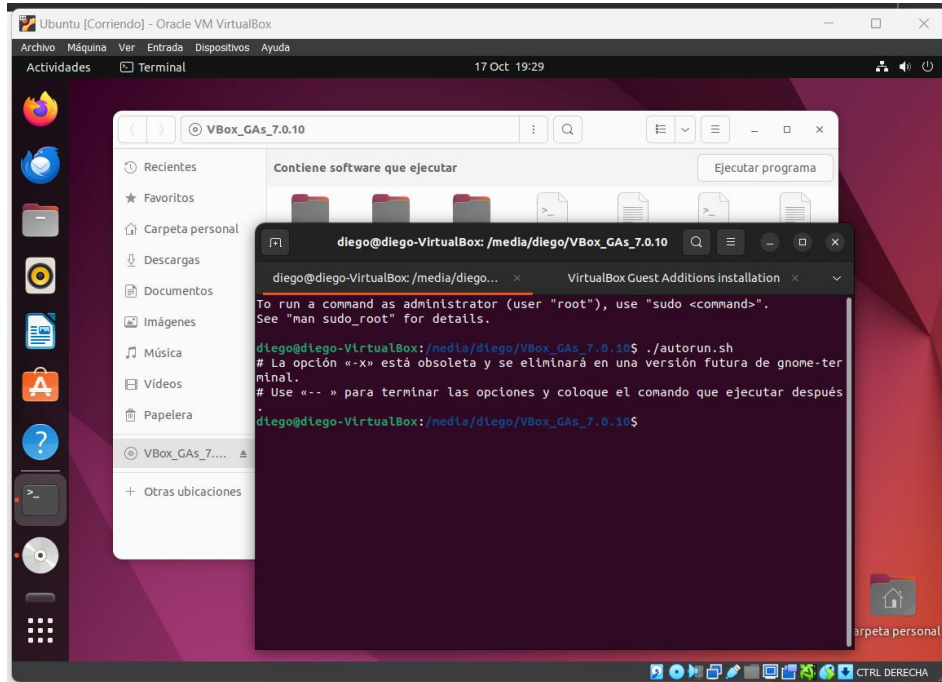
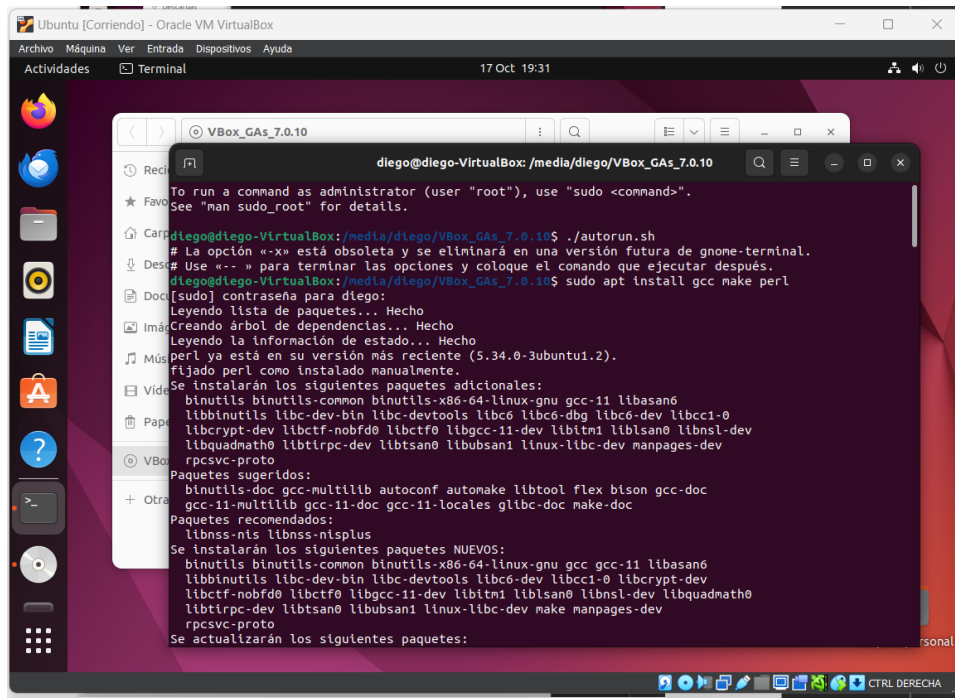


