

Hier ein kleines Tutorial wie ich meinen SmartMirror umgesetzt habe, natürlich ist das Projekt noch beliebig erweiterbar, aber hier erstmal das Basis Tutorial was gerne in weiteren Beiträgen beliebig erweitert werden darf, was mich sehr freuen würde.

Auf Instagram haben vielleicht schon viele das obige Bild gesehen, by the Way lasst mir gern ein Follow auf [Instagram SchimmerMediaHD](#) da 😊

Dort sieht man auf jeden Fall das ich für meinen SmartMirror die Software "**MagicMirror²**" verwende und demnach wird das nachfolgende Tutorial in Verbindung mit einem [RaspBerry Pi 3 B+](#) und eben dieser Software sein 😊

Benötigte Sachen

- [Raspberry Pi 3B+](#) (Kit mit allem was ihr braucht)
- [Display \(LTN173KT01 / LP173WD1 / B173RW01\)](#)
- [Display Controller \(Für LTN173KT01 / LP173WD1 / B173RW01\)](#)
- [Maus & Tastatur](#) (Falls noch nicht vorhanden)
- [IKEA Ribba Bilderrahmen \(61x91cm\)](#) (bei IKEA günstiger)
- [Spiegelfolie mit Rakel \(meist im Lieferumfang enthalten\) \(67x100cm\)](#)

Benötigte Software

- [Etcher](#)
- [Raspbian](#) (Das neuste mit Desktop)
- [MagicMirror²](#) (wird mit Befehl auf dem Raspberry Pi installiert)

Optionale Software

- [SD Formater](#)
- [FileZilla](#)
- [Putty](#)

1. Installieren des neuen Raspbian mit Desktop

- Als erstes [Etcher](#) und das neuste [Raspbian mit Desktop](#) herunterladen
 - dann Etcher installieren und das Raspbian Image entpacken
 - Die Micro SD Karte mit einem [SD Adapter](#) in den PC stecken
 - öffne Etcher und wähle das zuvor entpackte Raspbian Image aus
 - wähle als zweites in der Software das Entsprechende Laufwerk mit der Micro SD Karte aus
 - Nochmal alles überprüfen & wenn alles stimmt auf "**Flash!**" klicken.
-

2. Raspberry Pi starten & Einrichten

- Micro SD Karte in Raspberry Pi schieben & Netzwerk, Tastatur, Maus, Netzteil & Monitor anschließen

- Entweder über Schalter am Netzteil (verlinktes Set) oder durch einstecken des Netzteils Einschalten

- einloggen mit User: **pi** und dem Passwort: **raspbian**

- Raspberry Pi auf den neusten Stand bringen mit folgender Kommando Zeile

Code

1. `sudo apt-get update && sudo apt-get upgrade`

- dann noch die Firmware auf den neusten Stand bringen mit folgender Kommando Zeile

