



Hochschule Neubrandenburg  
University of Applied Sciences

Hochschule Neubrandenburg  
Master – Geoinformatik und Geodäsie  
Modul VMGG33 – Informatik-Projekt  
1. Semester

# Anwenderdokumentation

## Entwicklung einer konfigurierbarer Sensorstation auf der Basis von Raspberry Pi

Autor:	Tino Schuldt
Betreuer:	Dipl.-Inform. Jörg Schäfer Prof. Dr.-Ing. Andreas Wehrenpfennig
Tag der Einreichung:	18.01.2016

# Inhaltsverzeichnis

<b>Inhaltsverzeichnis.....</b>	<b>II</b>
<b>1 Einführung.....</b>	<b>1</b>
1.1 Nutzung.....	1
1.2 Anwendung .....	1
1.3 Erster Start .....	1
1.4 Sensoren verbinden .....	1
<b>2 Sensorstation.....</b>	<b>3</b>
2.1 Installation der Sensoren.....	3
2.2 Sensorstation nutzen.....	3
2.3 Automatisierte Aufzeichnung .....	4
2.4 Speicherung der CSV-Daten .....	5
<b>3 Webbasierte Datenabfrage .....</b>	<b>6</b>
3.1 RESTful-Webservice .....	6
<b>Quellenverzeichnis .....</b>	<b>III</b>
<b>Abbildungsverzeichnis.....</b>	<b>III</b>
<b>Tabellenverzeichnis.....</b>	<b>III</b>

# 1 Einführung

## 1.1 Nutzung

Die Sensorstation dient zum Auslesen von Messdaten durch beliebige Sensoren am Raspberry Pi.

## 1.2 Anwendung

Für die Anwendung werden Sensoren benötigt, die von der Sensorstation unterstützt werden. Zurzeit werden folgende Sensoren unterstützt:

- DS18B20 (Temperatursensor)
- DHT22 (Temperatur- & Feuchtigkeitssensor)
- HCSR04 (Entfernungssensor durch Ultraschall)

## 1.3 Erster Start

Bevor die Sensorstation genutzt werden kann, müssen alle Skripte mit Ausführungsrechten versehen werden. Dazu wird das Terminal geöffnet, in das Projektverzeichnis navigiert anschließend wird der folgender Befehl ausgeführt:

- `chmod +x . R`

## 1.4 Sensoren verbinden

Um die Sensoren mit dem Raspberry Pi zu verbinden, wird die GPIO-Schnittstelle (General Purpose Input/Output) verwendet. In der Abb. 1.1 wird die Anordnung der Pins von 1 bis 40 dargestellt. In der Tabelle 1.1 wird die Pinbelegung des Raspberry Pi dargestellt.

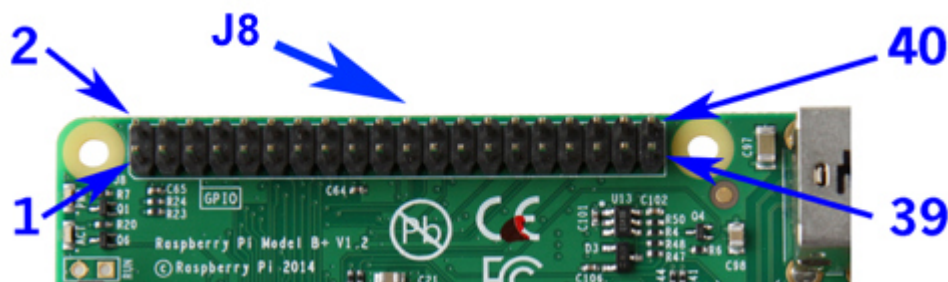


Abb. 1.1: Anordnung der Pins (übernommen von [PiJ15])

