

# Cómo montar un servidor Blynk legacy local

- OJO ESTO NO TIENES QUE HACERLO
- Raspberry: Qué hay que hacer para empezar
- Raspberry: Poner un Sistema Operativo
- Raspberry: Conectarlo a la red
- Raspberry: Conexión SSH
- Raspberry: Cambiar usuario y contraseña
- Raspberry: Apagar
- Raspberry: VNC
- Raspberry: Instalar Java 8
- Raspberry: Instalar y ejecutar el servidor Blynk local
- En Windows Instalar Java 11
- En Windows Instalar Blynk local
- En Windows Ejecutar Blynkserver
- En Linux

# OJO ESTO NO TIENES QUE HACERLO

Este apartado es para tu conocimiento NO LO TIENES QUE HACER, sólo para que sepas como CATEDU ha montado el servidor en la RASPBERRY por si te interesa seguir los mismos pasos

Como CATEDU ya te proporciona un servidor Blynk legacy **NO ES NECESARIO PARA EL CURSO**

Pero fuera del curso de Catedu, tienes que hacerlo.

Financiado por el Ministerio de Educación y Formación Profesional y por la Unión Europea - NextGenerationEU



# Raspberry: Qué hay que hacer para empezar

Lo primero que tenemos que conseguir es:

- Tener una Raspberry con el sistema operativo propio de la raspberry instalado, antes se llamaba Raspbian, ahora Raspberry Pi OS
- Comunicarnos con la Raspberry por comandos SSH
- Una vez hecho esto que es lo básico instalaremos el servidor Blynk



Ahora vamos a ver unas páginas para

- Instalar un **sistema operativo** en una Raspberry
- Una vez instalado hay que **conectarlo a la red wifi** y **ponerle una IP FIJA**
- Para **comunicarse vía comandos SSH**, aconsejamos hacerlo de forma remota, pues lo contrario implica tener siempre una pantalla, teclado y ratón conectado en la Raspberry

## ¿Qué Raspberry habría que comprar?

Aconsejamos una que tenga la Wi-fi incorporada, nosotros hemos hecho pruebas con la Raspberry Pi 3B+ y con la Raspberry Pi4 y no encontramos diferencias de velocidad para estos propósitos de robótica, luego por la mitad de precio aconsejamos la Raspberry Pi 3B



# Raspberry: Poner un Sistema Operativo

## ¿Es el único sistema operativo que se puede instalar?

**No**, podemos instalar otros sistemas operativos en nuestra Raspberry (evidentemente uno sólo cada vez, poner dos a la vez en la *rasp* sería forzarlo demasiado). Por ejemplo:

- Podemos instalar un **Linux-Ubuntu** ligero pero que se puede usar en una aula de informática (ver libro del curso de Aularagon Monta tu aula de informática con Raspberry).
- Si no podemos vivir sin Windows, hay un **Windows 10 especial** gratis (*sí has leído bien, los de Microsoft van entendiendo eso de si no puedes con ellos únete a ellos, pero no te hagas ilusiones, sólo vale para la Raspberry y no esperes una máquina que sustituya a tu PC va lentísimo*).
- Sistemas operativos para aplicaciones concretas:
  - Video juegos: Recalbox.
  - Dómótica y control Home Assistant
- Nuestro sistema operativo de este curso (y el mejor que "rula en la Rasp") pues está adaptado a la arquitectura ARM de la Raspberry es **RASPBIAN** o actualmente lo llaman **RASPBERRY PI OS**

Hay dos opciones, elige la que quieras (nuestra preferida es la A):

## Opción A Instalar el sistema operativo en la tarjeta sin NOOBS

Con esta opción no tienes que encender la Raspberry.

# Opción B Instalar la Raspbian en la tarjeta con NOOBS

Con esta opción SI que tienes que encender la Rasbperry con pantalla, teclado y ratón

Esto es válido para cualquier ordenador: Descargar la ISO de un Sistema Operativo, grabarlo en un pincho de forma autoarrancable, pero por curiosidad, no es tan fácil como en la RASPBERRY  
Se añade esta presentación como curiosidad:

# Raspberry: Conectarlo a la red

## Opción A La Raspberry con pantalla, teclado y ratón

Esta es más fácil, pero tienes que encender la Raspberry con teclado, pantalla y ratón:

## Opción B La Raspberry sin pantalla, teclado y ratón

Con esta opción es **más difícil**, pero no tienes que encender la Raspberry para la configuración, sólo manipular los ficheros de la tarjeta micro SD.

### Texto que hay que poner en interfaces

```
auto wlan0
iface wlan0 inet static
address 192.168.1.xxx
gateway 192.168.1.1
netmask 255.255.255.0
wpa-ssid el_nombre_tu_red_wifi
wpa-psk la_contraseña_de_tu_wifi
dns-nameservers 8.8.8.8 188.244.82.1
```

## Texto que hay que poner en wpa\_supplicant.conf

```
# /etc/wpa_supplicant/wpa_supplicant.conf
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
network={
    ssid="nombre de tu router o SSID"
    psk="tu contraseña del wi-fi"
    key_mgmt=WPA-PSK
}
```



# Raspberry: Conexión SSH

Controlar Raspberry pi de forma remota y de forma textual es muy rápido y eficaz, sobre todo para la robótica con AlphaBot.

**SSH** (Secure Shell) según [Wikipedia](#) es el nombre de un protocolo y del programa que lo implementa, y sirve para acceder servidores privados a través de una puerta trasera. Te recomiendo visitar [esta página](#) de Luis Llamas.

Recuerda que lo que hemos hecho es:

- Hemos conectado la Raspberry a la Wifi
- Hemos activado el protocolo SSH y VNC en la raspberry
- Hemos fijado una IP y la sabemos para poder acceder a él.

Ya podemos trabajar por SSH con la Raspberry, no nos hace falta que conectemos la raspberry a una pantalla, teclado, etc....

**YA PODEMOS ENCENDER LA RASPBERRY** si has elegido las opciones A de [3.-Raspbian](#) y [4.-Conectando](#) aún no habrás encendido la Raspberry.

## Si trabajas con Windows

No lo tiene nativo, tenemos que instalar [Putty](#)

Y rellenamos los campos de la IP (la que hayas determinado, en la imagen es 192.168.1.111), puerto y el protocolo SSH, es recomendable grabar la sesión para tenerlo en futuras entradas.