En primer lugar se tiene que hacer la instalación del complemento **guest additions**, para instalarlo habría que ir a la pestaña superior en la que pone “dispositivos” y desde ahí insertamos el CD virtual de **guest additions**, el cual se monta en la máquina virtual como si le insertásemos un CD real.



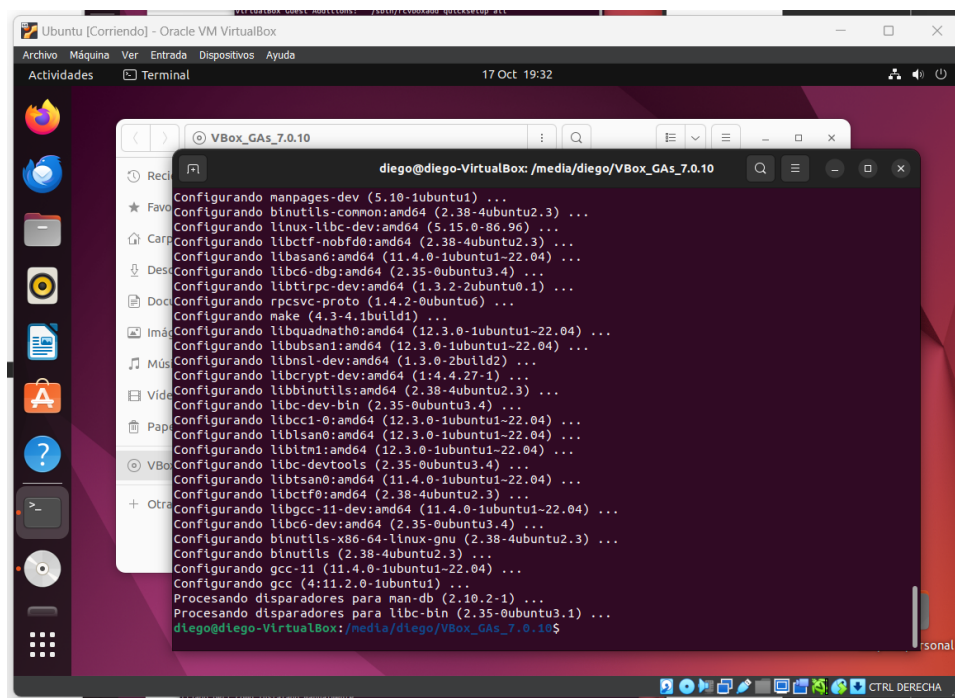
Una vez hecho el paso anterior tenemos que ejecutar el comando **./autorun.sh**



The screenshot shows a terminal window titled 'diego@diego-VirtualBox: /media/diego/VBox_GAs_7.0.10'. The user has run the command 'sudo apt install gcc make perl'. The terminal output shows the following steps:

