# Raspberry pi remote connection

# Remote programming

Bovenstaande werkwijze is omslachtig en niet echt gebruiksvriendelijk, het is altijd leuker om met je eigen laptop aan de slag te kunnen. Met behulp van onderstaande tools is het werken met de py een stuk productiever en aangenamer.

## Mac users: via je mac werken op je raspi

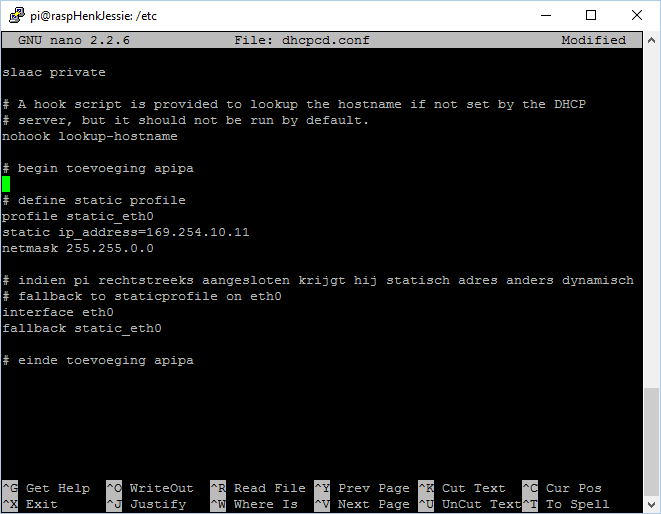
**Stap 1:**

Maak een opstelling met scherm, toetsenbord en muis

Hang je raspi aan het scherm met een hdmi kabel

Via de file /etc/dhcpcd.conf kan je een statisch ip adres toevoegen.

Helemaal onderaan de file voeg je een paar regels toe.



Verbinden we nu de laptop rechtstreeks met de pi, dan krijgt de pi het statisch adres 169.254.10.11.

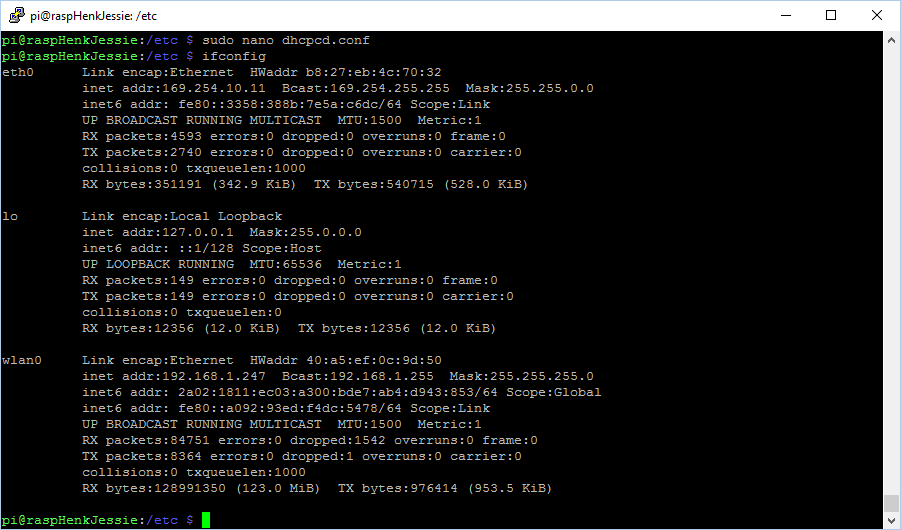
Verbinden we de pi met een router, dan zal hij een dynamisch adres krijgen.

Nu de pi eens heropstarten en we zijn klaar. Nu kunnen we de pi rechtstreeks met onze laptop verbinden. Via de wifi stick kan de pi connectie maken met het internet.

Via het volgende commando kunnen we de netwerkinstellingen van de pi opnieuw toekennen.



Via het commando : ifconfig kan je alle instellingen nog eens controleren.



Het keyboard, muis en scherm heb je nu niet meer nodig.

**Stap 2: APIPA op mac**

Zet je wifi af op mac

Hang je raspi met netwerkkabel aan je mac

(voor meer info over APIPA: zie verder)

Ga naar Terminal

Via SSH (secure shell), een encrypted network protocol kan je remote inloggen op de pi.

Tik: ssh

pi@169.254.10.11



Figure 1: voor de raspi met de naam pi

Geef je paswoord in



Figure 2: paswoord intikken

(tip: Je krijgt niets te zien op je scherm als je paswoord intikt)

Je kan nu alles doen op je mac vb 

Of netwerkinstellingen checken met ifconfig

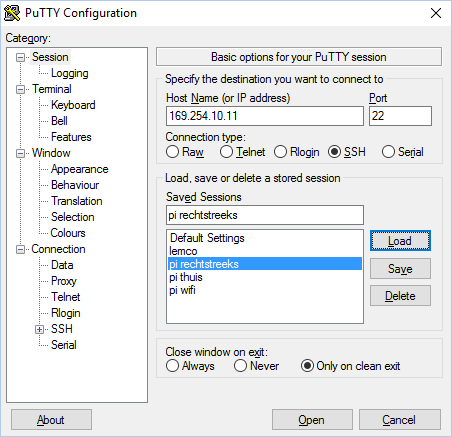


Figure 3: ifconfig

je kan ook je wifi instellen van thuis (moet ik nog doen)

## PC users: gebruik Putty

Via SSH (secure shell), een encrypted network protocol kan je remote inloggen op de pi.



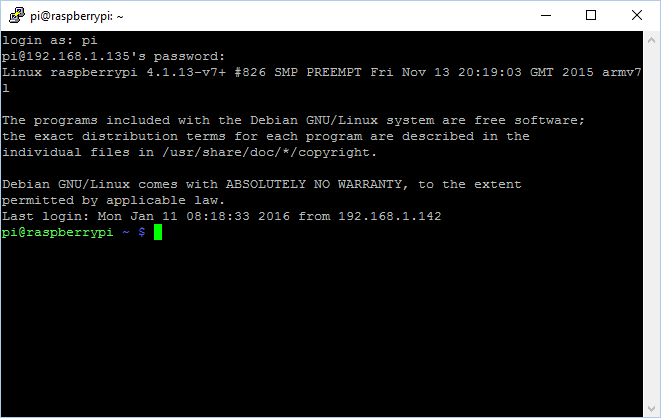
Via de shell kan je nu de pi perfect gaan overnemen. Wel opletten als je het paswoord intikt krijg je niks van feedback terug.

