Konzeption und Implementierung von SmartHome

Schritt-für-Schritt-Anleitung

Inhaltsverzeichnis

[1. Setup Raspberry Pi 1](#_Toc93352995)

[1.1. Raspberry Pi Imager und Setup 1](#_Toc93352996)

[1.2. FHEM Setup 3](#_Toc93352997)

[1.2.1. Alternatives FHEM Setup 4](#_Toc93352998)

[1.3. FTUI Setup 5](#_Toc93352999)

[1.4. Sicherungskonzept 5](#_Toc93353000)

[1.5. Konfiguration einer CA und HTTPS 7](#_Toc93353001)

[2. Implementation in FHEM 13](#_Toc93353002)

[2.1. Funktion Weihnachtsbeleuchtung 13](#_Toc93353003)

[2.1.1. Flashen und einrichten der Funksteckdose 13](#_Toc93353004)

[2.1.2. Manueller Betrieb 14](#_Toc93353005)

[2.1.3. Betriebsmodus Feste Betriebszeit 14](#_Toc93353006)

[2.1.4. Betriebsmodus Sonnenaufgang und Sonnenuntergang 15](#_Toc93353007)

[2.1.5. Betriebsmodus Sonnenaufgang und Sonnenuntergang mit Uhrzeiten 15](#_Toc93353008)

[2.1.6. Betriebsmodus Anwesenheitserkennung 16](#_Toc93353009)

[2.1.7. Betriebsmodusauswahl 17](#_Toc93353010)

[3.1. Funktion Sturmwarnung 18](#_Toc93353011)

[3.1.1. Einrichten eines Bots zum Senden von Telegram Nachrichten über FHEM 18](#_Toc93353012)

[3.1.2. Sensordaten auslesen 21](#_Toc93353013)

[3.1.3. Anlegen der Dummy-Devices für die Eingabe der Schwellenwerte und für den Schalter der Sturmwetterwarnung 22](#_Toc93353014)

[3.1.4. Hinzufügen von Do-If-Abfragen 23](#_Toc93353015)

[3.2. Funktion Photovoltaik/Balkonkraftwerk 25](#_Toc93353016)

[3.2.1. Einrichten einer DB zum Speichern von Daten 26](#_Toc93353017)

[3.2.2. Einfaches Heizen 28](#_Toc93353018)

[3.2.3. Optimiertes Heizen 29](#_Toc93353019)

[3.2.4. Betriebsmodullogik Heizen 33](#_Toc93353020)

[3.2.5. Warmwasserboiler 34](#_Toc93353021)

[3.2.6. Kombination zwischen Warmwasserboiler und Infrarotheizung 37](#_Toc93353022)

[4. FTUI erstellen 38](#_Toc93353023)

[5. HTML-Seiten Konfiguration 39](#_Toc93353024)

[6. Optionale Aufgabe: Müllplan 40](#_Toc93353025)

# Setup Raspberry Pi

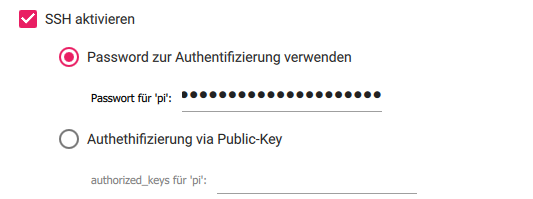
Um das Setup für den Raspberry Pi einzurichten benötigst du:

* Raspberry Pi
* SD-Karte (mind. 16 GB)
* SD-Karten-Slot am PC

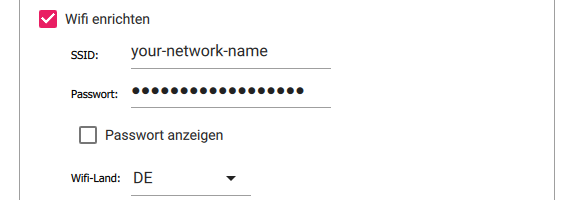
Hast du alles, um das Setup einzurichten, so kannst du der nachfolgenden Anleitung folgen. Du hast aber auch die Möglichkeit einer Videoanleitung zu folgen. Diese findest du auf der Website [www.raspberrypi.com/software/](http://www.raspberrypi.com/software/) mit dem Schrift-Link „Watch our 45-second video“ im Text unter der Überschrift „Install Raspberry Pi OS using Raspberry Pi Imager“.

## Raspberry Pi Imager und Setup

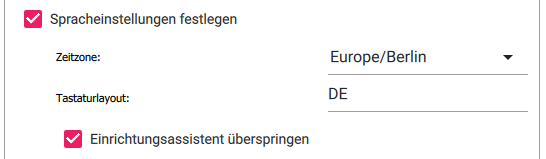
1. Rufe die Website [www.raspberrypi.com/software/](http://www.raspberrypi.com/software/) auf
2. Lade den Raspberry Pi Imager für dein Betriebssystem (Windows/macOS/Ubuntu for x86) die passende Version herunter.
3. Stecke die SD-Karte in den SD-Kartenleser deines Computers.
4. Installiere den Raspberry Pi Imager, indem du die heruntergeladene .exe-Datei durch einen Doppelklick startest.
   1. Wähle das Betriebssystem „Raspberry PI OS (32 Bit)“ aus.
   2. Drücke anschließend die Tasten Strg+shift+x, sodass sich ein weiteres Fenster mit „Erweiterten Optionen“ öffnet.
      1. Setze den Hacken bei „SSH aktivieren“. Wähle ein Passwort für die Authentifizierung aus und lege ein Passwort für den Raspberry Pi fest.



* + 1. Scrolle in den Erweiterten Optionen nach unten und richte dein Wifi ein. Trage hierfür unter SSID den Netzwerk Name und unter Passwort das Netzwerkpasswort ein



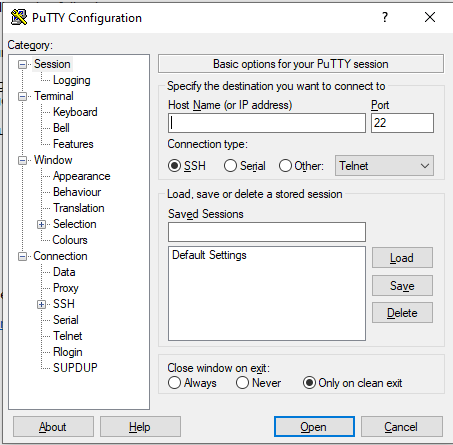
* + 1. Lege die Spracheinstellungen fest



* + 1. Speicher deine Änderungen.
  1. Wähle die entsprechende SD-Karte, die du nutzt, aus.
  2. Führe den Button „Schreiben“ durch Anklicken aus und klicke dich mit der weiter Taste durch die Installation bis du am Ende angekommen bist und der Installationsteil abgeschlossen ist.
     1. Sollten noch Daten auf der SD-Karte vorhanden sein, drücken sie in der Auswahl, dass alle vorherigen Daten gelöscht werden dürfen. Bitte beachten sie, dass nach einer Überschreibung keine Daten wiederhergestellt werden können.

1. Entferne die SD-Karte aus dem SD-Kartenleser deines Computers und stecke sie in den Raspberry Pi. Den Raspberry Pi musst du folgend an das Stromnetz, an einen Bildschirm mit HDMI-Eingang (Computer), sowie vorteilhaft an eine Maus und eine Tastatur anschließen.
   1. Warte bis der Raspberry Pi sich mit deinem WLAN verbunden hat.
2. Lade dir auf dem PC/Laptoop PuTTY von der Website [www.putty.org/](http://www.putty.org/) herunter und installiere dies.
3. Starte den PuTTY und verbinde ihn mit dem Raspberry Pi.
4. Trage unter Host Name (or IP address) entweder den Gerätenamen (siehe 8.1) des Raspberry Pi oder die IP-Adresse (siehe 8.2) ein.
   1. Gib unter Host Name (or IP address) „raspberrypi“ ein. Folgende Meldung erscheint, welche du mit „Accept“ bestätigen musst. Ein Bild, das Text, Screenshot, Monitor enthält.

      Automatisch generierte Beschreibung
   2. Alternativ: Öffne die Kommandozeile auf dem RaspberryPi und führe den Befehl: ifconfig aus. Dort kannst du nun die IP-Adresse des Raspberry Pi’s ablesen.
   3. Trage den Wert 22 unter Port ein.
   4. Wähle SSH bei Connection type aus.



1. PuTTY-Eingaben:
   1. Login as: pi à Enter
   2. Password: Passwort aus Schritt 4.2.1. à Enter



* 1. Eingabe 1: sudo apt update à Enter
  2. Eingabe 2: sudo apt upgrade à Enter (ggfs. zusätzliche Bestätigung mit „Y“ à Enter)

## FHEM Setup

Wir bieten im Rahmen der Schritt-für-Schritt Anleitung zwei Möglichkeiten an die von der Projektgruppe entwickelte FHEM Konfiguration auf einem Raspberry PI zu installieren. Es besteht die Wahl zwischen einem Setup via Skript, detailliert in 1.2.1 FHEM Setup via Skript beschrieben, oder manuell, wie in 1.2.2 FHEM Setup manuell beschrieben. Das Setup via Skript führt die Installation von FHEM, der erarbeiteten Konfiguration, sowie der Einrichtung von Datenbank und Geräten automatisch aus, wobei nur wenige Eingaben des Benutzers nötig sind. Das Skript arbeitet die Schritte der Schritt-für-Schritt-Anleitung ab. Wird die Installation via Skript gewählt, so muss zumindest das install.sh Skript ausgeführt werden. Vorteil der Installation via Skript ist die immense Zeit- und Aufwandsersparnis.

### FHEM Setup via Skript

Die Installation via Skript bedingt das Ausführen des Hauptinstallationsskriptes. Ergänzende kleinere Skripte, die zusätzliche Funktionalität installieren sind bereitgestellt. Diese können nur genutzt werden, wenn das Hauptinstallationsskript zuvor ausgeführt worden ist. Zudem benötigen die ergänzenden Skripte händische Tätigkeiten, welche ausgeführt werden müssen, bevor die Skripte gestartet werden. Teilweise sind auch interaktive Skripte vorhanden, welche bedient werden müssen.

Nach dem Aufsetzen des Pis muss eine SSH Verbindung zum Pi aufgebaut werden. Die Skripte werden bei AWS gehostet und von dort bezogen. Die Skripte sind in einem Archiv und gezippt, somit komprimiert.

Das gezippte Archiv wird heruntergeladen:

Curl?  
wget https://s3.eu-central-1.amazonaws.com/www.fhemchumbucket.com/install\_fhem.tar.gz

#### Das Hauptinstallationsskript

### FHEM Setup manuell

Um das FHEM Setup herzustellen, musst du auf Basis der vorherigen Ausgangssituation weitermachen. Solltest du den PuTTY bereits geschlossen haben, so beginnst du bei Kapitel 1.1 mit dem Schritt 7. Solltest du soweit sein kannst du mit den folgenden Schritten weitermachen.

1. Befehl:

sudo wget -O- https://debian.fhem.de/archive.key | gpg --dearmor | sudo tee /usr/share/keyrings/debianfhemde-archive-keyring.gpg

1. Wechsel ins Verzeichnis “/etc/apt/“. Hierhin gelangst du mit dem cd Befehl ( „cd /etc/apt/“ --< Enter) ins Verzeichnis.
2. Öffne die Datei „sources.list“. Hierfür kannst du den Befehl „sudo nano sources.list“ nutzen.
   1. Füge in die geöffnete Datei

deb [signed-by=/usr/share/keyrings/debianfhemde-archive-keyring.gpg] https://debian.fhem.de/nightly/ / ein.

* 1. Speicher die Datei mit der Taste F3 à Enter
  2. Schließe die Datei mit Strg + x

1. Befehl: sudo apt update
2. Befehl: sudo apt install fhem (ggfs. zusätzliche Bestätigung mit „Y“ à Enter)

### Alternatives FHEM Setup

Bei der Installation wie in 1.2. beschrieben stellte sich beim Testen heraus, dass diese zeitweise unzuverlässig funktionierte. In diesem Fall ist ein alternativer Installationsweg möglich, der folglich beschrieben wird:

Auf dem RaspberryPi mit sudo-Rechten ausführen: (sudo –i)

apt -y install perl-base libdevice-serialport-perl libwww-perl libio-socket-ssl-perl libcgi-pm-perl libjson-perl sqlite3 libdbd-sqlite3-perl libtext-diff-perl libtimedate-perl libmail-imapclient-perl libgd-graph-perl libtext-csv-perl libxml-simple-perl liblist-moreutils-perl fonts-liberation libimage-librsvg-perl libgd-text-perl libsocket6-perl libio-socket-inet6-perl libmime-base64-perl libimage-info-perl libusb-1.0-0-dev libnet-server-perl libdate-manip-perl libhtml-treebuilder-xpath-perl libmojolicious-perl libxml-bare-perl libauthen-oath-perl libconvert-base32-perl libmodule-pluggable-perl libnet-bonjour-perl libcrypt-urandom-perl nodejs npm libnet-dbus-perl wget libfuse2 libffi-dev make gcc-avr avrdude avr-libc

Download des Pakets mit wget, danach Installation mit dpkg. Im Anschluss wird ein User namens fhem erstellt. Der Copy-Befehl kopiert eine Service Datei in die Systemumgebung um die Steuerung mit Systemctl zu ermöglich. Der Deamon lädt anschließend die Datei ein via dem reload.

wget <http://fhem.de/fhem-6.1.deb>

dpkg -i fhem-6.1.deb

useradd --system --home /opt/fhem --gid dialout --shell /bin/false fhem

cp /opt/fhem/contrib/init-scripts/fhem.service /etc/systemd/system/

systemctl daemon-reload

## FTUI Setup

Um das FTUI Setup herzustellen kannst du der Anleitung auf der Website <https://wiki.fhem.de/wiki/FHEM_Tablet_UI> folgen oder du folgst den unten beschriebenen Schritten.