Se abre una ventana pidiendo el usuario **pi** y contraseña por defecto **raspberry**

## Si trabajas con Linux

Linux tiene de forma nativa el protocolo SSH, lo activamos en el icono en negro que es el **Terminal** :



Supongamos que la IP fija que le hemos asignado a la Raspberry la IP 192.168.1.131 Tecleamos

```
ssh pi@192.168.1.131
```

Te pide usuario y contraseña, por defecto es usuario pi y la contraseña *raspberrypi*, la tecleas **no la puedes ver**, te tiene que salir algo así en la pantalla:

**Cuando te sale pi@raspberrypi: es que ya está preparado para recibir órdenes**

```
cat@cat-Aspire-E5-571 ~ $ ssh pi@192.168.1.131
pi@192.168.1.131's password:
Linux raspberrypi 4.9.80-v7+ #1098 SMP Fri Mar 9 19:11:42 GMT 2018 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

Last login: Fri Mar 23 18:03:09 2018 from 192.168.1.131
pi@raspberrypi:~ $
```

## Nota por si te pasa

Si después de un intento fallido, o realizar la conexión con otra computadora.... te sale este error al intentar conectarte por ssh:

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ @ 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 the RSA host key has just been changed.
The fingerprint for the RSA key sent by the remote host is
Please contact your system administrator. Add correct host key in
/home/user/.ssh/known_hosts to get rid of this message. Offending key in