Via de shell kan je perfect alle configuraties aanpassen, scripts uitvoeren,…

Wil je gebruik maken van het keypad heb je de onderstaande instelling nodig. Vergeet niet je configuratie op te slaan.



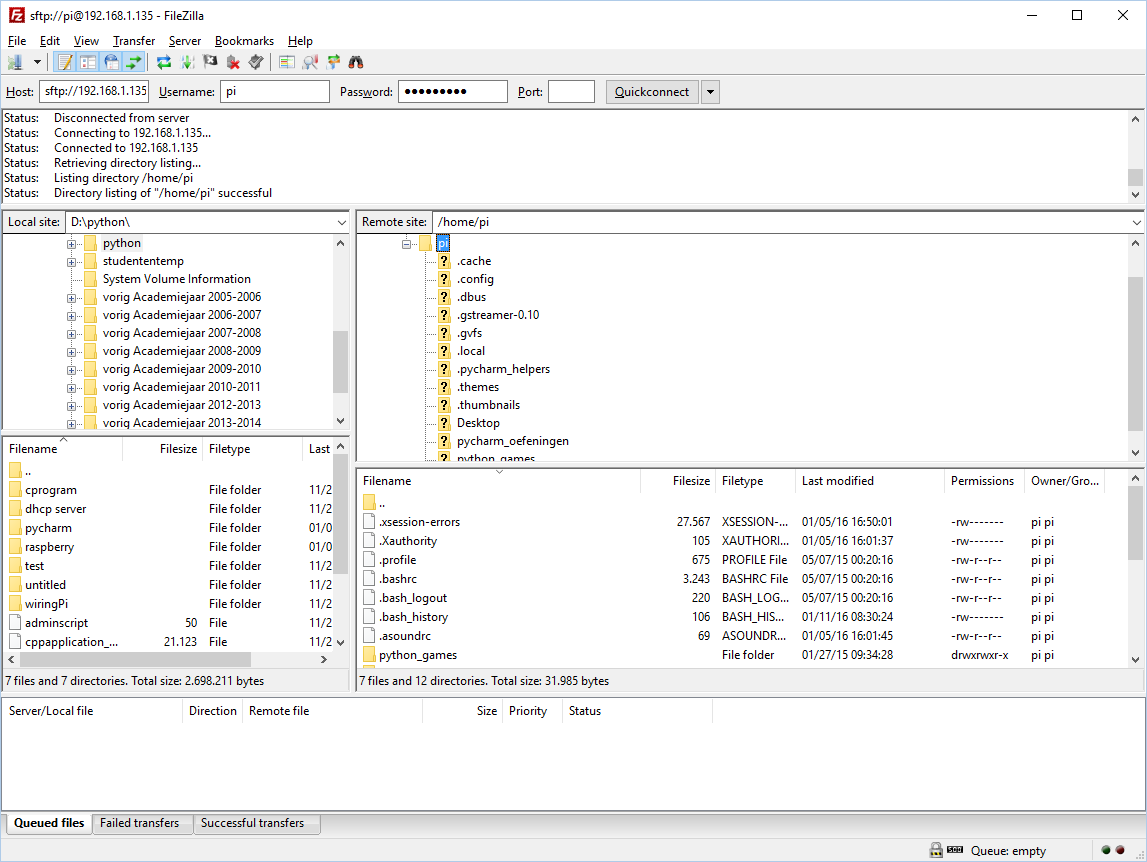
Het keyboard, muis en scherm heb je nu niet meer nodig. Thuis werkt deze opstelling zeer goed. Gewoon de pi verbinden met een ethernetkabel, surfen naar de router waar je het ip adres van je pi kan terugvinden en je kan aan de slag.



## FileZilla

Het kopiëren en verplaatsen van files blijft wel nog steeds lastig. Ftp tools kunnen ons daarbij helpen.

B.v. FileZilla



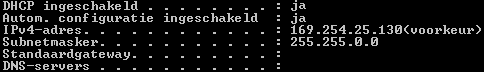
## Configuratie van raspberry pi voor de howest

Het is zeer vervelend om telkens het ip adres van de raspberry pi opnieuw te moeten instellen. We moeten dit doen voor de interpreter van pycharms, ftp, putty,…

Volgende instellingen zorgen ervoor dat we door een statisch ip adres in te stellen, via de ethernet poort kunnen connecteren via secure shell met de pi. Via de usb wifi dongle kan de pi verbinding maken met het internet. Thuis werkt dit zeer vlot. Verbinding maken met het howest netwerk is niet zo eenvoudig omdat er met certificaten gewerkt wordt. Voorlopig denken we er aan dat elke docent een eigen acces point zal meebrengen waar de ganse klas kan op inloggen.

## Afbeeldingsresultaat voor pippa middletonAPIPA (Merci Hans, ik had enkel nog maar van Pippa gehoord)

**APIPA** is een afkorting die staat voor **Automatic Private IP Addressing**. Met APIPA zullen DHCP-clients op een [Microsoft](https://nl.wikipedia.org/wiki/Microsoft)-[besturingssysteem](https://nl.wikipedia.org/wiki/Besturingssysteem" \o "Besturingssysteem)zichzelf automatisch configureren met een [IP-adres](https://nl.wikipedia.org/wiki/IP-adres) en een [subnetmask](https://nl.wikipedia.org/wiki/Subnetmask" \o "Subnetmask) als er geen DHCP-server beschikbaar is en geen vast IP-adres is ingesteld.

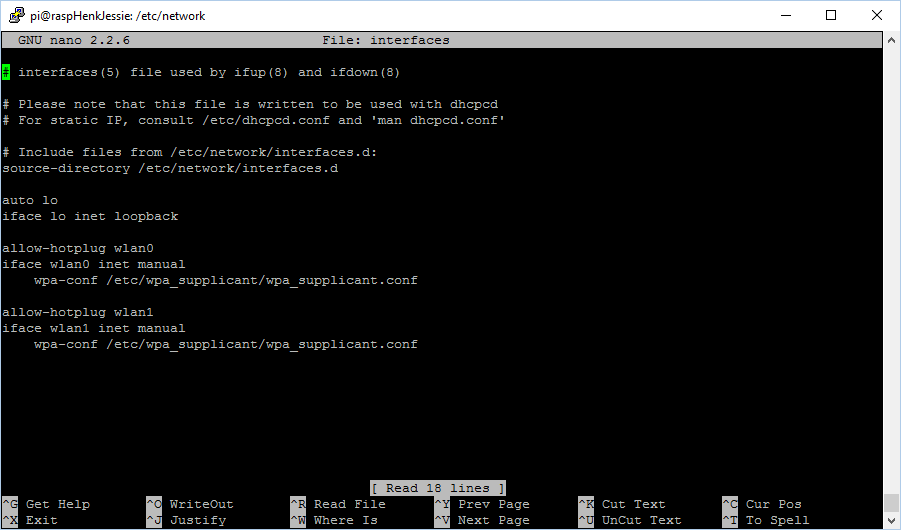
[](https://commons.wikimedia.org/wiki/File:APIPA.png)