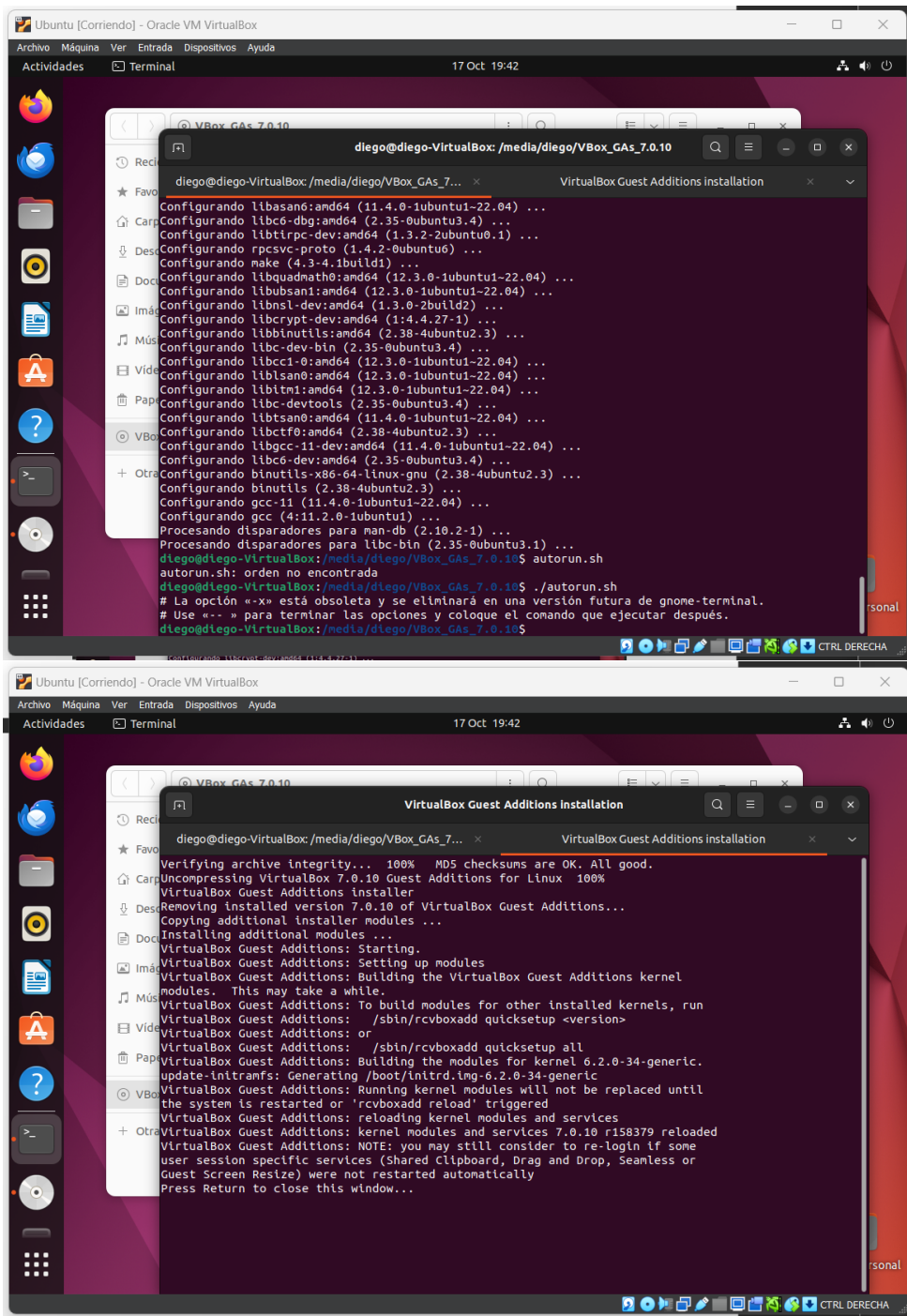
```
diego@diego-VirtualBox: /media/diego/VBox_GAs_7.0.10$ ./autorun.sh
# La opción «-x» está obsoleta y se eliminará en una versión futura de gnome-terminal.
Des# Use «-» para terminar las opciones y coloque el comando que ejecutar después.
diego@diego-VirtualBox: /media/diego/VBox_GAs_7.0.10$ sudo apt install gcc make perl
[sudo] contraseña para diego:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
perl ya está en su versión más reciente (5.34.0-3ubuntu1.2).
El paquete perl está fijado como instalado manualmente.
Se instalarán los siguientes paquetes adicionales:
binutils binutils-common binutils-x86-64-linux-gnu gcc-11 libasan6
libbinutils libc-dev-bin libc-devtools libc6 libc6-dbg libc6-dev libcc1-0
libcrypt-dev libctf-nobfd0 libctf0 libgcc-11-dev libitm1 liblsan0 libnsl-dev
libquadmath0 libtirpc-dev libubsan0 libubsan1 linux-libc-dev manpages-dev
rpcsvc-proto
Paquetes sugeridos:
binutils-doc gcc-multilib autoconf automake libtool flex bison gcc-doc
gcc-11-multilib gcc-11-doc gcc-11-locales glibc-doc make-doc
Paquetes recomendados:
libnss-nis libnss-nisplus
Se instalarán los siguientes paquetes NUEVOS:
binutils binutils-common binutils-x86-64-linux-gnu gcc gcc-11 libasan6
libbinutils libc-dev-bin libc-devtools libc6-dev libcc1-0 libcrypt-dev
libctf-nobfd0 libctf0 libgcc-11-dev libitm1 liblsan0 libnsl-dev libquadmath0
libtirpc-dev libubsan0 libubsan1 linux-libc-dev make manpages-dev
rpcsvc-proto
Se actualizarán los siguientes paquetes:
```

