

Izdelava spletne strani za merjenje temperature in vlage z Raspberry Pi

Raspberry Pi

Raspberry Pi (RPi) je mikroračunalnik v velikosti kreditne kartice. Tretja generacija računalnika Raspberry je opremljena z zmogljivim štirijedrnim procesorjem ARM Cortex-A7 s 4 x 1,2 GHz. S to nadgradnjo vam računalnik Pi® 3 nudi bistveno večjo zmogljivost pri določenih aplikacijah. Za hitro in enostavno medsebojno povezavo sta prvič integrirana WLAN in Bluetooth Low Energy. Razporeditev in priključki računalnika Raspberry Pi® 3 so identični kot pri Raspberry Pi® 2, model B in Raspberry Pi® B+.

Dodatna prednost računalnika Raspberry Pi® 3 je njegova absolutna vsestranskost. Lahko ga uporabite kot mini računalnik v svoji dnevni sobi ter prek HDMI-izhoda predvajate Full HD videe ali pa ga prek vrat GPIO povežete z ostalimi tiskanimi vezji in udejanjite krmilne sisteme.

Namesto na trdi disk Raspberry Pi® 3 zaganja in shranjuje podatke prek kartice microSD, ki jo vstavite v režo za kartico microSD na spodnji strani tiskanega vezja. Napajanje enostavno poteka prek priključka Micro USB.



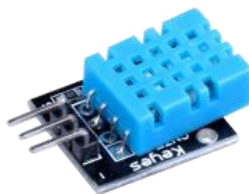
Slika 1 Mikroračunalnik Raspberry Pi

Digitalni senzor DHT11

Digitalni senzor temperature in vlage DHT11 ima območje merjenja temperature od 0 do 50 °C, vlage pa od 20 do 90 %. Senzor ima tri priključke:

- napajanje (5V),
- GND (-) in
- signal (S).

Deluje pri napetosti 5V, njegove meritve pa so na desetinko natančne.



Slika 2 Digitalni senzor DHT11

Priprava Raspberry Pi

Da bo Raspberry Pi uporaben, je potrebno nanj namestiti ustrezen operacijski sistem in programsko opremo.

Priporočen operacijski sistem za uporabo z Raspberry Pi se imenuje Raspbian. Raspbian je različica GNU/Linux, ki je zasnovana tako, da posebej dobro deluje z Raspberry Pi. Obstaja več možnosti, kako dobiti kopijo operacijskega sistema Raspbian.

Najprej prenesemo slikovno datoteko operacijskega sistema (**raspbian.img**) in jo zapišemo na SD kartico. SD kartico pred tem formatiramo (uporabimo lahko program **SD_Card_Formater**). Nato s programom **Etcher** naložimo slikovno datoteko na SD kartico. Ko je SD kartica pripravljena, jo vstavimo v Raspberry Pi in ga vključimo.

Za dostop od raspberry pi računalnika lahko uporabimo oddaljeni dostop, za prenos datotek na raspberry pa ftp protokol. Najprej moramo omogočiti SSH. V terminalskem oknu z ukazom **sudo raspi-config** zaženemo konfiguracijske nastavitve za Raspberry Pi. Izberemo **Interfacing Options** in omogočimo **SSH (Enable)** ter izberemo **Finish**. Nato namestimo še oddaljeni dostop:

```
sudo apt-get remove xrdp vnc4server tightvncserver
sudo apt-get install tightvncserver
sudo apt-get install xrdp
```

Da bomo lahko gostili spletno aplikacijo, moramo najprej namestiti spletni strežnik **Apache** in **PHP**:

```
sudo apt install apache2 -y
sudo apt install php -y
sudo service apache2 restart
```

Nato namestimo še **podatkovni strežnik** in **phpMyAdmin** ter kreiramo uporabnika za dostop do strežnika:

```
sudo apt install mariadb-server php-mysql -y
sudo service apache2 restart
sudo mysql_secure_installation
sudo apt install phpmyadmin -y
sudo phpenmod mysqli
sudo apt install php-mysql
sudo service apache2 restart
sudo ln -s /usr/share/phpmyadmin /var/www/html/phpmyadmin
sudo mysql -u root -p
MariaDB> CREATE USER 'username'@'localhost' IDENTIFIED BY password;
MariaDB > GRANT ALL PRIVILEGES ON *.* TO 'username'@'localhost';
MariaDB > FLUSH PRIVILEGES;
```

