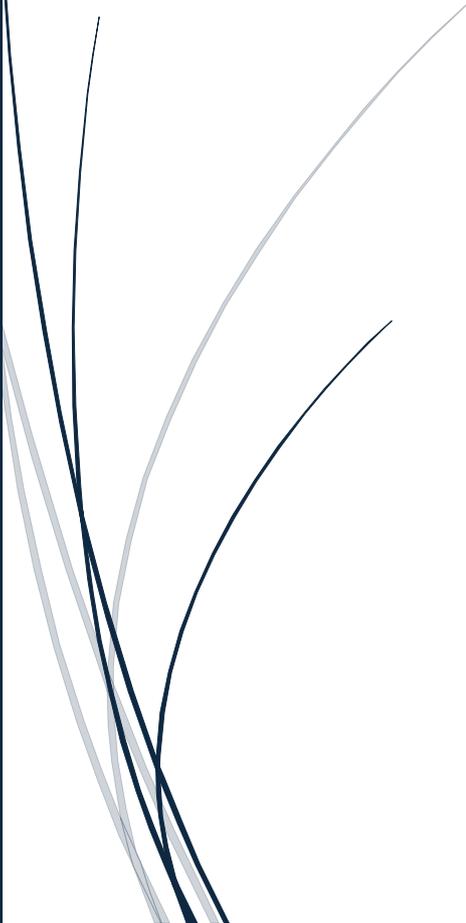




4/11/2024

# L1 Raspberry Pi



Pit Dieschbourg

## Inhaltsverzeichnis

Betriebssystem installieren.....	2
Hostnamen und weitere Einstellungen.....	7
SSH aktivieren .....	8
Statische IP-Adresse konfigurieren .....	10
SFTP Verbindung mit FileZilla .....	16
SSH Verbindung mit PuTTY .....	17
Automatisches Starten der Webseite .....	19
Dateien dem Benutzer www-data zuweisen.....	22
Aktualisieren der Softwarepakete .....	22
Netzwerküberwachungstool installieren .....	23
Passwörter im Passwortmanager speichern .....	24

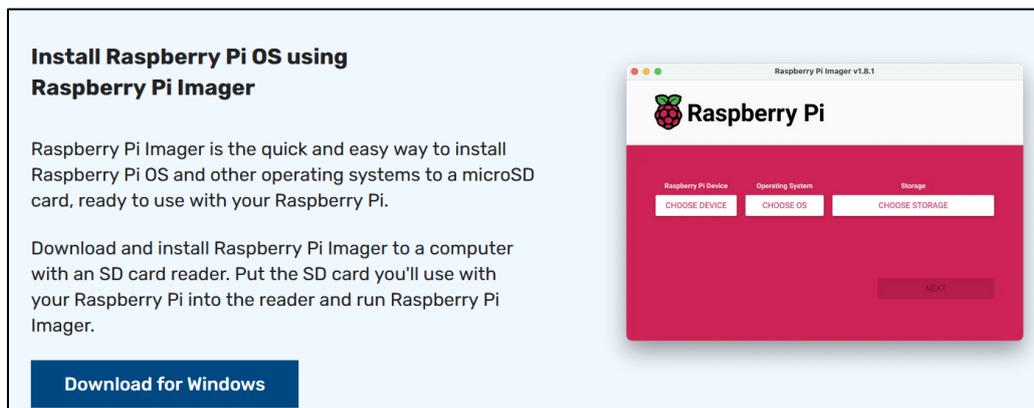
## Betriebssystem installieren

Als die Stunde angefangen hat haben wir von den Lehrern einige Sachen bekommen:

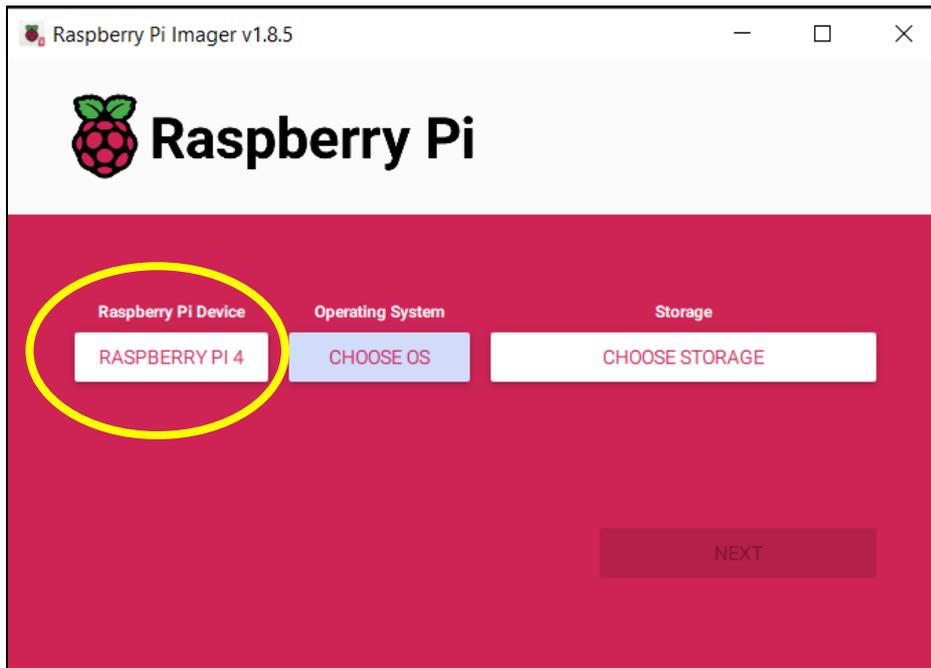
- Als erstes haben wir ein Raspberry Pi 4 Model B bekommen.
- Einen Touchdisplay den man auf den Raspberry Pi schrauben.
- Und ein Stromkabel damit der Raspberry Pi mit Strom versorgt wird.

Mit einem Link von der Userstory habe ich dann das Betriebssystem „FullpageOS“ heruntergeladen.

Dann habe ich auf der Webseite „raspberrypi.com“ den Imager heruntergeladen.



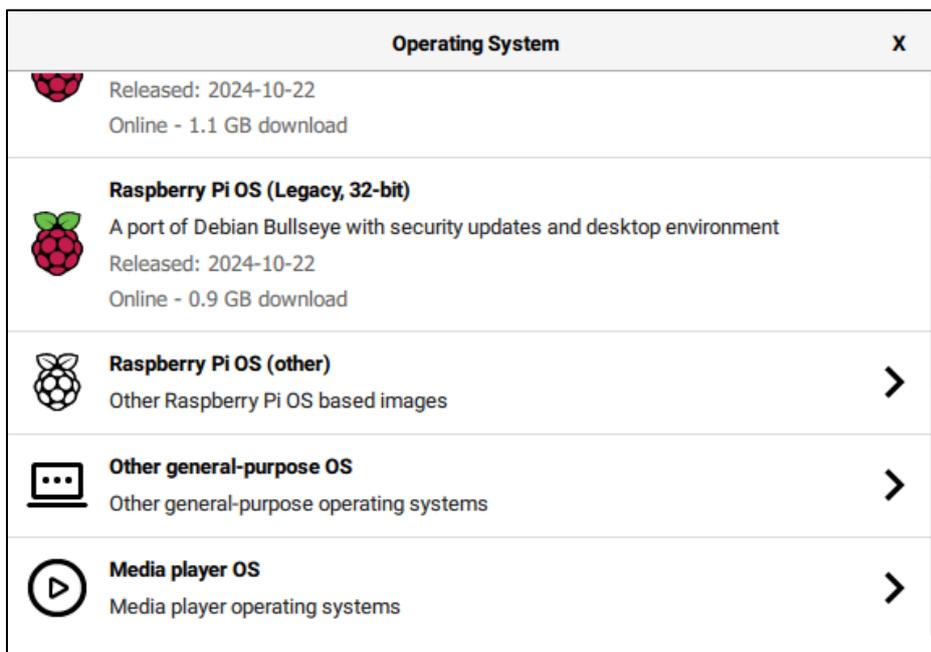
Anschließend wenn der Imager fertig heruntergeladen ist können sie ihn starten.



Bei der Auswahl „Raspberry Pi Device“ wählen sie „Raspberry Pi 4“ aus.

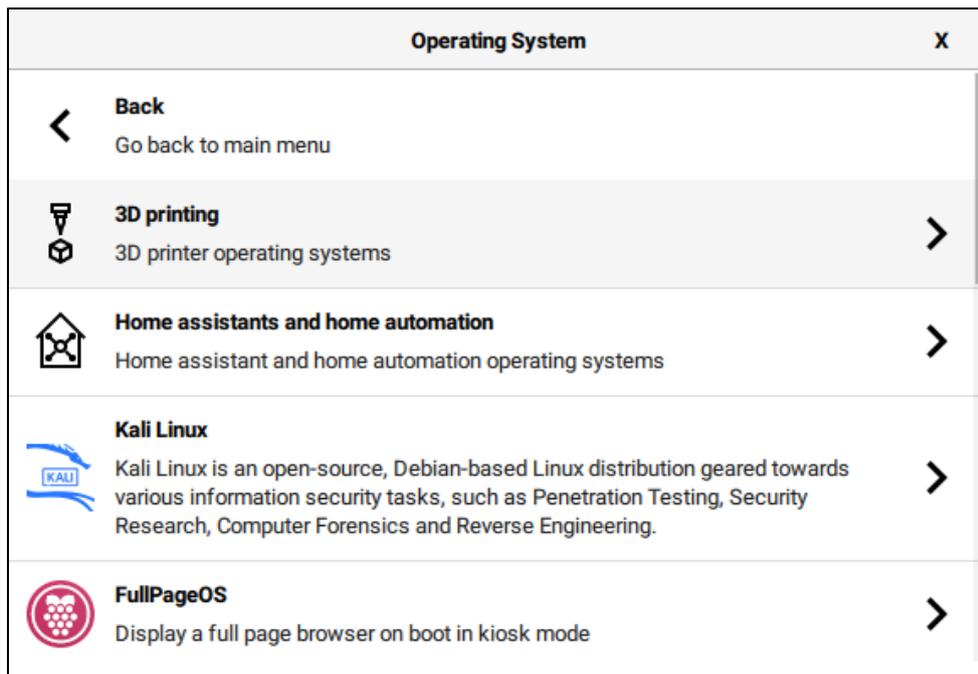


Bei „CHOOSE OS“ müssen sie das Betriebssystem auswählen.