WiringPi	Rev 2	Rev 1	Pins		Rev 1	Rev 2	WiringPi
	+ 3,3 V	+ 3,3 V	1		2	+ 5 V	
8	GPIO 2	GPIO 0	3		4	+ 5 V	
9	GPIO 3	GPIO 1	5		6	GND	
7	GPIO 4	GPIO 4	7		8	GPIO 14	15
	GND	GND	9		10	GPIO 15	16
0	GPIO 17	GPIO 17	11		12	GPIO 18	1
2	GPIO 27	GPIO 21	13		14	GND	
3	GPIO 22	GPIO 22	15		16	GPIO 23	4
	+ 3,3 V	+ 3,3 V	17		18	GPIO 24	5
12	GPIO 10	GPIO 10	19		20	GND	
13	GPIO 9	GPIO 9	21		22	GPIO 25	6
14	GPIO 11	GPIO 11	23		24	GPIO 8	10
	GND	GND	25		26	GPIO 7	11
30	SDA 0 (GPIO 0)		27		28	SCL 0 (GPIO 1)	31
21	GPIO 5		29		30	GND	
22	GPIO 6		31		32	GPIO 12	26
23	GPIO 13		33		34	GND	
24	GPIO 19		35		36	GPIO 16	27
25	GPIO 26		37		38	GPIO 20	28
	GND		39		40	GPIO 21	29

Tabelle 1.1: Pin-Belegung der GPIO-Schnittstelle

## 2 Sensorstation

### 2.1 Installation der Sensoren

Bevor die Sensorstation vollständig genutzt werden kann, müssen einige Sensoren wie z.B. der DS18B20 und DHT22 im System installiert werden. Die Sensorstation muss dafür mit Root-Rechten ausgeführt werden. Dazu wird das Terminal aufgerufen und im Projektverzeichnis folgender Befehl ausgeführt:

- `sudo ./setup.py`

### 2.2 Sensorstation nutzen

Wie in der Abb. 2.1 dargestellt, wird nach dem Start der Sensor ausgewählt. Nach der Wahl des Sensors kann dieser durch einen Klick auf den Button „Konfigurieren“ konfiguriert werden. Anschließend erscheint wie in Abb. 2.2 dargestellt ein neues Fenster für die Konfiguration. Wie der Sensor angeschlossen werden kann, wird in diesem Fenster auf der rechten Seite angezeigt. Auf der linken Seite müssen die genutzten GPIO-Nummern und das Intervall für die Abfrage der Werte angegeben werden. In der Mitte sind alle möglichen Aktionen der Sensoren dargestellt. Diese Aktionen sind in der Tabelle 2.1 kurz erläutert.

Die Sensorstation beschränkt sich auf die Nutzung von einem einzigen Sensor pro Typ. Mehrere Sensoren des gleichen Typs können nicht unterschieden werden.

Aktion	Beschreibung
<b>Sensor installieren</b>	Starten eine Installationsroutine, die notwendig für einige Sensoren sind. Benötigt dazu Root-Rechte!
<b>Sensor testen</b>	Prüft, ob der Sensor Werte liefert, und gibt diese als Message-Box aus.
<b>Aufzeichnung starten</b>	Startet dauerhaft das Auslesen des Sensors im eingegebenen Intervall, bis dieser gestoppt wird.
<b>Zeitgesteuerte Aufzeichnung</b>	Öffnet ein Fenster, in dem ein Job für die Aufzeichnung erstellt werden kann.
<b>Export als CSV-Datei</b>	Exportiert die Daten als CSV-Datei im Ordner „data“.
<b>Beenden</b>	Schließt das Fenster.