/home/user/.ssh/known_hosts:1 RSA host key for ras.mydomain.com has changed and you
have >requested strict checking. Host key verification failed.
```

Entonces antes de intentar conectarte por ssh, hay que obligar a la Raspberry que reinicie las claves de conexión de ssh, ejecuta esta orden :

Supongamos que la IP fija que le hemos asignado a la Raspberry en el punto 4 es 192.168.1.131  
Tecleamos

**ssh-keygen -R 192.168.1.131**



# Raspberry: Cambiar usuario y contraseña

El usuario por defecto es **pi** y su contraseña es **raspberry** pero lo podemos cambiar perfectamente

## Opción A remota con SSH

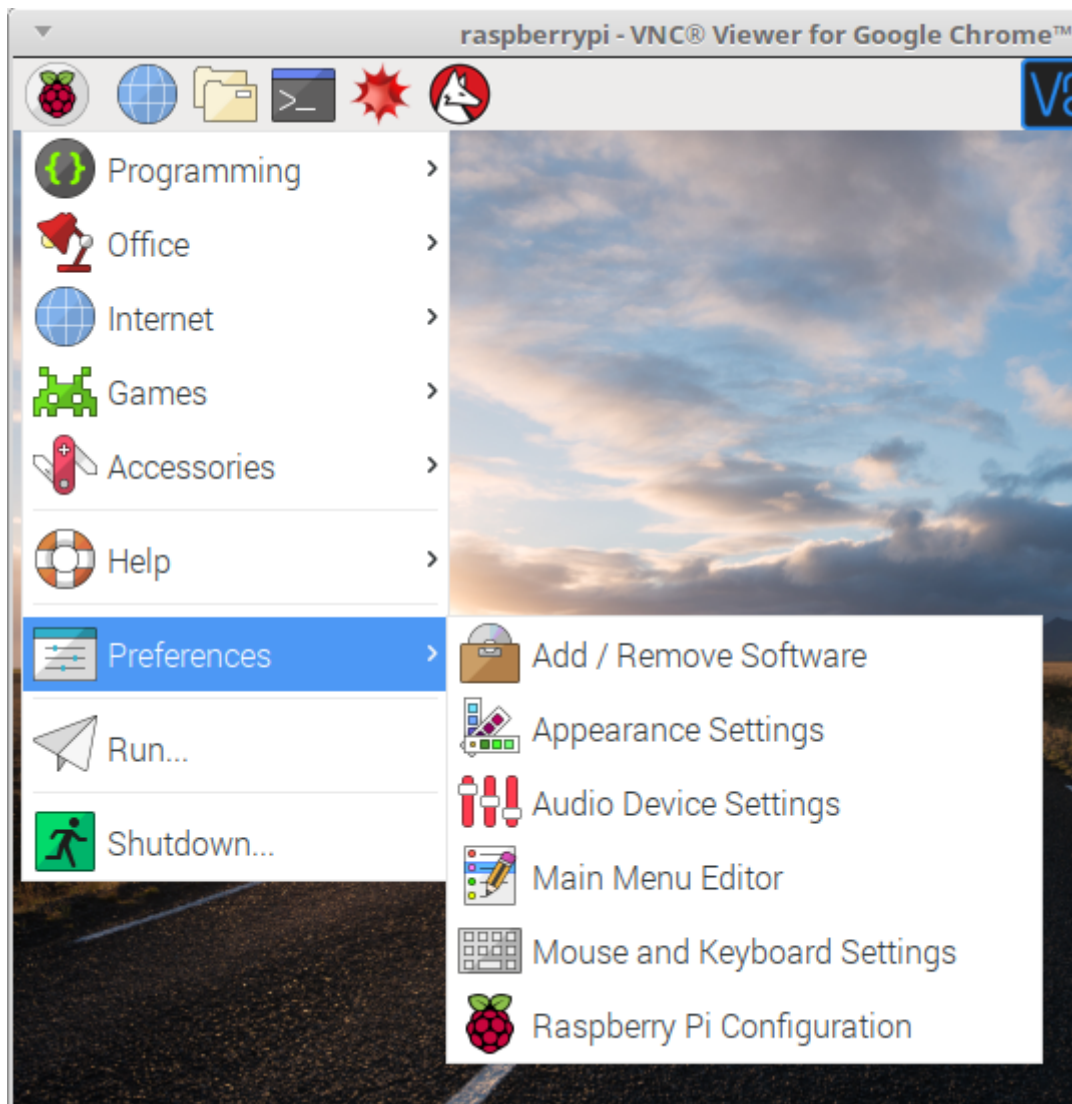
Entramos por SSH como hemos aprendido, y tecleamos

**sudo passwd pi**

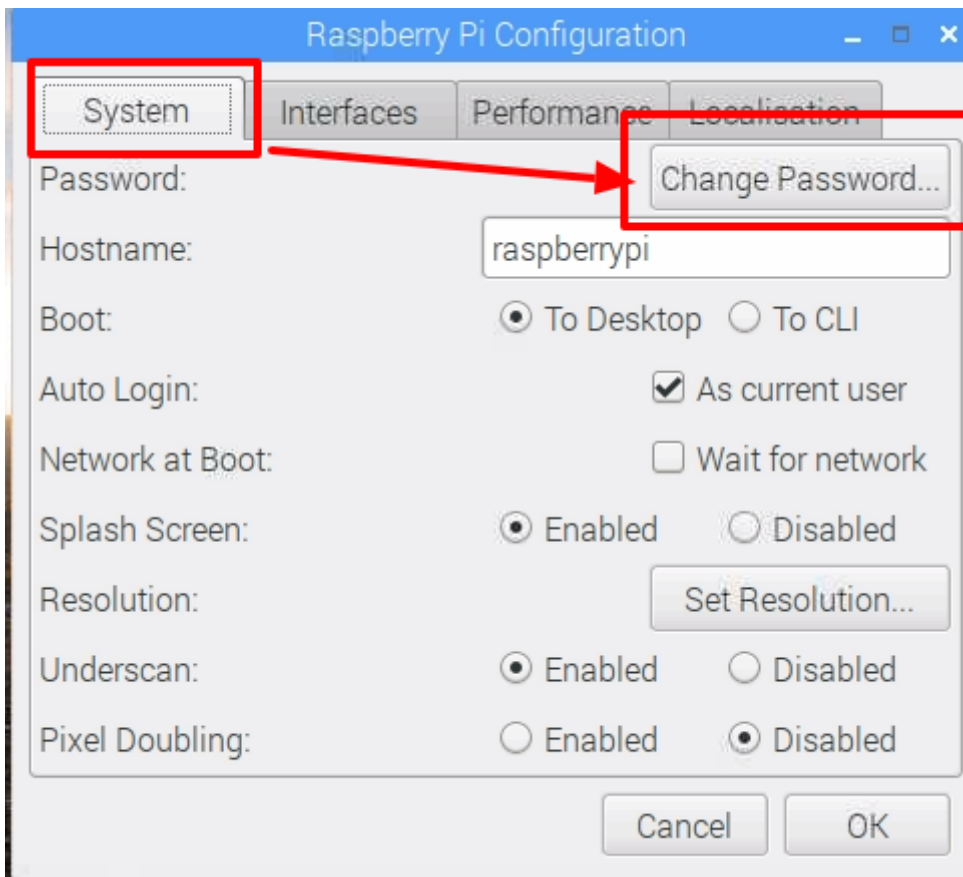
y nos pide enseguida la contraseña, la tenemos que insertar dos veces *no se puede ver*.

## Opción B local por pantalla

Entramos por VNC o conectamos una pantalla, teclado y ratón a la raspberry y en **Preferencias-RaspberryPi configuration**



Y entramos en **System**



# Raspberry: Apagar

Apagar de forma caliente (es decir simplemente quitando de la fuente de alimentación la corriente) daña al sistema operativo Raspbian dejando ficheros corruptos.

## 7.1 Opción de forma remota con SSH

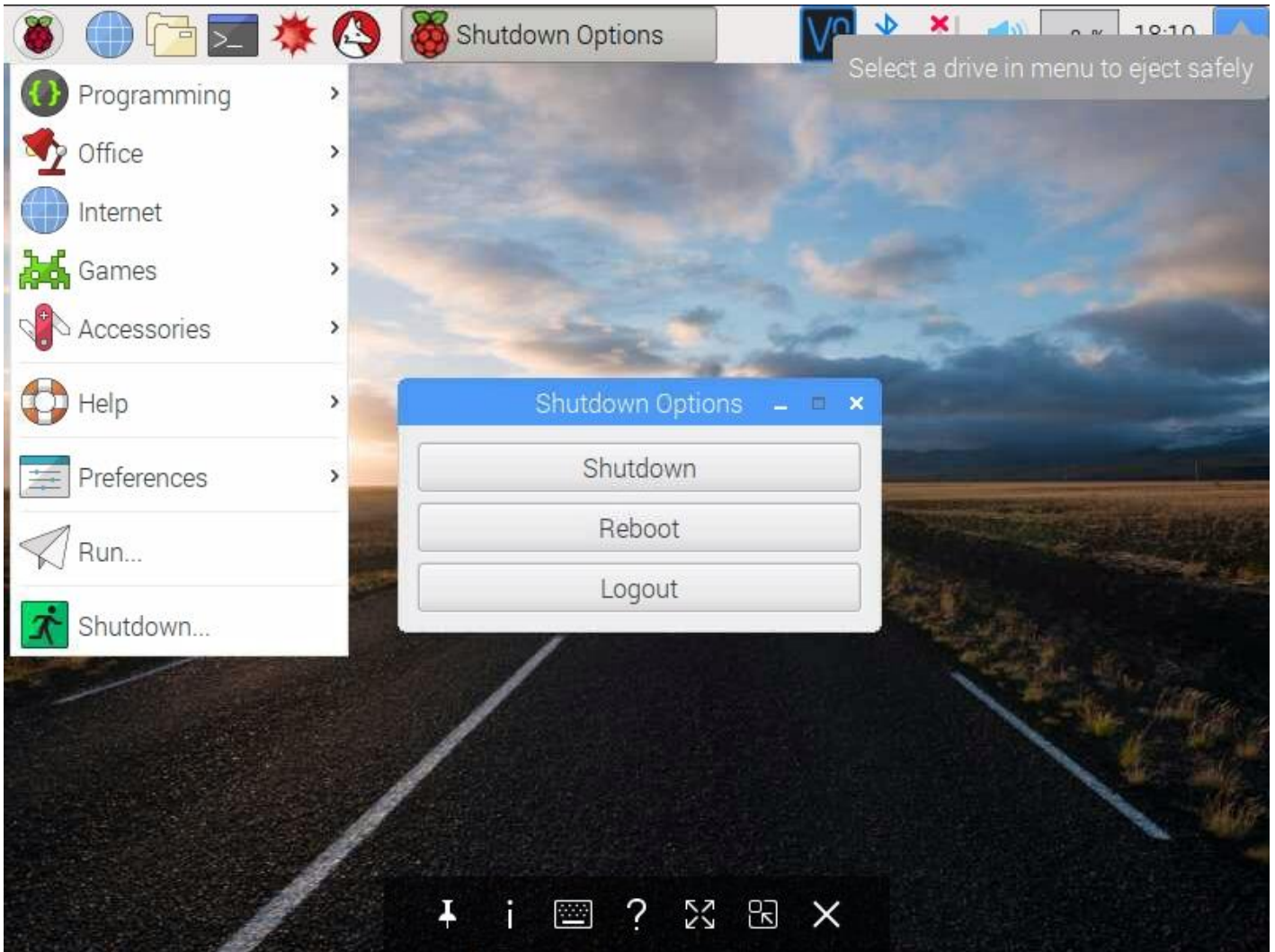
Muy fácil, teclea