1. Öffne die FHEM-Befehlszeile. Kopiere nacheinander in der vorgegebenen Reihenfolge die Befehle, füge sie in die FHEM-Befehlszeile ein und führe sie aus.
   1. Befehl: „update all <https://raw.githubusercontent.com/knowthelist/fhem-tablet-ui/master/controls_fhemtabletui.txt>“
   2. Befehl: define TABLETUI HTTPSRV ftui/ ./www/tablet/ Tablet-UI
   3. Befehl: attr WEB longpoll websocket“ (ggf. Beispieldatei mit dem Befehl: „sudo cp -a /opt/fhem/www/tablet/index-example.html /opt/fhem/www/tablet/index.html kopieren und auf dem Pi über PuTTy ausführen)
   4. Änderungen speichern mit „Save config“
2. Starte FHEM neu und füge den Befehl shutdown restart zur Ausführung in die FHEM-Befehlszeile ein.
3. Nach Neustart ist FTUI über den Link in der FHEM-Oberfläche im Browser oder über **Fehler! Linkreferenz ungültig.** erreichbar

## Sicherungskonzept

Sicherung des gesamten Pi in die Dropbox einrichten

Alle Befehle werden als root ausgeführt (zu Beginn einmal sudo -i)!!

1. apt-get install libfuse2
2. pip3 install dbxfs
3. Verzeichnis anlegen
   1. cd /mnt/
   2. mkdir mydropbox
4. Verzeichnis mounten
   1. dbxfs /mnt/mydropbox
   2. Link nach „Go to“ kopieren und im Browser öffnen
   3. Mit Dropbox anmelden
   4. Authorization code kopieren und im Terminal-Fenster einfügen
   5. Mit Enter bestätigen
5. Backup Script anlegen
   1. Verzeichnis anlegen
      1. mkdir /home/fhem/scripts/
      2. nano
   2. Script anlegen
      1. Einfügen:

#!/bin/bash

# Define dates for today and 28 days (4 weeks ago)

today=$(date +%Y-%m-%d)

date2delete=$(date --date="$today -28 day" +%Y-%m-%d)

# Echo the defined dates

echo "$today"

echo "$date2delete"

# Create dd backup

sudo dd if=/dev/mmcblk0p2 of=/mnt/mydropbox/DHBW-Pi/backup-$today

# Compress backup file

sudo tar -cpzf /mnt/mydropbox/DHBW-Pi/backup-$today.tar.gz --one-file-system /mnt/mydropbox/DHBW-Pi/backup-$today

# Remove uncompressed backup

sudo rm /mnt/mydropbox/DHBW-Pi/backup-$today

# Remove 28 day old backup

sudo rm /mnt/mydropbox/DHBW-Pi/backup-$date2delete.tar.gz

* + 1. Speichern mit „Strg + O“
    2. „backup.sh“ als Name eingeben und mit Enter bestätigen
    3. Mit Strg + X den Editor verlassen
  1. Script ausführbar machen chmod +x backup.sh
  2. Cronjob anlegen
     1. crontab -e
     2. Am Ende der Datei einfügen:

0 0 \* \* \* bash /home/fhem/scripts/backup.sh

* + 1. Mit Strg + O speichern
    2. Mit Strg + X verlassen

Nun wird jeden Tag um 0:00 Uhr ein Backup in der Dropbox erstellt, welches nach vier Wochen (28 Tagen) beim nächsten Backup wieder gelöscht wird.

## Konfiguration einer CA und HTTPS

Das Einrichten von Zertifikaten und einer Certification Authority (CA) ist durch den Ablauf der folgenden Zeilen zu erreichen. Die genutzten Programme agieren hierbei interaktiv, die getätigten Eingaben sind beispielhaft und können abgeändert werden. Eingaben des Benutzers sind farblich markiert. Gerade bei der Angabe von Daten beim Erstellen von Zertifikaten ist darauf zu achten valide Angaben zu machen, welche sich im Optimalfall nicht unterscheiden. Die Angaben wiederholen sich teilweise. Beispielweise Angaben sind ebenfalls markiert. Beim Generieren des Serverkeys ist die Angabe der IP des PIs wichtig!   
Diese kann identifiziert werden via “sudo hostname –I", die erste IP ist einzutragen. In der openssl.cnf sind Angaben abzuändern. Die Datei ist im Zielzustand abgebildet und so abzuändern. Anleitungen des FHEM-Wiki sind hier ausführlicher und aktueller als der Dargestellte Ablauf, die Links sind an entsprechender Stelle eingefügt.

https://wiki.fhem.de/wiki/FHEM\_mit\_HTTPS\_SSL-Zertifikat\_und\_eine\_eigene\_Zertifizierungsstelle

Verbinden mit dem PI via SSH.

pi@raspberrypi:~# sudo -i

root@raspberrypi:~/ca# apt install libio-socket-ssl-perl -y

root@raspberrypi:~/ca# apt install libwww-perl -y

root@raspberrypi:~# mkdir /root/ca

root@raspberrypi:~# cd /root/ca

root@raspberrypi:~/ca# openssl req -new -x509 -newkey rsa:2048 -keyout fhemSSL.pem -out cacert.pem -days 3650

Generating a RSA private key

...........................................+++++

.....................................................................................................................+++++

writing new private key to 'fhemSSL.pem'

Enter PEM pass phrase: #redbull

Verifying - Enter PEM pass phrase: #redbull

-----

You are about to be asked to enter information that will be incorporated

into your certificate request.

What you are about to enter is what is called a Distinguished Name or a DN.

There are quite a few fields but you can leave some blank

For some fields there will be a default value,

If you enter '.', the field will be left blank.

-----

Country Name (2 letter code) [AU]:DE

State or Province Name (full name) [Some-State]:Baden-Württemberg

Locality Name (eg, city) []:Stuttgart

Organization Name (eg, company) [Internet Widgits Pty Ltd]:DHBW

Organizational Unit Name (eg, section) []:WWI2019D

Common Name (e.g. server FQDN or YOUR name) []:Gruppe2

Email Address []:Gruppe2.WWI2019D@DHBW.de

root@raspberrypi:~/ca# ls -la

total 16

drwx------ 7 root root 4096 Jan 5 11:54 ..

-rw------- 1 root root 1854 Jan 5 11:56 fhemSSL.pem

-rw-r--r-- 1 root root 1489 Jan 5 11:57 cacert.pem

drwxr-xr-x 2 root root 4096 Jan 5 11:57 .

root@raspberrypi:~/ca# chmod 600 fhemSSL.pem

root@raspberrypi:~/ca# openssl genrsa -out serverkey.pem -aes128 2048

Generating RSA private key, 2048 bit long modulus (2 primes)

.............................................................................+++++

..........................................+++++

e is 65537 (0x010001)

Enter pass phrase for serverkey.pem: #password

Verifying - Enter pass phrase for serverkey.pem: #password

root@raspberrypi:~/ca# openssl rsa -in serverkey.pem -out serverkey.pem

Enter pass phrase for serverkey.pem: #jaja

root@raspberrypi:~/ca# openssl req -new -key serverkey.pem -out req.pem -nodes

You are about to be asked to enter information that will be incorporated

into your certificate request.

What you are about to enter is what is called a Distinguished Name or a DN.

There are quite a few fields but you can leave some blank

For some fields there will be a default value,

If you enter '.', the field will be left blank.

-----

Country Name (2 letter code) [AU]:DE

State or Province Name (full name) [Some-State]:Baden-Württemberg

Locality Name (eg, city) []:Stuttgart

Organization Name (eg, company) [Internet Widgits Pty Ltd]:DHBW

Organizational Unit Name (eg, section) []:WWI2019D

Common Name (e.g. server FQDN or YOUR name) []:192.168.178.81

Email Address []:admin@192.168.178.81

Please enter the following 'extra' attributes

to be sent with your certificate request

A challenge password []:

An optional company name []:

root@raspberrypi:~/ca# ls -la

total 24

drwx------ 7 root root 4096 Jan 5 11:54 ..

-rw------- 1 root root 1854 Jan 5 11:56 fhemSSL.pem

-rw-r--r-- 1 root root 1489 Jan 5 11:57 cacert.pem

-rw------- 1 root root 1679 Jan 5 12:05 serverkey.pem

-rw-r--r-- 1 root root 1082 Jan 5 12:11 req.pem

drwxr-xr-x 2 root root 4096 Jan 5 12:11 .

root@raspberrypi:~/ca# nano /etc/ssl/openssl.cnf

[ CA\_default ]

dir = /root/ca # Where everything is kept

certs = $dir/certs # Where the issued certs are kept

crl\_dir = $dir/crl # Where the issued crl are kept

database = $dir/index.txt # database index file.

#unique\_subject = no # Set to 'no' to allow creation of

# several certs with same subject.

new\_certs\_dir = $dir # default place for new certs.

certificate = $dir/cacert.pem # The CA certificate

serial = $dir/serial # The current serial number

crlnumber = $dir/crlnumber # the current crl number

# must be commented out to leave a V1 CRL

crl = $dir/crl.pem # The current CRL

private\_key = $dir/fhemSSL.pem # The private key

RANDFILE = $dir/.rand

x509\_extensions = usr\_cert # The extensions to add to the cert

# Comment out the following two lines for the "traditional"

# (and highly broken) format.

name\_opt = ca\_default # Subject Name options

cert\_opt = ca\_default # Certificate field options

# Extension copying option: use with caution.

# copy\_extensions = copy

# Extensions to add to a CRL. Note: Netscape communicator chokes on V2 CRLs

# so this is commented out by default to leave a V1 CRL.

# crlnumber must also be commented out to leave a V1 CRL.

# crl\_extensions = crl\_ext

default\_days = 3650 # how long to certify for

default\_crl\_days= 30 # how long before next CRL

default\_md = default # use public key default MD

preserve = no # keep passed DN ordering

# A few difference way of specifying how similar the request should look

# For type CA, the listed attributes must be the same, and the optional

# and supplied fields are just that :-)

policy = policy\_match

Hier Speichern via Strg + O und schließen mit Strg + X.

root@raspberrypi:~/ca# echo 01 > serial

root@raspberrypi:~/ca# touch index.txt

root@raspberrypi:~/ca# openssl ca -in req.pem -notext -out servercert.pem

Enter pass phrase for ./cakey.pem: #redbull

...

Certificate is to be certified until Jan 05 10:45:36 2032 GMT (3650 days)

Sign the certificate? [y/n]: y

1 out of 1 certificate requests certified, commit? [y/n] y

Write out database with 1 new entries

Data Base Updated

---------------------------------------------------------------------------------------------------

Hier kopieren wir eine Auswahl der generierten Dateien in ein Verzeichnis von FHEM unter /opt/fhem/ und passen die Zugriffsrechte an.

https://wiki.fhem.de/wiki/Raspberry\_Pi\_%26\_HTTPS

root@raspberrypi:~/ca# mkdir /opt/fhem/certs

root@raspberrypi:~/ca# cd /opt/fhem/certs

root@raspberrypi:~/ca# cp serverkey.pem /opt/fhem/certs

root@raspberrypi:~/ca# cp servercert.pem /opt/fhem/certs

root@raspberrypi:~/ca# sudo chown -R fhem:dialout /opt/fhem/certs/

---------------------------------------------------------------------------------------------------

Anschließend sind zwei Zeilen auf der Oberfläche von FHEM einzugeben, um HTTPS zu aktivieren:

In FHEMWEB:

attr WEB sslVersion TLSv12:!SSLv3

attr WEB HTTPS 1

Ab nun ist die Oberfläche mit https://... aufrufen. Manche Browser bereiten Probleme beim Aufruf von selbstsignierten Zertifikaten. In diesem Fall ist hier eine Handlungsempfehlung aufgeführt: <https://wiki.fhem.de/wiki/FHEM_mit_HTTPS_SSL-Zertifikat_und_eine_eigene_Zertifizierungsstelle#Einbinden_der_root_CA_in_die_Zertifizierungsstelle_des_Browsers> Die entsprechende Datei cacert.pm lässt sich via scp auf einen Windows-PC kopieren.

Hierfür: Auf dem PI:

Sudo cp /root/ca/cacert.pm /home/pi/

Sudo chown pi:pi /home/pi/cacert.pm

Auf einem Windows PC in der PowerShell oder CMD mit der IP des PIs( sudo hostname –I):

Scp [pi@IP://home/pi/cacert.pm](mailto:pi@IP://home/pi/cacert.pm) .