Tabelle 2.1: Aktionen der Sensoren



Abb. 2.1: Auswählen des Sensors

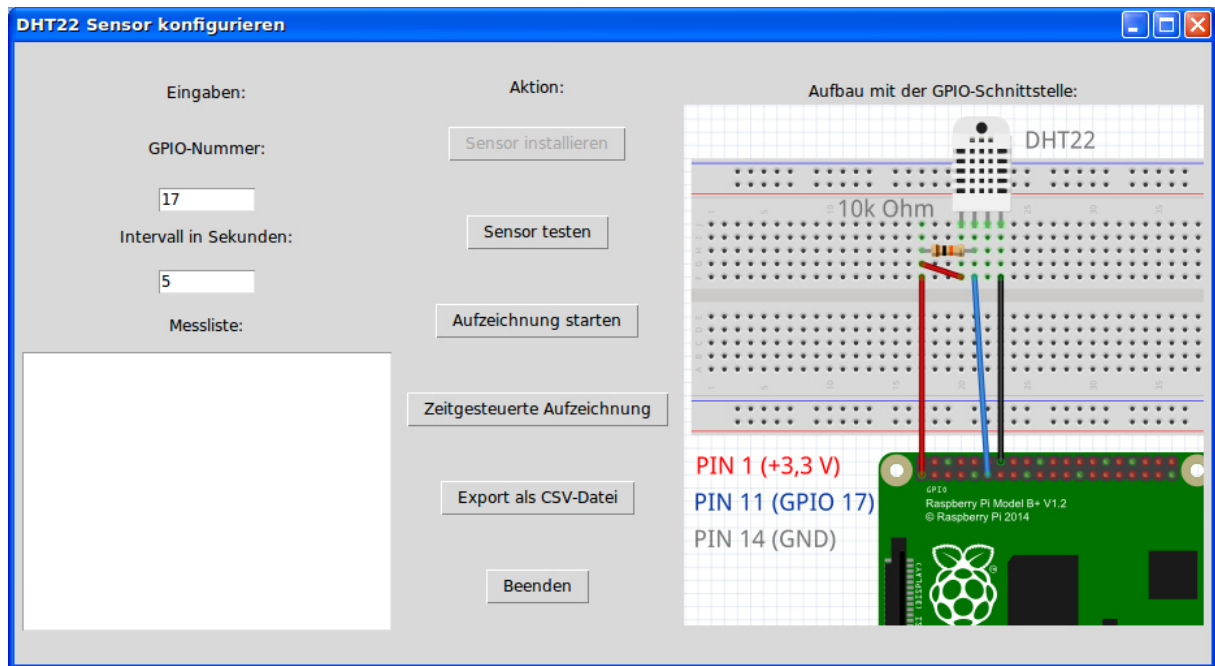


Abb. 2.2: Sensor DHT22 konfigurieren

### 2.3 Automatisierte Aufzeichnung

Durch einen Klick auf den Button „Zeitgesteuerte Aufzeichnung“ erscheint wie in der Abb. 2.3 dargestellt ein neues Fenster, in dem Jobs erstellt werden können. Zuerst muss eine Startzeit festgelegt werden, an dem die Messung automatisch beginnt. Diese Einstellung kann durch fünf Angaben (Monat, Wochentag, Tag, Stunde und Minute) spezifiziert werden. Danach muss die Laufzeit der Messung nach dem Start in Minuten angegeben werden. Einige Beispiele für die Nutzung sind in der Tabelle 2.2 dargestellt.

Um einen eingestellten Job zu übernehmen, wird der Button „Neuen Job“ genutzt. Anschließend werden alle aktiven Jobs im unteren Feld angezeigt. Alle Jobs können mit dem Button „Alle Jobs entfernen“ entfernt werden.

Aktion	Laufzeit[Min]	Monat	Wochentag	Tag	Stunde	Min.
Jeden Abend um 20:30 Uhr bis 21:30 Uhr	60	*	*	*	20	30
Jeden Montag um 12:00 Uhr bis 14:00 Uhr	120	*	Mo.	*	12	00
Jede volle Stunde für 10 Minuten	10	*	*	*	*	00
Den ganzen Tag am 24.12	1440	12	*	24	00	00

Tabelle 2.2: Beispiele für die Benutzung der zeitgesteuerten Aufzeichnung

Abb. 2.3: Zeitgesteuerte Aufzeichnung

## 2.4 Speicherung der CSV-Daten

Um die Daten beispielsweise mit Excel zu visualisieren, können die Messdaten als CSV-Datei (Comma Separated Values) gespeichert werden. Eine CSV-Datei beinhaltet eine Kopfzeile mit einer Beschreibung der Spalten und dann zeilenweise alle Messdaten, die durch ein Trennzeichen getrennt sind.

Das folgende Beispiel stellt den Inhalt einer CSV-Datei (Sensor: DHT22) dar:

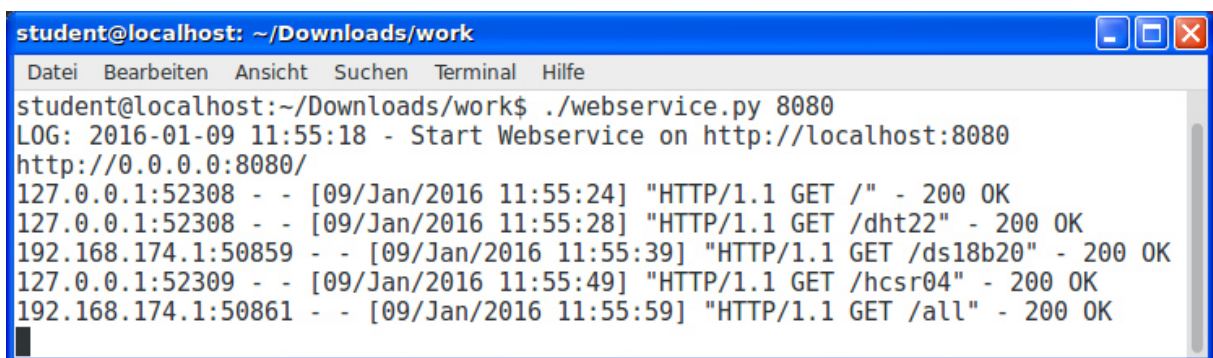
```
timestamp,temperature,humidity
2015-12-27 11:14:43,20.543,50.123
2015-12-27 11:14:48,20.551,50.132
```