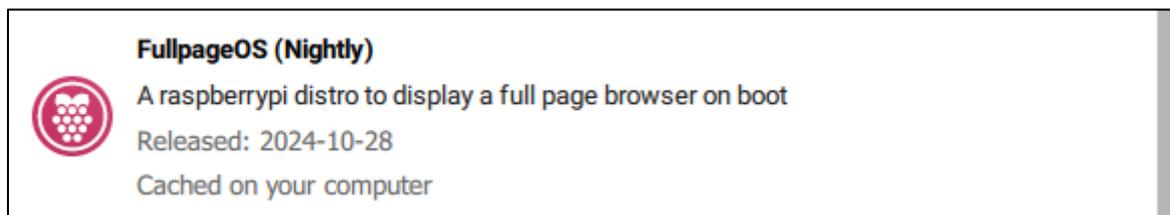


Dort scrollen sie ein bisschen runter bis sie „Other specific-purpose OS“ sehen.

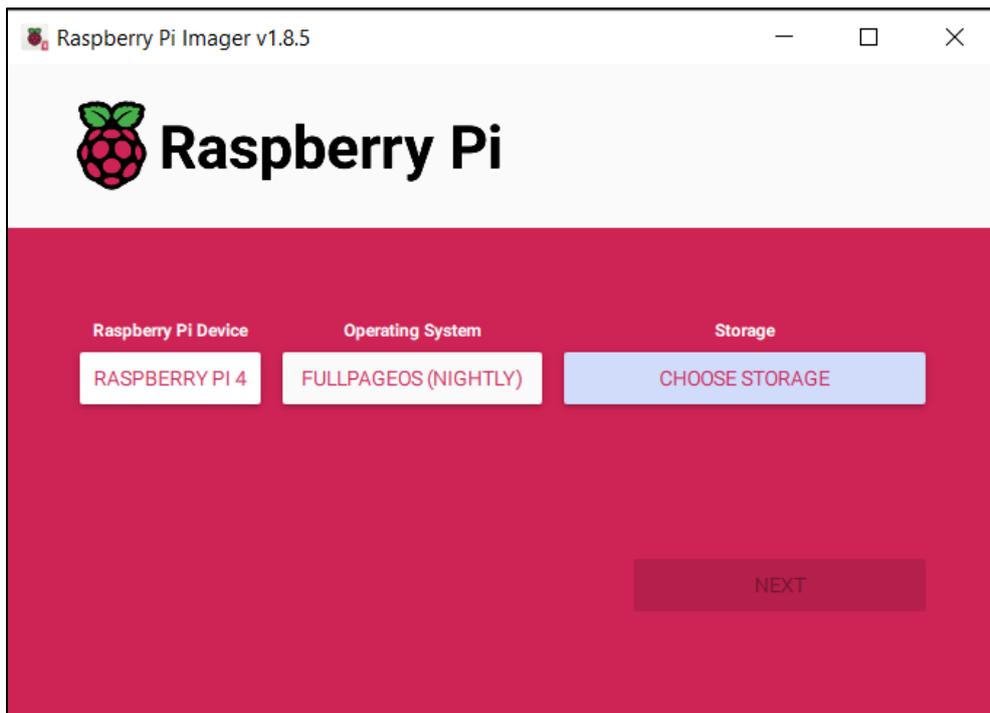
Und wählen dies aus.



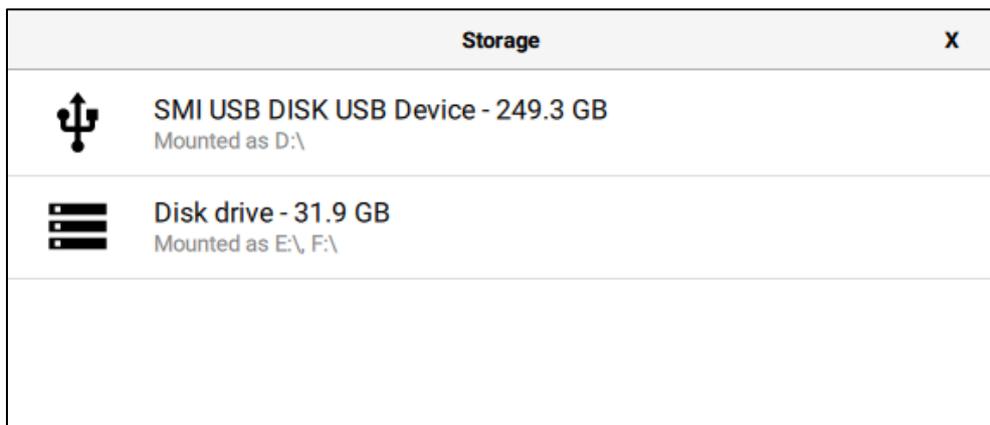
Wenn sie dann hier angelangt sind wählen sie ganz Oben das unterste aus „FullPageOS“.



Dort angekommen wählen sie dann „FullpageOS(Nightly)“ aus.

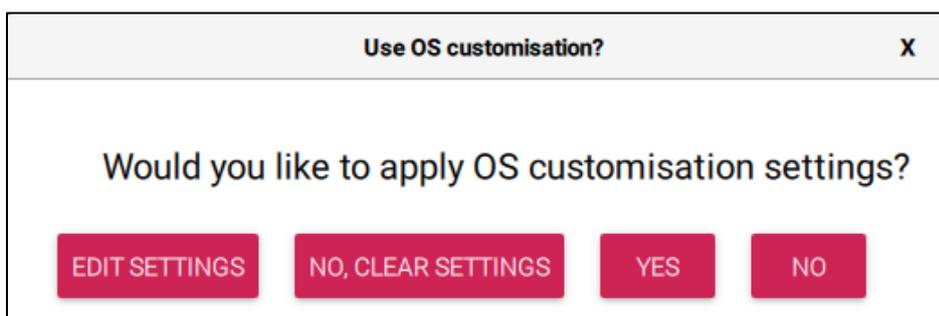


Als nächstes gehen sie dann zu „Storage“.



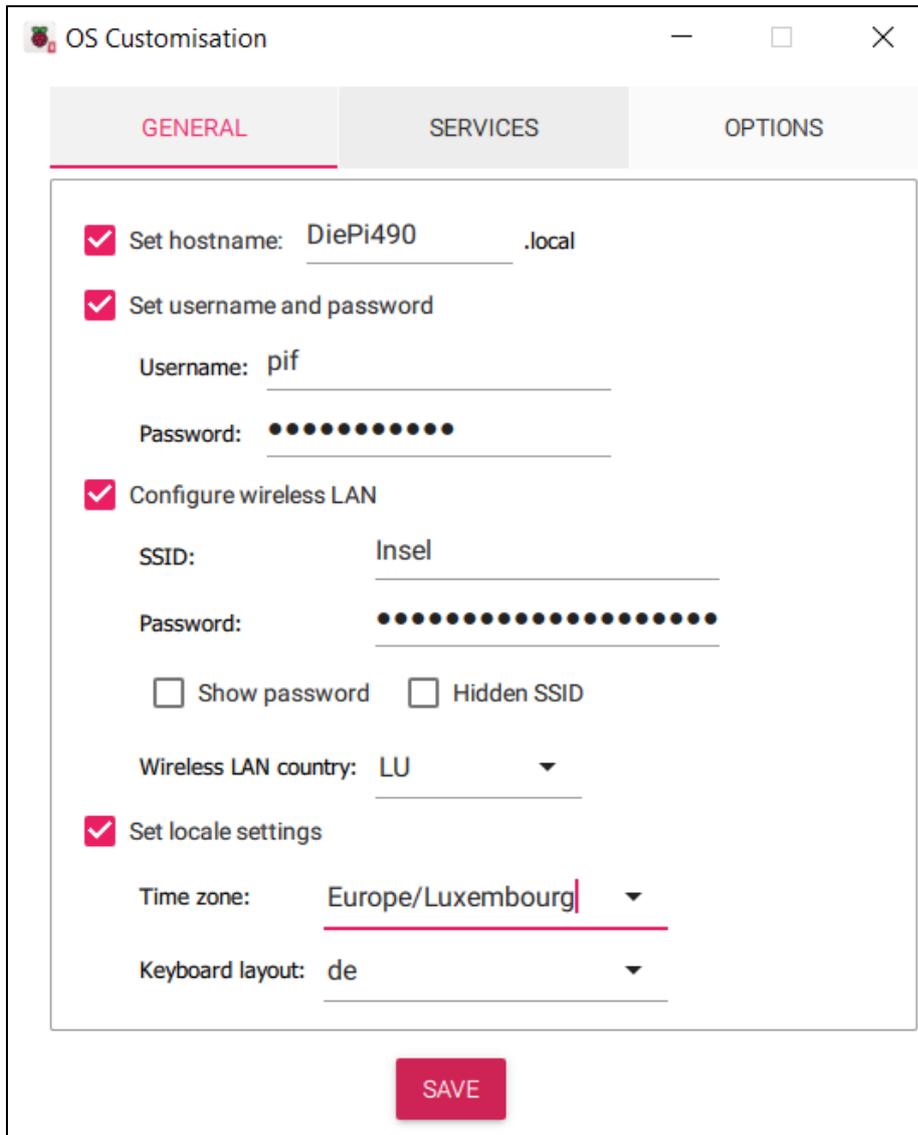
Hier wählen sie dann ihren Disc Drive aus.