Wanneer een DHCP-client opstart, wordt eerst gezocht naar een DHCP-server om een IP-adres en [subnetmask](https://nl.wikipedia.org/wiki/Subnetmask" \o "Subnetmask) op te vragen. Indien de client geen DHCP server kan bereiken gaat de client gebruikmaken van APIPA. [IANA](https://nl.wikipedia.org/wiki/Internet_Assigned_Numbers_Authority) (Internet Assigned Numbers Authority) heeft de reeks 169.254.0.1 t/m 169.254.255.254 gereserveerd voor APIPA-gebruik. Hierdoor is het gegarandeerd dat een APIPA-adres geen conflict zal krijgen met routeerbare adressen. De client zal uit deze reeks zichzelf van een IP adres voorzien. Tevens zal de client gebruik blijven maken van dit zelf-geconfigureerde IP-adres tot een DHCP-server beschikbaar is.

APIPA controleert om de 5 minuten op de aanwezigheid van een DHCP-server. Als eenmaal een DHCP-server is gevonden stopt APIPA. De DHCP-server vervangt het APIPA-adres en het subnetmask met dynamisch toegewezen adressen.

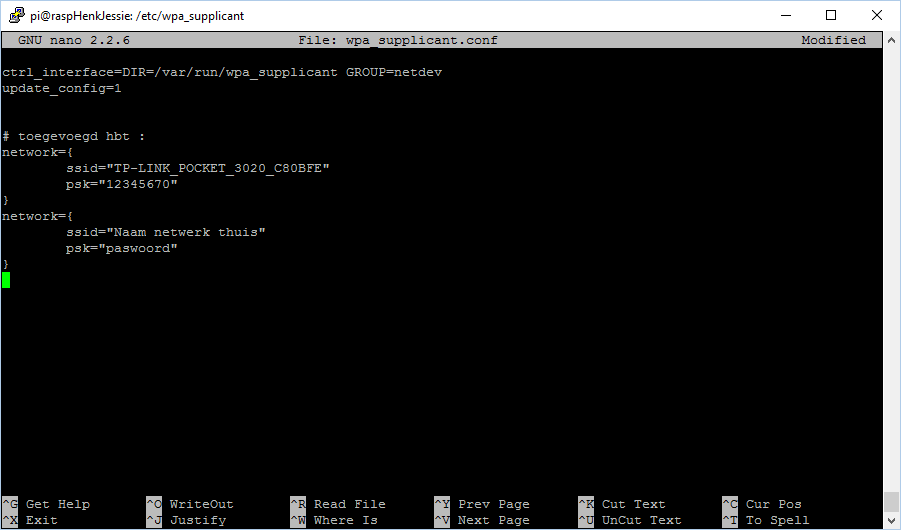
# Configuratie van Raspbian Jessie

Bij Jessie is het de bedoeling dat we de file /etc/network/interfaces ongemoeid laten.



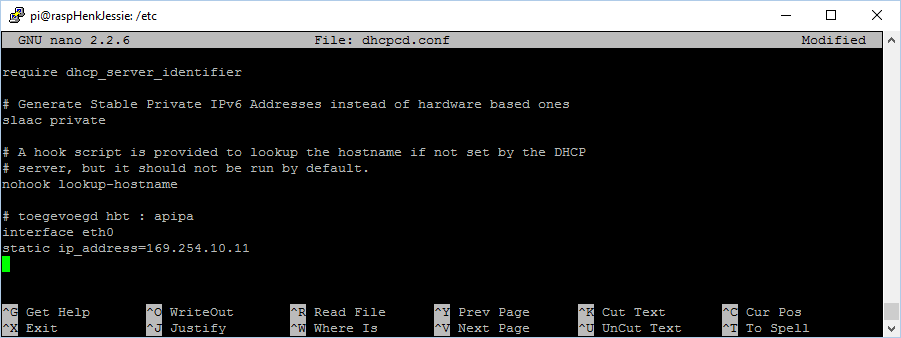
We zien wel dat er naar de file /etc/wpa\_supplicant/wpa\_supplicant.conf verwezen wordt voor de wifi-instellingen.

Via  kan je deze file aanpassen. Ik heb 2 wifi-netwerken toegevoegd.



Via de file /etc/dhcpcd.conf kan je een statisch ip adres toevoegen.

Helemaal onderaan de file voeg je 2 regels toe.

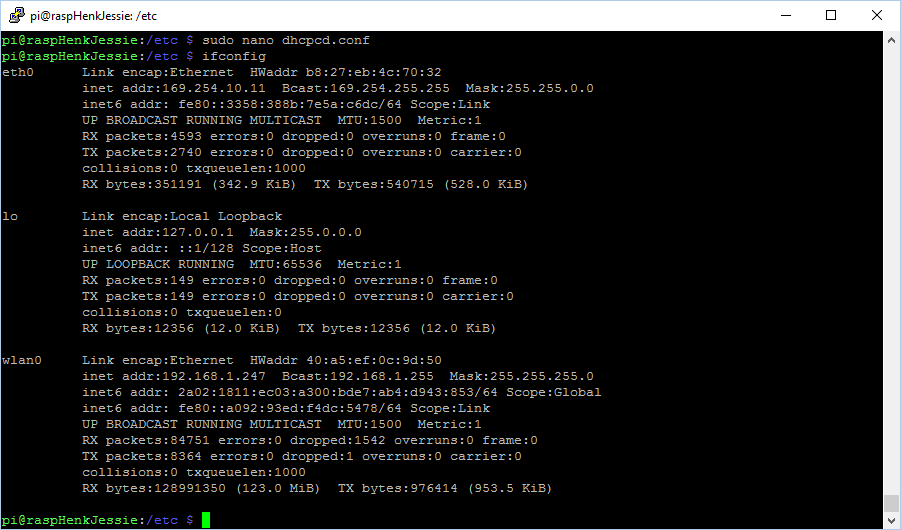


Nu de pi eens heropstarten en we zijn klaar. Nu kunnen we de pi rechtstreeks met onze laptop verbinden. Via de wifi stick kan de pi connectie maken met het internet.