Ahora tenemos que instalar GCC, Make y Perl, para ello tenemos que ejecutar el comando **sudo apt install gcc make perl**

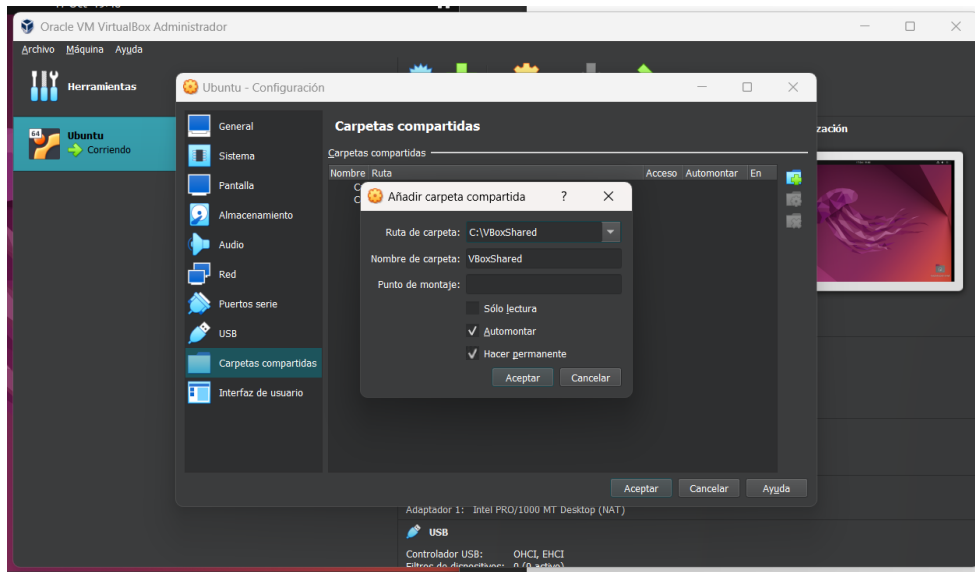


The screenshot shows the same terminal window as the previous one, but now displaying the configuration of the installed packages. The output is as follows:

