



**SPŠE**  
**JEČNÁ**

STŘEDNÍ  
PRŮMYSLOVÁ  
ŠKOLA  
ELEKTROTECHNICKÁ

# Meteostanice

Daniel Šíma

Anotace	3
Zadání	4
Popis Hardwaru	5
Zapojení	6
Popis Softwaru	7
Ukázky Kódu	8
Ukázka Webové Stránky	9
Závěr	10
Zdroje	11

## Anotace

Projekt meteostanice má za cíl měřit počasí. Naměřené hodnoty poté poslat do databáze a následně například na webovou stránku. Lze to použít pro monitorování teploty venku nebo měření vlhkosti v domácnosti.

## Annotation

The weather station project aims to measure the weather. Then send the measured values to a database and then for example to a website. This can be used to monitor outdoor temperature or humidity at home.

## Zadání

Cílem tohoto projektu je připojit sensor DHT11 k Raspberry Pi, a napsat script který tyto údaje pošle do databáze. Poté udělat webovou stránku která by tyto údaje zobrazila uživateli. Dále také zabezpečit škálovatelnost všeho. Například kdybychom chtěli připojit další senzory na měření tlaku, rychlosti větru a podobně.

## Assignment

The goal of this project is to connect the DHT11 sensor to a Raspberry Pi, write a script that sends the data to a database. Then make a website that would display this data to user. Furthermore, to ensure the scalability of everything. For example if we wanted to add additional sensors to measure pressure, wind speed and so on.

## Popis Hardwaru

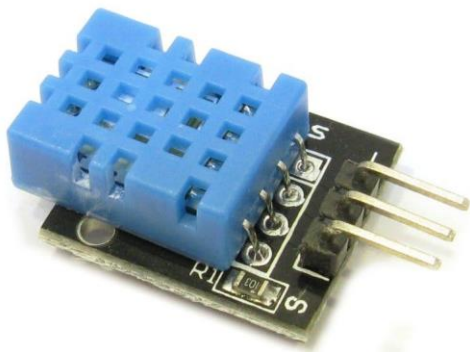
Pro měření teploty a vlhkosti použijeme senzor DHT11. Tento senzor je velice jednoduchý, nepřesný a pomalý, ale pro naše účely stačí. Senzor je vyrobený ze dvou částí, kapacitního čidla vlhkosti a termistoru. Obsahuje také velmi základní čip, který konvertuje analogové údaje na digitální.

Pro ovládání sensoru použijeme Raspberry Pi což je jednočipový počítač s operačním systémem Linux. Používám verzi 3B+, pravděpodobně by však vše fungovalo na starších verzích.

## Hardware Description

To measure temperature and humidity, we will use the DHT11 sensor. This sensor is very simple, inaccurate and slow but for our purposes it will be enough. The sensor is a two-part sensor, a capacitive humidity sensor and a thermistor. It also includes a basic chip that converts analog data to digital.

To control the sensor we use Raspberry Pi which is a single-chip computer with Linux operating system. I use the 3B+ version, but it would probably work on older versions.