Via het volgende commando kunnen we de netwerkinstellingen van de pi opnieuw toekennen.



Via het commando : ifconfig kan je alle instellingen nog eens controleren.



# Configuratie voor Wheezy (niet voor Jessie)

<http://weworkweplay.com/play/automatically-connect-a-raspberry-pi-to-a-wifi-network/>

sudo nano /etc/network/interfaces

auto lo

iface lo inet loopback

auto eth0

allow-hotplug eth0

iface eth0 inet static

address 169.254.10.11

netmask 255.255.0.0

auto wlan0

allow-hotplug wlan0

iface wlan0 inet dhcp

wpa-conf /etc/wpa\_supplicant/wpa\_supplicant.confwpa-conf

sudo nano /etc/wpa\_supplicant/wpa\_supplicant.conf

ctrl\_interface=DIR=/var/run/wpa\_supplicant GROUP=netdev

update\_config=1

network={

ssid="NETGEAR15"

psk="breezyjade909"

proto=RSN

key\_mgmt=WPA-PSK

pairwise=CCMP

auth\_alg=OPEN

}

network={

ssid="TP-LINK\_POCKET\_3020\_C80BFE"

psk="12345670"

proto=RSN

key\_mgmt=WPA-PSK

pairwise=CCMP

auth\_alg=OPEN

}

Via het commande : ifconfig kan je de ip-adressen van je wlan en eth0 checken.

Via het commando : sudo service networking restart kan je de ip-adressen opnieuw toekennnen.

## Pycharm

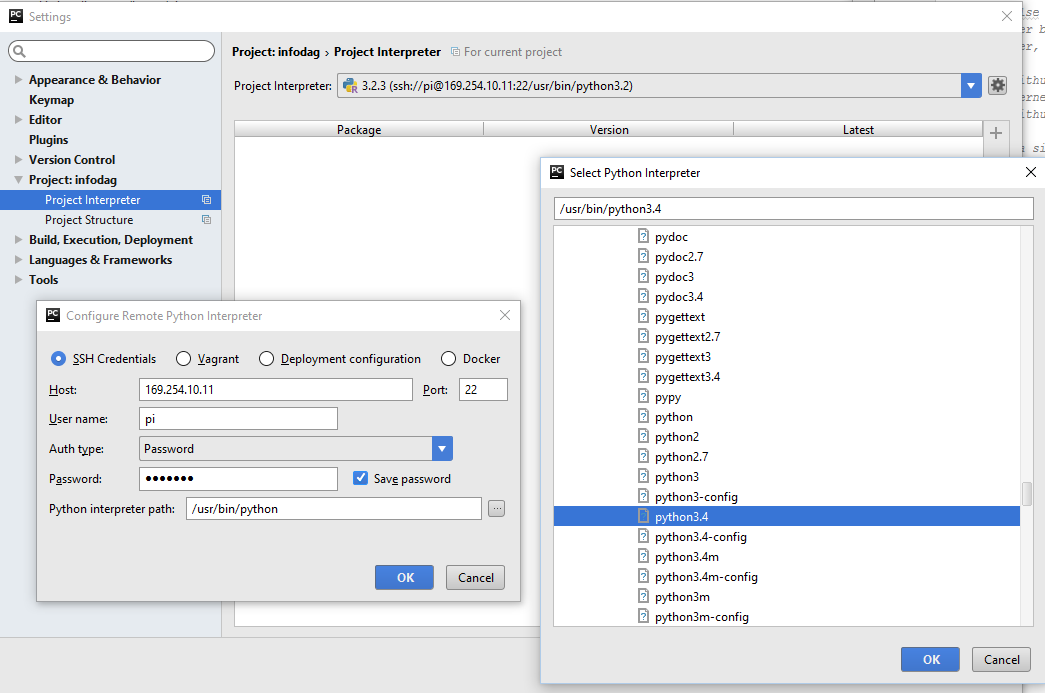
We zijn natuurlijk allen de luxe gewoon van een deftige IDE.

Een aantal softwareboeren biedt ons ook de mogelijkheid om aan remote programming te doen op de pi. Voor pycharm heb je wel de professional edition nodig.