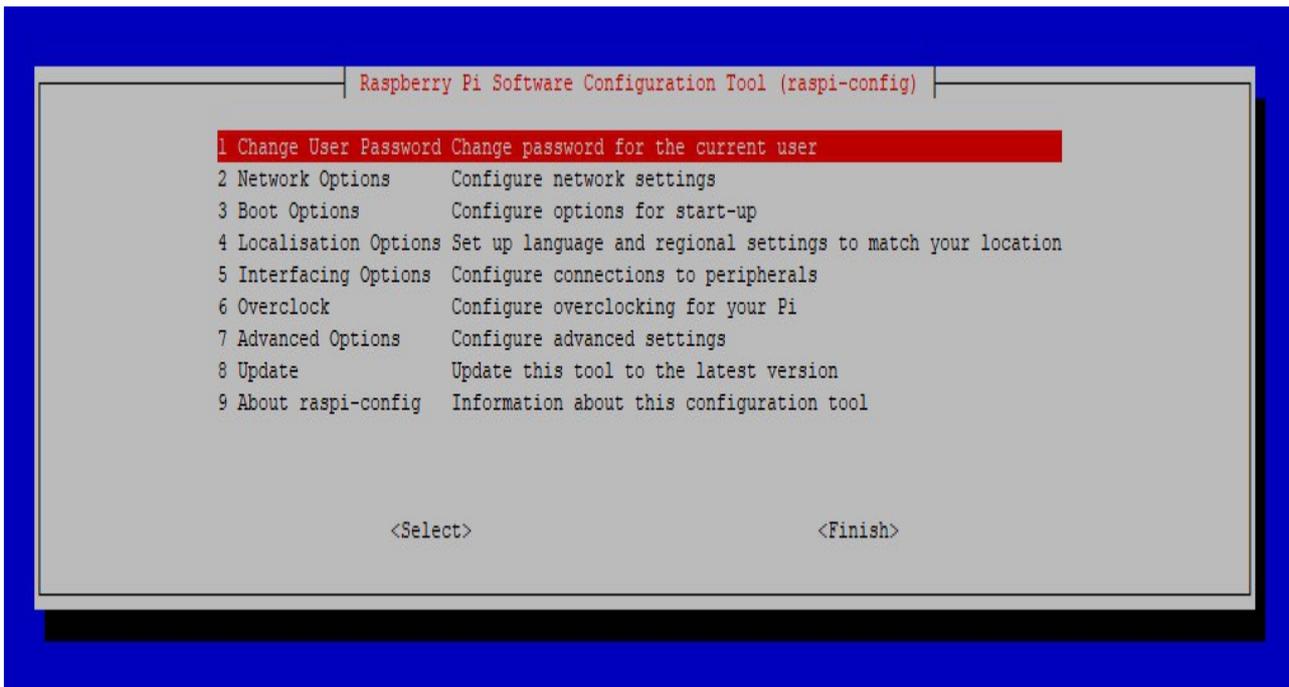
Code

1. `sudo rpi-update`

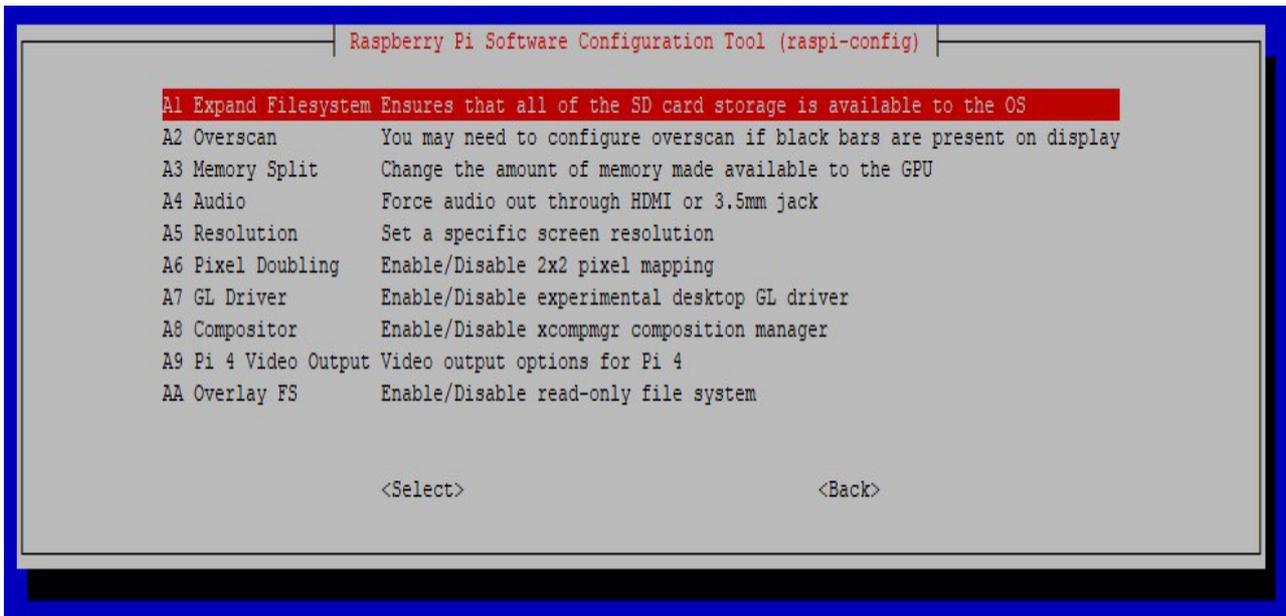
- nun müssen wir nur noch ein paar Einstellungen in der **raspi-config** machen. Um die Einstellungen zu öffnen folgenden Befehl eingeben

Code

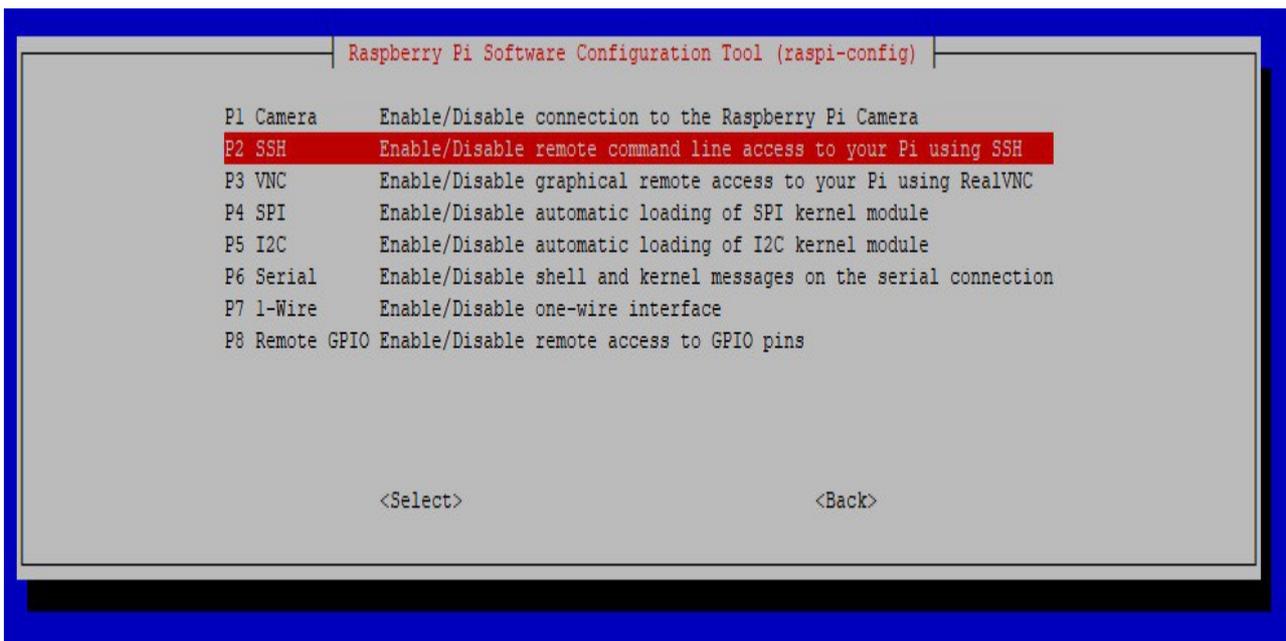
1. `sudo raspi-config`



- dann als erstes den Speicher auf die gesamte SD Karte ausweiten, dazu erst auf "**Advanced Options**" und dann "**Expent Filesystem**".



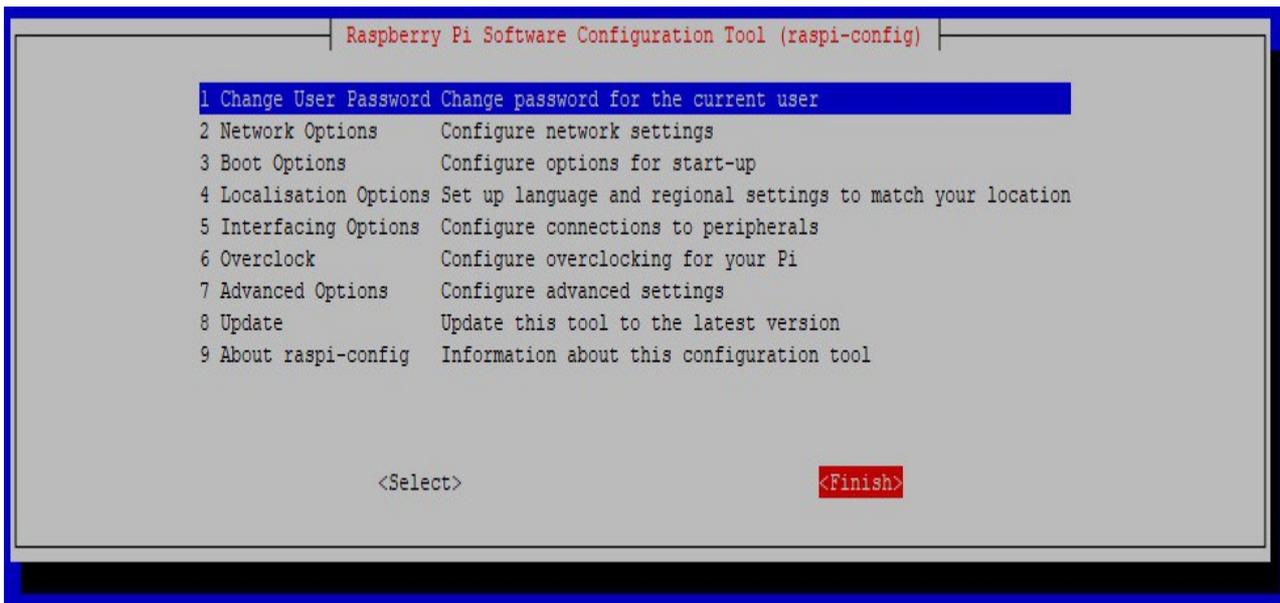
- um später eventuell nochmal Fernzugriff auf euren Pi zu haben noch **SSH** Freischalten, dazu erst auf "**Interfacing Options**" und anschließend auf "**P2 SSH**" um das ganze auf **enabled** mit "**Ja**" zu setzen.