Wenn sie das erledigt haben klicken sie unten rechts auf „Next“.



Hier wählen sie dann „Edit Settings“.

## Hostnamen und weitere Einstellungen



The screenshot shows the 'OS Customisation' window with the 'GENERAL' tab selected. The settings are as follows:

- Set hostname: DiePi490 .local
- Set username and password
  - Username: pif
  - Password: [masked]
- Configure wireless LAN
  - SSID: Insel
  - Password: [masked]
  - Show password  Hidden SSID
  - Wireless LAN country: LU
- Set locale settings
  - Time zone: Europe/Luxembourg
  - Keyboard layout: de

A red 'SAVE' button is located at the bottom center of the window.

Hier angekommen drücken sie auf das Kästchen bei „Set hostname“ und geben sie ihren IAM Namen ein.

Der Username lautet „pif“ und das Passwort frei wähl bar.

Als WLAN habe ich das WLAN „Insel“ genommen.

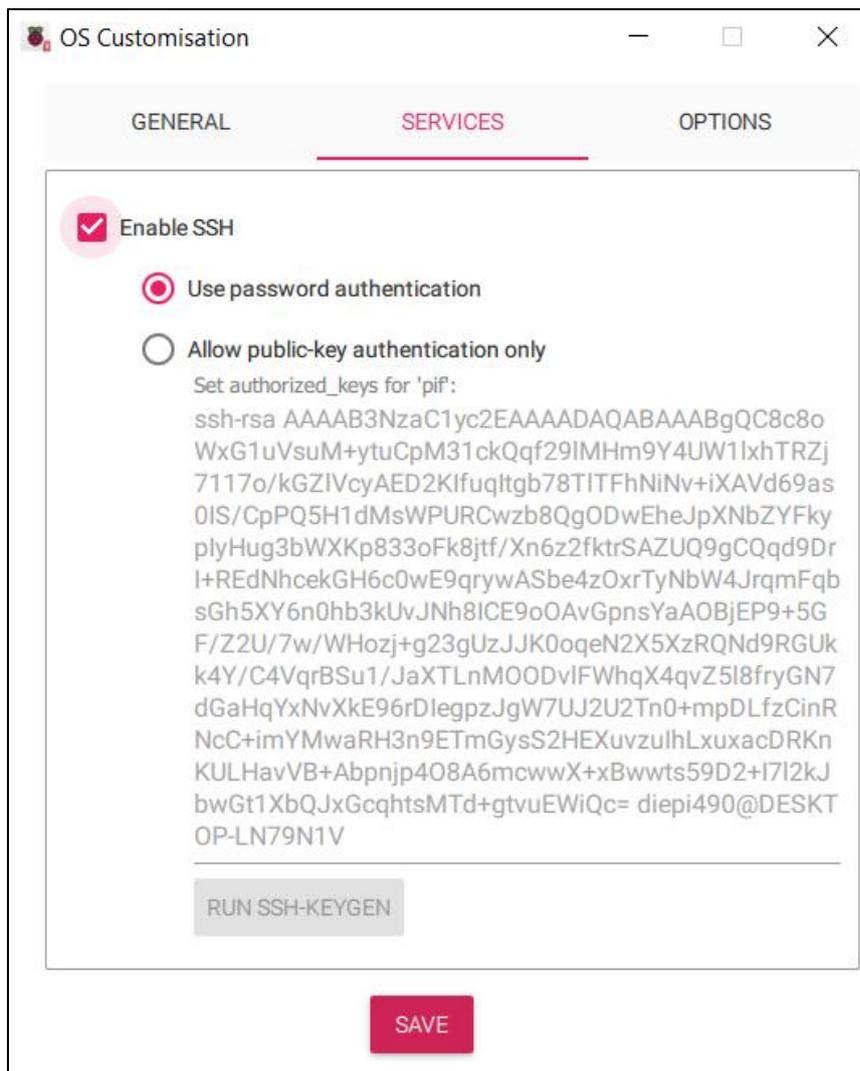
Bei „Wireless LAN Country“ habe ich „LU“ genommen.

Dann habe ich unten noch bei „Set locale settings“ das Kästchen bestätigt.

Bei der Zeit Zone habe ich „Europe/Luxembourg“ ausgewählt.

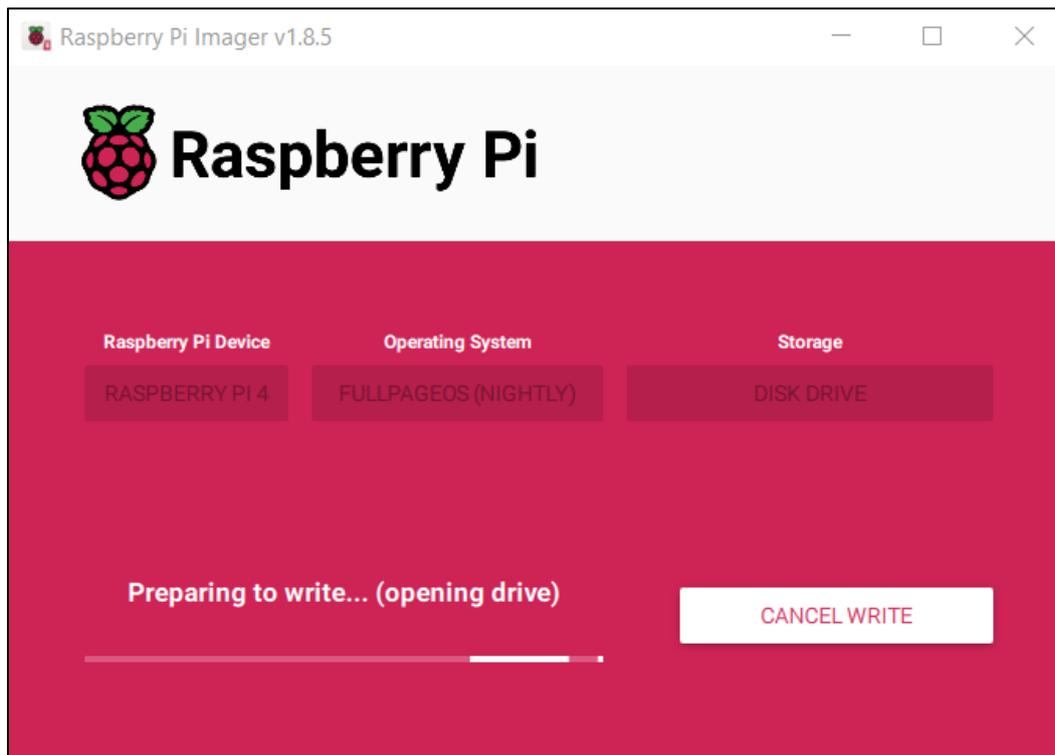
Und mein gewünschtes Keyboard Layout ausgewählt.

## SSH aktivieren



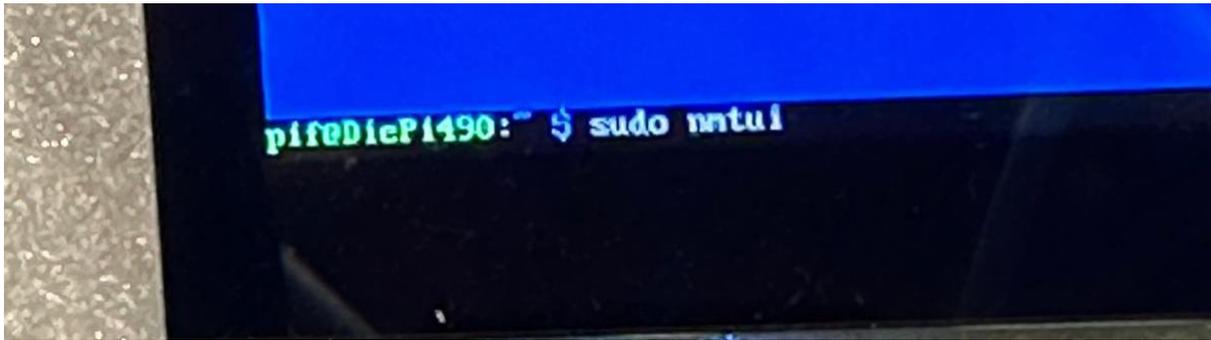
Auf der nächsten Seite bei „Services“ habe ich dann bei „Enable SSH“ das Kästchen ausgewählt damit SSH aktiviert ist und untendrunter „Use Password authentication“ ausgewählt.

Anschließend drücken sie unten auf „Save“.