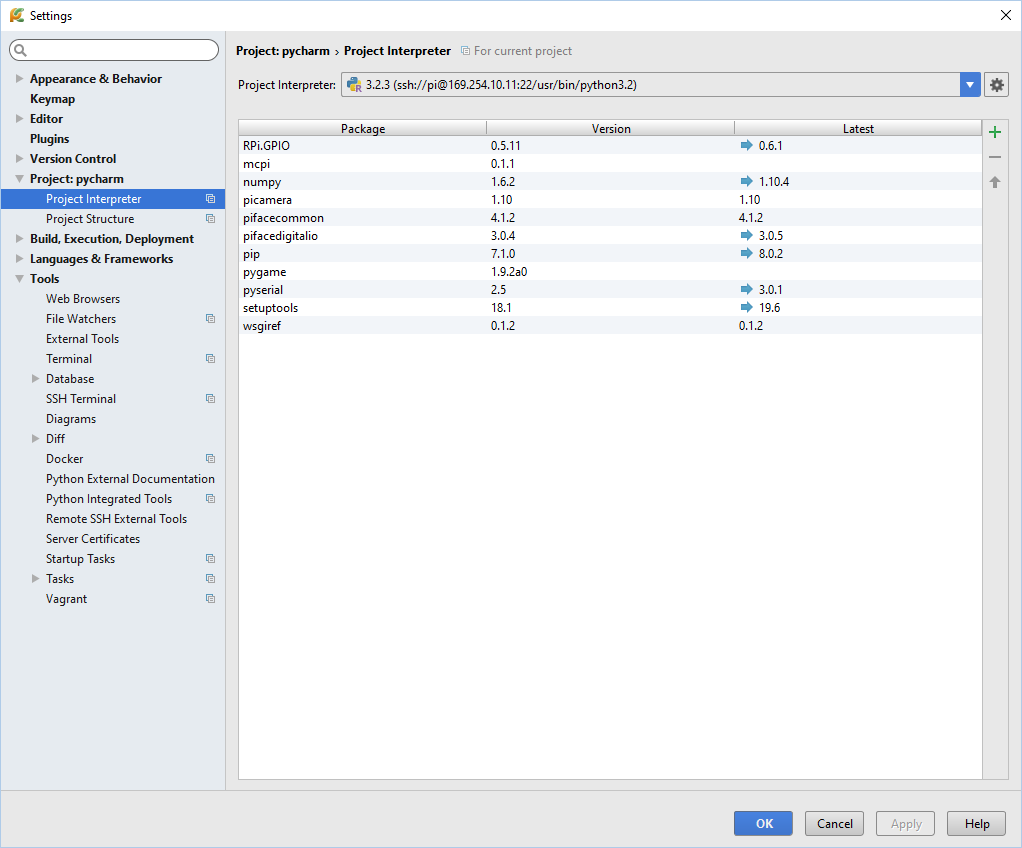
## Project interpreter

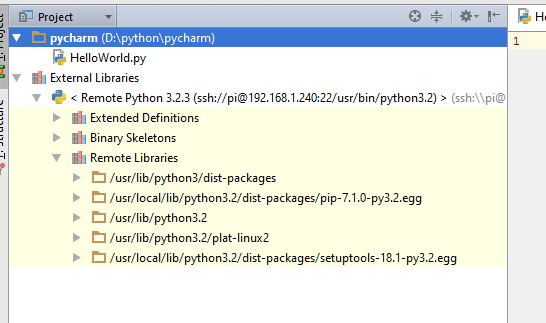
Onze laptop beschikt natuurlijk niet over de functionaliteiten van de pi. We hebben b.v. geen GPIO pinnen. Vandaar dat we gebruik gaan maken van de interpreter op de pi. We gaan dus eerst naar de settings van pycharm om een interpreter van de pi toe te voegen.

Eventueel moet je nog op de blauwe link “install packaging tools” klikken.



Als alles goed verloopt krijg je het volgende te zien :

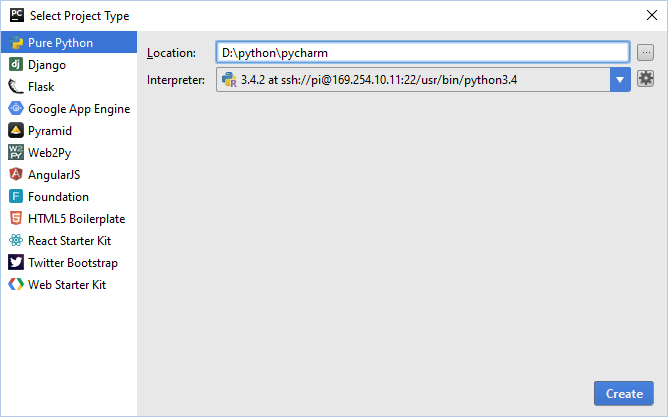




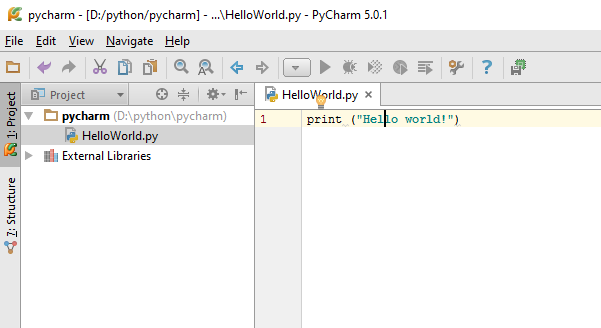
## Selecteer een new project

Kies een directory op je laptop

Kies een remote interpreter op de raspberry pi

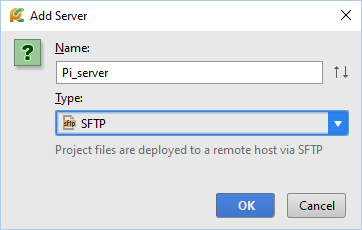


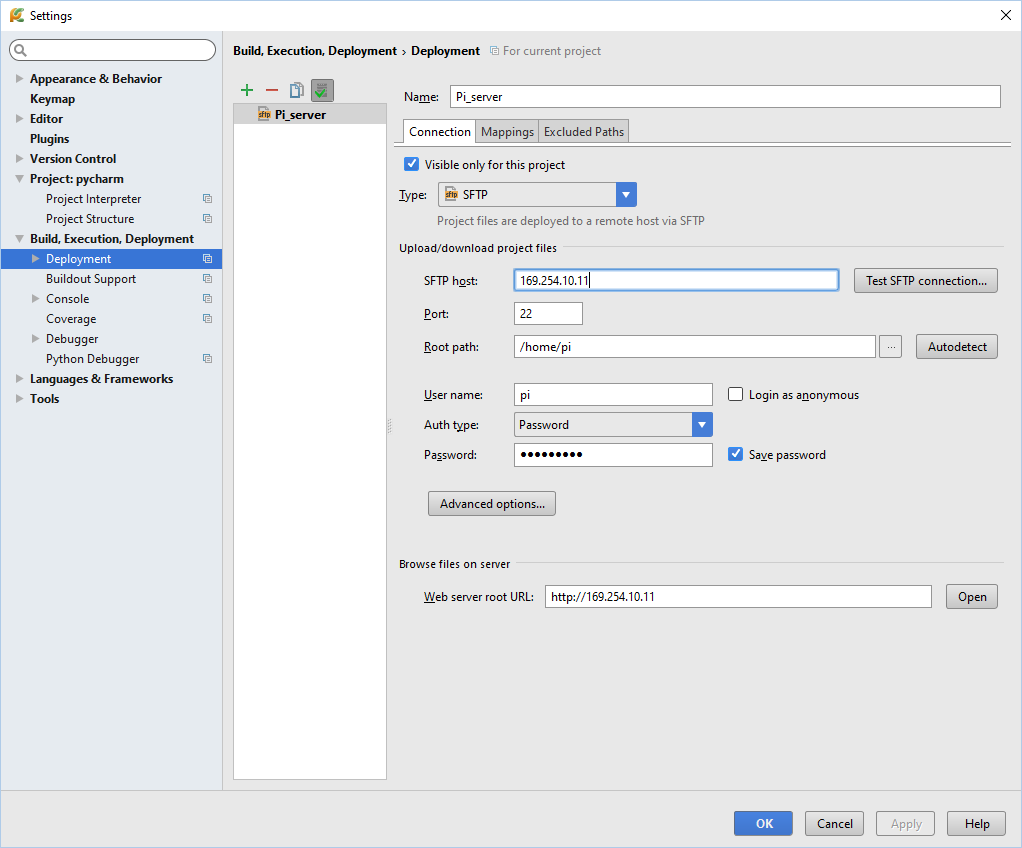
Nu kunnen we een nieuwe python file toevoegen aan het project