```
sudo poweroff
```

## 7.2 Opción local de forma gráfica

Igual que en otros sistemas operativos Windows o Linux tenemos que apagar desde el menú **Shutdown**



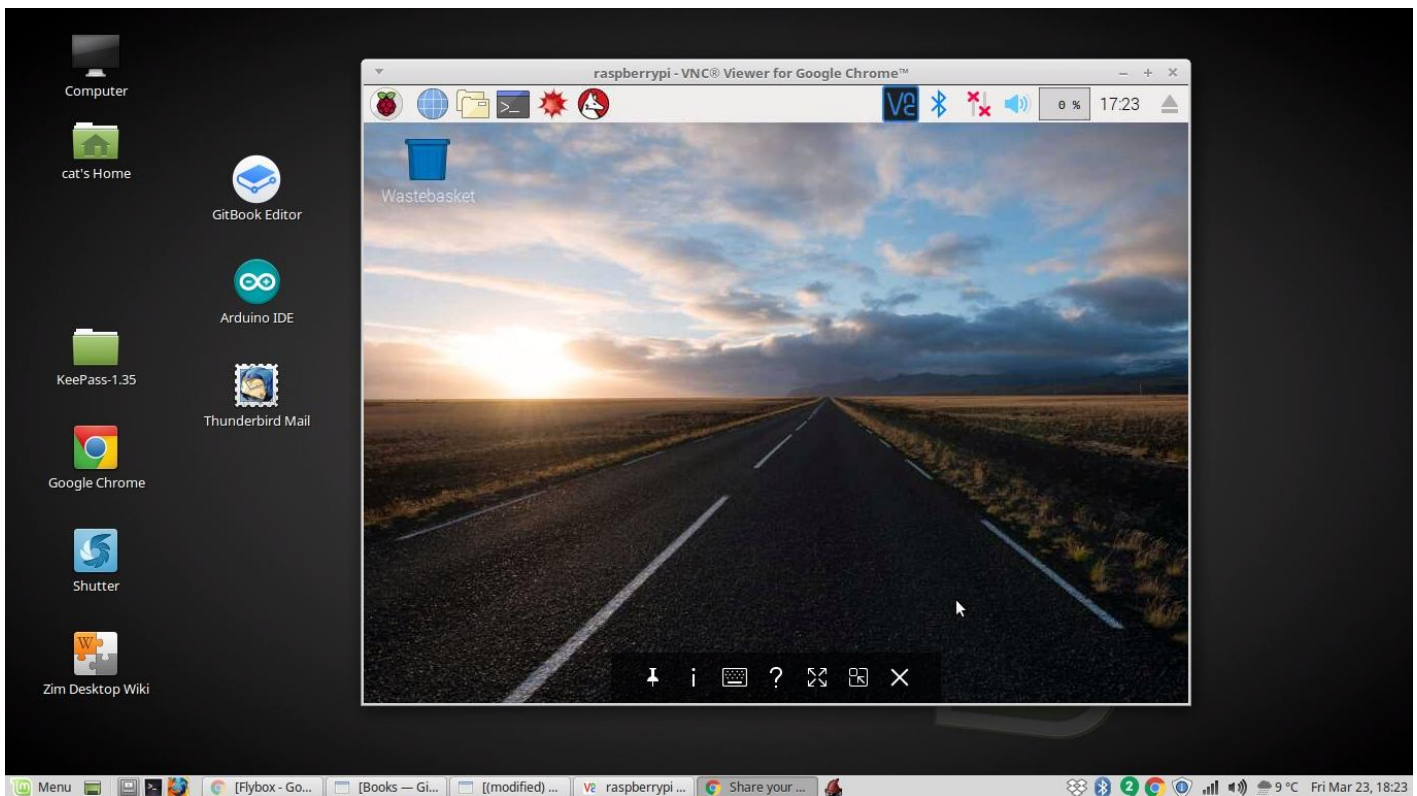


# Raspberry: VNC

VNC es un programa que nos permite la conexión remota con los servidores pero de forma gráfica. Es el sistema elegido por RASPBIAN pues ya lo tiene incorporado, pero no activado.

Hay que tener claro que son **dos programas**: \* **VNC SERVER** que es el programa que se ejecuta en la RASPBERRY y que aunque viene instalado, hay que activarlo. Es el programa que lanza el escritorio para que lo lea: \* **VNC VIEWER** es el programa que tenemos que tener en nuestro ordenador para ver lo que lanza VNC SERVER.

En la imagen podemos ver un pantallazo de mi ordenador, el escritorio negro es mi escritorio y el de la carretera es el de la Raspberry:



# Raspberry: Instalar Java 8

Una vez que tenemos la Raspberry funcionando y accesible por SSH o por VNC es el momento de instalar el servidor Blynk, primero hay que, para instalar Blynk, se necesita la **versión 8 de Java**

Seguimos las instrucciones de <https://github.com/Peterkn2001/blynk-server#blynk-server>

Entramos en la ventana de comandos SSH y ejecutamos las siguientes órdenes, si en algún momento nos pide confirmación [Y/n] es porque faltan descargar paquetes o confirmación de instalación, por lo tanto aceptamos.

Con esta orden busca las actualizaciones

**sudo update**

y ya ejecutamos que instala la versión 8 de java (curiosamente no funciona con la versión 64 bits de Raspberry OS sino con la de 32 bits)

**sudo apt install openjdk-8-jdk**

Por último una vez finalizado, comprobamos la versión que coge por defecto

**java -version**

Tiene que salir Openjdk version 1.8.0 etc...

## Si no sale esa versión...

Suele pasar que tenga ya instalado la versión 11, tenemos que obligar a que sea la 8 por defecto, para ello ejecutamos

**sudo update-alternatives --config java**

```

pi@raspberrypi:~ $ sudo update-alternatives --config java
Existen 2 opciones para la alternativa java (que provee /usr/bin/java).