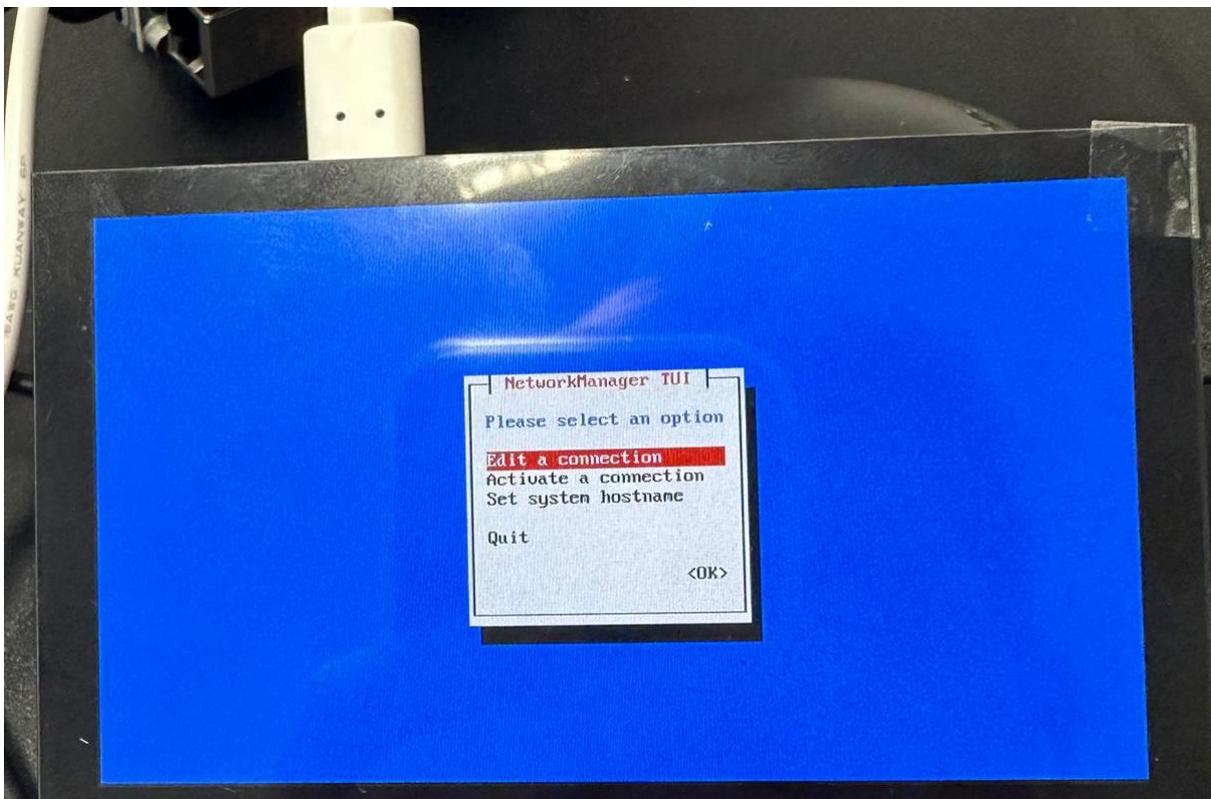


Anschließend wird es dann installiert.

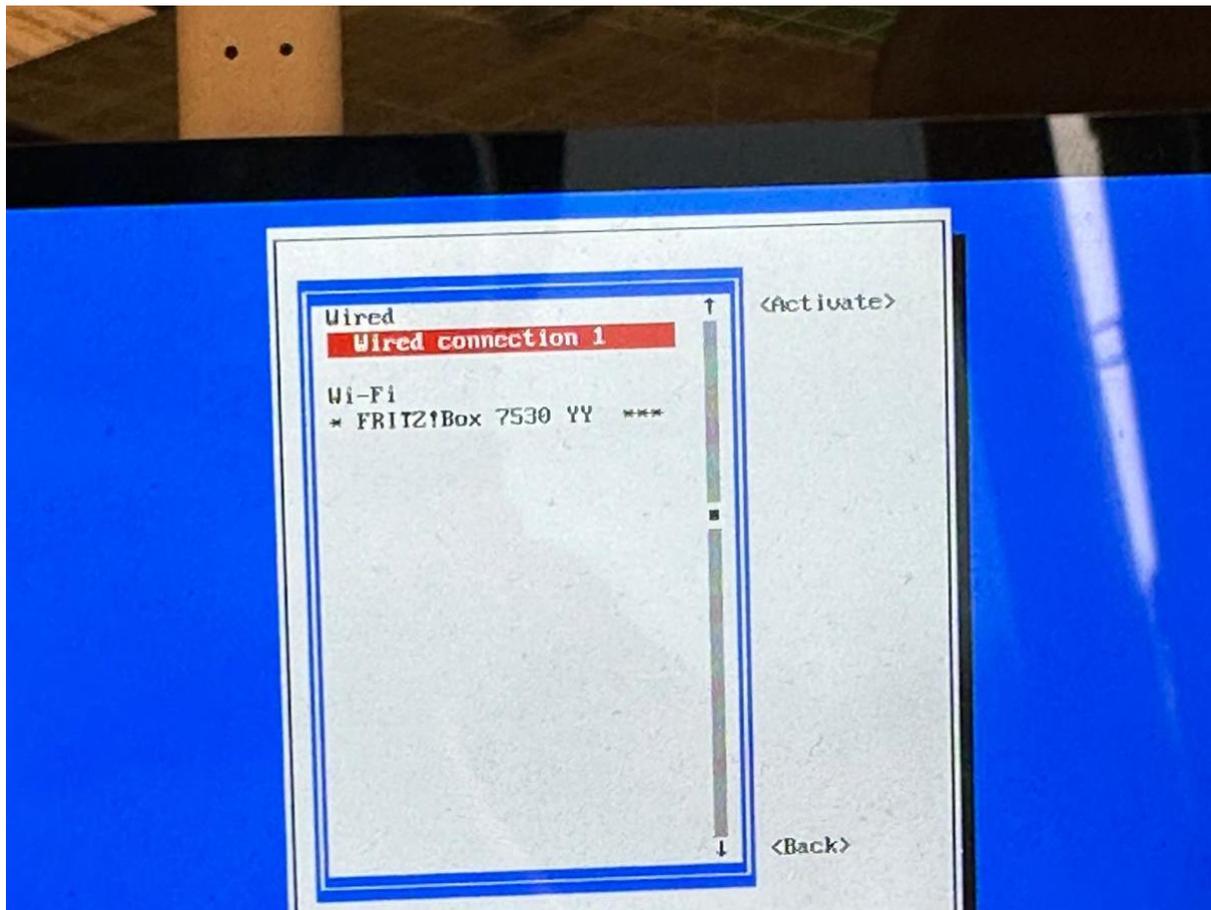
## Statische IP-Adresse konfigurieren



Um Anschließend dem Raspberry Pi eine statische zuzuweisen muss ich „sudo nmtui“ eingeben.

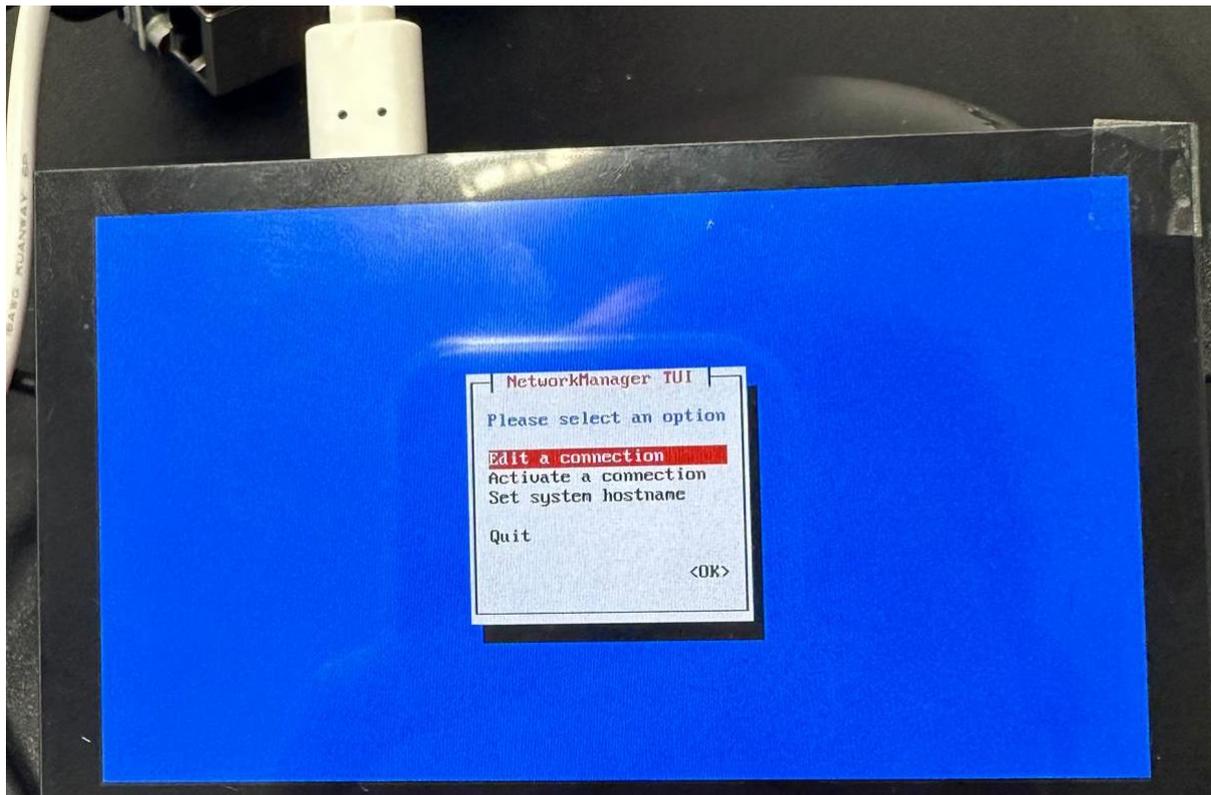


Dies führt dazu dass sich dies öffnet, hier muss ich zuerst ein neues Netzwerk hinzufügen, das tue ich indem ich die 2.Auswahl Möglichkeit „Activate a connection“ nehme.