- Dann noch die Location, Zeitzone und Tastatur auf Deutsch stellen, dazu einfach auf "**Localisation Options**" & dort dann die passenden Einstellungen vornehmen unter "**Change Locale**", "**Change Timezone**" & "**Change Keyboard Layout**"



- Zum Schluss könnt ihr noch optional euer Passwort ändern über "Change User Password" und dann das ganze mit "<Finish>" abschließen (hier kommt ihr mit der Tab Taste hin)



- dann einmal mit folgender Kommandozeile neustarten

Code

1. `sudo reboot`

3. MagicMirror² Software installieren

Nun installieren wir die MagicMirror² Software auf dem Raspberry Pi mit nur einer einzigen Befehl über die Kommandozeile. Wichtig hierbei ist das wir kein "sudo" davor eingeben, sonst bekommt man einen Fehler.

Code

1. `bash -c "$(curl -sL`

<https://raw.githubusercontent.com/MichMich/MagicMirror/master/installers/raspberry.sh>)"

oder ihr könnt auch das install Script nehmen (ist etwas mehr up to date)

Code

1. `bash -c "$(curl -sL https://www.dropbox.com/s/rh710ng0zvw3vp/raspberry.sh?dl=0)"`

4. Der erste Start vom Smart Mirror

- dafür als erstes das Verzeichnis wechseln um im MagicMirror Ordner zu sein mit...

Code

1. `cd ~/MagicMirror`

- danach Folgenden Befehl eingeben um den Bildschirm in einen Smart Mirror zu verwandeln

Code

1. `DISPLAY=:0 npm start`

- damit ist euer SmartMirror aber noch nicht fertig, ein paar Schritte folgen noch. Um weiter machen zu können am einfachsten einfach kurz den Stecker vom Pi ziehen und wieder rein stecken oder falls Ihr mein Empfohlenes Kit bestellt habt, eben den Schalter einmal kurz betätigen.

5. Display Drehen (falls Notwendig)

- Je nachdem ob Ihr euren Bildschirm hinter dem Spiegel Quer oder Hochkant verwendet wollt müsst ihr diesen noch drehen. Für das Querformat müsst ihr nichts mehr ändern, denn wie ihr ja schon gesehen habt, hat das ja bereits funktioniert. Wer allerdings, wie ich auch, das ganze hochkant nutzen möchte muss nach dem Login mit folgendem Befehl eine Config Datei öffnen...

Code

1. `sudo nano /boot/config.txt`

- und hier dann ganz nach unten scrollen (mit den Pfeiltasten) um dort folgenden Code einzufügen...

Code

1. `display_rotate=1`

- das ganze dann mit [**Strg+X**] und anschließend [**J**] und [**Enter**] bestätigen und speichern.

- anschließend noch ein reboot mit folgendem Befehl und euer Monitor zeigt alles vom Pi in Hochkant an.

Code

1. `reboot`

6. Standart Module Einstellen (Wetter/Uhrzeit/Kalender/etc.)

- als erstes öffnen wir die Datei wo wir die ganzen Einstellungen vornehmen können, dafür wechseln wir in das Verzeichnis mit dem Befehl...

Code

1. `cd ~/MagicMirror`

- und dann öffnen wir mit dem Editor "**nano**" die **config.js** Datei, das geht mit dem Befehl...

Code

1. nano config/config.js

- die "ipWhitelist" könnt Ihr auf "[]" stellen damit Ihr Zugriff von anderen Geräten aus habt, also so...

Code

1. ipWhitelist: ["127.0.0.1", "::ffff:127.0.0.1", "::1"], --> zu --> ipWhitelist: [],

- oben können wir dann noch die Sprache von "en" auf "de" stellen, also...

Code

1. language: "en", --> zu --> language: "de",

6.1 Positionen für Module

- top_bar
- top_left
- top_center
- top_right
- upper_third
- middle_center
- lower_thrid
- bottom_left
- bottom_center
- bottom_right
- bottom_bar
- fullscreen_above
- fullscreen_below

🌐 Das "**calendar**" Modul steht noch auf "**US Holidays**" und unten ist ein Kallender von "**calendarlabs.com**" verlinkt. Dies können wir viel interaktiver und cooler gestalten indem wir hier unseren persönlichen **Google Kalender** einbinden.

- dazu einfach nach dieser Anleitung "**Kalender abrufen (nur ansehen)**" vorgehen und im Google Konto einloggen, dann auf die "**Übersicht der Kalender**" und dort Links an der Seite euren Kalender auswählen. Am besten wählt ihr den aus, der auch auf eurem Smartphone genutzt wird, so synchronisieren sie sich automatisch.