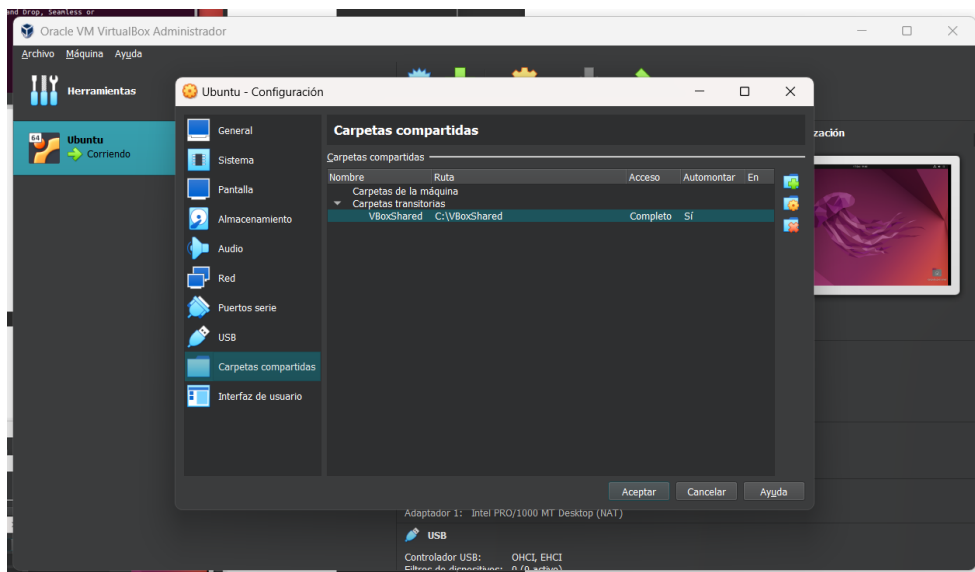
```
diego@diego-VirtualBox: /media/diego/VBox_GAs_7.0.10$
Configurando manpages-dev (5.10-1ubuntu1) ...
Configurando binutils-common:amd64 (2.38-4ubuntu2.3) ...
Configurando linux-libc-dev:amd64 (5.15.0-86.96) ...
Configurando libctf-nobfd0:amd64 (2.38-4ubuntu2.3) ...
Configurando libasan6:amd64 (11.4.0-1ubuntu1-22.04) ...
Configurando libc6-dbg:amd64 (2.35-0ubuntu3.4) ...
Configurando libtirpc-dev:amd64 (1.3.2-2ubuntu0.1) ...
Configurando rpcsvc-proto (1.4.2-0ubuntu6) ...
Configurando make (4.3-4.1build1) ...
Configurando libquadmath0:amd64 (12.3.0-1ubuntu1-22.04) ...
Configurando libubsan1:amd64 (12.3.0-1ubuntu1-22.04) ...
Configurando libnsl-dev:amd64 (1.3.0-2build2) ...
Configurando libcrypt-dev:amd64 (1:4.4.27-1) ...
Configurando libbinutils:amd64 (2.38-4ubuntu2.3) ...
Configurando libc-dev-bin (2.35-0ubuntu3.4) ...
Configurando libcc1-0:amd64 (12.3.0-1ubuntu1-22.04) ...
Configurando liblsan0:amd64 (12.3.0-1ubuntu1-22.04) ...
Configurando libitm1:amd64 (12.3.0-1ubuntu1-22.04) ...
Configurando libc-devtools (2.35-0ubuntu3.4) ...
Configurando libubsan0:amd64 (11.4.0-1ubuntu1-22.04) ...
Configurando libctf0:amd64 (2.38-4ubuntu2.3) ...
Configurando libgcc-11-dev:amd64 (11.4.0-1ubuntu1-22.04) ...
Configurando libc6-dev:amd64 (2.35-0ubuntu3.4) ...
Configurando binutils-x86-64-linux-gnu (2.38-4ubuntu2.3) ...
Configurando binutils (2.38-4ubuntu2.3) ...
Configurando gcc-11 (11.4.0-1ubuntu1-22.04) ...
Configurando gcc (4:11.2.0-1ubuntu1) ...
Procesando disparadores para man-db (2.10.2-1) ...
Procesando disparadores para libc-bin (2.35-0ubuntu3.1) ...
diego@diego-VirtualBox: /media/diego/VBox_GAs_7.0.10$
```