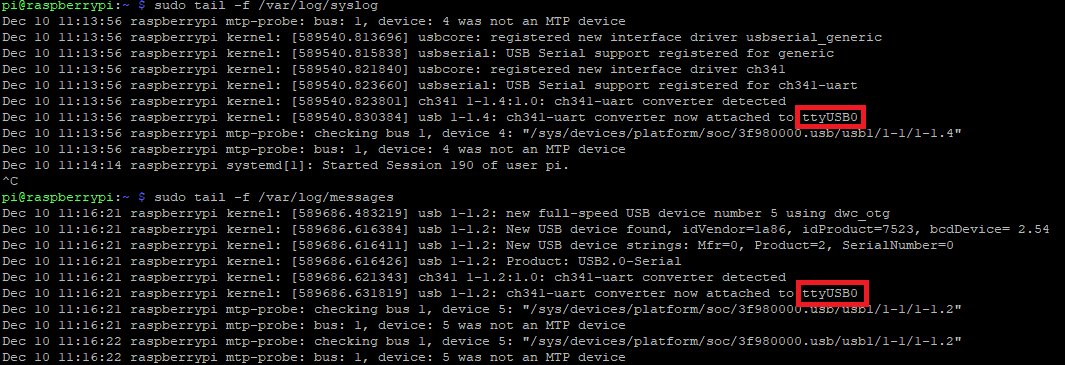
Die Datei wird nach Eingabe des Passworts des Users PI an die Stelle kopiert, an welcher sich der User in PowerShell/CMD momentan befindet. Wenn nicht verändert bei Start: C:/Users/<user>/

# Implementation in FHEM

## Funktion Weihnachtsbeleuchtung

### Flashen und einrichten der Funksteckdose

Auf dem Raspberry Pi:

1. Arduino in USB des Pi einstecken
2. Befehl sudo tail -f /var/log/syslog à STRG + C
3. Befehl sudo tail -f /var/log/message à STRG + C
4. Nach rot markierten Werten (sollte derselbe sein) suchen und merken  
   
5. Befehl wget <http://culfw.de/culfw-1.67.tar.gz>
6. Befehl tar xfv culfw-1.67.tar.gz
7. Befehl sudo apt-get install make gcc-avr avrdude avr-libc
8. Befehl cd culfw-1.67/Devices/nanoCUL/
9. Befehl make
10. Befehl make program

In FHEM:

1. Befehl define nanoCUL CUL /dev/ttyUSB0@38400 1234 (den roten Teil durch das Gemerkte aus Schritt 4 ersetzen)
2. Befehl attr nanoCUL rfmode HomeMatic
3. Befehl attr nanoCUL room 241\_Weihnachtsbeleuchtung
4. Befehl define funkLamp IT FF0000000F FF F0 (den roten Teil entsprechend des Drehschalters auf der Rückseite der Funksteckdose und <https://wiki.fhem.de/wiki/Intertechno_Code_Berechnung#Original_Intertechno_System> ersetzen. Auf der Funksteckdose ist hinten ein Drehschalter, mit dem die Frequenz ausgewählt werden kann. Diese kann frei gewählt werden. Diese wird durch eine Kombination aus Buchstaben + Zahl dargestellt und haben einen entsprechenden vierstelligen Code in der Tabelle haben. Die ersten vier Stellen des roten Teils werden durch die Code-Entsprechung des Buchstabens und die zweiten vier Stellen durch die Code-Entsprechung der Zahl ersetzt.)
5. Befehl attr funkLamp IODev nanoCUL
6. Befehl attr funkLamp model itswitch
7. Befehl attr funkLamp room 241\_Weihnachtsbeleuchtung

Funksteckdose in eine Steckdose stecken und innerhalb von 5 Sekunden in FHEM beim Gerät „funkLamp“ auf „on“ klicken (siehe Screenshot)

### Manueller Betrieb

Mit der Einrichtung der Funksteckdose und des CUL ist die manuelle Bedienung, bis auf die Verbindung zum Frontend, bereits gegeben.

### Betriebsmodus Feste Betriebszeit

1. Erstellen eines Erweiterungsmoduls „at“, um die Funklampe um 7 Uhr auszuschalten:

define a\_funkLamp\_morning\_off at \*07:00:00 set funkLamp off

1. Raum zuweisen:

attr a\_funkLamp\_morning\_off room 241\_Weihnachtsbeleuchtung

1. Erstellen eines weiteren Erweiterungsmoduls „at“, um die Funklampe um 17 Uhr einzuschalten:

define a\_funkLamp\_evening\_on at \*17:00:00 set funkLamp on

1. Raum zuweisen:

attr a\_funkLamp\_evening\_on room 241\_Weihnachtsbeleuchtung

### Betriebsmodus Sonnenaufgang und Sonnenuntergang

1. Koordinaten in FHEM hinterlegen (für die Daten von Sonnenauf- und Sonnenuntergang wird der Standort benötigt)

Unter Unsorted das Attribut global auswählen

Dort die Attribute latitude und longitude aus den gewählten Koordinaten hinzufügen

attr global latitude 48.773486

attr global longitude 9.170658



1. Einen DOIF-Typen erstellen (“doif\_funk\_sonne”) und die Sonnenaufgangs- bzw. Sonnenuntergangsfunktionalität einbetten, damit die Lampe bei Sonnenaufgang aus- und bei Sonnenuntergang eingeschaltet wird.

Define doif\_funk\_sonne doif ([{sunset("REAL")}])(set funkLamp on)DOELSEIF ([{sunrise("REAL")}])(set funkLamp off)

attr doif\_funk\_sonne room 241\_Weihnachtsbeleuchtung

attr doif\_funk\_sonne disable 1

### Betriebsmodus Sonnenaufgang und Sonnenuntergang mit Uhrzeiten

1. Einen DOIF-Typen erstellen (“doif\_funk\_szeit”), damit die Lampe bei Sonnenaufgang aus- und bei Sonnenuntergang eingeschaltet wird. Die Lampe geht jedoch spätestens um 7:30 Uhr aus.

define doif\_funk\_szeit DOIF ([{sunset("REAL")}])(set funkLamp on) DOELSEIF ([{sunrise\_abs("REAL",0,"00:00","07:30")}])(set funkLamp off)”

attr doif\_funk\_szeit room 241\_Weihnachtsbeleuchtung

attr doif\_funk\_szeit disable 1

### Betriebsmodus Anwesenheitserkennung

Einrichtung der FritzBox in FHEM:

1. IP-Adresse der FritzBox herausfinden (am besten über die Weboberfläche der FritzBox, häufig 192.168.178.1)
2. Skript einfügen
   * 1. In FHEM links auf “Edit files”
     2. Öffnen von “myUtilsTemplate.pm”
     3. Inhalt ersetzen durch den Inhalt von 99\_myUtils.txt, welches unter <https://github.com/doenisf/HomeAutomationProjektGruppe2> zu finden ist in dem UtilFiles Ordner
     4. Im Textfeld neben “Save as” “99\_myUtils.pm” eintragen und mit Klick auf “Save as” speichern
3. In FHEM:  
   define FritzBox FRITZBOX <IP>
   * 1. Bei gewissen FritzBoxen sind zusätzliche Schritte notwendig. Tests lassen die Vermutung zu, dass Provider wie Unitymedia Funktionalität blockieren. Sollte das Modul eine Meldung aufzeigen wie etwa “Keine Session ID” sind folgende Eingaben zu tätigen. In der FritzBox muss ein User angelegt werden mit Nutzernamen und Password.

attr FritzBox boxUser <username>

set FritzBox password <password>

Ggf. Kann es auch notwendig sein das Attribut allowTr064Command auf 1 zu setzen:

attr FritzBox allowTR064Command 1

1. Raum zuweisen:

attr FritzBox room 99\_Fritz

1. Herausfinden der MAC-Adresse des iPhones (am iPhone)
   1. Einstellungen -> "Allgemein" -> "Info" -> "WLAN-Adresse"
   2. Unter IOS “WLAN-Adresse” = “MAC-Adresse”
   3. MAC-Adresse merken (Format xx:xx:xx:xx:xx:xx)
2. In FHEM:  
   define iPhone PRESENCE function {checkAllFritzMACpresent("<MAC-Adresse>")} 01 01  
     
   <MAC-Adresse> ersetzen durch die Adresse aus Schritt 4b
3. Die Schritte 4. und 5. können für beliebig viele weitere iPhones wiederholt werden, es muss nur in Schritt 5. der Name des Geräts (hier “iPhone”) geändert werden.
4. Skript zur Anwesenheitserkennung (und Subroutine für 2.1.7 Betriebsmodusauswahl) einfügen

7.1. In FHEM links auf “Edit files”

* + 1. “myUtilsTemplate.pm” öffnen
    2. Inhalt ersetzen durch den Inhalt von 99\_myUtils\_241.txt, welches unter <https://github.com/doenisf/HomeAutomationProjektGruppe2> zu finden ist in dem UtilFiles Ordner
    3. Im Textfeld neben “Save as” “99\_myUtils\_241.pm” eintragen und mit Klick auf “Save as” speichern

1. In FHEM:  
   define a\_funkLamp\_anwesenheitsCheck at +\*00:00:01 {modusAnwesenheit("funkLamp", 17, 7)}

Raum zuweisen:

attr a\_funkLamp\_anwesenheitsCheck room 241\_Weihnachtsbeleuchtung

### Betriebsmodusauswahl

Diese Funktion verwendet eine Subroutine aus 99\_myUtils\_241.pm. Das Einfügen dieser Funktion wird unter 2.1.6 beschrieben.

Allgemeines:

1. Einen DUMMY-Typen erstellen, der für die Übermittlung des Betriebsmodus vom Frontend ins Backend fungiert.

define funkLampModus dummy

attr funkLampModus room 241\_Weihnachtsbeleuchtung

attr funkLampModus setList switch:festeZeit,Sonne,SZeit,Anwesenheit,Manuell

1. Einen DOIF-Typen (“doif\_funk\_general”) erstellen, um die verschiedenen Betriebsmodi anzutriggern.

define doif\_funk\_general DOIF ([funkLampModus:"festeZeit"])(attr doif\_funk\_sonne disable 1)(attr doif\_funk\_szeit disable 1)(attr a\_funkLamp\_anwesenheitsCheck disable 1)(attr a\_funkLamp\_evening\_on disable 0)(attr a\_funkLamp\_morning\_off disable 0)({checkTime("funkLamp", 17, 7)})DOELSEIF ([funkLampModus:"Sonne"])(attr doif\_funk\_sonne disable 0)(attr doif\_funk\_szeit disable 1)(attr a\_funkLamp\_anwesenheitsCheck disable 1)(attr a\_funkLamp\_evening\_on disable 1)(attr a\_funkLamp\_morning\_off disable 1)({checkTime("funkLamp", sunset\_abs(), sunrise\_abs())})DOELSEIF ([funkLampModus:"SZeit"])(attr doif\_funk\_sonne disable 1)(attr doif\_funk\_szeit disable 0)(attr a\_funkLamp\_anwesenheitsCheck disable 1)(attr a\_funkLamp\_evening\_on disable 1)(attr a\_funkLamp\_morning\_off disable 1)({checkTime("funkLamp", sunset\_abs(), sunrise\_abs("REAL",0,"00:00","07:30"))})DOELSEIF ([funkLampModus:"Anwesenheit"])(attr doif\_funk\_sonne disable 1)(attr doif\_funk\_szeit disable 1)(attr a\_funkLamp\_anwesenheitsCheck disable 0)(attr a\_funkLamp\_evening\_on disable 1)(attr a\_funkLamp\_morning\_off disable 1)DOELSEIF([funkLampModus:"Manuell"])(attr doif\_funk\_sonne disable 1)(attr doif\_funk\_szeit disable 1)(attr a\_funkLamp\_anwesenheitsCheck disable 1)(attr a\_funkLamp\_evening\_on disable 1)(attr a\_funkLamp\_morning\_off disable 1)

attr doif\_funk\_general room 241\_Weihnachtsbeleuchtung

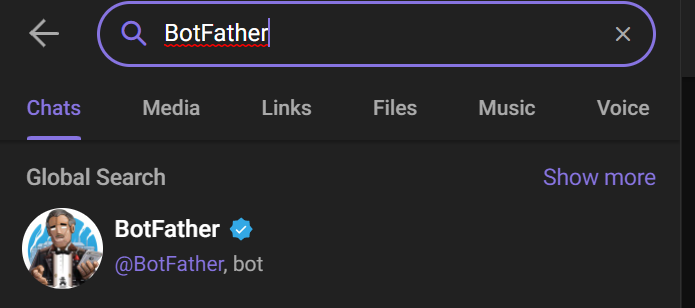
## Funktion Sturmwarnung

### Einrichten eines Bots zum Senden von Telegram Nachrichten über FHEM

Installation von Telegram auf seinem Smartphone. Einrichten eines Telegram Bots, Telegrambot in FHEM einrichten, um darüber an bestimmte Personennachrichten zu senden.