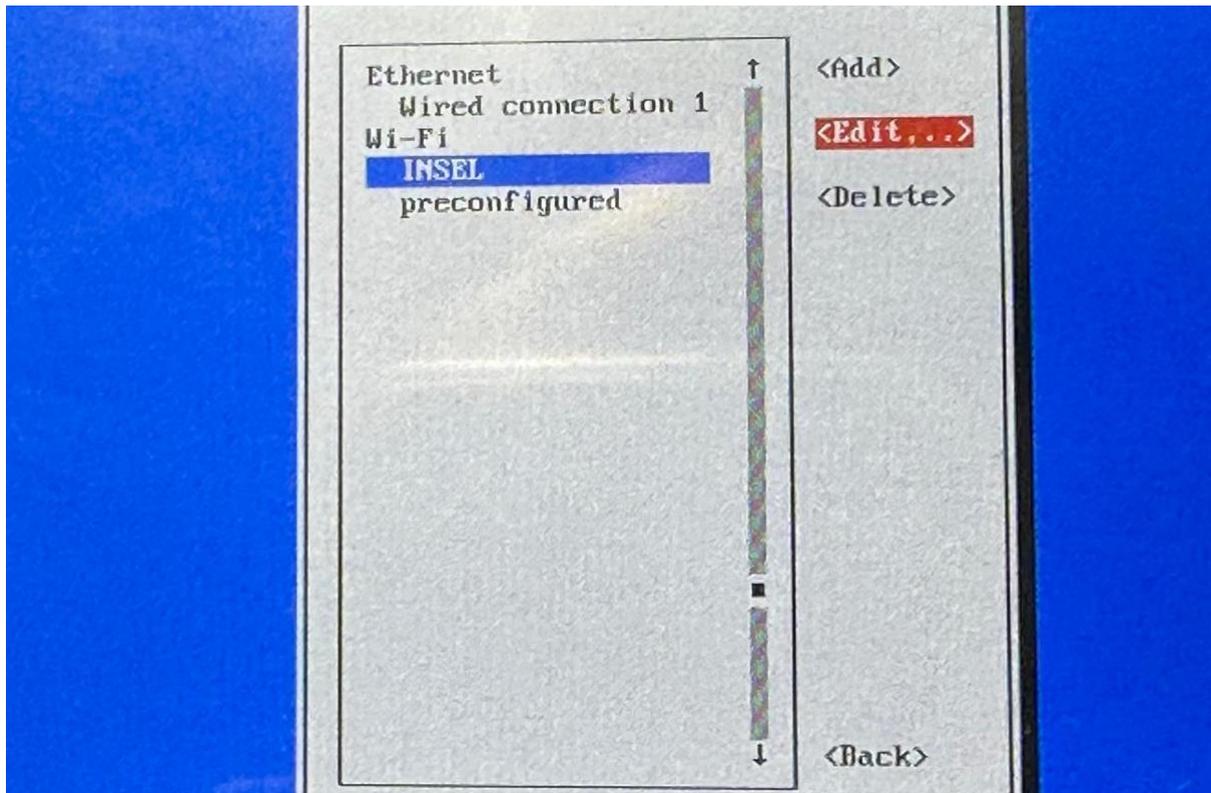


Hier füge ich nun ein Netzwerk hinzu.

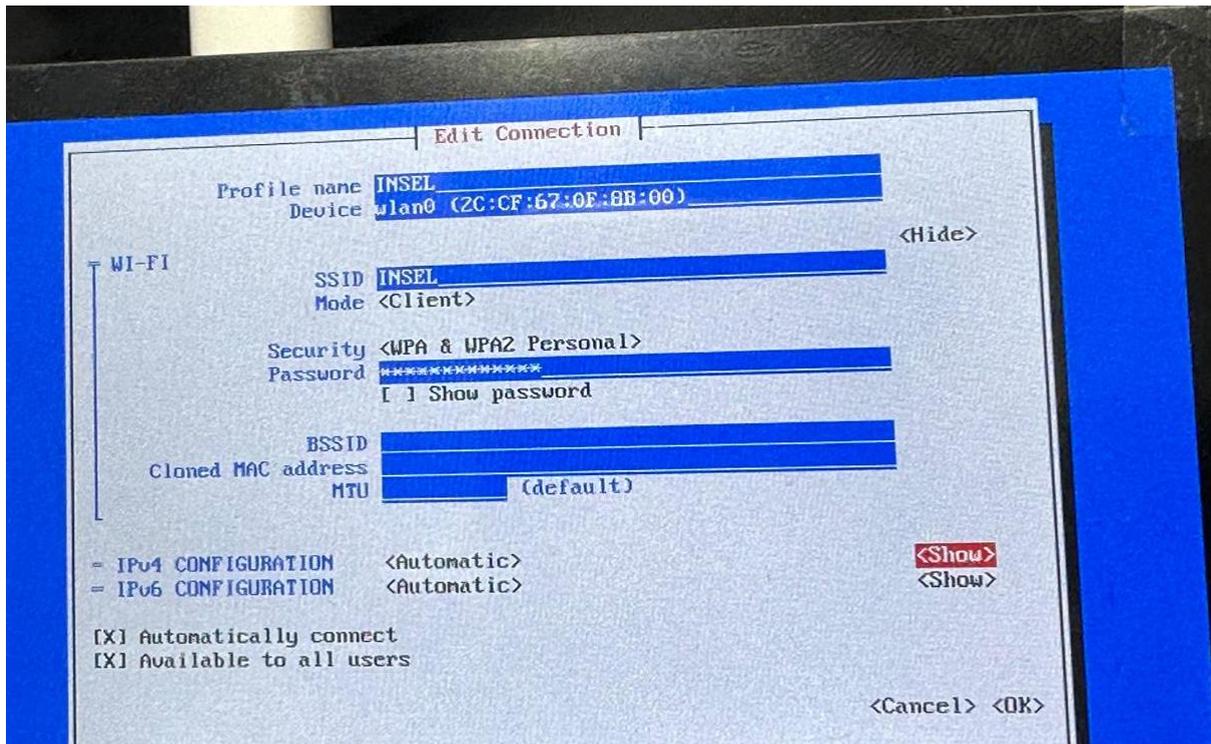
Das tue ich indem das Netzwerk auswähle indem ich mich befinde, anschließend mit der Auswahl werde ich nach dem Passwortes des Netzwerkes gefragt. Wenn ich das eingegeben habe kann ich unten rechts zu „Back“ gehen und „Enter“ drücken.



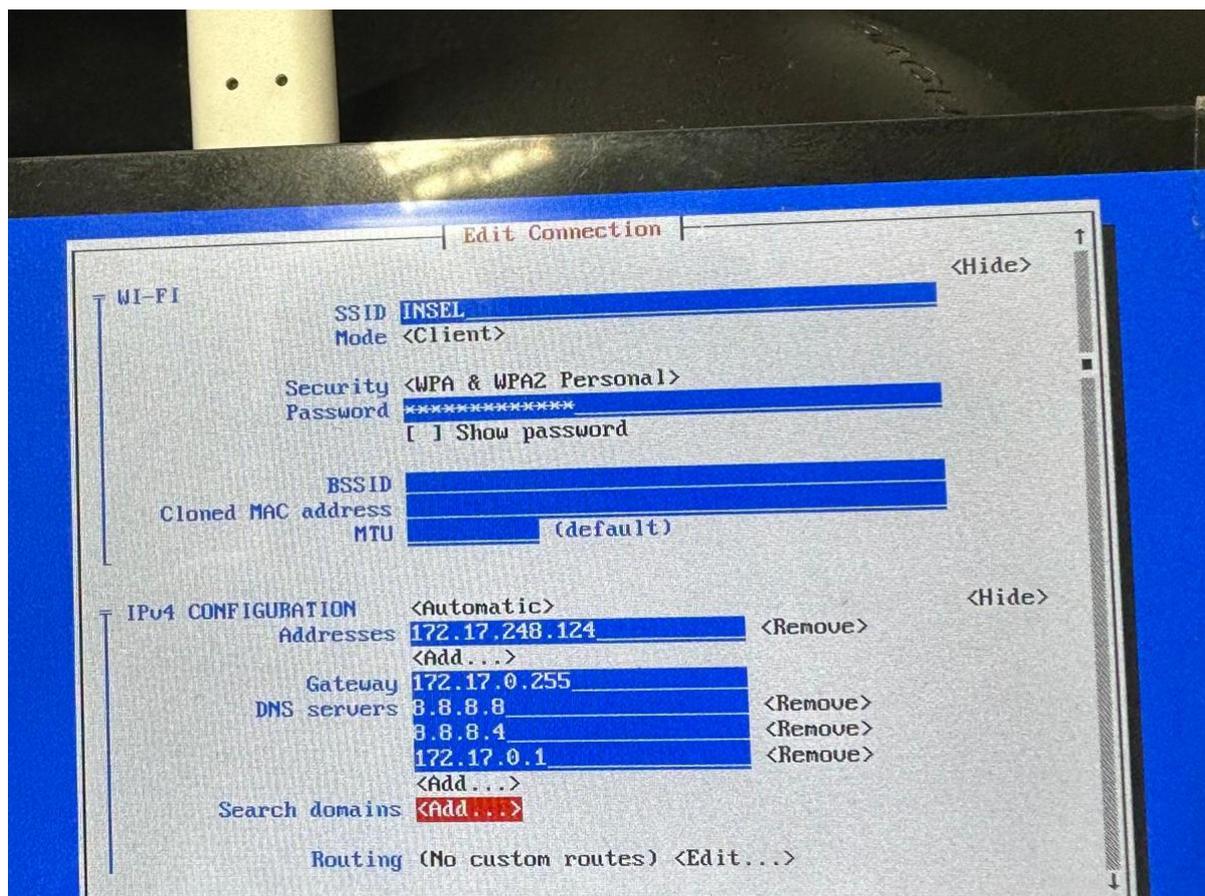
Nun sind wir wieder hier angelangt, dieses mal wähle ich die 1. Auswahl Möglichkeit aus um eine Statische IP-Adresse zu konfigurieren.



Hier angelangt wählen wir nun unser Netzwerk aus was wir vorhin hinzugefügt hatten. Dafür gehen wir runter zu unserem Netzwerk und wählen auf der rechten Seite „Edit“ aus und drücken „Enter“.