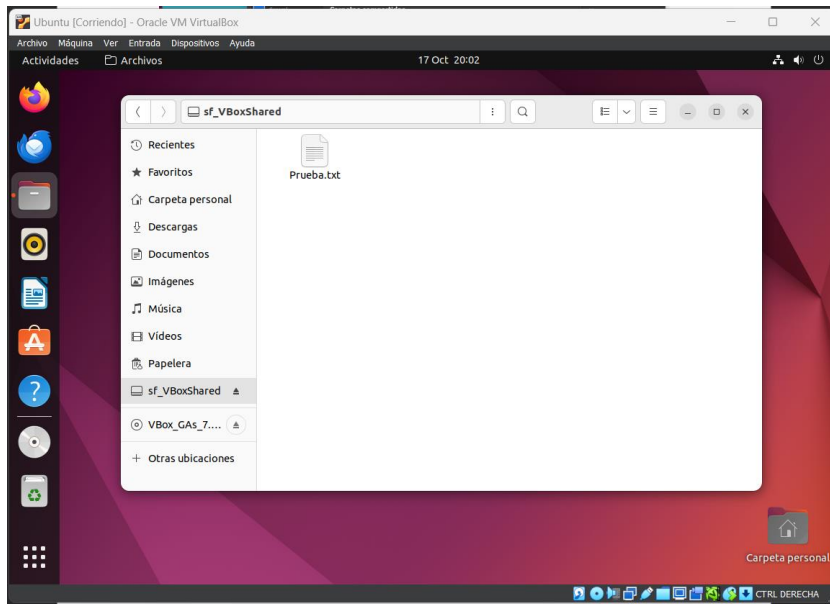


Una vez completada la ejecución de los comandos ya tendríamos instalado el complemento de **guest additions**.

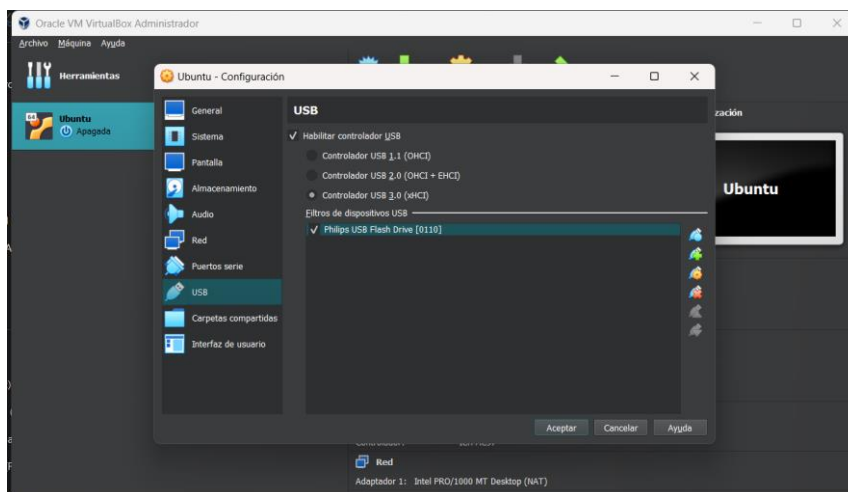


Una vez hecho lo anterior hay que hacer la **carpeta compartida**, para crear esta misma debemos crear en nuestro ordenador una carpeta, una vez hecho esto la seleccionamos desde la **configuración de carpetas compartidas** de VirtualBox para establecer su ruta en los ajustes, seleccionamos **automontar** y **hacer permanente** y le damos a aceptar.