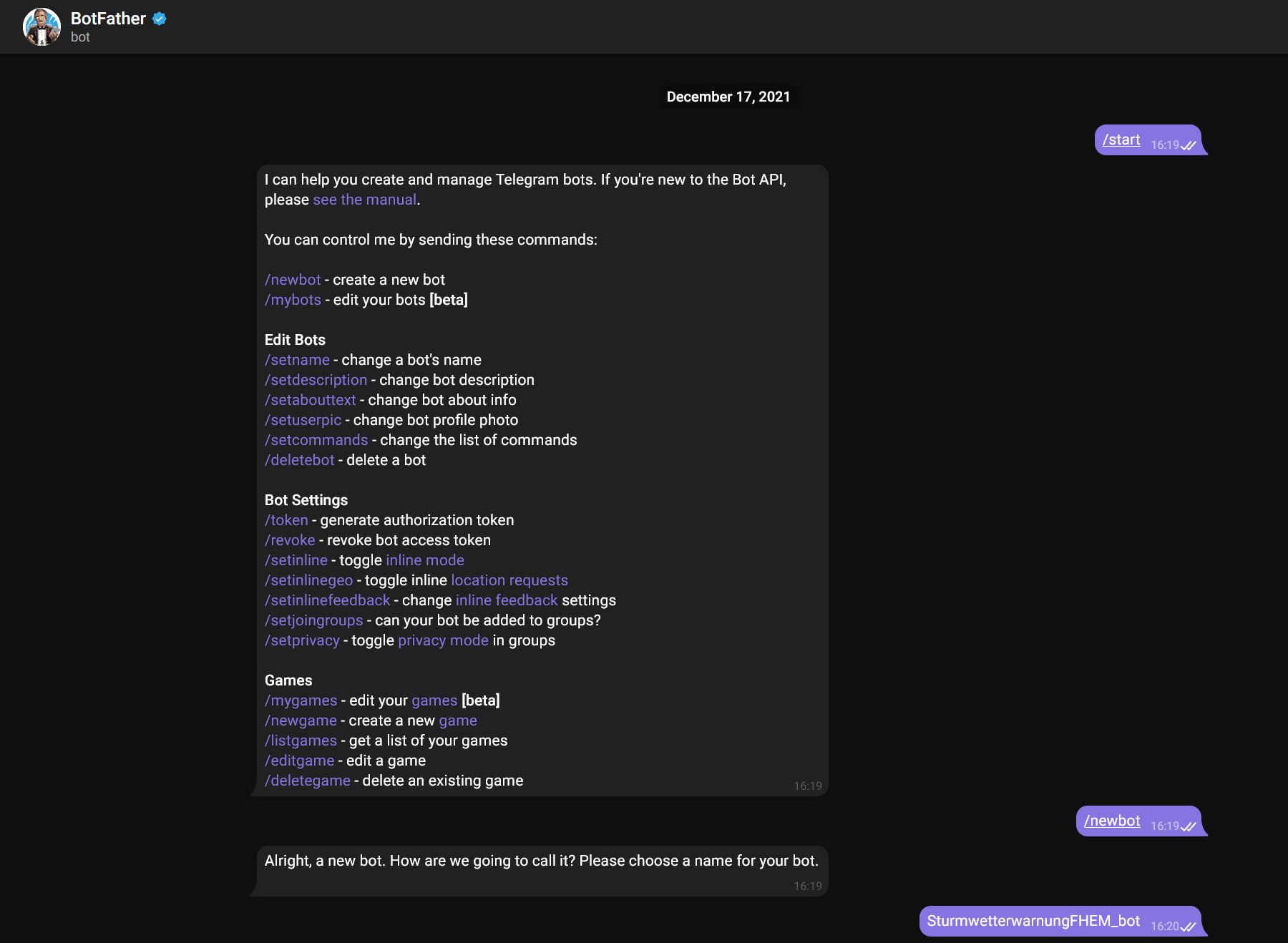
1. Installation von Telegram auf deinem Smartphone, dafür kann die Telegram App einfach über den Apple Store oder Google Play Store heruntergeladen werden.

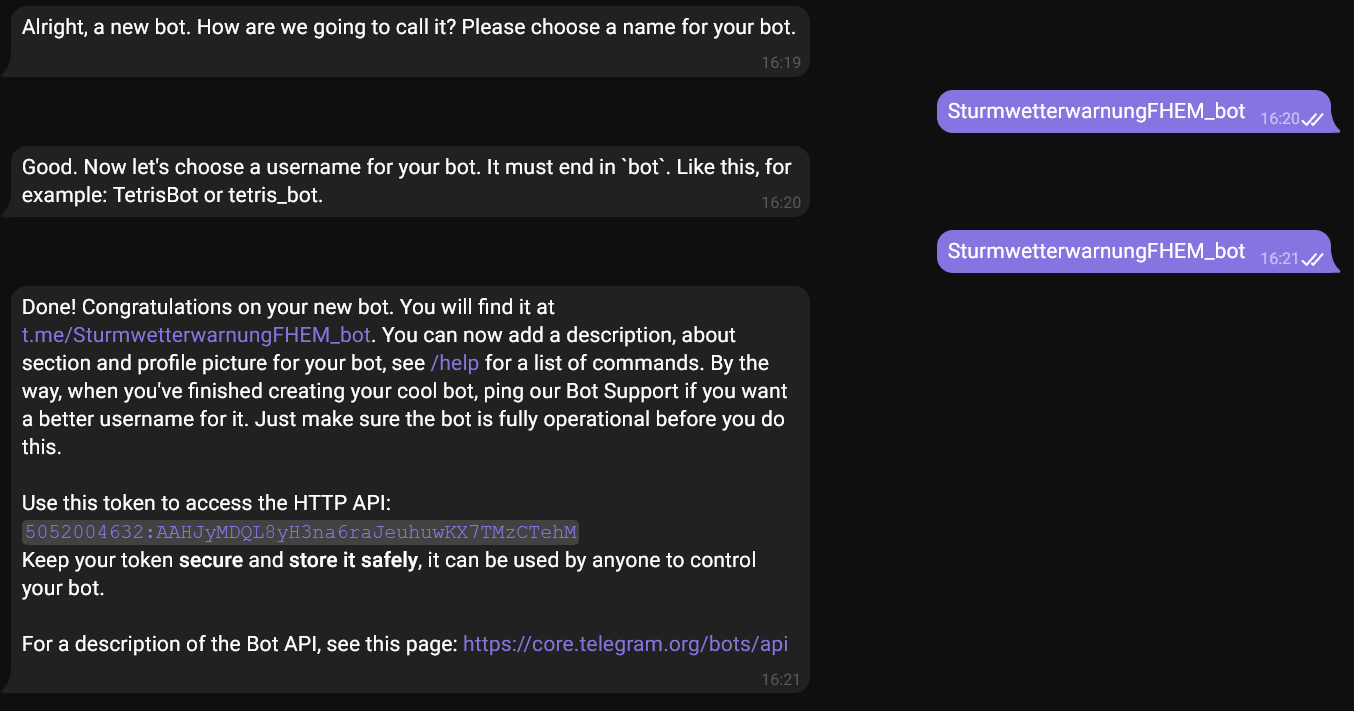


1. Erstellen eines eigenen Telegram Bots, dafür ist es nötig, sich zuerst in Telegram anzumelden und sich als Nutzer zu registrieren. Falls Sie keinen neuen Bot erstellen möchten, können Sie mit dem bereits durch das Projekt erstellen Bot weitermachen (Schritt 4).
2. Telegramm auf dem Handy öffnen oder Telegram Web verwenden.
   1. Anschreiben des BotFather, dafür einfach in Telegram nach “BotFather” suchen
   2. Starten des BotFathers und Erstellen eines neuen Bots indem man dem BotFather folgende Befehle im Chat schickt:

/start

/newbot



* 1. Nun vergeben sie den Namen ihres Bots, achten sie dabei darauf, dieser Name darf noch nicht vergeben sein und sollte auf “bot” enden. Sie bekommen dann einen HTTP API Token, dieser ist für die weitere Einrichtung in FHEM sehr wichtig. Bitte merken sie sich den Token und geben sie diesen nicht weiter! 

1. Einrichtung Telegram Bot in FHEM

Botname: SturmwetterwarnungFHEM\_bot

Token HTTP API: 5052004632:AAHJyMDQL8yH3na6raJeuhuwKX7TMzCTehM

Dazu geben sie den Befehl in FHEM ein:

define <name> TelegramBot <token>

Mit unserem Bot wäre das:

define SturmwetterwarnungFHEM\_bot TelegramBot 5086529595:AAGoVMH8AIwA\_RCBONvtUI6hft\_lYySYmRY

1. Einstellung des Attributes pollingTimeout und dem Raum “Sturmwetterwarnung” zuordnen.:

Polling Attribut:

attr <name> pollingTimeout 120

Raum Attribut:

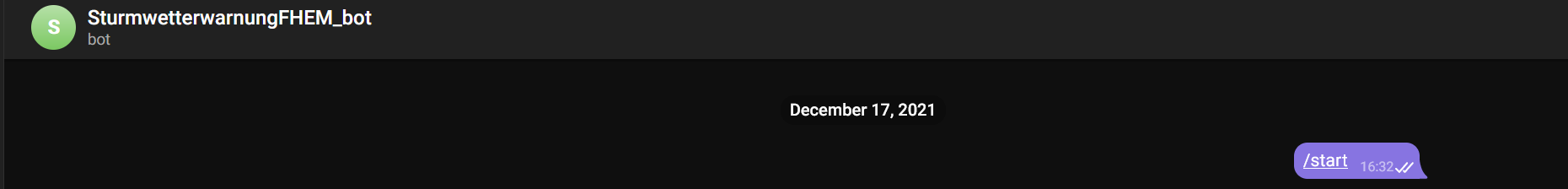
attr <name> room 242\_Sturmwetterwarnung

Mit unserem Bot wäre das:

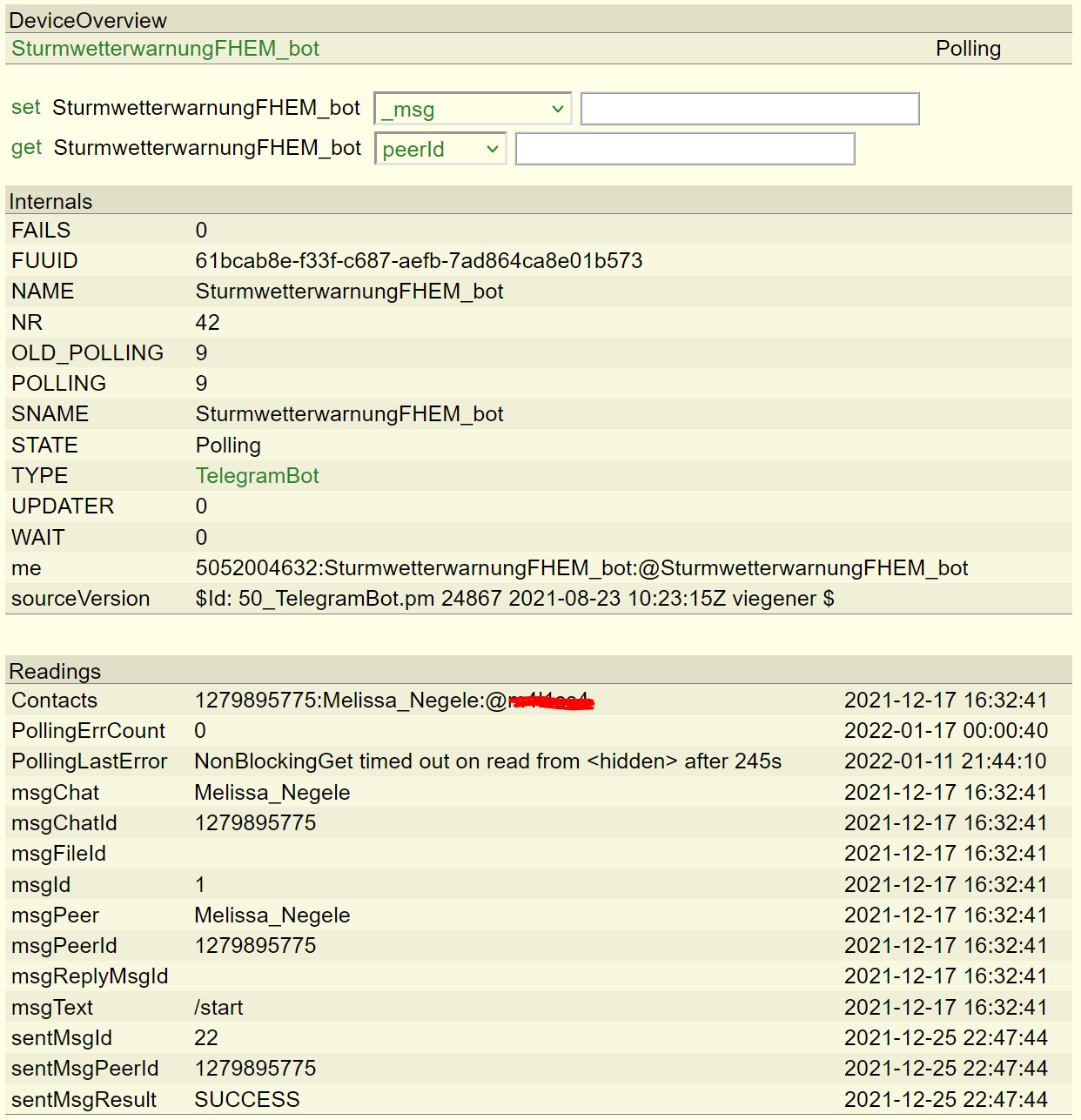
attr SturmwetterwarnungFHEM\_bot pollingTimeout 120

attr SturmwetterwarnungFHEM\_bot room 242\_Sturmwetterwarnung

1. Um den Bot zu starten und damit dieser ihnen eine Nachricht schreiben kann, müssen sie diesen über Telegram eine Nachricht senden.