- hier dann auf die **3 Punkte** und dann "**Einstellungen und Freigabe**" anklicken

+ Eintragen

SO 10 MO 11

November 2019 < > GMT+01

	S	M	D	M	D	F	S
44	27	28	29	30	31	1	2
45	3	4	5	6	7	8	9
46	10	11	12	13	14	15	16
47	17	18	19	20	21	22	23
48	24	25	26	27	28	29	30
49	1	2	3	4	5	6	7

Personen suchen

Meine Kalender

- Dennis Schimmer
- Erinnerungen
- Geburtstage
- Tasks

Weitere Kalender +

- Feiertage in Deutschland

Nur diesen zeigen

Einstellungen und Freigabe

+

hier dann etwas runter scrollen bis zu dem Punkt "**Kalender integrieren**" und hier könnt ihr euch den Link bei "**Privatadresse im iCal-Format**" kopieren, denn so könnt Ihr den Kalender einbinden ohne ihn für die Öffentlichkeit freigeben zu müssen.

Importieren & Exportieren

Einstellungen für meine Kalender

Dennis Schimmer

Kalendereinstellungen

Zugriffsberechtigungen

Für bestimmte Personen freigeben

Terminbenachrichtigungen

Benachrichtigungen für ganztägige Termine

Allgemeine Benachrichtigungen

Kalender integrieren

Kalender entfernen

Geburtstage

Einstellungen für weitere Kalender

Feiertage in Deutschland

Kalender integrieren

Kalender-ID
[redacted]@gmail.com

Öffentliche URL zu diesem Kalender
https://calendar.google.com/calendar/ical/[redacted]

Mit dieser URL können Sie über einen Webbrowser auf den Kalender zugreifen.

Code einbetten
<iframe src="https://calendar.google.com/calendar/ical/[redacted]">

Mit diesem Code können Sie den Kalender in eine Website einbetten.
Sie können den Code individuell anpassen oder mehrere Kalender einbetten.

Anpassen

Öffentliche Adresse im iCal-Format
https://calendar.google.com/calendar/ical/[redacted]

Mit dieser Adresse können Sie von anderen Anwendungen aus auf diesen Kalender zugreifen.
Achtung: Diese Adresse funktioniert nur, wenn Sie den Kalender öffentlich gemacht haben.

Privatadresse im iCal-Format
https://calendar.google.com/calendar/ical/[redacted]

Mit dieser Adresse können Sie von anderen Anwendungen aus auf den Kalender zugreifen, ohne ihn öffentlich zu machen.
Achtung: Geben Sie diese Adresse nur an Personen weiter, die alle Termindetails für diesen Kalender sehen dürfen.
Sie können diese Adresse zurücksetzen und damit die aktuelle ungültig machen.

Zurücksetzen

- nun könnt Ihr in der Config den **"header:"** umbenennen. So zum Beispiel...

Code

1. header: "US Holidays", --> zu --> header: "Mein Kalender",

- dann wird **"Mein Kalender"** als Überschrift von dem Modul angezeigt. Nun brauchen wir nur noch bei **"url:"** die eben kopierte URL von unserem Privaten Google Kalender eintragen, was dann in etwa so aussieht...

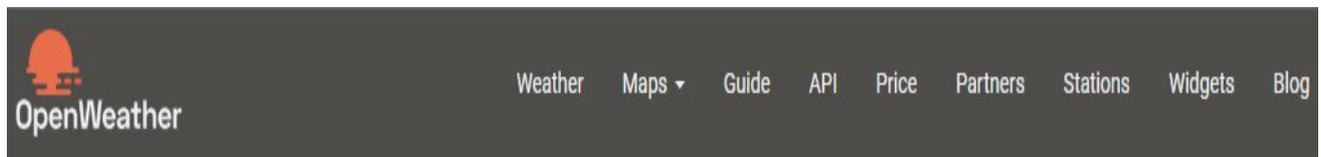
Code

1. url: "https://calendar.google.com/calendar/ical/EuerAccount@gmail.com/private-123456789/basic.ics"

☛ Das "currentweather" Modul richten wir als nächstes ein. Dazu müssen wir uns auf

"openweathermap.org" einen Account erstellen. Wichtig hierbei ist das Ihr einen Haken bei "System news (API usage alert, system update, temporary system shutdown, etc)" setzt, denn wir benötigen einen API Zugang.

- den "API Zugang" bekommt Ihr indem ihr bei "Price" auf "[Free - Get API key and Start](#)" klickt



Price

[Home](#) / [Price](#)

Subscribe to current weather, forecasts, and historical data collections and enjoy our fast, simple API! Please also read [How to buy](#) before you subscribe.

[Current weather & forecasts collection](#) and [Historical weather collection](#) are provided separately from each other.

Current weather and forecasts collection

To get access to weather history, please see our "[Historical weather collection](#)"

	Free	Startup	Developer	Professional	Enterprise
Price per month <small>Price is fixed, no other hidden costs (VAT is not included)</small>	Free	40 USD / month	180 USD / month	470 USD / month	2,000 USD / month
Subscribe	Get API key and Start	Subscribe	Subscribe	Subscribe	Subscribe
Calls per minute (no more than)	60	600	3,000	30,000	200,000
Current weather API	✓	✓	✓	✓	✓
4 days/hourly forecast API ^{NEW}	-	-	✓	✓	✓

- unter "API" und dann "API keys" findet ihr dann "[Euren API Key](#)". Diesen könnt Ihr dann schon mal zwischenspeichern.

You can generate as many API keys as needed for your subscription. We accumulate the total load from all of them.

Key	Name		Create key
	Default	 	<input type="text" value="* Name"/> <input type="text"/> <input type="button" value="Generate"/>

- nun müsst Ihr nur noch auf "**Weather**" nach "**Eurer Stadt oder Eurem Dorf**" suchen und oben aus dem Browser den Code kopieren. In meinem Fall ist es "2924201".