Obrázek / Image 1 - Sensor DHT11



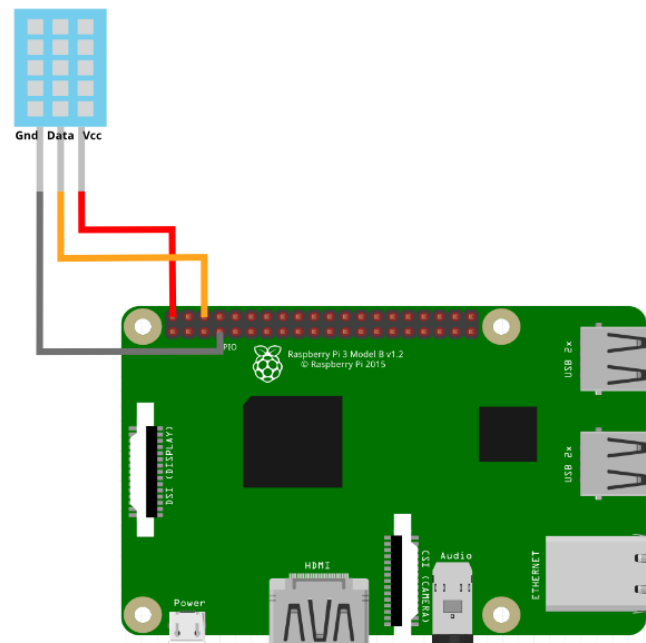
Obrázek / Image 2 - Raspberry Pi 3B+

## Zapojení

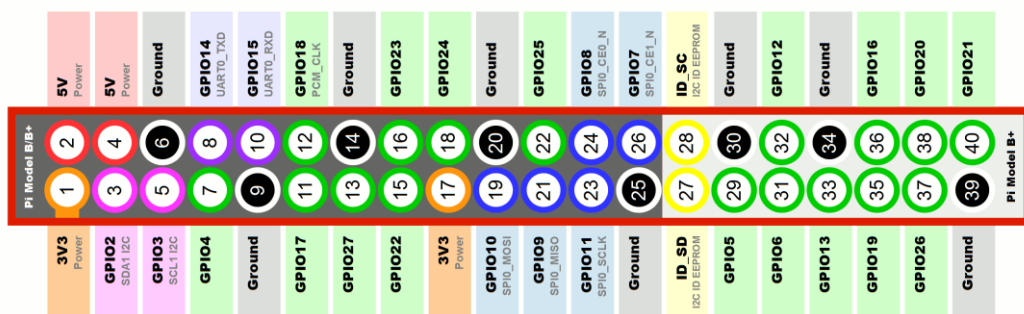
Sensor má 3 piny, uzemnění, data a napájení. Ty se připojí k Raspberry Pi podle následujícího schématu. Popřípadě je možno použít jiné piny podle obrázku 4.

## Connection

Sensor has 3 pins, ground, data and power. These will connect to Raspberry Pi according to the following scheme. Alternatively, any pins according to image 4 may be used.



Obrázek / Image 3 - Scheme



Obrázek / Image 4 - Pinout

## Popis Softwaru

Jako první budeme potřebovat databázi, nejjednodušší cesta je si pronajmout službu RDS od Amazonu, která je zdarma na 12 měsíců a poté pay as you go, neboli zaplatíte jen to co využijete.

Pro sejmutí a odeslání dat ze sensoru do databáze použijeme skript v Pythonu. Využijeme také knihovny Adafruit\_DHT a \_mysql. Tento skript se nastaví jako Cron Job běžící každou hodinu.

Webová stránka pak běží na virtuálním stroji EC2 také od Amazonu. Fungoval by však i například hosting od 000Webhost který podporuje dynamické stránky. Používá HTML, CSS, Javascript a několik knihoven například Chart.js pro grafy počasí a nebo Materialize pro uhlazení designu stránky.

## Software Description

First we need a database, the easiest way is to rent Amazon RDS service which is free for 12 months and after that you pay as you go.

We use a Python script to capture and send sensor data to a database. We will also use Adafruit\_DHT and \_mysql libraries. This script will be set up as a Cron Job running hourly.

The website also runs from Amazon on the EC2 virtual machine. However, hosting from 000Webhost would also work as it supports dynamic sites. We use HTML, CSS, Javascript, and several libraries, such as Chart.js for weather charts, or Materialize to polish the design of the website.

## Ukázky Kódu / Code Samples

Toto je ukázkou MySQL kódu na vytvoření tabulky.

This is a sample MySQL code to create a table.

```
create table weather_Station (  
    ID int not null AUTO_INCREMENT,  
    date date not null,  
    temperature int not null,  
    humidity int not null,  
    PRIMARY KEY (ID)  
);
```

Toto je ukázkou Python kódu který odesílá data ze sensoru do databáze.

This is a Python code sample that sends data from the sensor to the database.

```
connection = _mysql.connect(host, user, passwd, db, port)  
  
humidity, temperature = Adafruit_DHT.read_retry(sensor, pin)  
  
connection.query('insert into`weather_dashboard`  
(date,temperature,humidity) ' + 'values(\'\' + date +'\','+  
str(int(temperature)) +','+ str(int(humidity)) +')')
```

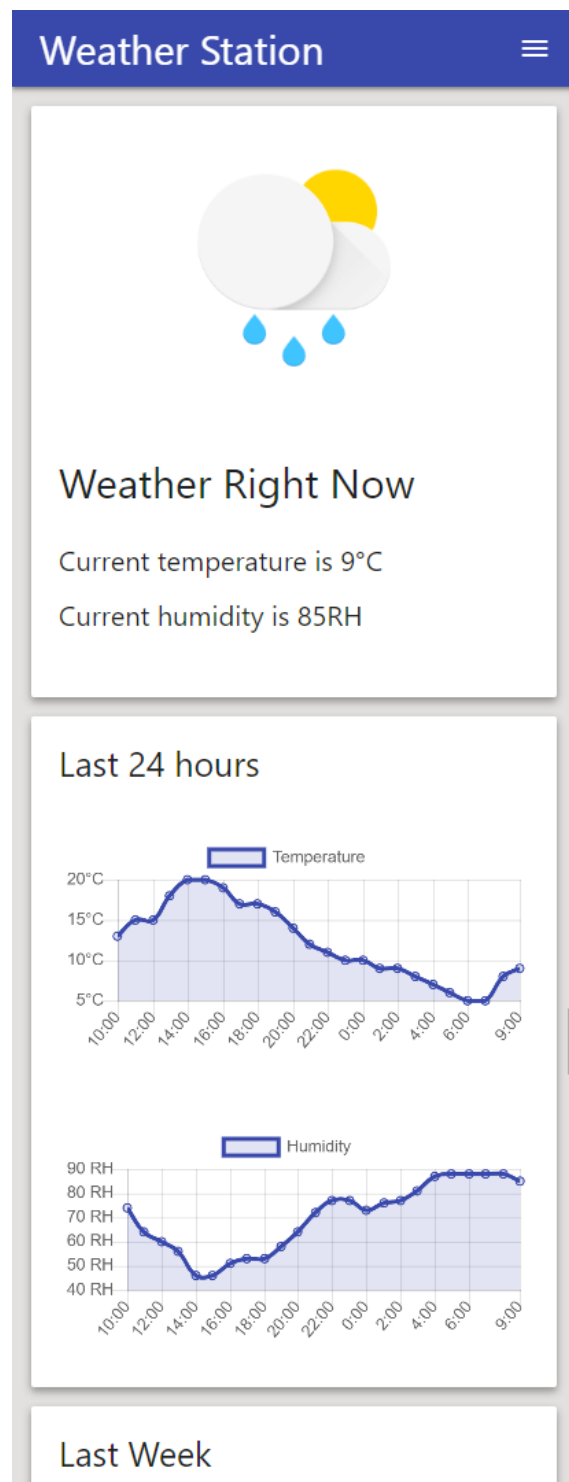
Toto je ukázkou Javascript kódu který žádá PHP o data z databáze.