Nach dem Starten ist der Telegram bot in FHEM aktiv und sie sehen nun in den Readings ihren Nutzernamen



1. Fügen sie das Attribut defaultPeer hinzu und hinterlegen sie dort ihren Benutzernamen (Nur Benutzer die unter defaultPeer sind können Nachrichten über FHEM empfangen).

attr <Botname> defaultPeer @<Ihr Benutzername>

Bei uns wäre dies:

attr SturmwetterwarnungFHEM\_bot defaultPeer @XXXXX

1. Um neue Benutzer über die FHEM UI Oberfläche hinzuzufügen, müssen die neuen Benutzer zuerst den TelegramBot anschreiben, danach können sie über die Oberfläche hinzugefügt werden. Sollten sie einen anderen Bot Namen verwenden als den vorgegebenen, müssen sie gegeben falls für die Anzeige der Benutzernamen auch die Frontend Datei ändern und dort den vergebenen Bot Namen durch den von ihnen neu angelegten ersetzen.



### Sensordaten auslesen

Verbindung zwischen FHEM und Sensor Hochdorf herstellen

Weitere Beschreibung:

Anlegen eines neuen Devices und auslesen der Windgeschwindigkeit sowie der Böen-Stärke von der Seite: <https://measurements.mobile-alerts.eu/Home/SensorsOverview?phoneid=285142992122>

1. Neues Device “SensorDaten” erzeugen und Verbindung zwischen FHEM und Sensor Hochdorf über das HTTPMOD Modul herstellen. Eine Prüfung der Verbindung findet im Intervall 60 sek. statt.

define SensorDaten HTTPMOD <https://measurements.mobile->alerts.eu/Home/SensorsOverview?phoneid=285142992122 60

1. HTTPMOD ruft den Sensor Hochdorf über einen eigenen, leider stark veralteten Browser auf, dieser kann moderne Webseiten nicht mehr anzeigen. Deswegen ist es nötig zu sagen, HTTPMOD soll die Webseite über den neueren Mozilla Browser aufrufen.

attr SensorDaten reading01Regex requestHeader1  
attr SensorDaten requestHeader1 User-Agent: Mozilla/5.0 (Windows NT 6.0)

1. Auslesen der Windgeschwindigkeit vom Sensor Hochdorf über das HTTPMOD Modul

attr SensorDaten reading02Name Windgeschwindigkeit

attr SensorDaten reading02Regex Windgeschwindigkeit[\S\s\r\n]\*?>([\-\d\d\,\d]+)

attr SensorDaten reading02OExpr join ".", (split /,/, $val)

1. Auslesen der Böen-Stärke vom Sensor Hochdorf über das HTTPMOD Modul

attr SensorDaten reading03Name Böe

attr SensorDaten reading03Regex B&#246[\S\s\r\n]\*?>([\-\d\d\,\d]+)

attr SensorDaten reading03OExpr join ".", (split /,/, $val)

1. Auslesen der Windrichtung vom Sensor Hochdorf über das HTTPMOD Modul

attr SensorDaten reading04Name Windrichtung

attr SensorDaten reading04Regex Windrichtung[\S\s\r\n]\*?>(?|(Norde)|(Nordnordost)|(Nordost)|(Ostnordost)|(Ost)|(Ostsüdost)|(Südost)|(Südsüdost)|(Süd)|(Südsüdwest)|(Südwest)|(Westsüdwest)|(West)|(Westnordwest)|(Nordwest)|(Nordnordwest))

1. Dem SensorDaten einem Raum zuweisen

attr SensorDaten room 242\_Sturmwetterwarnung

1. Optional: Icon in FHEM hinzufügen

attr SensorDaten icon weather\_wind\_speed@SkyBlue

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

### Anlegen der Dummy-Devices für die Eingabe der Schwellenwerte und für den Schalter der Sturmwetterwarnung

Um die Schwellwerte eingeben und die Sturmwarnung ein- und ausschalten zu können, werden Dummy-Devices angelegt. Bis auf ein Attribut beim Schalter für die Sturmwarnung müssen keine weiteren Attribute oder Readings angelegt werden.

Anlegen des Dummy-Devices für den Schwellwert der Windböe; Dazu muss folgender Befehl in FHEM ausgeführt werden:

Anlegen des Dummy-Devices für den Schwellwert der Windgeschwindigkeit und der Böen-Stärke; Dazu muss folgender Befehl in FHEM ausgeführt werden:

define Schwellwert\_Windstaerke dummy

attr Schwellwert\_Windstaerke room 242\_Sturmwetterwarnung

set Schwellwert\_Windstaerke 200

define Schwellwert\_Windboe dummy

attr Schwellwert\_Windboe room 242\_Sturmwetterwarnung

set Schwellwert\_Windboe 200

Anlegen des Dummy-Devices für den Schalter der Sturmwarnung. Dazu müssen folgende Befehle nacheinander in FHEM ausgeführt werden:

define SturmwarnungSchalter dummy

attr SturmwarnungSchalter room 242\_Sturmwetterwarnung

attr SturmwarnungSchalter webCmd on:off

define [MerkerSchalterBoe](http://192.168.0.214:8083/fhem?detail=MerkerSchalterBoe) dummy

attr [MerkerSchalterBoe](http://192.168.0.214:8083/fhem?detail=MerkerSchalterBoe) room 242\_Sturmwetterwarnung

attr [MerkerSchalterBoe](http://192.168.0.214:8083/fhem?detail=MerkerSchalterBoe) webCmd on:off

setreading MerkerSchalterBoe state off

define [MerkerSchalterWindstaerke](http://192.168.0.214:8083/fhem?detail=MerkerSchalterWindstaerke) dummy

attr [MerkerSchalterWindstaerke](http://192.168.0.214:8083/fhem?detail=MerkerSchalterWindstaerke) room 242\_Sturmwetterwarnung

attr [MerkerSchalterWindstaerke](http://192.168.0.214:8083/fhem?detail=MerkerSchalterWindstaerke) webCmd on:off

setreading MerkerSchalterWindstaerke state off

### Hinzufügen von Do-If-Abfragen

Aufpassen, hier muss der bei einem neu angelegten Bot dieser in den DO-If-Abfragen ersetzt werden durch den Namen, unter dem sie ihren Bot in FHEM angelegt haben. (Rot markiert)

1. Anlegen eines Devices vom Typ DOIF für das Überschreiten des Windstärke Schwellenwerts:

define UeberschreitenSchwellenwertWindstaerke DOIF ([Schwellwert\_Windstaerke:state] < [SensorDaten:Windgeschwindigkeit] and [SturmwarnungSchalter:state] eq "on" and [MerkerSchalterWindstaerke:state] eq "off") (set SturmwetterwarnungFHEM\_bot message Achtung! Windgeschwindigkeit wurde überschritten. Aktuelle Windgeschwindigkeit: [SensorDaten:Windgeschwindigkeit] km/h)(set MerkerSchalterWindstaerke on)

attr UeberschreitenSchwellenwertWindstaerke room 242\_Sturmwetterwarnung

1. Anlegen eines Devices vom Typ DOIF für das Überschreiten des Böen Schwellenwerts:

define UeberschreitenSchwellenwertBoe DOIF ([Schwellwert\_Windboe:state] < [SensorDaten:Böe] and [SturmwarnungSchalter:state] eq "on" and [MerkerSchalterBoe:state] eq "off") (set SturmwetterwarnungFHEM\_bot message Achtung! Böengeschwindigkeit wurde überschritten. Aktuelle Böengeschwindigkeit: [SensorDaten:Böe] km/h) (set MerkerSchalterBoe on)

attr UeberschreitenSchwellenwertBoe room 242\_Sturmwetterwarnung

1. Anlegen eines Devices vom Typ DOIF für das Entwarnen (Böe):

define UnterschreitenSchwellenwertBoe DOIF ([Schwellwert\_Windboe:state] > ([SensorDaten:Böe]\*1.3) and [SturmwarnungSchalter:state] eq "on" and [MerkerSchalterBoe:state] eq "on") (set SturmwetterwarnungFHEM\_bot message Die Wetterlage hat sich beruhigt. Aktuelle Böengeschwindigkeit: [SensorDaten:Böe] km/h)(set MerkerSchalterBoe off)

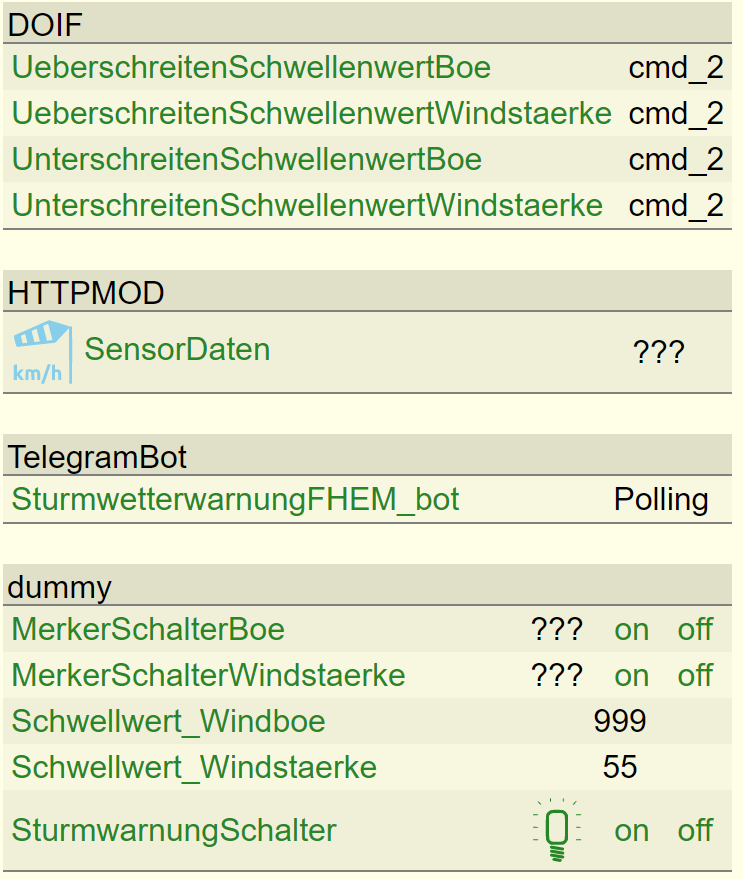
attr UnterschreitenSchwellenwertBoe room 242\_Sturmwetterwarnung

1. Anlegen eines Devices vom Typ DOIF für das Entwarnen (Windstärke):

define UnterschreitenSchwellenwertWindstaerke DOIF ([Schwellwert\_Windstaerke:state] > ([SensorDaten:Windgeschwindigkeit]\*1.3) and [SturmwarnungSchalter:state] eq "on" and [MerkerSchalterWindstaerke:state] eq "on") (set SturmwetterwarnungFHEM\_bot message Die Wetterlage hat sich beruhigt. Aktuelle Windgeschwindigkeit: [SensorDaten:Windgeschwindigkeit] km/h)(set MerkerSchalterWindstaerke off)

attr UnterschreitenSchwellenwertWindstaerke room 242\_Sturmwetterwarnung

Anbei ein Screenshot aller angelegten Devices im Raum 242\_Sturmwetterwarnung;



## Funktion Photovoltaik/Balkonkraftwerk

Anleitung Einrichtung Shelly

Shelly mit WiFi verbinden

1. Shelly Cloud App runterladen
2. Shelly Cloud Account erstellen und/oder anmelden
3. Raum in Shelly Cloud App erstellen
4. Smartphone mit WiFi-Access Point des Shelly verbinden
   1. Access Point startet entweder beim Verbinden des Shelly mit dem Strom oder alternativ

durch 10 Sekunden gedrückt halten des Buttons auf der Rückseite des Shelly (Shelly ist in der Unterputzdose)

1. In Shelly Cloud App neues Device hinzufügen
2. Im Schritt Include new devices in your wi-fi network Name (SSID) sowie Passwort des Netzwerks eingeben, in das der Shelly eingebunden werden soll
3. In Dropdown-Liste „Shelly25“ auswählen
4. Name für Shelly vergeben und mit Shelly Cloud verbinden

Shelly in FHEM einbinden:

INFO: der zweite Kanal des Shelly ist auf Grund begrenzter Hardware doppelt belegt. Zum Testen des kombinierten Betriebes von Wasserboiler und Infrarotheizung kann eines der beiden Geräte auf den anderen Kanal umgestellt werden ( attr <gerätename> defchannel 0 ). Im tatsächlichen Gebrauch würden mehrere Shellys zum Einsatz kommen. Diese werden auf die gleiche Art eingerichtet, bis auf die entsprechend andere IP-Adresse im Befehl.