openweathermap.org/city/2924201

BIOS Settings

Support Center

Weather in your city

Sign In

Sign Up

°C °F



Weather

Maps

Guide

API

Price

Partners

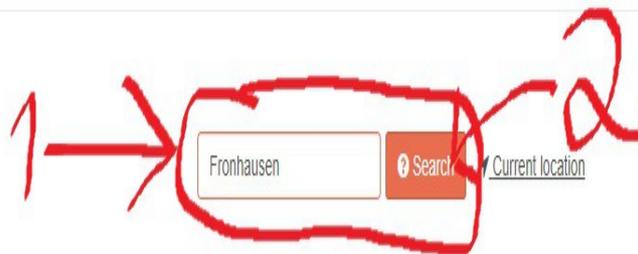
Stations

Widgets

Blog

Weather forecast

Home / Weather forecast



Current weather and forecasts in your city

Weather in Fronhausen, DE

Main Daily Hourly Chart Map

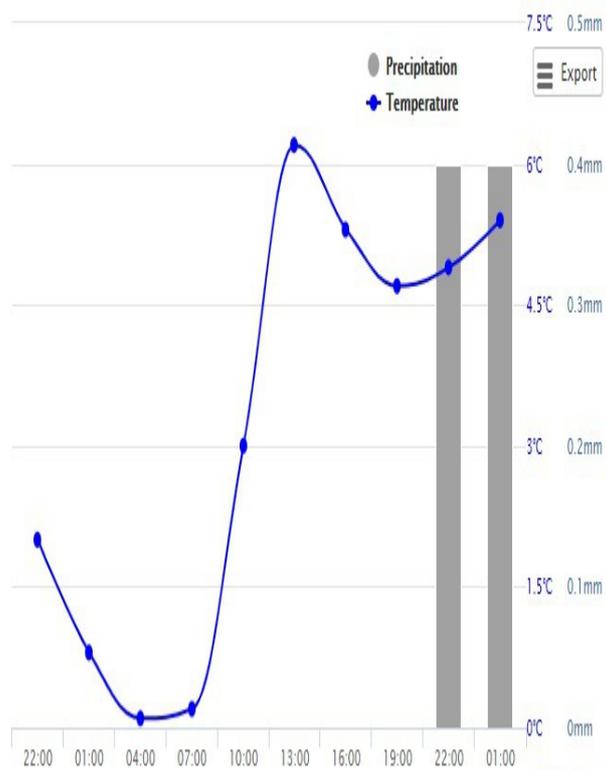
Weather and forecasts in Fronhausen, DE

3 °C

Scattered clouds

19:12 Nov 16 Wrong data?

Wind	Gentle Breeze, 4.1 m/s, South (190)
Cloudiness	Scattered clouds
Pressure	1012 hpa
Humidity	93 %
Sunrise	07:41
Sunset	16:38
Geo coords	[50.7, 8.7]



The weather forecast is displayed in

Highcharts.com

- jetzt müssen wir die Daten nur noch in unsrer Config Datei wie folgt einfügen...

Code

1. location: "New York", --> zu --> location: "Eure Stadt oder Euer Dorf",
2. locationID: "", --> zu --> locationID: "Eure LocationID (bei mir 2924201)",
3. appid: "YOUR_OPENWEATHER_API_KEY" --> zu --> appid: "Euer vorher kopierter API Key"

☛ Beim Modul "**weatherforecast**" brauchen wir die selben Daten wie oben beim "**currentweather**" Modul. Daher ändern wir hier das ganze einfach wie folgt ab...

Code

1. location: "New York", --> zu --> location: "Eure Stadt oder Euer Dorf",
2. locationID: "", --> zu --> locationID: "Eure LocationID (bei mir 2924201)",
3. appid: "YOUR_OPENWEATHER_API_KEY" --> zu --> appid: "Euer vorher kopierter API Key"

☛ Das "**newsfeed**" Modul ist natürlich für viele auch interessant und hier habt Ihr verschiedene Möglichkeiten beliebige **RSS Feeds** zu hinterlegen.

- Für die Supporter meines YouTube Kanals, oder generell falls ihr einen Lieblings YouTuber habt könnt ihr euch mit dem "**YouTube RSS Extractor**" einen **RSS Link** von eurem Lieblings Kanal bekommen, der euch dann immer über die neusten Videos auf dem jeweiligen Kanal informiert. Hier folgt der RSS Feed Link meines YouTube Kanals...

Code

1. https://www.youtube.com/feeds/videos.xml?channel_id=UC3PCSf8gzYLVBdqs1-SMmRA

- oder falls Ihr an den neusten Nachrichten interessiert seid könnt Ihr auch den **RSS Feed** von ARD nehmen...

Code

1. <http://www.tagesschau.de/xml/rss2>

- und dann das ganze einfach wie Folgt in der Config Datei abändern...

Code

1. title: "New York Times", --> zu --> title: "Euer YouTuber / oder ARD News",
2. url: "http://www.nytimes.com/services/xml/rss/nyt/HomePage.xml" --> url: "RSS Link"

☛ **Config Datei Speichern**

Zum Speichern der Config Datei nachdem Ihr diese bearbeitet habt, müsst ihr nur [**Strg+X**] drücken und dann das ganze mit [**J**] und anschließendem [**Enter**] bestätigen. schon sind eure Einstellungen gespeichert.

7. Drittanbieter Module Einstellen (ioBroker, Alexa, Xiaomi, PV-Anlage, IP-Cam)

diesen Punkt 7. könnt Ihr überspringen falls ihr keine Drittanbieter Module installieren wollt und euch der bisherige Stand der Information eures SmartMirros ausreicht. Dann könnt Ihr einfach

direkt bei Punkt 8. weitermachen.

🔗 Xiaomi Aqara SmartHome Modul (MMM-xiaomi)

- Um dieses Modul zu installieren müsst ihr erstmal in das **MagicMirror Verzeichnis** navigieren mit...

Code

1. `cd ~/MagicMirror/modules`

- und dann könnt ihr einfach mit dem "**git clone**" Befehl euch das Modul herunterladen...

Code

1. `git clone https://github.com/mirko3000/MMM-xiaomi.git`

- installieren geht dann einfach indem Ihr in das eben kopierte Verzeichnis wechselt mit...

Code

1. `cd ~/MagicMirror/modules/MMM-xiaomi`

und dann das ganze installieren (falls es so nicht geht dann mit "sudo " davor)

Code

1. `npm install`

- danach müssen wir noch in der "**Config.js**" Datei etwas hinzufügen, dazu wechseln wir erstmal das Verzeichnis mit...

Code

1. `cd ~/MagicMirror`

und dann...

Code

1. `nano config/config.js`

- in der Datei müssen wir dann etwas einfügen um das Modul zu platzieren, die "**position**" ist variable und ihr müsst die **IP-Adresse** eure **Xiaomi Gateway** sowie den dazu gehörigen **Key** und dann von den einzelnen ZigBee Komponenten die **Device ID** wissen. Hier dazu mein Beispiel...