  Selección    Ruta
-----
0             /usr/lib/jvm/java-11-openjdk-armhf/bin/java      1111      modo a
automático
1             /usr/lib/jvm/java-11-openjdk-armhf/bin/java      1111      modo m
anual
* 2           /usr/lib/jvm/java-8-openjdk-armhf/jre/bin/java    1081      modo m
anual

Pulse <Intro> para mantener el valor por omisión [*] o pulse un número de selección:

```

Si seleccionamos el 2 nos lo pondrá como por defecto (sale con un (\*). Podemos comprobar que ya nos sale la versión con la orden :

### java -version

Pero esto no nos sirve, pues al reiniciar la Raspberry volverá a la versión 11.

Para ponerlo por defecto, edita el fichero *environment*

### sudo nano /etc/environment

Y si en la orden **sudo update-alternatives --config java** salía que la carpeta donde está la versión 8 es */usr/lib/jvm/java-8-openjdk-armhf/jre/bin/java* luego tenemos que poner en el fichero *environment* :

**JAVA\_HOME="/usr/lib/jvm/java-8-openjdk-armhf/jre/bin/java"**

Guardar, reiniciar

### sudo reboot

Y comprueba

### java -version

Y tiene que salir :

```

pi@raspberrypi:~/Blynk $ java -version
openjdk version "1.8.0_212"
OpenJDK Runtime Environment (build 1.8.0_212-8u212-b01-1+rpil-b01)
OpenJDK Client VM (build 25.212-b01, mixed mode)

```

# Para más cosas

Por ejemplo, desinstalarlo, volver a la 11, etc.. visitar esta página :

<https://phoenixnap.com/kb/install-java-raspberry-pi>

# Raspberry: Instalar y ejecutar el servidor Blynk local

## 1 Descarga Blynk

Crea una carpeta en tu directorio home, por ejemplo Blynk

**mkdir Blynk**

Y descarga Blynk :

```
wget "https://github.com/Peterkn2001/blynk-server/releases/download/v0.41.16/server-0.41.16-java8.jar"
```

El enlace <https://github.com/Peterkn2001/blynk-server/releases/download/v0.41.16/server-0.41.16-java8.jar> conviene actualizarlo, recomendamos visitar la página

<https://github.com/Peterkn2001/blynk-server#blynk-server> y coger la última versión de aquí

## GETTING STARTED

### Blynk server

Blynk Server is an Open-Source [Netty](#) based Java server, responsible for forwarding messages between Blynk mobile application and various microcontroller boards and SBCs (i.e. Arduino, Raspberry Pi. etc).

Download latest server build [here](#).

release **v0.41.17** downloads **6.2k**

### Requirements

- Java 8/11 required (OpenJDK, Oracle)
- Any OS that can run java
- At least 30 MB of RAM (could be less with tuning)
- Open ports 9443 (for app and hardware with ssl), 8080 (for hardware without ssl)

## 2 Configurar server.properties

Necesitamos crear un fichero de configuración para las diferentes opciones que queremos en nuestro servidor Blynk.

Entra en la carpeta creada Blynk y crea el fichero *server.properties*.

**cd Blynk**

**sudo nano server.properties**

A continuación se muestra un posible contenido de server.properties.

```
initial.energy=1000000
allow.reading.widget.without.active.app=false
user.message.quota.limit=100
logs.folder=./logs
user.dashboard.max.limit=100
lcd.strings.pool.size=6
server.ssl.key=./server_embedded.key
webhooks.response.size.limit=96
hardware.mqtt.port=8440
table.rows.pool.size=100
terminal.strings.pool.size=25
admin.email=admin@blynk.cc
admin.rootPath=/admin
user.widget.max.size.limit=20
listen.address=
blocking.processor.thread.pool.limit=6
stats.print.worker.period=60000
enable.db=false
force.port.80.for.csv=false
enable.raw.db.data.store=true
restore.host=blynk-cloud.com
csv.export.data.points.max=43200
restore=false
user.profile.max.size=256
allow.store.ip=true
allowed.administrator.ips=0.0.0.0/0,::/0
net.interface=eth
webhooks.frequency.user.quota.limit=1000
http.port=8080
```

```
web.request.max.size=524288
user.devices.limit=50
async.logger.ring.buffer.size=2048
user.tags.limit=100
server.ssl.key.pass=
admin.pass=admin
hard.socket.idle.timeout=10
product.name=Blynk
data.folder=/Path
map.strings.pool.size=25
profile.save.worker.period=60000
https.port=9443
log.level=info
server.ssl.cert=./server_embedded.crt
force.port.80.for.redirect=true
notifications.queue.limit=2000
notifications.frequency.user.quota.limit=5
server.host=192.168.137.1
app.ssl.port=8443
hardware.default.port=8442
hardware.ssl.port=8441
hardware.mqtt.port=8440
```

Para la explicación de cada línea, aconsejo consultar esta [página](#) en el apartado *Configuración avanzada del servidor local*

## 3 Ejecutar el servidor Blynk local

En la Raspberry por comandos SSH, entramos en la carpeta donde hemos creado el servidor Blynk

### **cd Blynk**

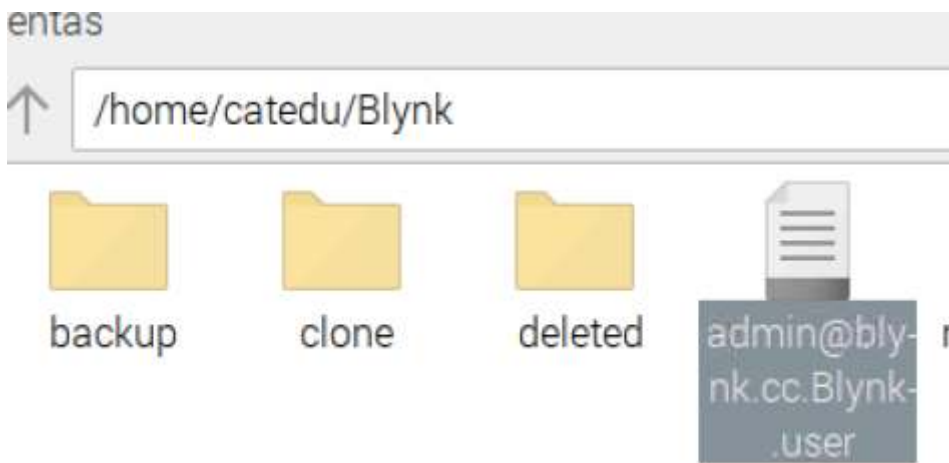
Y ejecutamos el servidor Blynk instalado, pero que cargue la configuración de server.properties que en nuestro caso como el la Raspberry el usuario se llama catedu la carpeta es catedu:

**java -jar server-0.41.16-java8.jar -dataFolder /home/catedu/Blynk -serverConfig /home/catedu/Blynk/server.properties**

Curiosamente la primera vez que ejecutas esta instrucción te sale los datos del usuario

**admin@blynk.cc y su contraseña sin encriptar** que por defecto es admin. Si no has tomado nota, ejecutar la instrucción anterior no sirve de nada pues ya ha creado el fichero texto de este usuario.





La única forma de que te vuelva a mostrar la contraseña es borrar el fichero texto y ejecutar otra vez la orden `java -jar server....`

## 4 Que la orden de ejecución se haga automáticamente cada vez que se reinicie la Raspberry

Para no repetir estos dos comandos `cd Blynk` y `java -jar server-0.41.16-java8.jar -dataFolder /home/catedu/Blynk -serverConfig /home/catedu/Blynk/server.properties` cada vez que reiniciamos la Raspberry puedes generar un script para que lo ejecute automáticamente, puedes ver buenos tutoriales en Internet.

En internet puedes ver varios métodos:

**Método1** es entrar en `/etc` y editar el fichero `rc.local` y añadir esta línea