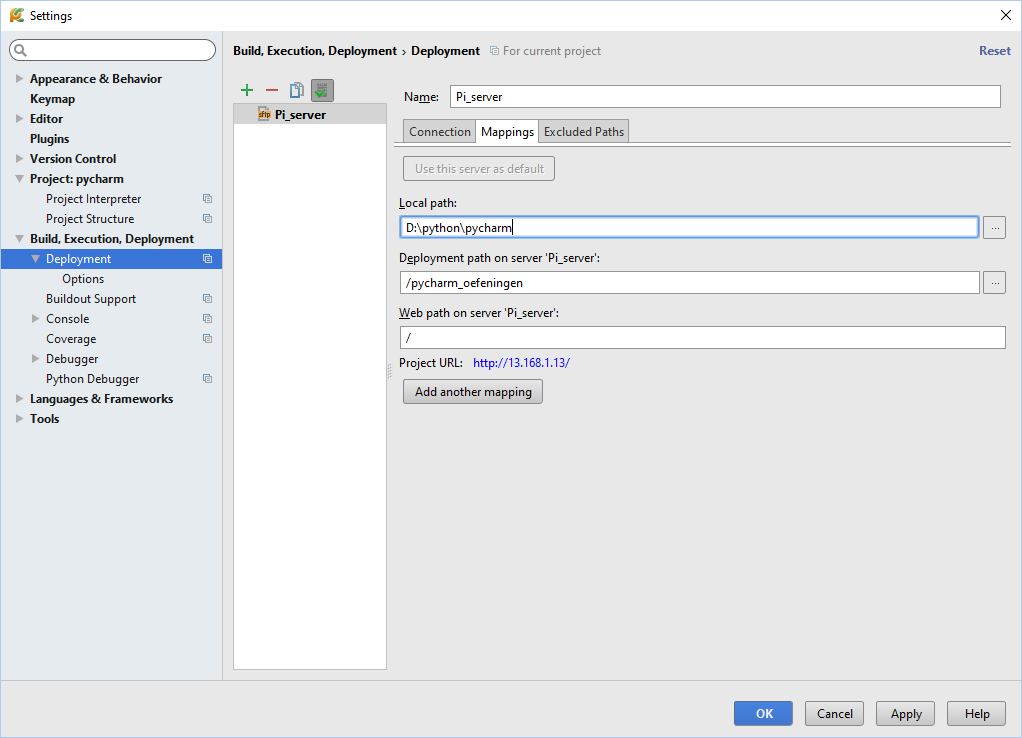


Nu deze file uploaden naar de pi. Daarvoor moet je de onderstaande instellingen in orde brengen.

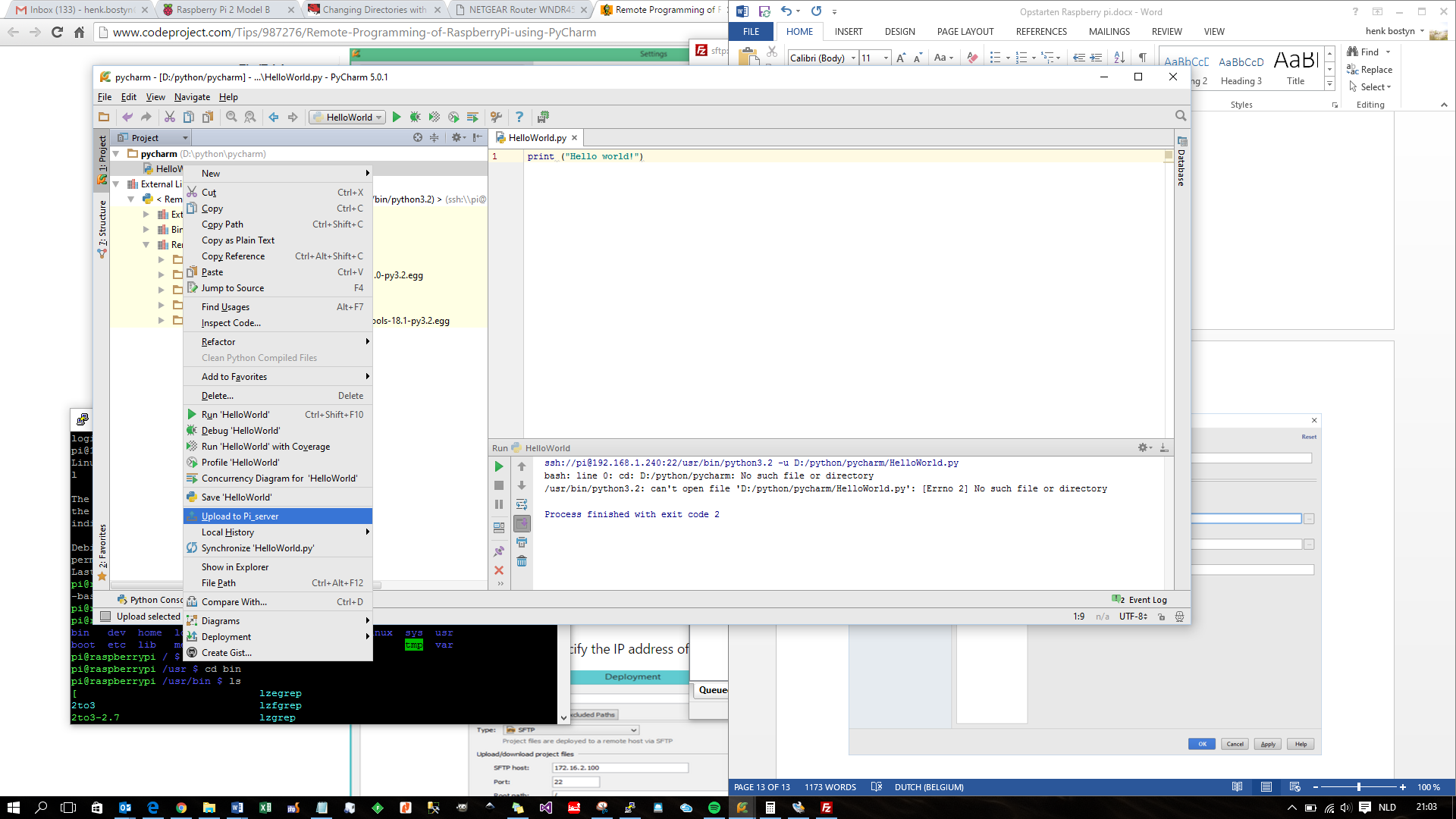
Ga naar de settings van het project => build, execution en deployment en voeg een server toe.