Code

1. {
2. module: 'MMM-xiaomi',
3. position: 'top_left',
4. header: 'Temperatur und Luftfeuchtigkeit', // This is optional
5. config: {
6. gatewayIP: 'Ip-Adresse eurer Gateway',
7. outsideSensorId: 'Key eurer Gateway',
8. showWindow: true,
9. showVentilation: true,
10. showLights: false,
11. audioNotifications: true,
12. rooms: [
13. {
14. name: 'Wohnzimmer',
15. sortOrder: 10,

```
16.devices : ['168d0101c95z51']
17.},
18.{
19.name: 'Schlafzimmer',
20.sortOrder: 20,
21.devices : ['168d0101c95z51']
22.},
23.{
24.name: 'Küche',
25.sortOrder: 20,
26.devices : ['168d0101c95z51']
27.},
28.{
29.name: 'Badezimmer',
30.sortOrder: 20,
31.devices : ['168d0101c95z51']
32.},
33.{
34.name: 'Kinderzimmer',
35.sortOrder: 20,
36.devices : ['168d0101c95z51']
37.},
38.{
39.name: 'Gästezimmer',
40.sortOrder: 20,
41.devices : ['168d0101c95z51']
42.},
43.]
44.}
45.},
```

Alles anzeigen

- für den Fall das Ihr das **Token** (Xiaomi Gateway Key) und die einzelnen **Device ID's** nicht kennt, dann könnt ihr das mit "[miio Tool](#)" ermitteln. Zum installieren dieses Tools folgendes in der Kommandozeile eingeben...

Code

1. `npm install -g miio`

- und nachdem ihr das intsalliert habt könnt ihr das Tool ausführen und nach allen Devices im netwerk suchen mit...

Code

1. `miio discover`

der Output sieht dann in etwa so aus, man sieht eigentlich direkt da wo das **Token** dabei steht ist die **Gateway** und "**lumi.weather**" ist dann quasi ein **Temperatur un Luftfeuchtigkeits Sensor...**

Code

1. Device ID: 12345678
2. Model info: Unknown
3. Address: 192.168.178.99
4. Token: ???
5. Support: Unknown

6. Device ID: 12345678
7. Model info: lumi.gateway.v3
8. Address: 192.168.178.91
9. Token: 82uf97i4fi5f4i5d45401k97c47ce470 via auto-token
- 10.Support: At least basic
- 11.Device ID: 179d5001a95f70
- 12.Model info: lumi.weather
- 13.Address: Owned by miiio:12345678
- 14.Token: Automatic via parent device
- 15.Support: At least basic

Alles anzeigen

- Zum Speichern dann wieder nur [**Strg+X**] drücken und dann das ganze mit [**J**] und anschließendem [**Enter**] bestätigen. schon sind eure Einstellungen gespeichert.

☛ schöneres Wetter Modul (MMM-OpenmapWeather)

- da ich ein Grafisch etwas ansprechenderes Wetter Modul wollte als das was Standardmäßig mit dabei ist habe ich mich bei den [Modulen](#) mal etwas umgesehen und das Modul hier gefunden, zum installieren auch hier wieder in das "**MagicMirror**" Verzeichnis wechseln mit...

Code

1. cd ~/MagicMirror/modules

- und dann könnt ihr einfach mit dem "**git clone**" Befehl euch das Modul herunterladen...

Code

1. git clone https://github.com/sathyarajv/MMM-OpenmapWeather.git

- danach müssen wir noch in der "**Config.js**" Datei etwas hinzufügen, dazu öffnen wir die Datei mit dem Nano Editor...

Code

1. nano config/config.js

- hier fügen wir jetzt einfach folgenden Code ein, bzw. ersetzen den Code vom Standard Wetter Modul. Der Vorteil bei diesem von mir ausgewähltem Modul ist das wir hier den selben **API Key** nutzen können wie bei dem Standard Modul, denn das ganze läuft hier auch über

OpenWeatherMap , oben habe ich schon beschrieben wie ihr dort den Key und eure Location Daten bekommen könnt, diese könnt Ihr einfach hier ersetzen...

Code

1. {
2. module: "MMM-OpenmapWeather",
3. position: "top_right", // This can be any of the regions.
4. // Best results in left or right regions.
5. config: {
6. // See 'Configuration options' for more information.
7. location: "Fronhausen",
8. locationID: "2924201", //Location ID from http://openweathermap.org/help/city_list.txt
9. appid: "Euer-API-Key-von-OpenWeatherMap.org", //openweathermap.org API key
- 10.colorIcon: true
- 11.}
- 12.},

```

13.{
14.module: "weatherforecast",
15.position: "top_right",
16.header: "Wettervorhersage für",
17.config: {
18.location: "Fronhausen",
19.locationID: "2924201", //ID from http://bulk.openweathermap.org/sample/city.list.json.gz;
   unzip the gz file and find your city
20.appid: "Euer-API-Key-von-OpenWeatherMap.org"
21.}
22.},

```

Alles anzeigen

- Zum Speichern dann wieder nur [**Strg+X**] drücken und dann das ganze mit [**J**] und anschließendem [**Enter**] bestätigen. schon sind eure Einstellungen gespeichert.

🌞 PV-Anlage Modul (MMM-SolarEdge)

- ich habe seit Mitte 2019 meine [PV Anlage \(9,6kWp\) mit Stromspeicher von Sonnen \(12,5kWh\)](#), diese arbeitet mit einem Wechselrichter von **SolarEdge** weshalb dieses Modul hier für mich absolut passend ist. Insallation ist hier auch wieder ganz einfach, wieder in das **MagicMirror** Verzeichnis wechseln...

Code

```
1. cd ~/MagicMirror/modules
```

- und dann könnt ihr einfach mit dem "**git clone**" Befehl euch das Modul herunterladen...

Code

```
1. git clone https://github.com/bertieuk/MMM-SolarEdge.git
```

- nachdem das ganze installiert ist benötigt ihr natürlich noch eure "**SiteID**" und einen "**API Key**" zu eurem [SolarEdge Portal](#). Sprecht zur Not nochmal mit eurem Installateur (in meinem Fall [Firma Sames Solar](#)), falls Ihr noch keinen Zugang habt, die können euch dabei helfen wenn ihr es selbst nicht hin bekommt. Wenn Ihr dann einen Zugang habt einfach über das [SolarEdge Portal](#) Einloggen...

Herzlich Willkommen in der SolarEdge-Monitoring-Plattform

Nutzername:

Passwort:

Ich bin kein Roboter.



reCAPTCHA

[Datenschutzerklärung](#) - [Nutzungsbedingungen](#)

Login

Merken

[Passwort vergessen?](#)



Neuer Installateur? [Hier klicken](#)

Neuer Eigentümer? Wenden Sie sich für die Einrichtung Ihres Kontos an Ihren Installateur

[Öffentliche Anlagen](#)

[Beispielkonto](#)

© SolarEdge 2019. Alle Rechte vorbehalten.