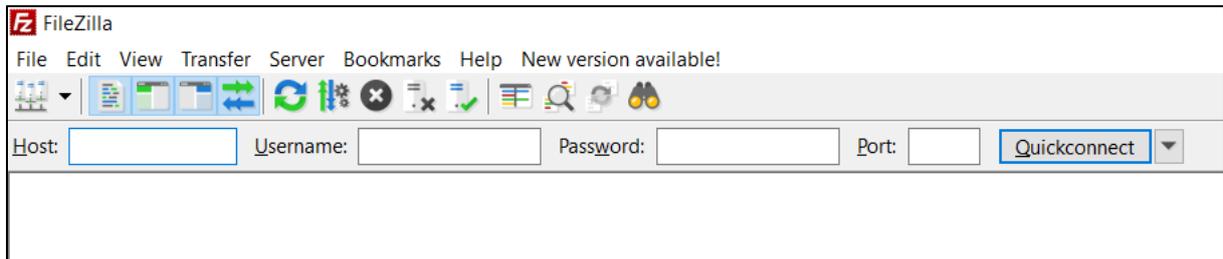
Hier gehen sie dann ein bisschen runter zu „IPv4 CONFUIGRATION“ und wählen rechts „Show“ aus.



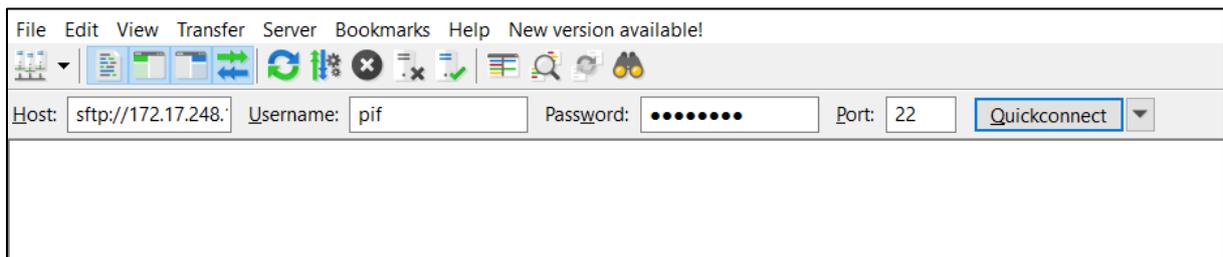
Dann gebe ich dort meine IP-Adresse, mein Gateway und meine 3 DNS Server ein die ich vom meinem Lehrer bekomme. Dann scrolle ich ganz runter und auf der rechten Seite wähle ich „OK“ aus und drücke „Enter“.

Nach diesen Einstellungen kann ich wieder zum Terminal zurückkehren.

## SFTP Verbindung mit FileZilla

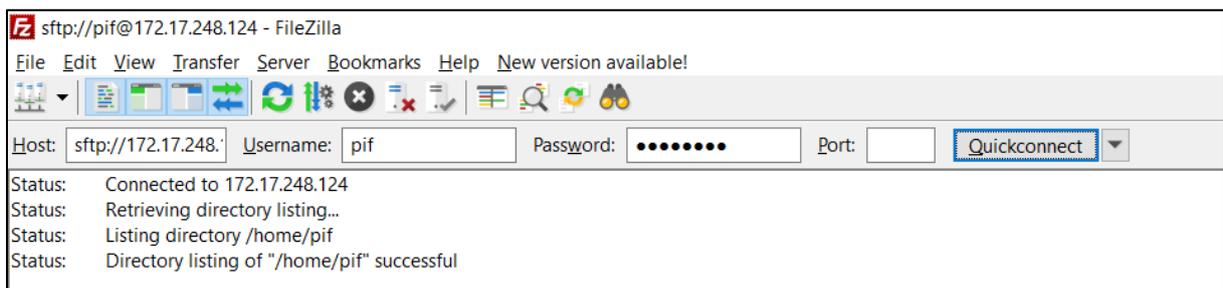


Um die SFTP Verbindung zu testen habe ich **FileZilla** geöffnet.



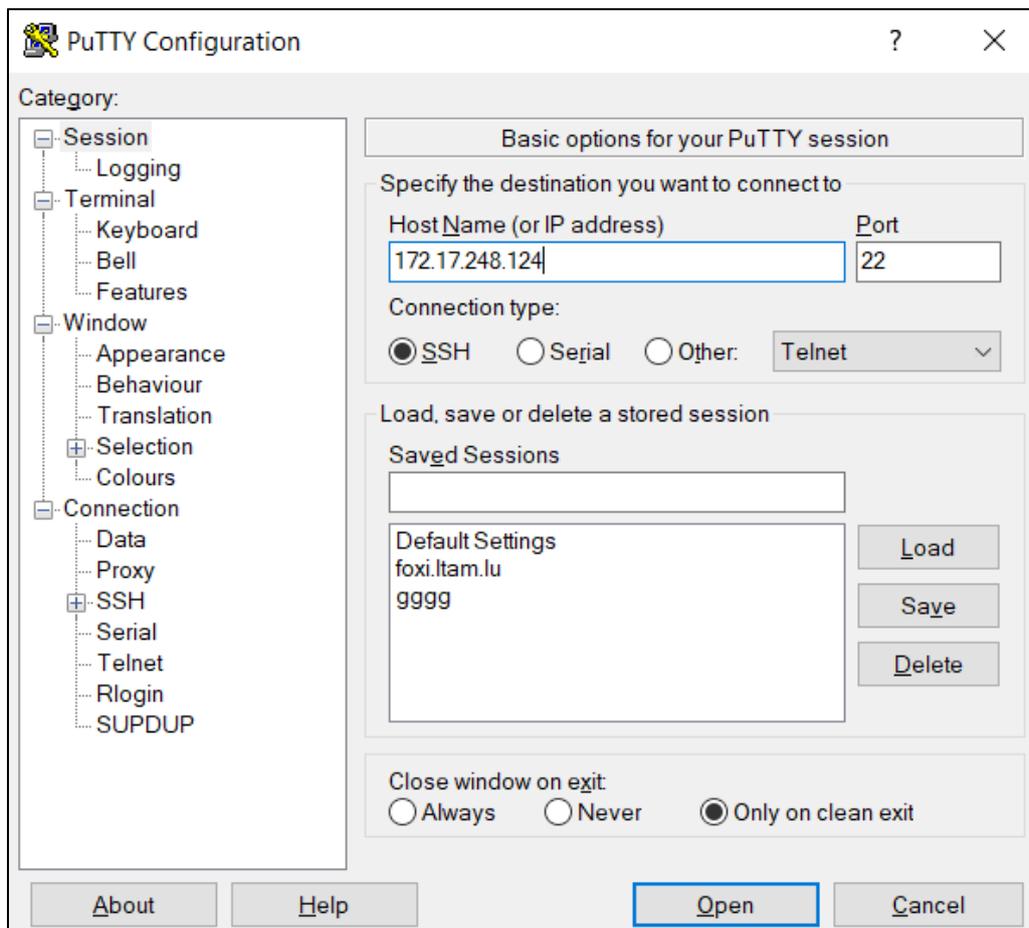
Dort habe ich dann meine Zugangsdaten eingeben.

- Bei „Host“ muss man „sftp://IP-Adresse“ schreiben.
- Bei „Username“ muss man den Namen des Benutzers schreiben, wie in diesem Fall „pif“.
- Bei „Password“ muss man das Passwort des Benutzers eingeben.
- Als letzten muss man die Port Nummer eingeben welche in diesem Fall „22“ ist.

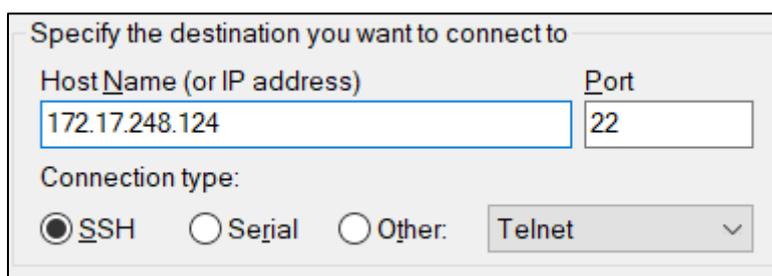


Und hier die Bestätigung dass die Verbindung mit SFTP auf mein Raspberry Pi funktioniert hat.

## SSH Verbindung mit PuTTY



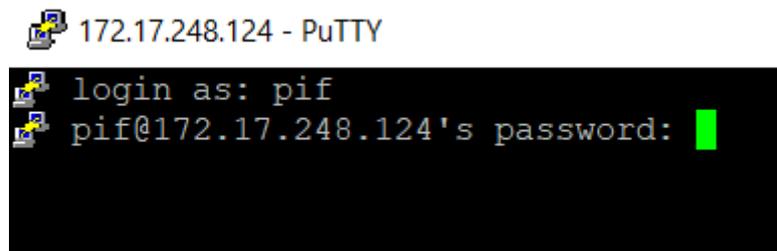
Um die Verbindung mit SSH zu testen habe ich **Putty** geöffnet.



Dann habe ich auf der rechten Seite von Putty meine IP-Adresse von Raspberry eingegeben und bei „Port“ 22 eingegeben.

Dann habe ich unten drunter als „Connection type“ SSH ausgewählt.

Als ich das erledigt hatte habe ich unten rechts auf „Open“ gedrückt.