```
java -jar server-0.41.16-java8.jar -dataFolder /home/catedu/Blynk -serverConfig /home/catedu/Blynk/server.properties &
```

**Método 2** usar el comando `crontab -e` y poner al final la siguiente línea `@reboot java java -jar server-0.41.16-java8.jar -dataFolder /home/catedu/Blynk -serverConfig /home/catedu/Blynk/server.properties`

## 5 Probarlo

Si la IP de la Raspberry es 192.168.1.112 entonces entramos en:

<https://192.168.1.112:9443/admin>

**Vale, ya estoy ¿y ahora qué?**

Seguramente te saldrá la siguiente advertencia por el certificado SSL, dale a **Configuración avanzada** y luego a **Acceder a (la IP del servidor Blynk Legacy) sitio no seguro**



## La conexión no es privada

Es posible que los atacantes estén intentando robar tu información de **192.168.43.111** (por ejemplo, contraseñas, mensajes o tarjetas de crédito). [Más información](#)

NET::ERR\_CERT\_AUTHORITY\_INVALID



Para disfrutar del máximo nivel de seguridad en Chrome, [activa la protección mejorada](#).

Ocultar configuración avanzada

1

Volver para estar a salvo

Este servidor no ha podido probar que su dominio es **192.168.43.111**, el sistema operativo de tu ordenador no confía en su certificado de seguridad. Este problema puede deberse a una configuración incorrecta o a que un atacante haya interceptado la conexión.

[Acceder a 192.168.43.111 \(sitio no seguro\)](#)

2

Si quieres generar certificados SSL propios para que no salga la anterior pantalla consulta [aquí](#)

Tienes que entrar con el usuario y contraseña fijada en **server.properties** :

# Default admin name and password. Will be created on initial server start

admin.email=admin@blynk.cc

admin.pass=admin

Si quieres cambiar la contraseña, tienes que hacerlo como un usuario normal en la página de administración tal y como hemos visto en el capítulo [Entrando en el Blynk local: El panel de control](#)

## 6 Para saber más :

- <https://github.com/Peterkn2001/blynk-server#blynk-server>
- [Intalación de Blynk : How to Install a Blynk Local Server on Raspberry Pi](#)
- [Configuración de server.properties.](#)

## 7 Para saber más : Configurar mail.properties

Este aparatado ya comentamos que no lo aconsejamos, pues los alumnos no suelen tener email y la APP ya no permite crear usuarios con email, pero si queremos que envíe los tokens por email, hay que crear este fichero para que el servidor envíe por email los tokens de los proyectos

Entra en la carpeta creada Blynk y crea el fichero *mail.properties*.

**cd Blynk**

**sudo nano mail.properties**

A continuación se muestra una muestra del posible contenido de mail.properties :