1. Erster Channel (Balkonkraftwerk) einbinden
   1. Befehl in FHEM-Kommandozeile define balkonkraftwerk Shelly <Shelly-IP> (IP des Shelly kann über den Router des Netzwerks rausgefunden werden)
   2. Befehl attr balkonkraftwerk model shelly2.5
   3. Befehl attr balkonkraftwerk mode relay
   4. Befehl attr balkonkraftwerk defchannel 0
   5. Befehl attr balkonkraftwerk room 243\_Balkonkraftwerk
2. Zweiten Channel (Infrarotheizung) einbinden
   1. Befehl in FHEM-Kommandozeile define infrarotheizung Shelly <Shelly-IP> IP des Shelly kann über den Router des Netzwerks rausgefunden werden)

Befehl attr infrarotheizung model shelly2.5

Befehl attr infrarotheizung mode relay

Befehl attr infrarotheizung defchannel 1

Befehl attr infrarotheizung room 243\_Balkonkraftwerk

1. Zweiten Channel erneut (Wasserboiler) einbinden
   1. Befehl in FHEM-Kommandozeile define wasserBoiler Shelly <Shelly-IP> (IP des Shelly kann über den Router des Netzwerks rausgefunden werden)
   2. Befehl attr wasserBoiler model shelly2.5
   3. Befehl attr wasserBoiler mode relay
   4. Befehl attr wasserBoiler defchannel 1
   5. Befehl attr wasserBoiler room 243\_Wasserboiler

### Einrichten einer DB zum Speichern von Daten

Die Daten werden in einer SQLite DB gespeichert, welche noch installiert werden muss.   
Dafür mit dem PI verbinden und auf der Shell folgende Abhängigkeiten installieren:

sudo apt install -y sqlite3 libdbi-perl libdbd-sqlite3-perl

Es lässt sich nun eine DB erstellen und aufrufen via:

sudo sqlite3 /opt/fhem/fhem.db

Anschließend findet man sich in der sqlite Console wieder, in diese folgende Zeilen einfügen und mit Enter abschicken, jeweils bis zum Semikolon:

CREATE TABLE history (TIMESTAMP TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP, DEVICE varchar(64), TYPE varchar(64), EVENT varchar(512), READING varchar(64), VALUE varchar(128), UNIT varchar(32));

CREATE TABLE current (TIMESTAMP TIMESTAMP, DEVICE varchar(64), TYPE varchar(64), EVENT varchar(512), READING varchar(64), VALUE varchar(128), UNIT varchar(32));

CREATE INDEX Search\_Idx ON `history` (DEVICE, READING, TIMESTAMP);

Die sqlite Console schliest sich mit dem Befehl:

.exit

Ändern der Rechte und Zugriffsmodifikationen:

sudo chown fhem /opt/fhem/fhem.db

sudo chmod 600 /opt/fhem/fhem.db

Anlegen einer Datei, über die die DB Konfiguriert wird:

sudo nano /opt/fhem/db.conf

In nano den folgenden Textblock einfügen, anschließend mit Strg + O speichern und mit Strg + X schließen:

%dbconfig= (

connection => "SQLite:dbname=/opt/fhem/fhem.db",

user => "",

password => ""

);

Zum Abschluss der Arbeiten auf dem PI wurd FHEM neu gestartet:

sudo systemctl restart fhem

In FHEM Werden nun noch Readings angelegt und die DB vorstellig gemacht. Dazu wie gewohnt in die Befehlszeile von FHEM eingeben:

setreading balkonkraftwerk sendData [balkonkraftwerk:power\_0]

define logdb DbLog /opt/fhem/db.conf balkonkraftwerk:sendData.\*

define timerSetReading at +\*00:01:00 setreading balkonkraftwerk sendData [balkonkraftwerk:power\_0]

Speichern via Save.

Funktionalität, welche Daten akkumuliert und in die DB speichert wird durch DBrep bereitgestellt. Das Modul wird definiert für Wochen und Monate:

define repBalkonkraftwerkDaily DbRep logdb

attr repBalkonkraftwerkDaily reading sendData

attr repBalkonkraftwerkDaily aggregation day

attr repBalkonkraftwerkDaily device balkonkraftwerk

attr repBalkonkraftwerkDaily timeDiffToNow d:1 FullDay

define repDaily at \*23:59:59 set repBalkonkraftwerkDaily sumValue writeToDBSingle

define repBalkonkraftwerkMonthly DbRep logdb

attr repBalkonkraftwerkMonthly reading sendData

attr repBalkonkraftwerkMonthly aggregation month

attr repBalkonkraftwerkMonthly device balkonkraftwerk

attr repBalkonkraftwerkMonthly timeDiffToNow d:31 FullDay

define repMonthly at \*23:59:59 {if ((strftime "%d",localtime time+86400) eq "01") {set repBalkonkraftwerkMonthly sumValue writeToDBSingle}}

Anschließend Speichern.

### Einfaches Heizen

1. In das Device “balkonkraftwerk” Readings hinzufügen.

setreading balkonkraftwerk energieSchwellwert 300

1. Anlegen eines DOIFs, das in einem regelmäßigen Zeitabstand prüft, ob die erzeugte Energie den Schwellenwert überschreitet bzw. unterschreitet und die Infrarotheizung demnach ein- oder ausschaltet

Dazu folgende Befehle ausführen:

define doif\_UeberschreitenEnergieSchwellenwert DOIF ([+([doif\_UeberschreitenEnergieSchwellenwert:intervall]\*60)] and [balkonkraftwerk:power\_0] > [balkonkraftwerk:energieSchwellwert] and [infrarotheizung:relay\_1] eq “off”) (set infrarotheizung on) DOELSEIF ([+([doif\_UeberschreitenEnergieSchwellenwert:intervall]\*60)] and [balkonkraftwerk:power\_0] < [balkonkraftwerk:energieSchwellwert] and [infrarotheizung:relay\_1] eq “on”) (set infrarotheizung off)

attr doif\_UeberschreitenEnergieSchwellenwert room 243\_Balkonkraftwerk

attr doif\_UeberschreitenEnergieSchwellenwert disable 1

“etreading doif\_UeberschreitenEnergieSchwellenwert intervall 5

attr doif\_UeberschreitenEnergieSchwellenwert disable 0

attr doif\_UeberschreitenEnergieSchwellenwert do always

attr doif\_UeberschreitenEnergieSchwellenwert disable 1

1. Um den Überhitzungsschutz umzusetzen, werden zwei DOIFs benötigt.

Der erste schaltet die Infrarotheizung aus, wenn diese eine bestimmte Zeit ununterbrochen läuft.

Dazu folgende Befehle ausführen:

define doif\_HeizungEinfach DOIF ([infrarotheizung:relay\_1] eq "on")(set infrarotheizung off)(attr doif\_UeberschreitenEnergieSchwellenwert disable 1)(setreading doif\_HeizungEinfach disable 1)

attr doif\_HeizungEinfach room 243\_Balkonkraftwerk

setreading doif\_HeizungEinfach disable 1

setreading doif\_HeizungEinfach on\_time 120

attr doif\_HeizungEinfach do resetwait

attr doif\_HeizungEinfach wait [doif\_HeizungEinfach:on\_time]\*60

attr doif\_HeizungEinfach disable 1

Das zweite DOIF stellt sicher, dass die Heizung für eine bestimmte Zeit ausgeschaltet bleibt.

Dazu folgende Befehle ausführen:

define doif\_disableTimerEinfach DOIF ([doif\_HeizungEinfach:disable] eq "1")(attr doif\_UeberschreitenEnergieSchwellenwert disable 0)(setreading doif\_HeizungEinfach disable 0)

attr doif\_disableTimerEinfach room 243\_Balkonkraftwerk”

attr doif\_disableTimerEinfach disable 1

setreading doif\_disableTimerEinfach disabled {AttrVal(“doif\_UeberschreitenEnergieSchwellenWert”,”disable”,2)}

-> wenn 2 zurückgegeben wird, existiert das Attribut nicht

setreading doif\_disableTimerEinfach off\_time 15

attr doif\_disableTimerEinfach do resetwait

attr doif\_disableTimerEinfach wait [doif\_disableTimerEinfach:off\_time]\*60

### Optimiertes Heizen

#### Sensordaten auslesen

Verbindung zwischen FHEM und Sensor Hobbyraum herstellen

Weitere Beschreibung:

Anlegen eines neuen Device’s und auslesen der Temperatur sowie des Zeitstempels von der Seite: <https://measurements.mobile-alerts.eu/Home/MeasurementDetails?deviceid=035DD29D94C5&vendorid=244DD836-16DE-465E-B265-B3F1596A26D4&appbundle=de.synertronixx.remotemonitor>

* + - 1. Neues Device “Temperatursensor” erzeugen und Verbindung zwischen FHEM und Sensor Hobbyraum Messung über das HTTPMOD Modul herstellen. Eine Prüfung der Verbindung findet im Intervall 420sec. Statt (Der Sensor aktualisiert seine Zeit nur alle 7min, deswegen ist eine mehrfache Prüfung nicht notwendig).

define Temperatursensor HTTPMOD <https://measurements.mobile-alerts.eu/Home/MeasurementDetails?deviceid=035DD29D94C5&vendorid=244DD836-16DE-465E-B265-B3F1596A26D4&appbundle=de.synertronixx.remotemonitor> 420

* + - 1. HTTPMOD ruft den Sensor Hochdorf über einen eigenen, leider stark veralteten Browser auf, dieser kann moderne Webseiten nicht mehr anzeigen. Deswegen ist es nötig zu sagen, HTTPMOD soll die Webseite über den neueren Mozilla Browser aufrufen.

attr Temperatursensor reading01Regex requestHeader1

attr Temperatursensor requestHeader1 User-Agent: Mozilla/5.0 (Windows NT 6.0)

* + - 1. Cookies auf der Webseite aktivieren

attr Temperatursensor enableCookies 1

* + - 1. Auslesen der Temperatur vom Sensor Hobbyraum Messung über das HTTPMOD Modul

attr Temperatursensor reading02Name temperature

attr Temperatursensor reading02Expr join ".", (split /,/, $val)

attr Temperatursensor reading02Regex ([^>]\d\*[,]\d)

1. Auslesen des Zeitstempels vom Sensor Hobbyraum Messung über das HTTPMOD Modul

attr Temperatursensor reading03Name timestamp

attr Temperatursensor reading03Regex ([^>]\d+[.]\d+[.]\d+\s\d+[:]\d+[:]\d+)

1. Auslesen der Luftfeuchtigkeit vom Sensor Hobbyraum Messung über das HTTPMOD Modul

attr Temperatursensor reading04Name humidity

attr Temperatursensor reading04Regex ([^>]\d\*[,]\d)[%]

attr Temperatursensor reading04Expr join ".", (split /,/, $val)

1. Auslesen bei Änderungen aktivieren

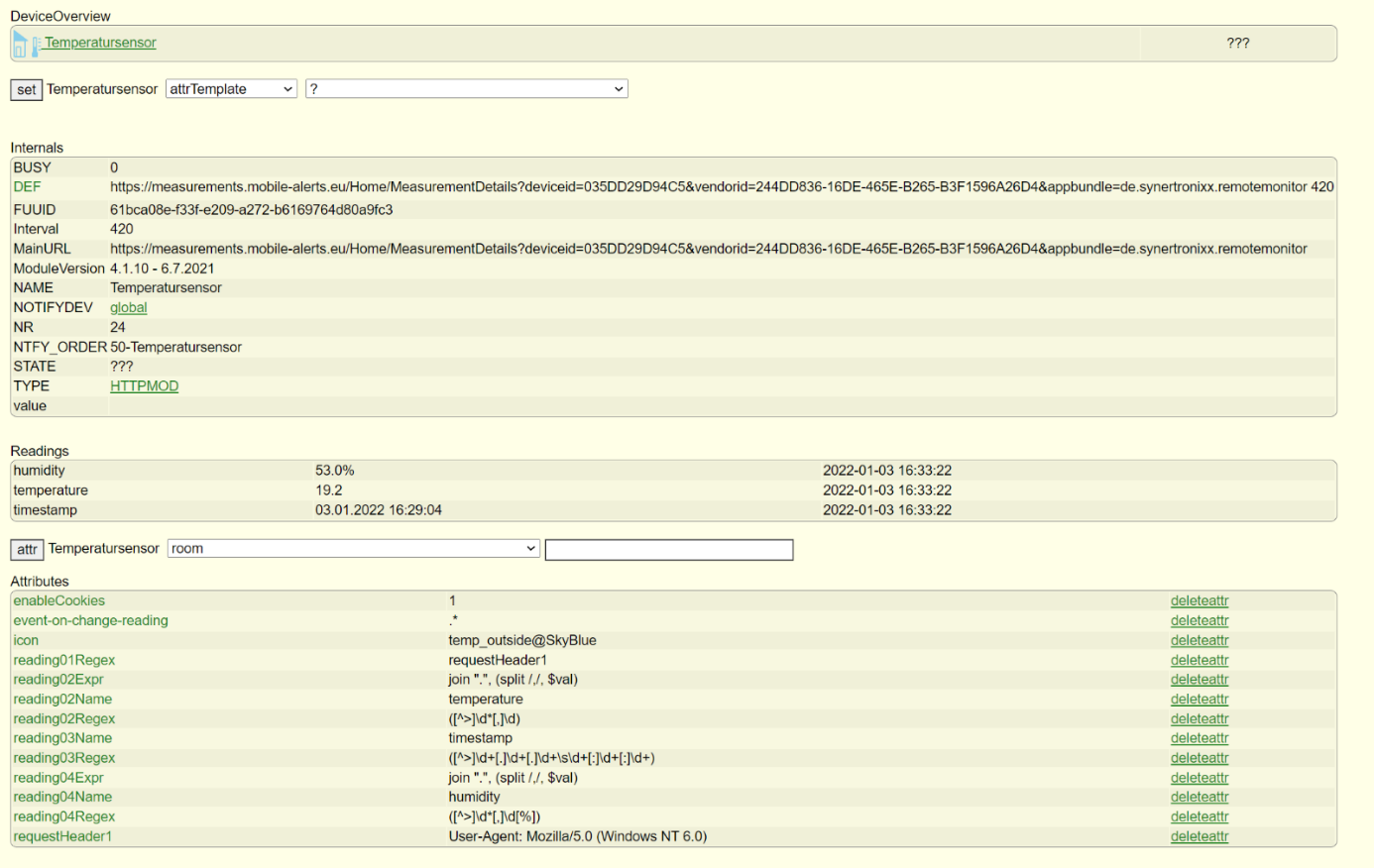
attr Temperatursensor event-on-change-reading .\*

1. Icon in FHEM hinzufügen

attr Temperatursensor icon temp\_outside@SkyBlue

1. Raum hinzufügen

attr Temperatursensor room 243\_Balkonkraftwerk



#### Logik Optimiertes Heizen

Anlegen von Dummy-Devices für die Eingabe der Schwellwerte und für den Schalter für das Optimierte Heizen

Die Schwellwerte für das Aus- und Einschalten kommen dabei einmal vom Temperatursensor und von der FHEM UI Übersicht.