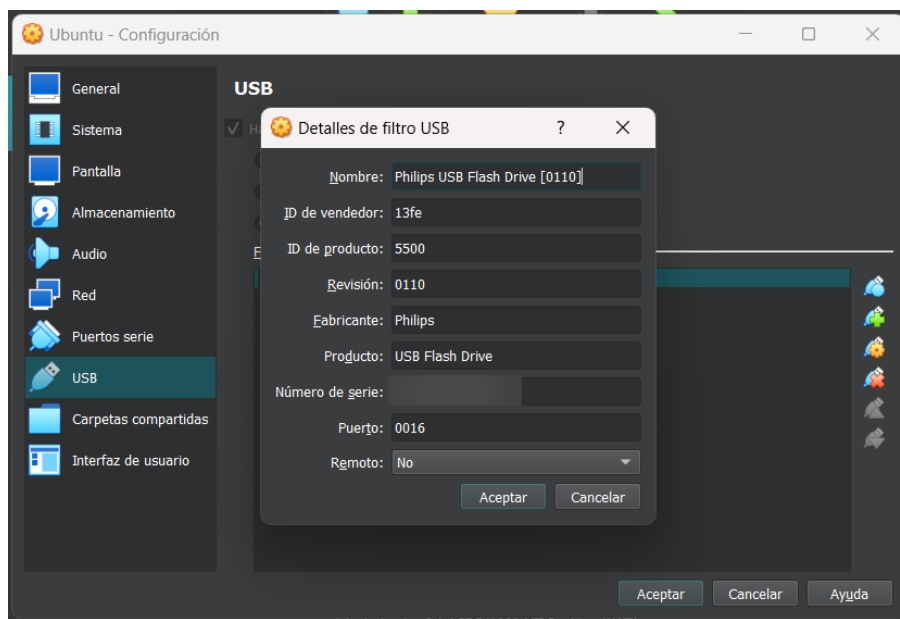
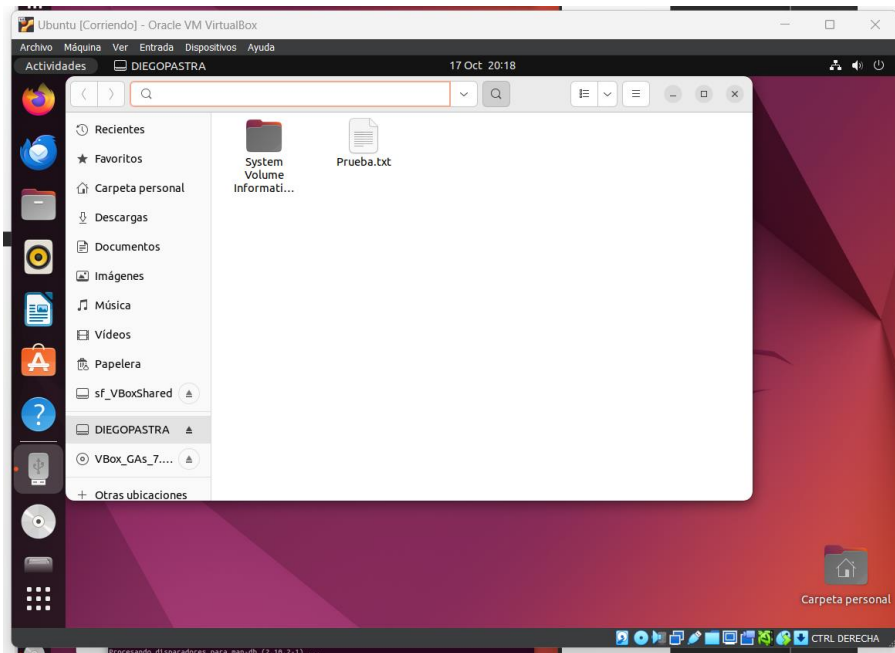