```
mail.smtp.auth=true
mail.smtp.starttls.enable=true
mail.smtp.host=smtp.gmail.com
mail.smtp.port=587
mail.smtp.username=Your EMAIL ID
mail.smtp.password=Password
```

Utilizando **Your EMAIL ID** y **Password** los datos de una cuenta de gmail tuya. En esa cuenta tienes que permitir accesos no seguros. Aquí se muestra dónde está en la pantalla de configuración de Gmail :

Inicio1

Información personal

Datos y personalización

Seguridad2

Contactos e información compartida

Pagos y suscripciones

Información general

Gestionar dispositivos

Gestionar i

Acceso de aplicaciones poco seguras3

Tu cuenta es vulnerable porque permites el acceso de aplicaciones y dispositivos que utilizan una tecnología de inicio de sesión poco segura. Para mantener tu cuenta protegida, Google desactivará automáticamente este ajuste si no se utiliza.

Activado4

Desactivar acceso (opción recomendada)

Para saber más

- [Configuración mail](#)

# En Windows Instalar Java 11

Entramos en la web de Oracle - Java - Technologies - JavaSE y descargamos e instalamos el servidor Java 11, si nuestro ordenador no es viejo, seguro que soportará la versión 64

<https://www.oracle.com/es/java/technologies/javase/jdk11-archive-downloads.html>

ORACLE

Q

Ver cuentas

Comunicarse con

Productos

Sectores








Recursos

Clientes

Partners

Desarrolladores

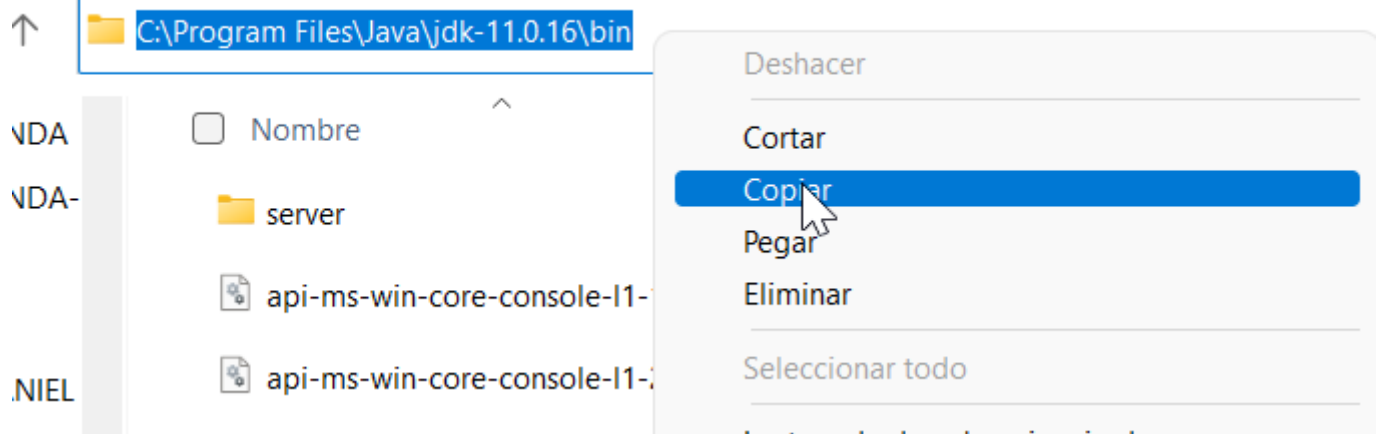
Eventos

|                                  |           |  |
|----------------------------------|-----------|--|
| macOS Arm 64 Compressed Archive  | 153.35 MB |  <a href="#">jdk-11.0.16_macos-aarch64_bin.tar.gz</a>   |
| macOS Arm 64 DMG Installer       | 152.83 MB |  <a href="#">jdk-11.0.16_macos-aarch64_bin.dmg</a>      |
| macOS x64 Compressed Archive     | 155.47 MB |  <a href="#">jdk-11.0.16_macos-x64_bin.tar.gz</a>       |
| macOS x64 DMG Installer          | 154.95 MB |  <a href="#">jdk-11.0.16_macos-x64_bin.dmg</a>          |
| Solaris SPARC Compressed Archive | 184.75 MB |  <a href="#">jdk-11.0.16_solaris-sparcv9_bin.tar.gz</a> |
| Windows x64 Installer            | 140.55 MB |  <a href="#">jdk-11.0.16_windows-x64_bin.exe</a>      |
| Windows x64 Compressed Archive   | 158.30 MB |  <a href="#">jdk-11.0.16_windows-x64_bin.zip</a>      |

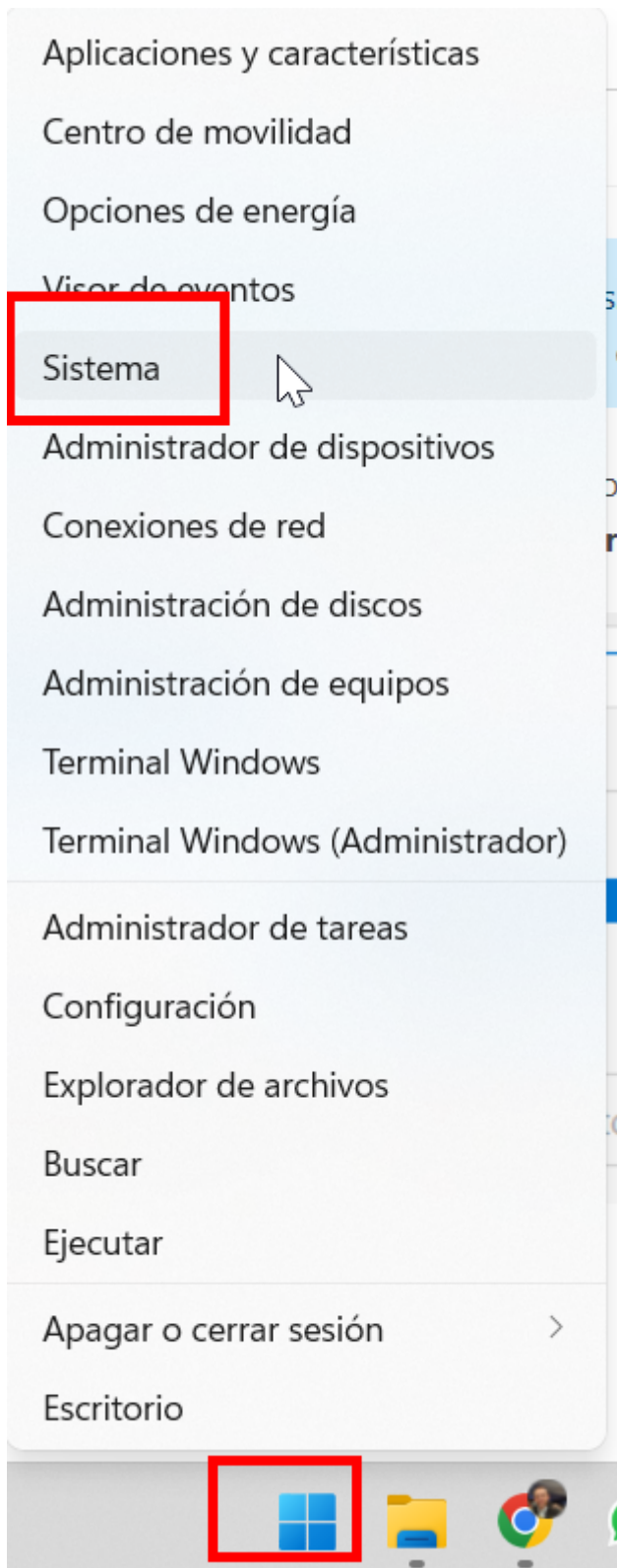
Seguramente para la descarga Oracle nos pide que nos registremos. No entendemos el por qué pero efectivamente es necesario proporcionar un email y a ese email envía un enlace para el registro, Una vez registrado ya permite la descarga

Instalamos el servidor Java descargado.