This is a sample Javascript code that asks PHP for data from the database.

```
var oReq = new XMLHttpRequest();  
oReq.open("get", "db.php", true);  
oReq.send();  
  
oReq.onreadystatechange = function () {  
    if (oReq.readyState === 4) {  
  
        database = JSON.parse(this.responseText);
```



## Ukázka Webové Stránky



2019-04-07 02:00:02	9	77
2019-04-07 01:00:02	9	76
2019-04-07 00:00:02	10	73
2019-04-06 23:00:02	10	77
2019-04-06 22:00:02	11	77
2019-04-06 21:00:02	12	72
2019-04-06 20:00:02	14	64
2019-04-06 19:00:02	16	58
2019-04-06 18:00:02	17	53
2019-04-06 17:00:01	17	53
2019-04-06 16:00:01	19	51
2019-04-06 15:00:02	20	46
2019-04-06 14:00:01	20	46
2019-04-06 13:00:01	18	56
2019-04-06 12:00:01	15	60
2019-04-06 11:00:01	15	64
2019-04-06 10:00:02	13	74

All data is collected by my Raspberry Pi using the DHT11 sensor once per hour.

Temperature readings are between 0-50°C with  $\pm 2^\circ\text{C}$  accuracy.

Humidity readings are between 20-80RH with  $\pm 5\text{RH}$  accuracy.

© 2019 Daniel Šfma

## Závěr

S projektem nebyly žádné větší problémy vzhledem k znalostem Linuxu a webových stránek ze školy a k mým zkušenostem s Raspberry Pi. Jediné s čím jsem zkušenosti neměl bylo PHP a jeho implementování s Javascriptem. Oproti ostatním projektům jsem to měl velice jednoduché a vše šlo velice rychle a bez větších problémů.

## Conclusion

There were no major problems with the project due to the knowledge of Linux and web development from school and my experience with Raspberry Pi. The only thing I had no experience with was PHP and its implementation with Javascript. Compared to other projects I had it very simple and everything went very fast and without any major problems.

## Zdroje / Sources

<https://learn.adafruit.com/dht/overview>

<http://www.circuitbasics.com/how-to-set-up-the-dht11-humidity-sensor-on-the-raspberry-pi/>

<http://mysql-python.sourceforge.net/MySQLdb.html>

<https://www.geeksforgeeks.org/crontab-in-linux-with-examples/>

[https://www.w3schools.com/php/php\\_mysql\\_connect.asp](https://www.w3schools.com/php/php_mysql_connect.asp)

[https://www.w3schools.com/js/js\\_ajax\\_php.asp](https://www.w3schools.com/js/js_ajax_php.asp)

<https://www.chartjs.org/doc>

<https://github.com/cferdinandi/smooth-scroll>

<https://materializecss.com/getting-started.html>

<https://www.w3schools.com>

<https://stackoverflow.com>

Obrázek / Image 1

[https://hackster.imgix.net/uploads/attachments/665283/blob\\_akU3N5knLQ.blob?auto=compress%2Cformat&w=900&h=675&fit=min](https://hackster.imgix.net/uploads/attachments/665283/blob_akU3N5knLQ.blob?auto=compress%2Cformat&w=900&h=675&fit=min)

Obrázek / Image 2

<https://www.raspberrypi.org/app/uploads/2017/05/Raspberry-Pi-3-462x322.jpg>

Obrázek / Image 3

<http://www.markomarinkovic.in.rs/assets/img/dht11.png>

Obrázek / Image 4

<https://www.raspberrypi-spy.co.uk/wp-content/uploads/2012/06/Raspberry-Pi-GPIO-Layout-Model-B-Plus-rotated-2700x900.png>

Kód: <https://drive.google.com/drive/folders/1miw6mB3uLNKj0EfKxwX-ENT2kh4YEFTI>