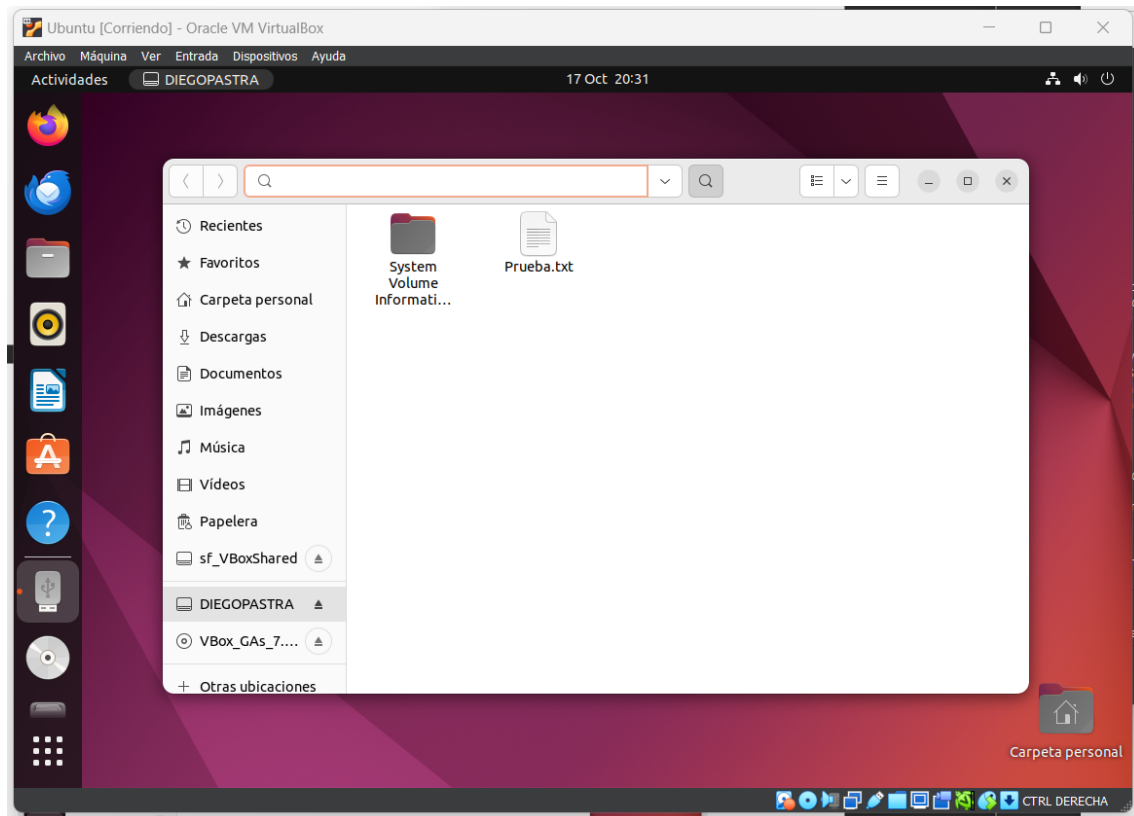
Una vez hemos logrado que nos aparezca la carpeta compartida y que nos deje entrar en ella (después de probar varios métodos facilitados en el pdf por el profesor), tenemos que **probar si funciona**, para esto metemos por ejemplo un archivo **.txt** cualquiera y debería de aparecer **tanto en nuestra carpeta de nuestro PC y como en la de nuestra máquina virtual** con Ubuntu.



A continuación, desde la **configuración USB** de la máquina virtual, añadimos un **pendrive**, con el objetivo de que nuestra MV con Ubuntu nos lo detecte y lo monte, y así poder transferir datos del pendrive a Ubuntu y viceversa.



Después nos aseguramos de tener añadido un **filtro USB**



Y volvemos a probar a ver si funciona el pendrive con el filtro

Posibles usos para los filtros USB en máquinas virtuales:

Los filtros USB en las MV son útiles porque con ellos se puede gestionar/controlar que dispositivos USB se pueden o no conectar a la máquina virtual, esto implica que se puede mejorar la seguridad de las MV con esta utilidad, además que simplifica las conexiones ya que si queremos conectar un USB a la MV en diferentes ocasiones no tendremos que volverlo a añadir, sino que simplemente funcionaria. Por lo tanto los filtros USB combinan utilidades de seguridad y comodidad.