox:~$ sudo apt-get install mssql-tools unixodbc-dev
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  autoconf automake autotools-dev binutils binutils-common
  binutils-x86-64-linux-gnu cpp-9 gcc gcc-10-base gcc-9 gcc-9-base libasan5
  libatomic1 libbinutils libc-dev-bin libc6 libc6-dbg libc6-dev libcc1-0
  libcrypt-dev libctf-nobfd0 libctf0 libgcc-9-dev libgcc-s1 libgomp1 libitm1
  liblsan0 libltdl-dev libodbc1 libquadmath0 libstdc+++6 libtool libtsan0
  libubsan1 linux-libc-dev m4 manpages-dev msodbcsql17 odbcinst
  odbcinst1debian2 unixodbc
Paquetes sugeridos:
  autoconf-archive gnu-standards autoconf-doc gettext binutils-doc
  gcc-9-locales gcc-multilib make flex bison gcc-doc gcc-9-multilib gcc-9-doc
  glibc-doc libtool-doc unixodbc-bin gfortran | fortran95-compiler gcj-jdk
  m4-doc
Se instalarán los siguientes paquetes NUEVOS:
  autoconf automake autotools-dev binutils binutils-common
  binutils-x86-64-linux-gnu gcc gcc-9 libasan5 libbinutils libc-dev-bin
  libc6-dev libcrypt-dev libctf-nobfd0 libctf0 libgcc-9-dev libitm1 liblsan0
  libltdl-dev libodbc1 libquadmath0 libtool libtsan0 libubsan1 linux-libc-dev
  m4 manpages-dev msodbcsql17 mssql-tools odbcinst odbcinst1debian2 unixodbc
  unixodbc-dev
Se actualizarán los siguientes paquetes:
  cpp-9 gcc-10-base gcc-9-base libatomic1 libc6 libc6-dbg libcc1-0 libgcc-s1
  libgomp1 libstdc+++6
10 actualizados, 33 nuevos se instalarán, 0 para eliminar y 349 no actualizados
```



Para poder acceder a sqlcmd desde cualquier ubicación, añadimos la ubicación al PATH:

```
echo 'export PATH="$PATH:/opt/mssql-tools/bin"' >> ~/.bashrc
source ~/.bashrc
```

Ahora entramos con nuestro usuario y vamos a crear la base de datos, añadir una tabla y un registro en la misma.

Oracle VM VirtualBox Administrador

Ubuntu\_BBDD2 [Running]

Actividades Terminal 15 de oct 13:59

borja@borja-VirtualBox: ~