## 3 Webbasierte Datenabfrage

### 3.1 RESTful-Webservice

Alle Messdaten können als Webservice plattformübergreifend bereitgestellt werden. Hierfür muss die Datei „webservice.py“ im Terminal ausgeführt werden. Für den ersten Start muss dieses Skript mit Root-Rechten ausgeführt werden. Anschließend wird das benötigte Paket „python-webpy“ installiert. Der Standardport zum kommunizieren mit dem Webservice ist 8080. Um diesen Port zu verändern, kann das Skript mit einem Parameter gestartet werden.

- `sudo ./webservice.py 8080`
- `./webservice.py 8080`



```
student@localhost: ~/Downloads/work
Datei Bearbeiten Ansicht Suchen Terminal Hilfe
student@localhost:~/Downloads/work$ ./webservice.py 8080
LOG: 2016-01-09 11:55:18 - Start Webservice on http://localhost:8080
http://0.0.0.0:8080/
127.0.0.1:52308 - - [09/Jan/2016 11:55:24] "HTTP/1.1 GET /" - 200 OK
127.0.0.1:52308 - - [09/Jan/2016 11:55:28] "HTTP/1.1 GET /dht22" - 200 OK
192.168.174.1:50859 - - [09/Jan/2016 11:55:39] "HTTP/1.1 GET /ds18b20" - 200 OK
127.0.0.1:52309 - - [09/Jan/2016 11:55:49] "HTTP/1.1 GET /hcsr04" - 200 OK
192.168.174.1:50861 - - [09/Jan/2016 11:55:59] "HTTP/1.1 GET /all" - 200 OK
```

Abb. 3.1: Webservice im Terminal starten

Nach dem Start kann der Client mit dem Webservice kommunizieren. Zum Testen wird der Webbrowser „Icweasel“ genutzt und die URL „localhost:8080“ aufgerufen. Hier erscheint eine Webseite, indem die verfügbaren Sensoren aufgelistet sind. Mit dem Verzeichnis z.B. „/dht22“ werden nur die Messwerte des jeweiligen Sensors aus der SQLite-Datenbank ausgelesen. Um alle Auszulesen wird das Verzeichnis „/all“ genutzt. Weitere Parameter z.B. rows, pretty und calc können genutzt werden, um die Ausgabe zu verändern. Die Beschreibung der einzelnen Parameter sind in der Tabelle 3.1 dargestellt. Der Aufruf mit dem Webbrowser ist in der Abb. 3.2 dargestellt.

Aufruf / Parameter	Beschreibung
/dht22	Zeigt die Werte des Sensors DHT22.
/ds18b20	Zeigt die Werte des Sensors DS18B20.
/hcsr04	Zeigt die Werte des Sensors HCSR04.
/all	Zeigt die Werte aller Sensoren an.
?rows=3	Zeigt die letzten drei Werte an.
?calc	Berechnet den Mittelwert, Maximalen und Minimalen Wert über alle Messwerte und gibt diese aus.
?pretty	Formatiert das JSON in leserlicher Form (Für den Produktiven Einsatz ungeeignet)
?rows=3&calc&pretty	Kombinierte Abfrage der drei Parameter mit dem &-Zeichen