Anlegen der Dummy-Devices für das Ein und Ausschalten des Optimierten Heizen; Dazu müssen folgende Befehle nacheinander in FHEM ausgeführt werden:

1. define OptimiertesHeizenSchalter dummy

attr OptimiertesHeizenSchalter room 243\_Balkonkraftwerk

attr OptimiertesHeizenSchalter webCmd on:off

1. define Schwellwert\_Temperatur\_Ein dummy

attr Schwellwert\_Temperatur\_Ein room 243\_Balkonkraftwerk

set Schwellwert\_Temperatur\_Ein 19

1. define Schwellwert\_Temperatur\_Aus dummy

attr Schwellwert\_Temperatur\_Aus room 243\_Balkonkraftwerk

set Schwellwert\_Temperatur\_Aus 21

1. define Schwellwert\_Luftfeuchtigkeit\_Ein dummy

attr Schwellwert\_Luftfeuchtigkeit\_Ein room 243\_Balkonkraftwerk

set Schwellwert\_Luftfeuchtigkeit\_Ein 75

1. define Schwellwert\_Luftfeuchtigkeit\_Aus dummy

attr Schwellwert\_Luftfeuchtigkeit\_Aus room 243\_Balkonkraftwerk

set Schwellwert\_Luftfeuchtigkeit\_Aus 70

#### Hinzufügen von Do-If-Abfragen

Anlegen eines Devices vom Typ DOIF für das Einschalten der Infrarotheizung nach unterschreiten der Schwellwert Temperatur oder überschreiten des Schwellwertes Luftfeuchtigkeit

define EinschaltenOptimiertesHeizen DOIF (([Schwellwert\_Temperatur\_Ein:state] > [Temperatursensor:temperature] or [Schwellwert\_Luftfeuchtigkeit\_Ein:state] < [Temperatursensor:humidity]) and [OptimiertesHeizenSchalter:state] eq "on" and [+([doif\_UeberschreitenEnergieSchwellenwert:intervall]\*60)] and [balkonkraftwerk:power\_0] > [balkonkraftwerk:energieSchwellwert] and [infrarotheizung:relay\_1] eq "off")(set infrarotheizung on)

attr EinschaltenOptimiertesHeizen room 243\_Balkonkraftwerk

attr EinschaltenOptimiertesHeizen do always

setreading EinschaltenOptimiertesHeizen EnergyTimeRange 60

attr EinschaltenOptimiertesHeizen disable 1

Anlegen eines Devices vom Typ DOIF für das Ausschalten der Infrarotheizung nach überschreiten des Schwellwert Temperatur oder unterschreiten des Schwellwertes Luftfeuchtigkeit

define AusschaltenOptimiertesHeizen DOIF ([Schwellwert\_Temperatur\_Aus:state] < [Temperatursensor:temperature] and [Schwellwert\_Luftfeuchtigkeit\_Aus:state] > [Temperatursensor:humidity] and [OptimiertesHeizenSchalter:state] eq "on" and [+([doif\_UeberschreitenEnergieSchwellenwert:intervall]\*60)]) (set infrarotheizung off)

attr AusschaltenOptimiertesHeizen room 243\_Balkonkraftwerk

attr AusschaltenOptimiertesHeizen do always

setreading AusschaltenOptimiertesHeizen EnergyTimeRange 60

attr AusschaltenOptimiertesHeizen disable 1

Um den Überhitzungsschutz umzusetzen, werden zwei DOIFs benötigt.

Der erste schaltet die Infrarotheizung aus, wenn diese eine bestimmte Zeit ununterbrochen läuft.

Dazu folgende Befehle ausführen:

define doif\_HeizungOptimierend DOIF ([infrarotheizung:relay\_1] eq "on")(set infrarotheizung off)(attr EinschaltenOptimiertesHeizen disable 1)(setreading doif\_HeizungOptimierend disable 1)

attr doif\_HeizungOptimierend room 243\_Balkonkraftwerk

setreading doif\_HeizungOptimierend disable 1

attr doif\_HeizungOptimierend do resetwait

attr doif\_HeizungOptimierend wait [doif\_HeizungEinfach:on\_time]\*60

attr doif\_HeizungOptimierend disable 1

Das zweite DOIF stellt sicher, dass die Heizung für eine bestimmte Zeit ausgeschaltet bleibt.

Dazu folgende Befehle ausführen:

define doif\_disableTimerOptimierend DOIF ([doif\_HeizungOptimierend:disable] eq "1")(attr EinschaltenOptimiertesHeizen disable 0)(setreading doif\_HeizungOptimierend disable 0)

attr doif\_disableTimerOptimierend room 243\_Balkonkraftwerk”

attr doif\_disableTimerOptimierend disable 1”

setreading doif\_disableTimerOptimierend disabled {AttrVal(“doif\_UeberschreitenEnergieSchwellenWert”,”disable”,2)}

-> wenn 2 zurückgegeben wird, existiert das Attribut nicht

attr doif\_disableTimerOptimierend do resetwait

attr doif\_disableTimerOptimierend wait [doif\_disableTimerEinfach:off\_time]\*60

### Betriebsmodullogik Heizen

1. Erstellen eines dummy zur Kommunikation des ausgewählten Betriebsmodus vom Frontend ins Backend.

define heizenModus dummy

attr heizenModus room 243\_Balkonkraftwerk

attr heizenModus setList switch:einfach,optimierend,aus

1. Erstellen eines DOIF zur Steuerung der Betriebsmodi.

define doif\_heizenModus DOIF ([heizenModus:"einfach"]) (attr doif\_UeberschreitenEnergieSchwellenwert disable 0) (attr doif\_HeizungEinfach disable 0) (attr doif\_disableTimerEinfach disable 0) (attr doif\_HeizungOptimierend disable 1) (attr doif\_disableTimerOptimierend disable 1) (attr EinschaltenOptimiertesHeizen disable 1) (attr AusschaltenOptimiertesHeizen disable 1) (set OptimiertesHeizenSchalter off) DOELSEIF ([heizenModus:"optimierend"]) (attr EinschaltenOptimiertesHeizen disable 0) (attr AusschaltenOptimiertesHeizen disable 0) (attr doif\_HeizungEinfach disable 1) (attr doif\_disableTimerEinfach disable 1) (attr doif\_HeizungOptimierend disable 0) (attr doif\_disableTimerOptimierend disable 0) (set OptimiertesHeizenSchalter on) (attr doif\_UeberschreitenEnergieSchwellenwert disable 1) DOELSEIF ([heizenModus:"aus"]) (attr doif\_UeberschreitenEnergieSchwellenwert disable 1) (attr EinschaltenOptimiertesHeizen disable 1) (attr AusschaltenOptimiertesHeizen disable 1) (attr doif\_HeizungEinfach disable 1)(attr doif\_disableTimerEinfach disable 1) (attr doif\_HeizungOptimierend disable 1) (attr doif\_disableTimerOptimierend disable 1) (set infrarotheizung off) (set OptimiertesHeizenSchalter off)

attr doif\_heizenModus room 243\_Balkonkraftwerk

### Warmwasserboiler

#### Manueller Betrieb

Mit der Einrichtung des Shelly-Device wasserBoiler ist das manuelle Steuern bis auf die Anbindung ans Frontend bereits gewährleistet.

#### Vorgegebener Zeitraum

1. Skript einfügen
   1. In FHEM links auf “Edit files”
   2. Öffnen von “myUtilsTemplate.pm”
   3. Inhalt ersetzen durch den Inhalt von 99\_myUtils\_243.txt, welches unter <https://github.com/doenisf/HomeAutomationProjektGruppe2> zu finden ist in dem UtilFiles Ordner
   4. Öffne Putty auf deinem Rechner (falls du es nicht mehr offen hast, siehe 1.1)
   5. In dem Terminal den Befehl: „sudo cpan Time::Piece“ ausführen und anschließend mit yes bestätigen
   6. Im Textfeld neben “Save as” “99\_myUtils\_243.pm” eintragen und mit Klick auf “Save as” speichern

Define a\_boiler\_zeit\_on at \*00:00:10 set wasserBoiler on

Define a\_boiler\_zeit\_off at \*00:00:10 set wasserBoiler off

Setreading a\_boiler\_zeit\_on datum <dd-mm-yyyy>

à Datum am Ende muss in der Zukunft liegen

Setreading a\_boiler\_zeit\_off datum <dd-mm-yyyy>

à Datum am Ende muss in der Zukunft liegen

Modify a\_boiler\_zeit\_on \*{parseDate("a\_boiler\_zeit\_on")} set wasserBoiler on

attr a\_boiler\_zeit\_on room 243\_Wasserboiler

Modify a\_boiler\_zeit\_off \*{parseDate("a\_boiler\_zeit\_off")} set wasserBoiler off

attr a\_boiler\_zeit\_off room 243\_Wasserboiler

define doif\_update\_a\_boiler\_zeit\_on\_and\_off DOIF ([+60]) (modify a\_boiler\_zeit\_off \*{parseDate("a\_boiler\_zeit\_off")} set wasserBoiler off)(modify a\_boiler\_zeit\_on \*{parseDate("a\_boiler\_zeit\_on")} set wasserBoiler on

attr doif\_update\_a\_boiler\_zeit\_on\_and\_of do always

attr doif\_update\_a\_boiler\_zeit\_on\_and\_of room 243\_Wasserboiler

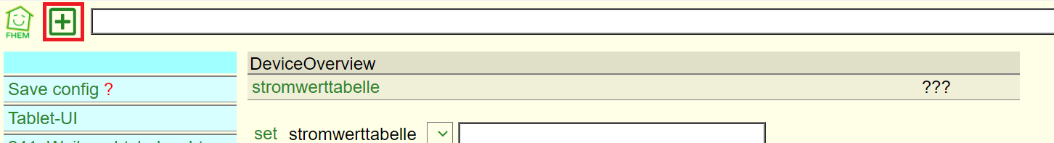
#### Stromwerttabelle

1. Dummy einrichten zum Speichern der Werte
   1. In FHEM:

define stromwerttabelle dummy

attr stromwerttabelle room 243\_Wasserboiler

* 1. Readings einrichten
     1. In FHEM oben links auf das “+”-Icon



* + 1. Folgendes einfügen

setreading stromwerttabelle t1 0

setreading stromwerttabelle t1\_on 00:00

setreading stromwerttabelle t1\_off 00:00

setreading stromwerttabelle t2 0

setreading stromwerttabelle t2\_on 00:00

setreading stromwerttabelle t2\_off 00:00

setreading stromwerttabelle t3 0

setreading stromwerttabelle t3\_on 00:00

setreading stromwerttabelle t3\_off 00:00

setreading stromwerttabelle t4 0

setreading stromwerttabelle t4\_on 00:00

setreading stromwerttabelle t4\_off 00:00

setreading stromwerttabelle t5 0

setreading stromwerttabelle t5\_on 00:00

setreading stromwerttabelle t5\_off 00:00

setreading stromwerttabelle t6 0

setreading stromwerttabelle t6\_on 00:00

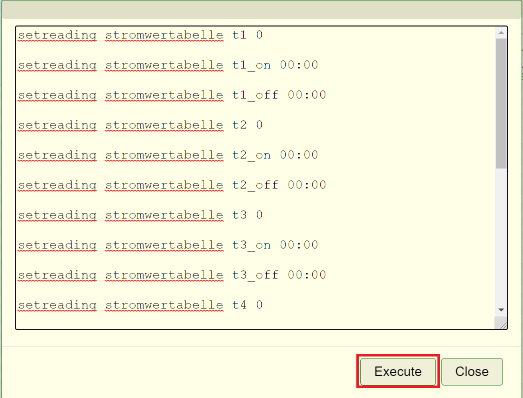
setreading stromwerttabelle t6\_off 00:00

setreading stromwerttabelle we 0

setreading stromwerttabelle we\_on 00:00

setreading stromwerttabelle we\_off 00:00

* + 1. Auf “Execute” drücken



1. In FHEM:  
   define a\_boiler\_stromwert at +\*00:00:10 {modusStromwerttabelle()} attr a\_boiler\_stromwert room 243\_Wasserboiler