Para que Windows encuentre la ruta del Java instalado cuando ejecutemos comandos, le tenemos que decir en qué camino está la carpeta del java instalado está. Para ello vamos a **C -> Archivos de programa -> Java -> Jdk11...-> carpeta bin** y copiamos la ruta :



Nos vamos al sistema



Y configuración avanzada del sistema - variables de entorno- path - editar - nueva y pegamos la ruta que hemos copiado anteriormente

[illegible]



# En Windows Instalar Blynk local


Descargamos el servidor Blynk server desde


<https://github.com/blynkkk/blynk-server/releases>



o desde <https://drive.google.com/file/d/1nPZxn-9Q9q0ceUhCiZGd4WRC1c8pufEQ/view?usp=sharing>

y lo pegamos en una carpeta, por ejemplo en c:\Blynk\_server

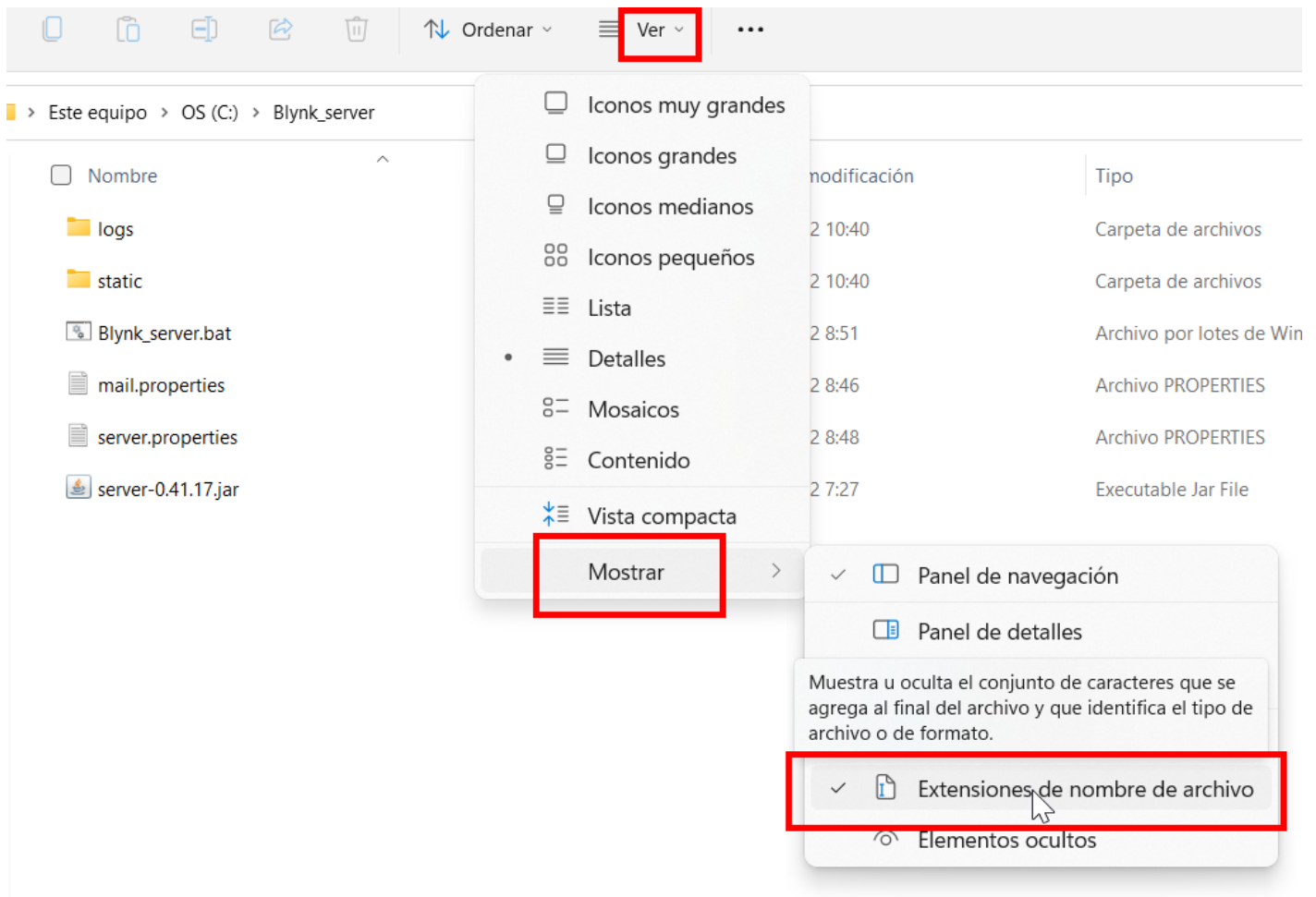
Creamos **mail.properties** y **server.properties** como ficheros nuevos, los editamos con un procesador de texto y pegamos el código que hemos explicado [aquí](#)

 mail.properties

 server.properties

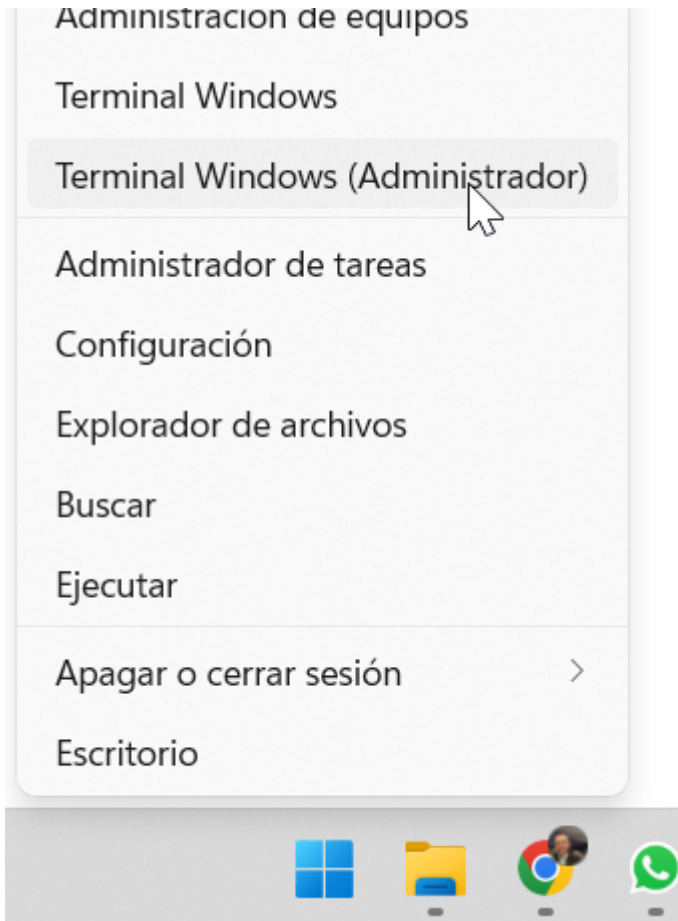
  server-0.41.17.jar

Ojo: Ten en cuenta que Windows le gusta colocar sus propias extensiones. Si editas un fichero con un procesador de textos, por ejemplo el bloc de notas, le añadirá .txt, si lo editas con Word le añadirá la extensión .docx, tienes que asegurarte que no se añada ninguna extensión, que los nombres de los ficheros sean **mail.properties** y **server.properties** y no otros. Ten activo la visualización de las extensiones para asegurarte que windows no añada las suyas



# En Windows Ejecutar Blynkserver

Entramos en comandos de windows como administrador



Y ejecutamos esta instrucción

**java -jar server-0.41.17.jar -dataFolder /Path**

de esta manera se crean las carpetas logs y static que nos dicen los errores que podemos encontrar

La primera pantalla nos dirá el nombre la url, el nombre de usuario y la contraseña para entrar en el servidor blynk

Windows PowerShell

Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Instale la versión más reciente de PowerShell para obtener nuevas características y mejoras.

PS C:\Users\ASUSI5> cd c:\Blynk\_server

PS C:\Blynk\_server> java -jar server-0.41.17.jar -dataFolder /Path

Blynk Server 0.41.18-SNAPSHOT successfully started.

All server output is stored in folder 'C:\Blynk\_server\.\logs' file.

Your Admin url is <https://192.168.137.1:9443/admin>

Your Admin login email is admin@blynk.cc

Your Admin password is admin

# En Linux

Seguimos las instrucciones de <https://github.com/Peterkn2001/blynk-server#blynk-server>

Pero si tu centro tiene Vitalinux, ponte en contacto con nosotros [soportecatedu@educa.aragon.es](mailto:soportecatedu@educa.aragon.es) para instalar en un ordenador Vitalinux el servidor Blynk de forma local