[Allgemeine Geschäftsbedingungen](#) | [Datenschutzerklärung](#)

- nachdem ihr dann eingeloggt seid, könnt ihr auch schon oben aus der Adressleiste die "SiteID" herauslesen, das ist die Nummer die nach dem "site/" steht, oder ihr schaut einfach rechts auf der Seite bei "Anlagenstatus" bei "ID"...

monitoring.solaredge.com/solaredge-web/p/site/.../dashboard

Home Willkommen Dennis Chris Schimmer

Anlagensuche (min. 3 Buchstaben eingeben): Schimmer-Fronhausen

Anlagenstatus ✓

ID

Leistung und Energieertrag

Tag Woche Monat Rechnungszyklus Jahr

01.11.2019 - 30.11.2019

Produktion: **116,95 kWh**

Wh

14 k

12 k

10 k

name status ✓

ID

Name Schimmer-Fronhausen

Adresse Kantstraße 22, Fronhausen, Hessen, Germany

Installiert

Letzte

Spitzenleistung 9,6 kWp

Bewölkt 3 °C

Gefühlte Temperatur: 1 °C

Wind SSE, 6 km/h

Luftfeuchtigkeit 93 %

Sonnenaufgang: 07:49

Sonnenuntergang: 16:32

Donnerstag Freitag Samstag

- jetzt brauchen wir noch die Entsprechende API dazu um das ganze dann auch mit unserem SmartMirror verbinden zu können, diese finden wir wenn wir oben auf das Zahnrad (**Admin**) klickt und dann dort auf "**Anlagenzugriff**", dort steht dann unten "**API-Zugriff**" dort einfach das Häkchen bei "**Ich habe die SolarEdge API AGBs gelesen, verstanden & stimme zu**" setzen und einen API Key generieren und kopieren. Die "**SiteID**" steht hier auch nochmal 😊

- Übersicht
- Auslegung
- Graphen
- Berichte
- Meldungen
- Admin**

Anlagensuche (min. 3 Buchstaben eingeben):

Schimmer-Frohnhausen

- Anlagendetails
- Anlagenzugriff**
- Erlös

Zugriffssteuerung

Öffentlicher Zugriff

Öffentlichen Zugriff aktivieren (Daten dieser Anlage können, ohne weitere Anmeldung, auf der Website von SolarEdge betrachtet werden)

Unter dem Namen:

Anzeigen:

Öffentliche Adresse:

Abbrechen

speichern

Kiosk-Ansicht

Erstellt einen Link für eine öffentliche Anzeige

Unter dem Namen:

Kiosk Adresse:

Abbrechen

speichern

API-Zugriff

Ich habe die SolarEdge API AGBs **gelesen, verstanden & stimme zu**

API Key:

Neuer API Key

Standort-ID:

[Lesen Sie das API-Handbuch](#)

Abbrechen

speichern

- die nun gewonne "SiteID" und der "API-Key" wird nun in der **Config.js** in einem Script eingetragen, dazu wechseln wir wieder in das **MagicMirror Verzeichnis** mit...

Code

1. cd ~/MagicMirror

und dann...

Code

1. nano config/config.js

- und hier dann eben die "SiteID" und den "API Key" in folgendem Script eintragen und in das **Config.js** einfügen.

Code

```
1. {
2.   module: 'MMM-SolarEdge',
3.   position: 'bottom_left',
4.   config: {
5.     apiKey: "#####", //Requires your own API Key
6.     siteId: "12345", //SolarEdge site ID
7.   }
8. },
```

- Zum Speichern dann wieder nur [**Strg+X**] drücken und dann das ganze mit [**J**] und anschließendem [**Enter**] bestätigen. schon sind eure Einstellungen gespeichert.

☼Live Wetter Karten Modul (MMM-Buienradar)

- Einrichtungs Tutorial Folgt

☼Alexa Steuerungs Modul (MMM-AlexaControl)

- Einrichtungs Tutorial Folgt

☼ioBroker Modul (MMM-ioBroker)

- Einrichtungs Tutorial Folgt

☼IP-Kamera Modul (MMM-RTSPStream)

- um das Modul zu installieren müsst Ihr einfach folgenden Befehl in die Kommandozeile kopieren oder per Hand eingeben...

Code

```
1. bash -c "$(curl -s https://raw.githubusercontent.com/shbatm/MMM-RTSPStream/develop/scripts/installer.sh)"
```

- danach müssen wir noch in der "**Config.js**" Datei etwas hinzufügen, dazu wechseln wir erstmal das Verzeichnis mit...

Code

1. cd ~/MagicMirror

und dann...

Code

1. nano config/config.js

- in der Datei müssen wir dann etwas einfügen um das Modul zu platzieren, die "**position**" ist variable und bei "**IP-Adresse-des-ONVIF-Streams**" trägst du die IP der Cam ein...

Code

1. {
2. module: "MMM-RTSPStream",
3. position: "middle_center",
4. config: {
5. autoStart: true,
6. rotateStreams: true,
7. rotateStreamTimeout: 10,
8. moduleWidth: 354,
9. moduleHeight: 240,
- 10.localPlayer: 'vlc',
- 11.remotePlayer: 'none',
- 12.showSnapWhenPaused: true,
- 13.remoteSnaps: false,
- 14.shutdownDelay: 12,
- 15.stream1: {
- 16.name: 'Haustür',
- 17.url: 'rtsp://IP-Adresse:554/onvif1',
- 18.frameRate: 'undefined',
- 19.width: undefined,
- 20.height: undefined,
- 21.}
- 22.},

Alles anzeigen

- das wäre jetzt ein Beispiel wie man das ganze mit einer Kamera einbinden kann, man kann aber auch mehrere Kameras eintragen die dann immer XX Sekunden gezeigt werden und dann kommt die nächste Kamera. Wenn Ihr das Modul installiert habt könnt Ihr auch einfach in eurem Browser folgendes Aufrufen und euer Script damit erstellen...

Code

1. <http://Raspberry-Pi-IP:8080/MMM-RTSPStream/config.html>

Step 1. MMM-RTSPStream Core Configuration Builder

Module Position

Module Header Text
Leave blank for no header.