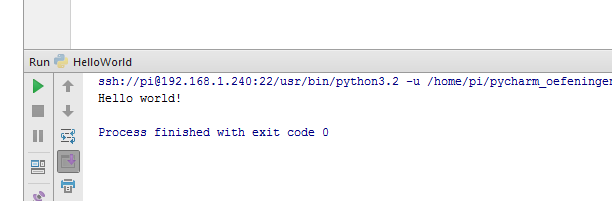






Nu moeten we onze file uploaden naar de py





Of opstarten in de pi zelf

## Sudo apllicatie runnen op de pi

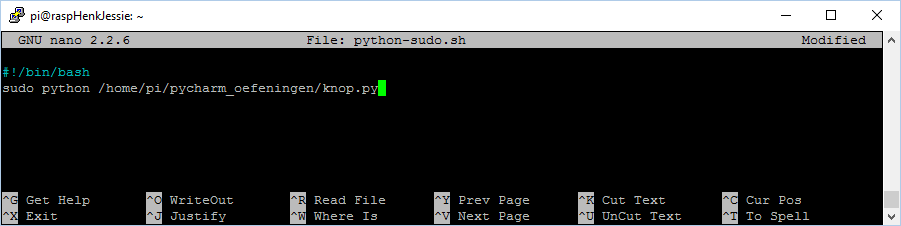
Nu kunnen we zeer vlot remote werken op de Pi via Pycharm. Er rest ons nog 1 probleem : namelijk het uitvoeren van een applicatie die sudo rechten nodig heeft.

Dat kunnen oplossen door het volgende script te maken op de pi :

Maak een map “bin” aan in de homedirectory.



Maak het script “python-sudo.sh” aan in de bin map.



#!/bin/bash (om te vertellen dat je een script maakt)

sudo python /home/pi/pycharm\_oefeningen/knop.py (om het script als sudo te runnen)

Je kan het script testen door het volgend commando in te geven :



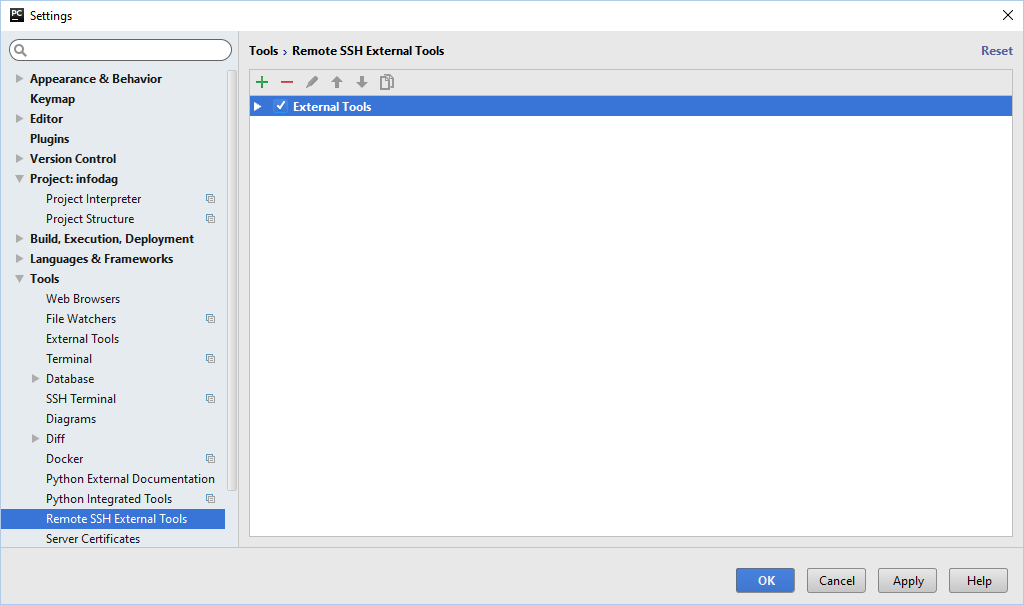
Vervang nu de naam van de file door “$@”

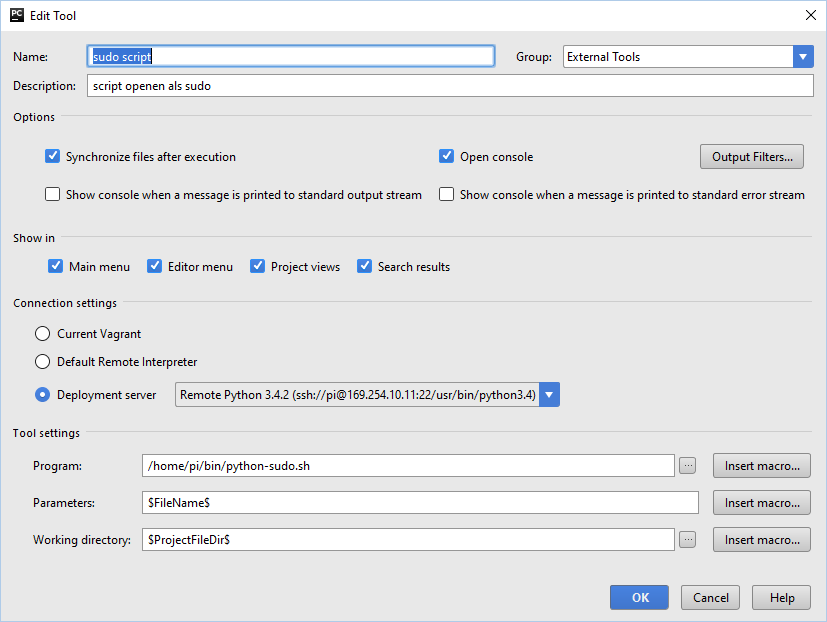


Geef de file volledige execute rechten :



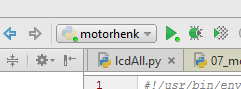
Ga naar settings in pycharm en voeg een remote ssh external tool toe.