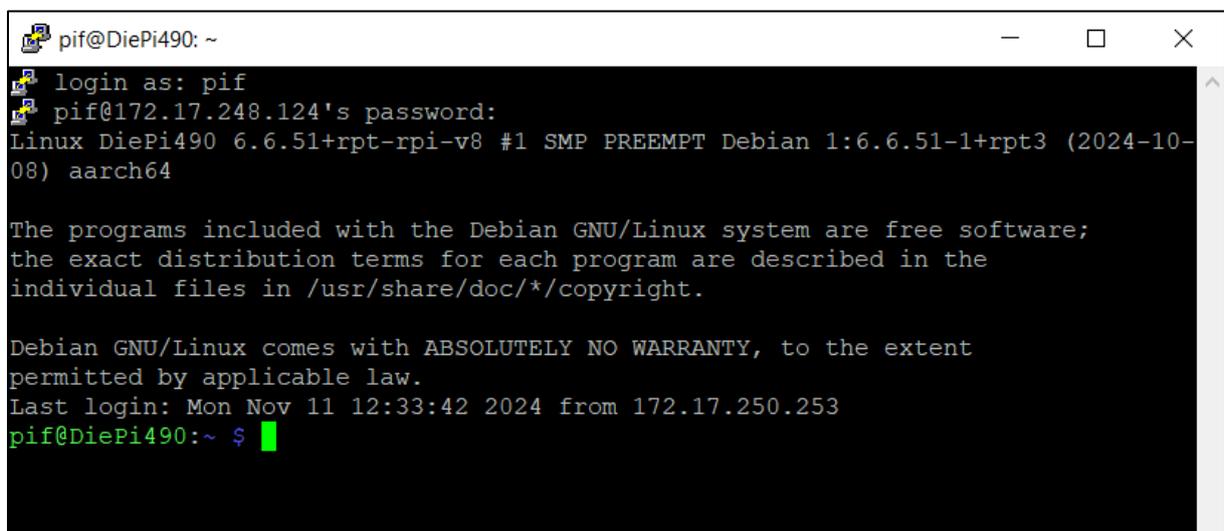
A terminal window titled "172.17.248.124 - PuTTY" showing a login prompt. The user enters "pif" for the login name and a redacted password for the password field.

```
login as: pif
pif@172.17.248.124's password: 
```

Dann hat sich der Terminal geöffnet.

Als login habe ich mein Benutzernamen eingegeben welcher „pif“ ist.

Und als Passwort mein Passwort was ich bei der Installation von Raspberry angegeben hatte.

A terminal window titled "pif@DiePi490: ~" showing the output of a successful login. The terminal displays system information, a copyright notice, and the last login time.

```
pif@DiePi490: ~
login as: pif
pif@172.17.248.124's password:
Linux DiePi490 6.6.51+rpt-rpi-v8 #1 SMP PREEMPT Debian 1:6.6.51-1+rpt3 (2024-10-08) aarch64

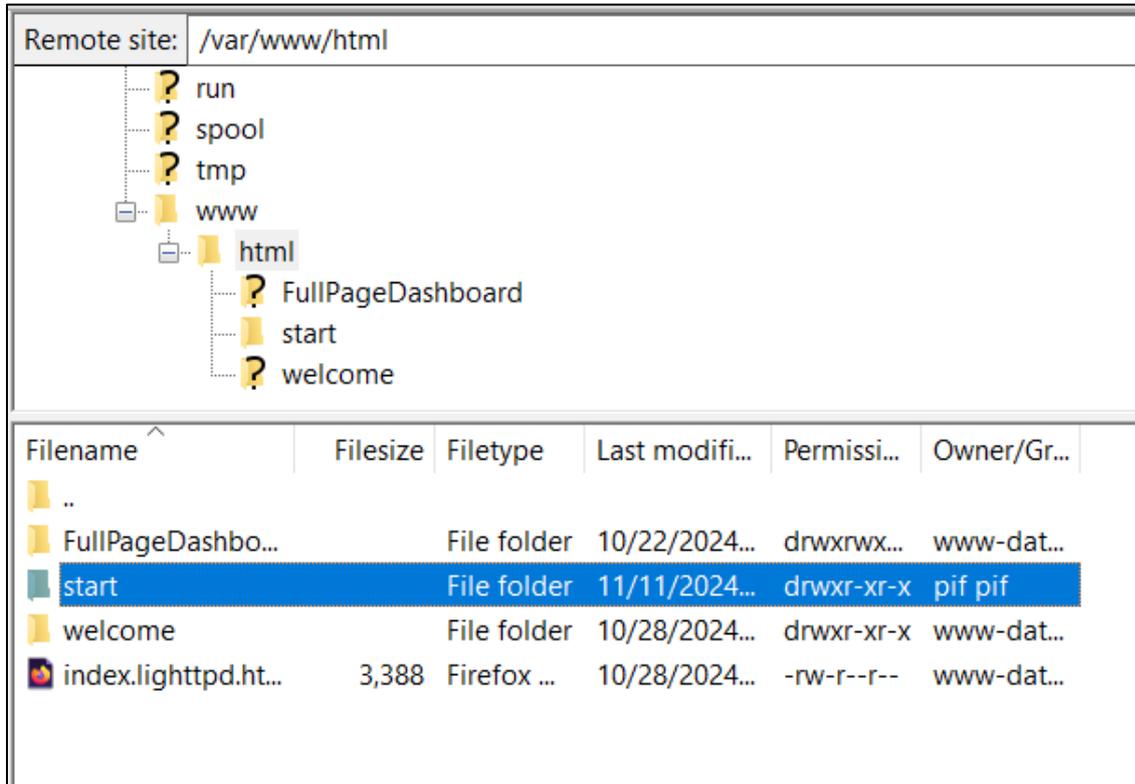
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Nov 11 12:33:42 2024 from 172.17.250.253
pif@DiePi490:~ $
```

Und hier die Bestätigung dass die Verbindung mit Putty funktioniert hat.

## Automatisches Starten der Webseite

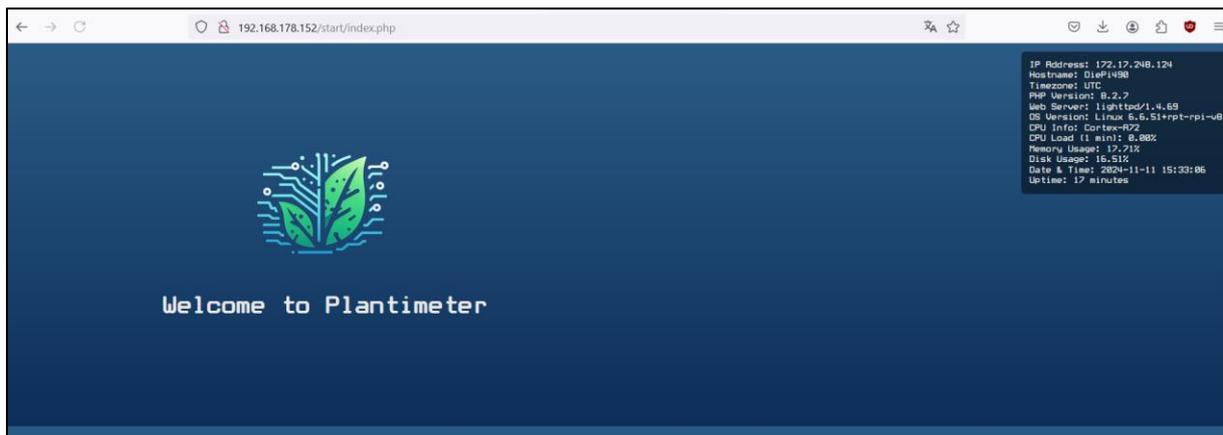
Die nächste Aufgabe die wir machen müssen ist dass, wenn wir den Raspberry Pi anschalten dass automatisch eine Webseite angezeigt wird.



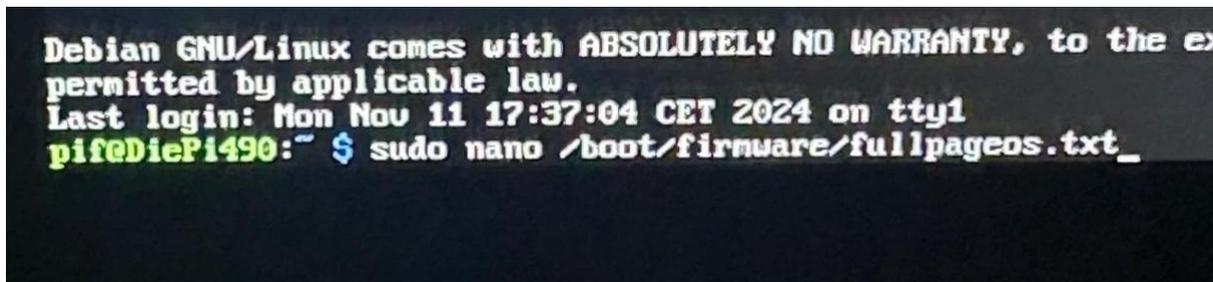
Deshalb gehen wir als erstes in FileZilla zu „/var/www/html/“ und erstellen dort den Ordner „start“.



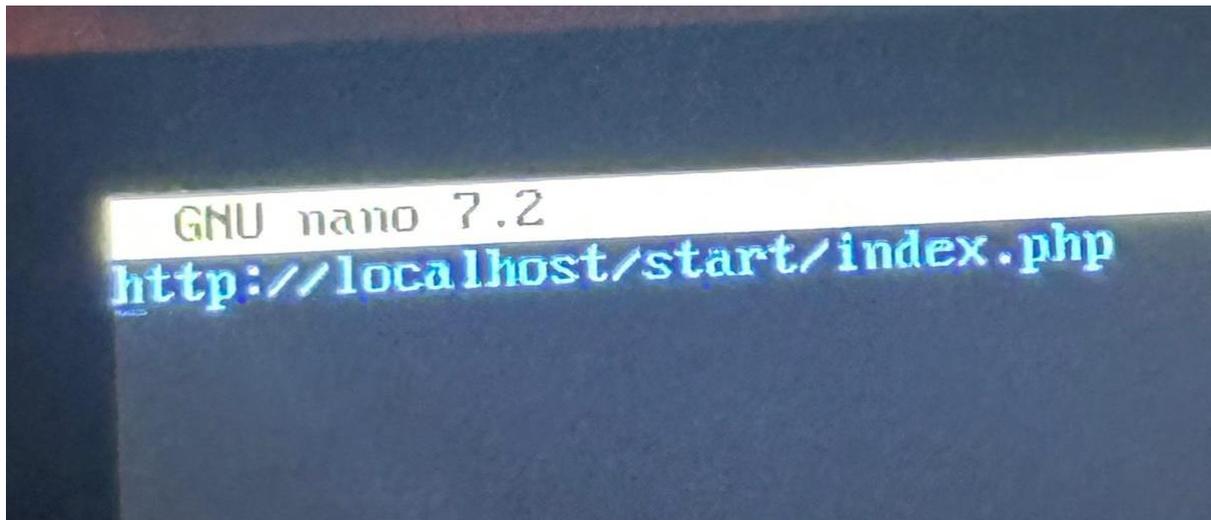
In den eben erstellten Ordner fügen wir jetzt die Dateien der Webseite ein die automatisch mit dem Starten von Raspberry Pi angezeigt werden soll.