Auto Start Streams? Yes
 No
Yes will start the streams automatically when the mirror starts, no will show snapshots (if available) or a blank window until the stream is started by a notification from another app or by clicking the play button.

Rotate Through Streams or Individual Windows? Rotate through all streams in a single window
 Display an individual window for each stream

Time (in sec) to show each stream
Only applies when the first option above is selected.

Show snapshot images when the stream(s) is paused Yes
 No
If your camera supports timed snapshot images via URL or saving to a file location, the module can display these when the stream is paused.

Local Screen Player to Use

Remote Browser/VNC Player to Use

Show Snapshots on Remote? Yes
 No
When using OMXPlayer for local player and None for remote player, no video will be visible if you open the Mirror on a remote web browser or view through VNC. Selecting yes for this option will show snapshots to the remote viewers (saves resources over using FFmpeg for Remote Player)

Module Width px
When rotateStreams is false and multiple streams are used, adjust this value to adjust the number of streams shown side by side. E.G. to show 2 streams side by side, this value should be = 2*(Stream Width + 2*1px (border) + 2*15px (margin))

Module Height px
Similar (but less critical) to moduleWidth. Adjust to the number of streams high to ensure other modules clear.

Module Offset - Only applies when using OMXPlayer/VLC.
On some displays, the video does not properly line up with the box on the screen. Entering a pixel value will shift the video over by that amount. Default: 0 Values: Any number (no units) by itself will adjust both top/left the same amount, or you can specify left & top adjustments separately (e.g. { left: -10, top: -10 })

Step 2. Video Stream Configurations

Number of Streams 1 2 3 4

Stream #1 Name
Enter the name for the camera

Feed #1 URL
The url of the RTSP stream. See this list for paths for some common security cameras. Also see below for how to test for a valid url Username and password should be passed in the url if required: rtsp://<username>-<password>@<hostname>-<port>-<path>

Network Protocol #1 TCP UDP

Feed #1 HD URL
The url of the HD RTSP stream for full screen OMXPlayer/VLC use.

Snapshot URL or File Path #1
The url or filepath of the snapshot images.

Snapshot #1 Type URL File Path

Snapshot #1 Refresh Rate seconds
Time between fetching new snapshot images

Stream #1 Width px
The width in px of the stream.

Stream #1 Height px
The height in px of the stream

Absolute Position
Required only for OMXPlayer or VLC. Use this to set an exact position for the video instead of getting it automatically. Format: "l top: XX, right: XX, bottom: XX, left: XX"

Step 3. Copy to MagicMirror Configuration File

Generate Config File

Configuration
Copy the above section to your configuration file.

- und nachdem Ihr eure Parameter eingegeben habt, könnt ihr das Script über "**Generate**" erzeugen und dann in eure **Config.js** rein kopieren.

- Zum Speichern dann wieder nur [**Strg+X**] drücken und dann das ganze mit [**J**] und anschließendem [**Enter**] bestätigen. schon sind eure Einstellungen gespeichert.

8. Bildschirm Rotation einstellen (Querformat oder Hochformat)

- Wenn Ihr den Bildschirm so wie ich im Video hochkant verwendet möchtet, dann geht das sehr leicht. Einfach in der Kommandozeile mit folgendem Befehl erstmal wieder ins Stammverzeichnis wechseln...

Code

1. `cd ~`

- dann als nächstes mit dem Folgenden Befehl und dem Nano Editor, die "**Config.txt**" im "**boot Verzeichnis**" aufrufen

Code

1. `sudo nano /boot/config.txt`

- und hier dann unten einfach folgende Zeile Einfügen. Ich verwende hier die Variable "**1**" am ende um das Bild 90° nach Rechts zu drehen. Ihr könnt aber bei bedarf auch weitere Variablen wie "**2**" oder "**3**" nehmen.

Code

1. `# Bildschirm drehen`
2. `display_rotate=1`

- Zum Speichern dann wieder nur [**Strg+X**] drücken und dann das ganze mit [**J**] und anschließendem [**Enter**] bestätigen. schon sind eure Einstellungen gespeichert.

9. Bildschirm dauerhaft AN lassen (kein Sleep Mode)

- damit Euer SmartMirror auch immer die gewünschten Informationen Anzeigen kann, solltet Ihr den Bildschirm daran hindern aus zu gehen. Das ist auch schnell gemacht. Einfach mit dem nano Editor die "**autostart**" Datei im "**LXDE-pi**" Verzeichnis öffnen mit Folgendem Befehl...

Code

1. `sudo nano ~/.config/lxpanel/LXDE-pi/autostart`

- und dann den Folgenden Inhalt in die Datei schreiben...

Code

1. `@xset s 0 0`
2. `@xset s noblank`
3. `@xset s noexpose`
4. `@xset dpms 0 0 0`

- Zum Speichern dann wieder nur [**Strg+X**] drücken und dann das ganze mit [**J**] und anschließendem [**Enter**] bestätigen. schon sind eure Einstellungen gespeichert.

10. SmartMirror automatisch starten beim booten vom Raspberry Pi

- als erstes mit dem Befehl ins Stammverzeichnis gehen

Code

1. cd ~

- dann diesen Befehl eintippen...

Code

1. pm2 startup

- dann mit dem nano Editor folgende Datei Erstellen mit dem Befehl...

Code

1. nano mm.sh

- und folgendes in die Datei rein schreiben...

Code

1. cd ~/MagicMirror
2. DISPLAY=:0 npm start

- Zum Speichern dann wieder nur [**Strg+X**] drücken und dann das ganze mit [**J**] und anschließendem [**Enter**] bestätigen. schon sind eure Einstellungen gespeichert.

- dann muss die Datei noch die entsprechenden Berechtigungen bekommen um auch ausführbar zu sein, das macht ihr mit dem befehl...

Code

1. chmod +x mm.sh

- dann das ganze einmal ausführen mit...

Code

1. pm2 start mm.sh

- und mit dem hier einmal speichern...

Code

1. pm2 save

- wenn alles fertig ist, noch einmal rebooten damit er alles einmal frisch lädt und Bäm...

Code

1. sudo reboot

und dann sollte nach dem starten automatisch euer SmartMirror starten und in meinem Fall sieht das ganze dann am ende so aus...

Quelle : <https://www.schimmer-media.de/forum/index.php?thread/3316-mein-smartmirror-mit-magicmirror%C2%B2-software-auf-einem-raspberry-und-ikea-ribba-r/&pageNo=1>