```
borja@borja-VirtualBox:~$ sqlcmd -S localhost -U SA -P Abcd1234.
1> CREATE DATABASE autoescuela;
2> GO
1> CREATE TABLE profesores;
2> GO
Msg 102, Level 15, State 1, Server borja-VirtualBox, Line 1
Sintaxis incorrecta cerca de ';'.
1> USE autoescuela;
2> GO
Se cambió el contexto de la base de datos a 'autoescuela'.
1> CREATE TABLE profesores (
2> ID INT PRIMARY KEY IDENTITY(1,1),
3> Nombre NVARCHAR(50),
4> Apellido NVARCHAR(50),
5> F_Nacimiento DATE;
6> Email NVARCHAR(100)
7> );
8> GO
Msg 102, Level 15, State 1, Server borja-VirtualBox, Line 5
Sintaxis incorrecta cerca de ';'.
1>
```

(50,44 MB)

Oracle VM VirtualBox Administrador

Ubuntu\_BBDD2 [Running]

Actividades Terminal 15 de oct 14:13

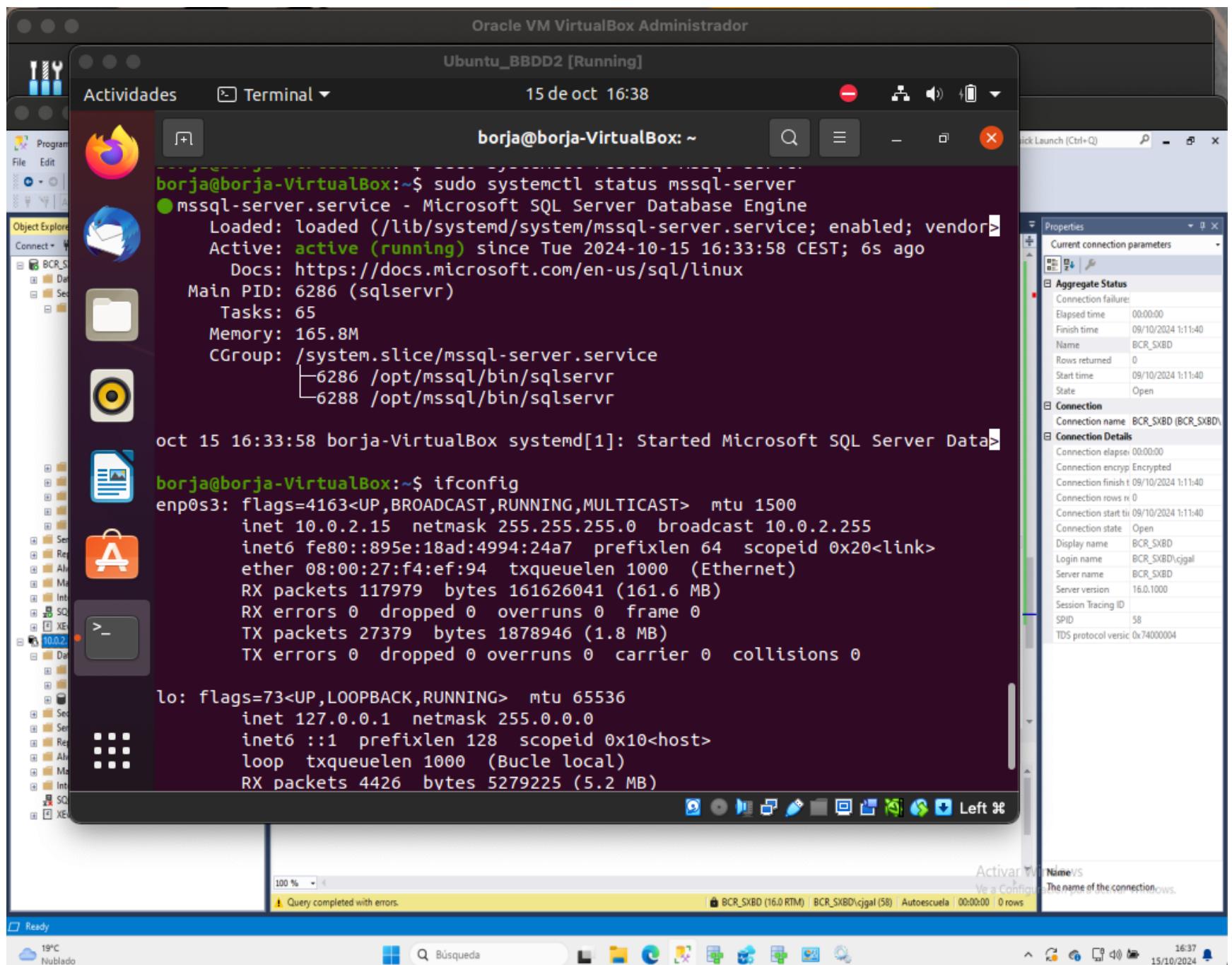
borja@borja-VirtualBox: ~

```
1> SELECT * FROM profesores;
2> GO
ID      Nombre          F_Nacimiento      Email        Apellido
-----  -----
(0 rows affected)
1> INSERT INTO profesores (Nombre, Apellido, F_Nacimiento, Email)
2> VALUES ('Borja', 'Costa', '1984-12-09', 'bcr@gmail.com');
3> GO
(1 rows affected)
1> SELECT * FROM profesores;
2> GO
ID      Nombre          F_Nacimiento      Email        Apellido
-----  -----
1      Borja           1984-12-09       bcr@gmail.com  Costa
(1 rows affected)
1>
```

(50,44 MB)

## Conectarse desde VM W10 al Servidor UBUNTU.

Primero comprobamos que el servidor en Ubuntu está activo y luego hacemos un ifconfig para ver la IP del ordenador.



Luego vamos a Windows, nos aseguramos que el servicio SSMS agent está funcionando y abrimos una conexión, en server name introducimos la dirección IP del servidor Ubuntu, usuario y contraseña introducimos los del servidor ubuntu, en este caso usuario "SA" y contraseña "Abcd1234.". El método de autenticación "SQL Server Authentication". Al presionar en conectar, vemos como se establece la conexión y se nos abre en el explorador del lateral el servidor Ubuntu con las bases de datos que tenemos en el.