Wenn ich auf dem Laptop <http://172.17.248.124/start/index.php> eingebe dann wird die „Index.php“ Seite aufgerufen



Damit die Mitgelieferte Webseite auch Automatisch mit dem Start von Raspberry Pi startet muss ich zuerst in das Verzeichnis „/boot/firmware/“ welches ich mit „sudo nano“ aufrufe. Dort drin öffne ich die Datei „fullpageos.txt“.



Dann füge ich hier den Pfad ein wo sich die Datei befindet mit der sich die Webseite automatisch öffnet wenn ich Raspberry Pi öffne.



Anschließend habe ich mein Raspberry Pi Neugestartet. Als der Neustart abgeschlossen war, sah ich die Webseite die automatisch gestartet wurde.

## Dateien dem Benutzer www-data zuweisen

Als nächstes sollte ich den Eigentümer aller Dateien im Ordner „start“ auf den Benutzer „www-data“ setzen

```
pif@DiePi490:~$ sudo chown -R www-data:www-data /var/www/html/start
chown: cannot access '/var/www/html/start': No such file or directory
pif@DiePi490:~$ sudo chown -R www-data:www-data /var/www/html/start/
chown: cannot access '/var/www/html/start/': No such file or directory
pif@DiePi490:~$ ls -l /var/www/html
total 16
drwxr-xr-x 7 www-data www-data 4096 Oct 22 14:56 FullPageDashboard
-rw-r--r-- 1 www-data www-data 3388 Oct 28 18:06 index.lighttpd.html
drwxr-xr-x 2 pif pif 4096 Nov 11 16:10 start
drwxr-xr-x 9 www-data www-data 4096 Oct 28 18:07 welcome
pif@DiePi490:~$ cd /var/www
pif@DiePi490:/var/www$ cd html
pif@DiePi490:/var/www/html$ cd start
pif@DiePi490:/var/www/html/start$ ls
index.php logo_tp.png nostats.html splash.png whitragt.woff
pif@DiePi490:/var/www/html/start$ cd
pif@DiePi490:~$ sudo chown -R www-data:www-data /var/www/html/start/
pif@DiePi490:~$ cd /var/www
pif@DiePi490:/var/www$ cd start
-bash: cd: start: No such file or directory
pif@DiePi490:/var/www$ cd html
pif@DiePi490:/var/www/html$ cd start
pif@DiePi490:/var/www/html/start$ ls -l
total 588
-rw-r--r-- 1 www-data www-data 4153 Nov 11 16:09 index.php
-rw-r--r-- 1 www-data www-data 240719 Nov 11 16:09 logo_tp.png
-rw-r--r-- 1 www-data www-data 1048 Nov 11 16:09 nostats.html
-rw-r--r-- 1 www-data www-data 329836 Nov 11 16:10 splash.png
-rw-r--r-- 1 www-data www-data 5096 Nov 11 16:09 whitragt.woff
pif@DiePi490:/var/www/html/start$
```

Dies habe ich mit dem Befehl „sudo chown -R www-data:www-data /var/www/html/start/“ getan.

## Aktualisieren der Softwarepakete

```
pif@DiePi490:~$ sudo apt-get update
Get:1 http://archive.raspberrypi.com/debian bookworm InRelease [39.2 kB]
Get:2 http://archive.raspberrypi.com/debian bookworm/main armhf Packages [541 kB]
Get:3 http://archive.raspberrypi.com/debian bookworm/main arm64 Packages [511 kB]
Get:4 http://raspbian.raspberrypi.com/raspbian bookworm InRelease [15.0 kB]
Get:5 http://raspbian.raspberrypi.com/raspbian bookworm/main armhf Packages [14.5 MB]
Fetched 15.6 MB in 10s (1,494 kB/s)
Reading package lists... Done
W: http://raspbian.raspberrypi.com/raspbian/dists/bookworm/InRelease: Key is stored in legacy trust
d.gpg keyring (/etc/apt/trusted.gpg), see the DEPRECATION section in apt-key(8) for details.
pif@DiePi490:~$
```

Und zum Schluss kann ich mit dem Befehl „sudo apt-get update“ alle Softwarepakete auf die neueste Version aktualisieren.

## Netzwerküberwachungstool installieren

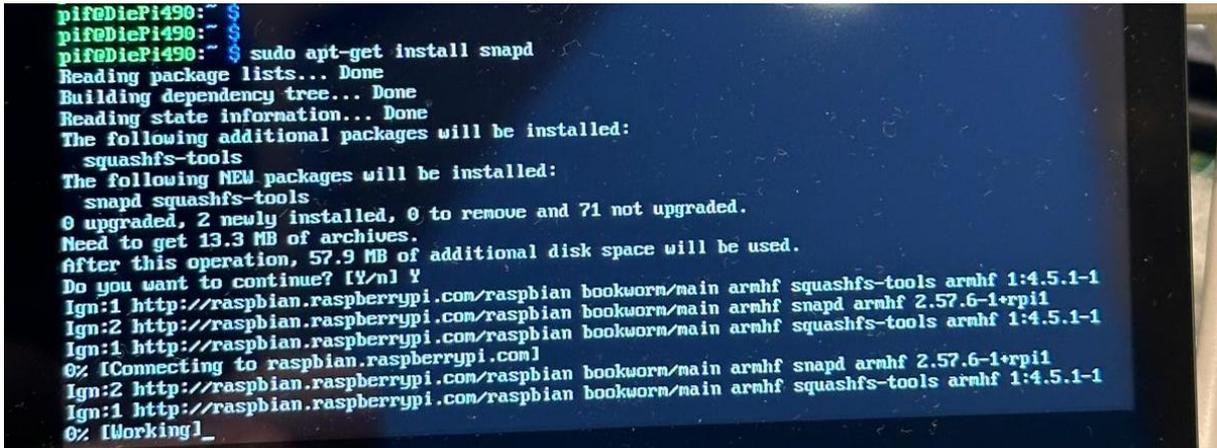
```
unstat
The following NEW packages will be installed:
unstat
0 upgraded, 1 newly installed, 0 to remove and 71 not upgraded.
Need to get 99.1 kB of archives.
After this operation, 268 kB of additional disk space will be used.
Get:1 http://mirror.de.leaseweb.net/raspbian/raspbian bookworm/main armhf unstat arm
kB1
Fetched 99.1 kB in 0s (216 kB/s)
Selecting previously unselected package unstat.
(Reading database ... 90961 files and directories currently installed.)
Preparing to unpack .../unstat_2.10-2_armhf.deb ...
Unpacking unstat (2.10-2) ...
Setting up unstat (2.10-2) ...
Created symlink /etc/systemd/system/multi-user.target.wants/unstat.service → /lib/sy
tat.service.
Processing triggers for man-db (2.11.2-2) ...
pifeDiePi490:~$
pifeDiePi490:~$
pifeDiePi490:~$
pifeDiePi490:~$
pifeDiePi490:~$
pifeDiePi490:~$
pifeDiePi490:~$ sudo apt-get install unstat
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
unstat is already the newest version (2.10-2).
0 upgraded, 0 newly installed, 0 to remove and 71 not upgraded.
pifeDiePi490:~$ _
```

Als nächstes möchte ich noch ein Netzwerküberwachungstool installieren dies mache ich mit dem Befehl „sudo apt-get install vnstat“.

```
pifeDiePi490:~$
pifeDiePi490:~$
pifeDiePi490:~$
pifeDiePi490:~$ sudo systemctl start vnstat_
pifeDiePi490:~$
```

Mit diesem Befehl starte ich anschließend „vnstat“.

## Passwörter im Passwortmanager speichern

A terminal window on a Raspberry Pi showing the command 'sudo apt-get install snapd' and its output. The output indicates that 'squashfs-tools' will be installed along with 'snapd'. It also shows the disk space requirements and the progress of the installation from the raspbian repository.

```
pif@DiePi490:~$  
pif@DiePi490:~$  
pif@DiePi490:~$ sudo apt-get install snapd  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following additional packages will be installed:  
  squashfs-tools  
The following NEW packages will be installed:  
  snapd squashfs-tools  
0 upgraded, 2 newly installed, 0 to remove and 71 not upgraded.  
Need to get 13.3 MB of archives.  
After this operation, 57.9 MB of additional disk space will be used.  
Do you want to continue? [Y/n] Y  
Ign:1 http://raspbian.raspberrypi.com/raspbian bookworm/main armhf squashfs-tools armhf 1:4.5.1-1  
Ign:2 http://raspbian.raspberrypi.com/raspbian bookworm/main armhf snapd armhf 2.57.6-1+rp1  
Ign:1 http://raspbian.raspberrypi.com/raspbian bookworm/main armhf squashfs-tools armhf 1:4.5.1-1  
0% [Connecting to raspbian.raspberrypi.com]  
Ign:2 http://raspbian.raspberrypi.com/raspbian bookworm/main armhf snapd armhf 2.57.6-1+rp1  
Ign:1 http://raspbian.raspberrypi.com/raspbian bookworm/main armhf squashfs-tools armhf 1:4.5.1-1  
0% [Working]
```

Mit dem Befehl „sudo apt-get install snapd“.