#### Schwellwert

Define doif\_boiler\_schwellwert DOIF ([+([doif\_boiler\_schwellwert:intervall]\*60)] and([balkonkraftwerk:power\_0] > [wasserBoiler:schwellwert]))(set wasserBoiler on)DOELSEIF ([+([doif\_boiler\_schwellwert:intervall]\*60)] and([balkonkraftwerk:power\_0] < [wasserBoiler:schwellwert]))(set wasserBoiler off)

Setreading wasserBoiler schwellwet 500 > Wert am Ende ist egal

attr doif\_boiler\_schwellwert room 243\_Wasserboiler

#### Unterstützende Funktion

Define doif\_wasserboiler\_general DOIF ([wasserBoilerModus:"manuell"])(attr a\_boiler\_zeit\_on disable 1)(attr a\_boiler\_zeit\_off disable 1)(attr doif\_boiler\_schwellwert disable 1)(attr a\_boiler\_stromwert disable 1)DOELSEIF ([wasserBoilerModus:"festeZeit"])(attr a\_boiler\_zeit\_on disable 0)(attr a\_boiler\_zeit\_off disable 0)(attr doif\_boiler\_schwellwert disable 1)(attr a\_boiler\_stromwert disable 1)DOELSEIF ([wasserBoilerModus:"stromwert"])(attr a\_boiler\_zeit\_on disable 1)(attr a\_boiler\_zeit\_off disable 1)(attr doif\_boiler\_schwellwert disable 1)(attr a\_boiler\_stromwert disable 0)DOELSEIF ([wasserBoilerModus:"schwellwert"])(attr a\_boiler\_zeit\_on disable 1)(attr a\_boiler\_zeit\_off disable 1)(attr doif\_boiler\_schwellwert disable 0)(attr a\_boiler\_stromwert disable 1)

Define wasserBoilerModus dummy

attr wasserBoilerModus room 243\_Wasserboiler

Attr wasserBoilerModus setList switch: manuell festeZeit stromwert schwell

attr doif\_wasserboiler\_general room 243\_Wasserboiler

### Kombination zwischen Warmwasserboiler und Infrarotheizung

Logik zur Kombination zwischen Infrarotheizung und Warmwasserboiler.

define doif\_kombinationHeizungBoiler01 DOIF ([balkonkraftwerk:power\_0] < [infrarotheizung:power\_1] and [wasserBoilerModus] eq "stromwert") (set infrarotheizung off) ( set wasserBoiler off) DOELSEIF ([balkonkraftwerk:power\_0] < [infrarotheizung:power\_1]) (set infrarotheizung off)

attr doif\_kombinationHeizungBoiler01 room 243\_Balkonkraftwerk

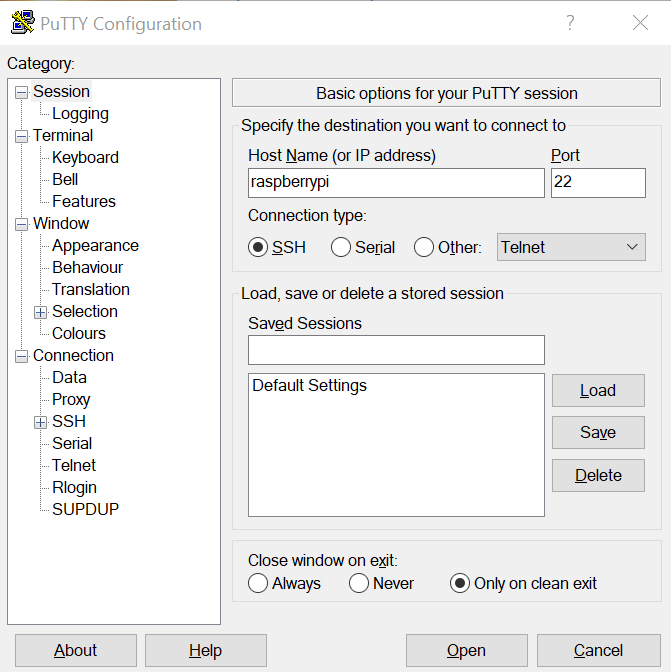
define doif\_kombinationHeizungBoiler02 DOIF ([balkonkraftwerk:power\_0] > [infrarotheizung:power\_1] and [balkonkraftwerk:power\_0] < ([infrarotheizung:power\_1] + [wasserBoiler:schwellwert]) and [wasserBoilerModus] eq "stromwert") (set wasserBoiler off)

attr doif\_kombinationHeizungBoiler02 room 243\_Balkonkraftwerk

# FTUI erstellen

Um die FTUI Oberflächen einrichten zu können ist es maßgeblich, dass die ein Kapitel 1. genannten Schritte vollständig durchgeführt wurden. (Siehe Setup Raspberry Pi).

1. Öffne Putty auf deinem Rechner.
2. Stelle eine Verbindung zu deinem Raspberrypi her, indem du in dem Feld Host Name den Host Namen „raspberrypi“ oder die IP-Adresse des Raspberrypi‘s eingibst.



Drücke auf „Open“ und die Verbindung wird hergestellt.

1. Gibt den Benutzernamen „pi“ und das von dir vergebene Passwort bei der Passwortabfrage ein.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

1. Navigiere zu dem Ordner, in dem die Oberflächen gespeichert sind. Gebe dafür den folgenden Befehl ein:

cd /opt/fhem/www/

1. Lösche den angelegten Ordner namens „tablet“ mit folgendem Befehl:

sudo rm -r tablet

1. Klone nun das von uns für dich erstellte Git-Repository indem du folgenden Befehl ausführst:

sudo git clone <https://github.com/tabascoel/tablet.git>

Nachdem du alle diese Schritte durchgeführt hast, kannst du Putty beenden und die Oberfläche wie nachfolgend beschrieben testen.

1. Nun öffne deine FHEM Oberfläche und starte FHEM neu, indem du folgenden Befehl in die FHEM Kommandozeile eingibst.

Shutdown restart

1. Nach Neustart sind alle FTUI-Oberflächen über den Link in der FHEM-Oberfläche namens „Tablet UI“ oder über **Fehler! Linkreferenz ungültig.** erreichbar.

Alle HTML, CSS und JavaScript Dokumente können auch unter dem nachfolgenden Link gefunden werden:

<https://github.com/tabascoel/tablet>

# HTML-Seiten Konfiguration

Durch das Klonen des Git Repositories sind die HTML Seiten eingerichtet. Achtung, wenn die Devices anders benannt wurden als in der Schritt-für-Schritt Anleitung vorgegeben, müssen die HTML-Dateien angepasst werden. Hierfür muss in der entsprechenden Datei das „data-device“ des entsprechendes Devices angepasst werden.



Abbildung 1: Frontend Beispiel Sturmwetterwarnung

Am Beispiel des „SturmwarnungSchalter“ zusehen, wenn dies im Backend geändert wird, muss dies auch im Frontend angepasst werden. Es muss der Name des Device im Frontend eingetragen werden.

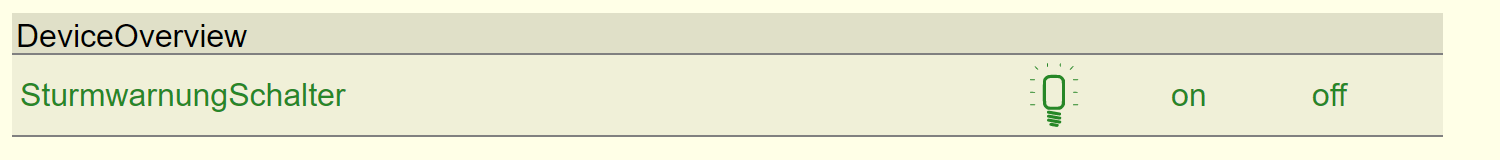


Abbildung 2: Backend Beispiel der Sturmwetterwarnung

# Optionale Aufgabe: Müllplan

Beim Müll Plan ist zu beachten, dass das Frontend auf die Readings vom Backend abgestimmt sind. Das heißt, wenn ein anderer Müll Plan verwendet wird, als der Müll Plan, der in der Schritt-für-Schritt Anleitung enthalten ist, müssen Anpassungen erfolgen.

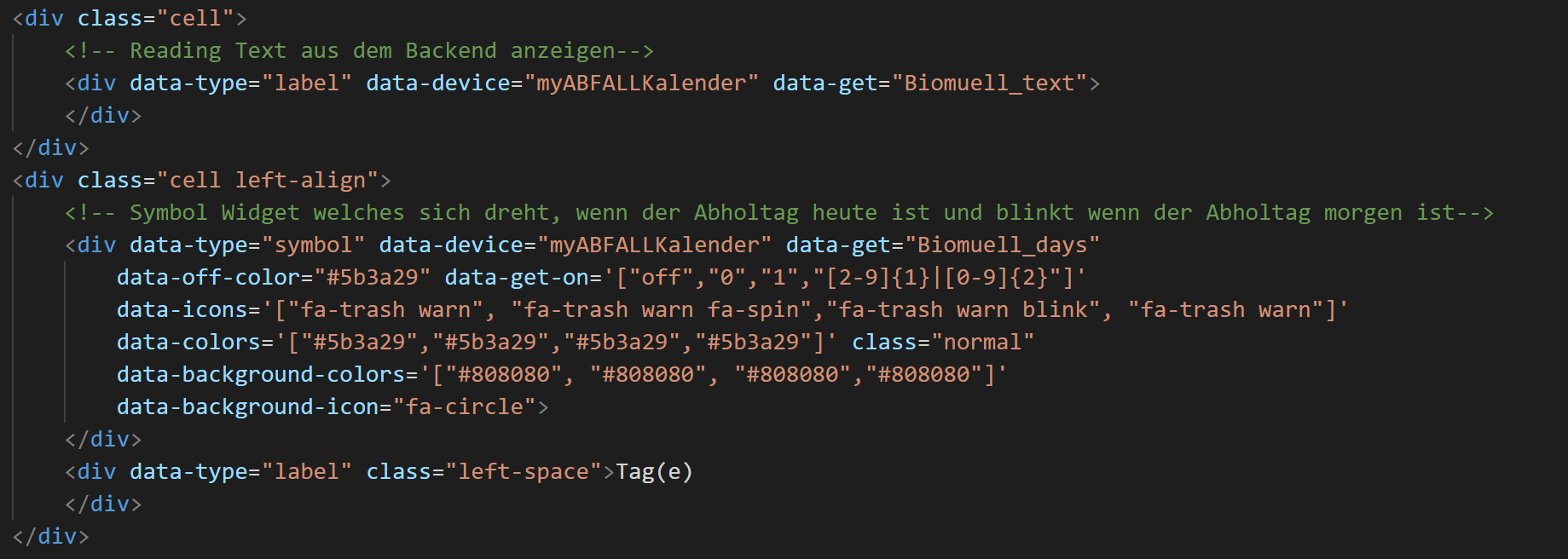


Abbildung 3: Frontend Beispiel Müll Plan

Am Beispiel des oben gezeigten Codes müssen folgende Befehle angepasst werden.

data-get = muss angepasst werden im Beispiel hier heißt das Feld, welches wir auslesen möchten „Biomuell\_text“ welches, aus dem Backend aus den Readings entnommen werden kann.



Abbildung 4: Backend Beispiel Müll Plan

Beispiel: <https://www.avl-ludwigsburg.de/fileadmin/Files/Abfallkalender/ICS/Privat/Privat_2022_Bietigheim-Stadt-und-Metterzimmern.ics>

Ggf. muss auf dem PI das Perl Modul zum Nutzen von HTTPS installiert werden. Hierfür auf der Shell eingeben:

sudo cpan install IO::Socket::SSL

Das Modul ABFALL in FHEM installieren via:

update all <https://raw.githubusercontent.com/uniqueck/fhem-abfall/master/controls_fhemabfall.txt>

Danach FHEM einmal neu starten via:

shutdown restart

Danach einen Kalender anlegen mit dem Link zum Abfallkalender via:

define abfallKalender Calendar ical url <https://www.avl-ludwigsburg.de/fileadmin/Files/Abfallkalender/ICS/Privat/Privat_2022_Bietigheim-Stadt-und-Metterzimmern.ics>

attr abfallKalender room 97\_Abfall

Hier für den Beispielkalender. Bei Bedarf den Link individuell abändern.

Anschließend das Modul ABFALL installieren mit der Referenz zum abfallKalender.

define myABFALLKalender ABFALL abfallKalender

attr myABFALLKalender room 97\_Abfall

Speichern mit Save config.

Mit dem schon angelegten Frontend-Part sind die Leerungstermine nun einsehbar. Die Korrektheit der Daten hängt davon ab, wie diese vom Bereitsteller zur Verfügung gestellt werden. Da es keinen Standard gibt kann sich dies unterscheiden und bedarf manuellen Aufwand zur Anpassung.