Tabelle 3.1: Parameter für den Aufruf des Webservices



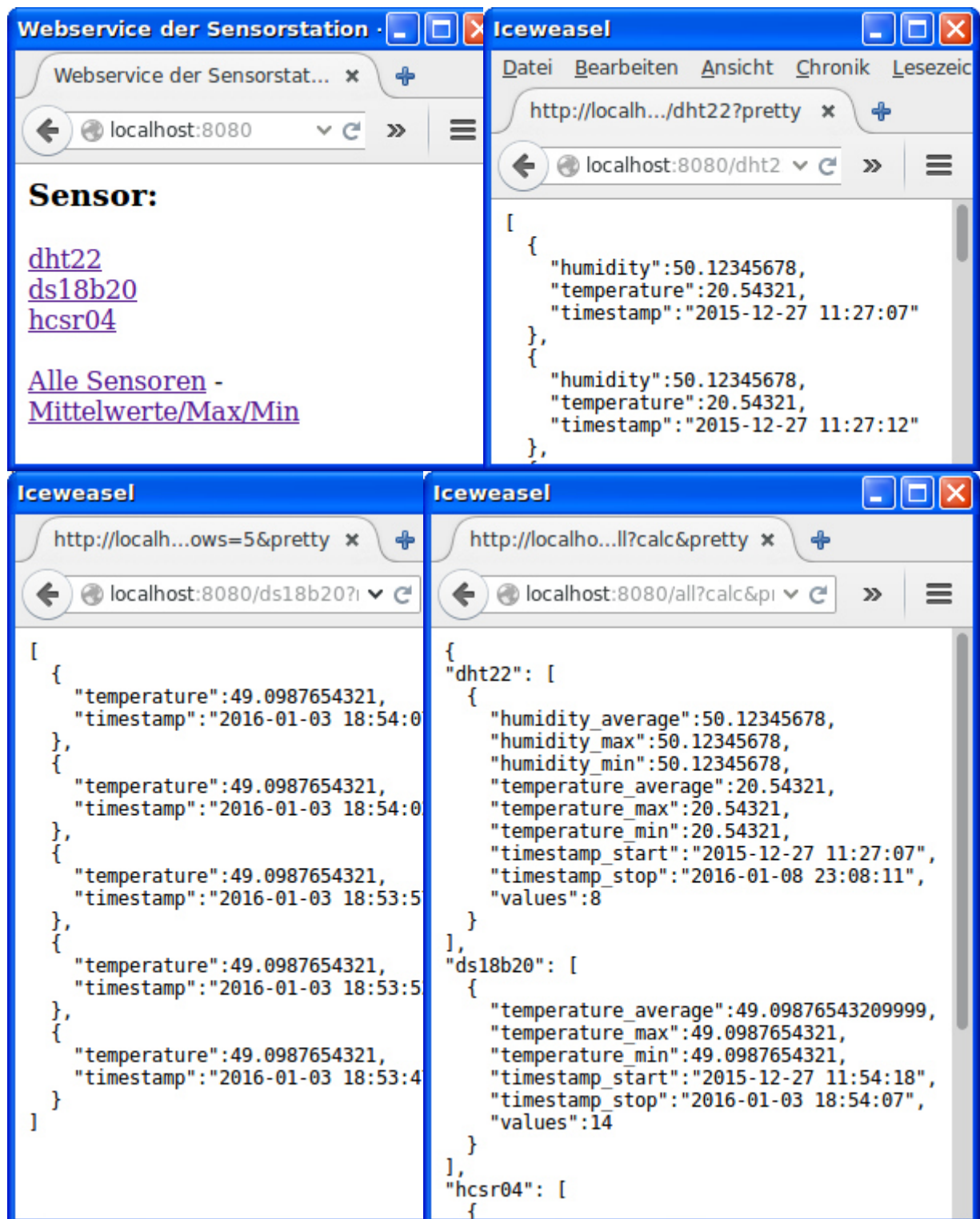


Abb. 3.2: Kommunikation mit dem Webservice

## Quellenverzeichnis

[PiJ15] The Pi4J Project; <http://pi4j.com/pins/model-b-plus.html> (13.10.2015).

## Abbildungsverzeichnis

Abb. 1.1: Anordnung der Pins (übernommen von [PiJ15]) .....	1
Abb. 2.1: Auswählen des Sensors .....	3
Abb. 2.2: Sensor DHT22 konfigurieren .....	4
Abb. 2.3: Zeitgesteuerte Aufzeichnung .....	5
Abb. 3.1: Webservice im Terminal starten.....	6
Abb. 3.2: Kommunikation mit dem Webservice .....	7

## Tabellenverzeichnis

Tabelle 1.1: Pin-Belegung der GPIO-Schnittstelle.....	2
Tabelle 2.1: Aktionen der Sensoren .....	3
Tabelle 2.2: Beispiele für die Benutzung der zeitgesteuerten Aufzeichnung .....	4
Tabelle 3.1: Parameter für den Aufruf des Webservices .....	6