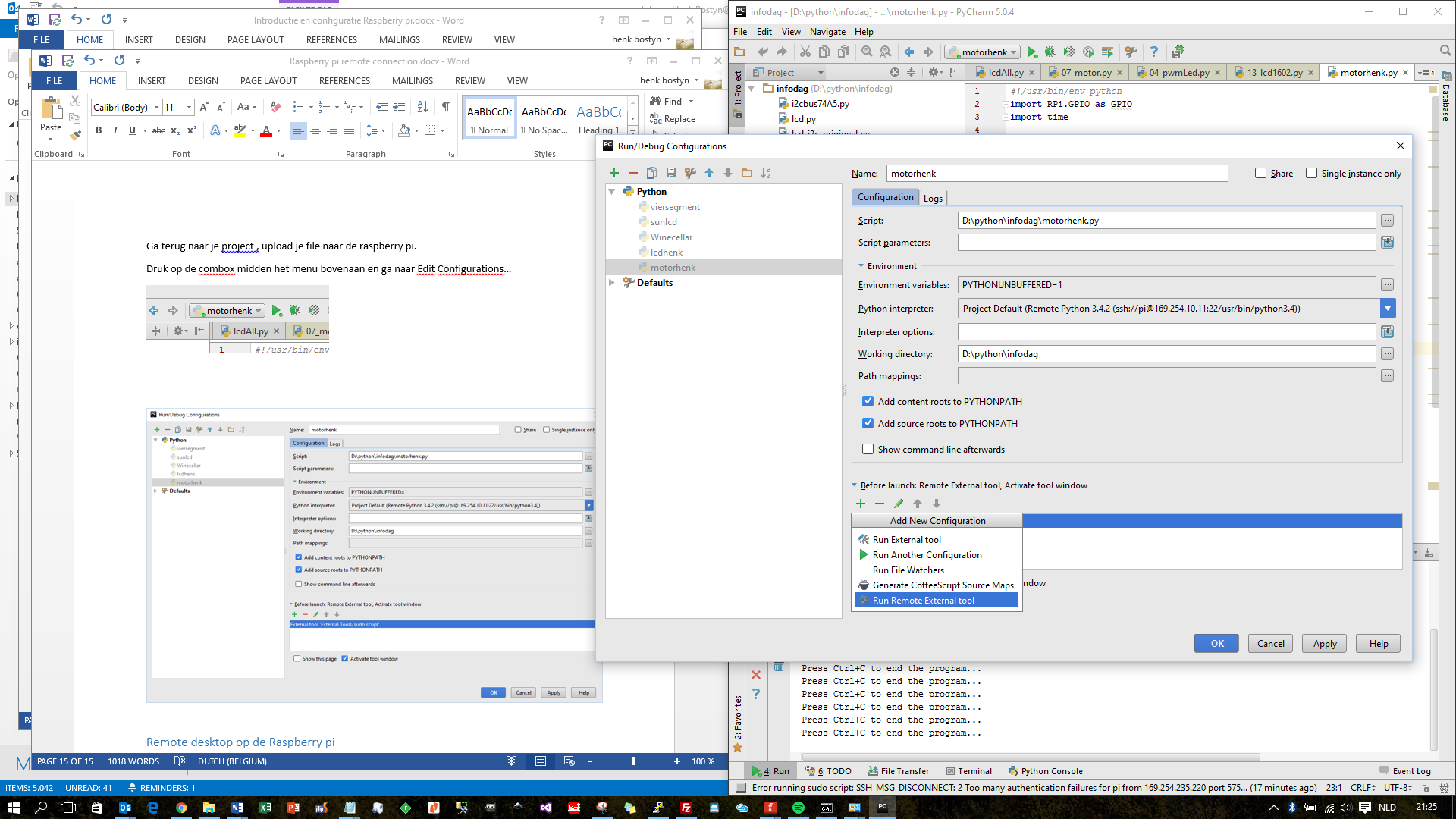


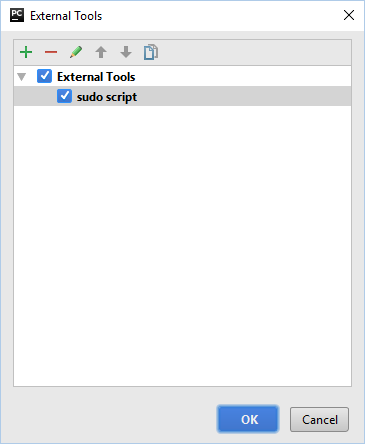
Ga terug naar je project , upload je file naar de raspberry pi.

Druk op de combox midden het menu bovenaan en ga naar Edit Configurations…

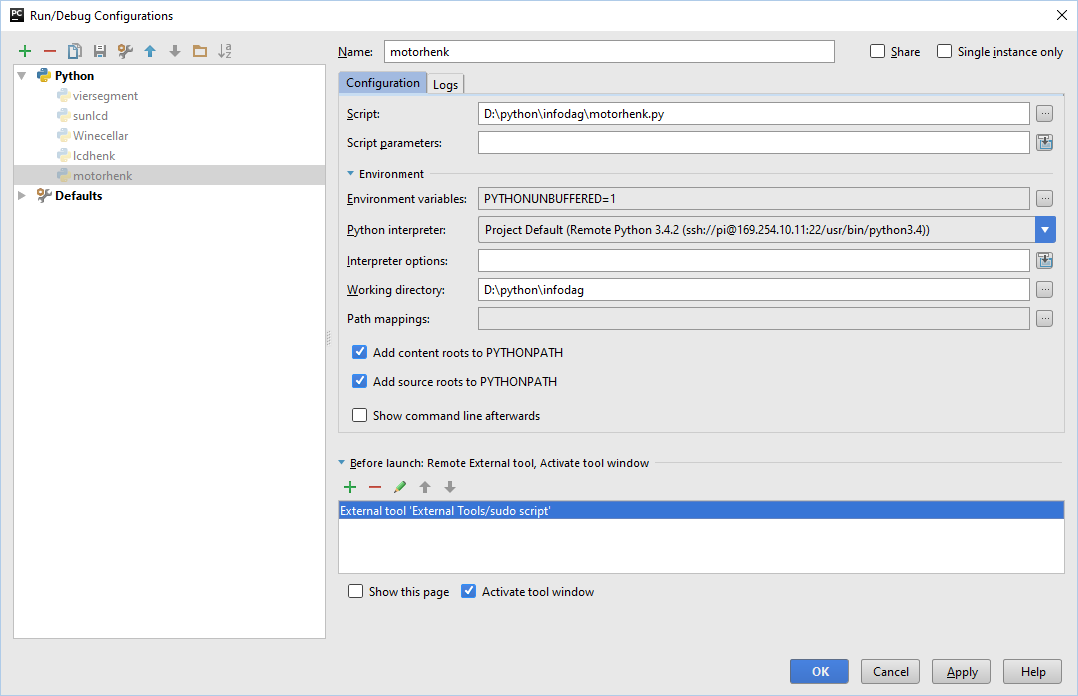


Voeg onderaan een “Run Remote External tool” toe.





Nu zou je onderstaand scherm te zien moeten krijgen en kan je het script als sudo runnen.



## Remote desktop op de Raspberry pi

Eventueel kan je ook het volledige scherm van de raspberry pi overnemen.

<https://www.raspberrypi.org/documentation/remote-access/vnc/>