Za delo s senzorji in ostalimi napravami, ki jih priključimo na Raspberry Pi, namestimo programski jezik **Python**, v katerem bomo programirali naprave.

```
sudo apt-get install python3-dev **default-libmysqlclient-dev**  
  
sudo pip3 install mysqlclient  
  
sudo apt install python3-mysql.connector  
  
pip install mysql-connector-python
```

Če želimo podatke s senzorja shraniti v podatkovno bazo, moramo v pythonov program vključiti ustrezne knjižnice:

```
import mysql.connector  
  
db = mysql.connector.connect(host="localhost", user="username", passwd="password",  
db="senzor_dht")
```

Spletno aplikacijo bomo ustvarili na svojem računalniku in jo bomo preko ftp protokola prenesli na Raspberry Pi. Korenska mapa spletnega strežnika **Apache** je `/var/www/html`. Za zapisanje v to mapo moramo spremeniti pravice:

```
ls -lh /var/www/  
  
sudo chown -R pi:www-data /var/www/html/  
  
sudo chmod -R 770 /var/www/html/  
  
ls -lh /var/www/
```


V naslednjem koraku pripravimo zbirko podatkov.

Priprava zbirke podatkov

Izdelamo novo zbirko podatkov z imenom **senzor_dht** in ustvarimo tabelo **meritve** z naslednjimi podatki:

- id (celo število, primarni ključ, auto_increment);
- datum (date);
- ura (time);
- temperatura (float);
- vlaga (float);
- prostor (varchar(30)).

Ustvarimo novega uporabnika: **PHPvaje, Februar2020!**. To vse naredimo z orodjem phpMyAdmin (v brskalnik vpišemo **localhost/phpmyadmin** in se prijavimo z uporabniškim imenom in geslom, ki smo ga določili pri kreiranju dostopa do strežnika).

Name	Type	Collation	Attributes	Null	Default	Comments	Extra
id 	int(11)			No	None		AUTO_INCREMENT
datum	date			No	None		
ura	time			No	None		
temperatura	float			No	None		
vlaga	float			No	None		
prostor	varchar(30)	utf8_slovenian_ci		No	None		

Slika 3 Tabela meritve

Izdelava spletne strani

Priprava vsebine

Spletno stran izdelamo v skriptnem jeziku PHP. Za oblikovanje spletne strani uporabimo ogrodje Bootstrap. Pod naslovno vrstico (Raspberry Pi Vaje) izdelamo tri stolpce:

- v prvem stolpcu je opis vaje,
- drugi stolpec prikazuje meritve v obliki tabele,
- tretji stolpec pa prikazuje zadnjo meritev.

V tretjem stolpcu je tudi gumb **Izmeri** s katerim preberemo trenutno temperaturo in vlago. Zraven je zapisan tudi datum in čas meritve.



Raspberry Pi Vaje

Merjenje temperature in vlažnosti

Za merjenje smo uporabili **Raspberry Pi** mikroračunalnik in senzor temperature in vlažnosti **DHT11**. V programskem jeziku **Python** smo napisali program za Raspberry Pi, ki bere podatke s senzorja in jih zapisuje v podatkovno bazo. Podatki se preberejo in zapišejo vsakih 30 minut.




Mikroračunalnik Raspberry Pi

Senzor temperature in vlažnosti DHT11

Meritve

Stran 1 od 12

#	Datum	Ura	Temperatura	Vlažnost
1	2020-02-28	17:32:56	22	11
2	2020-02-28	15:15:39	21	11
3	2020-02-28	13:39:49	21	11
4	2020-02-27	20:36:46	20	12
5	2020-02-27	17:39:00	20	12

◀ ◁ ▷ ▶

Zadnja meritev

Čas meritve:
2020-02-28 17:32:56

Temperatura:
22°C

Vlažnost:
11%

Izmeri

Slika 4 Prikaz spletne strani

Glavni bootstrap slog na spletni strani je **info**, za gumbje pa **dark**.

Priloga

Povezava z bazo:

```
function connect_mysql()
{
    $Streznik = "localhost";
    $Up_ime = "PHPvaje";
    $Geslo = "Februar2020!";
    $Baza = "senzor_dht";
    $conn = new mysqli($Streznik, $Up_ime, $Geslo, $Baza);
    return $conn;
}
```

Branje podatkov iz tabele:

```
function meritve($page, $per_page)
{
    $tabela = "";
    $offset = ($page-1)*$per_page;
    $conn = connect_mysql();
    $sql = "SELECT * FROM meritve ORDER BY id DESC LIMIT ".$offset.", ".$per_page;
    $result = $conn->query($sql);
    $tabela = $tabela."<table class='table table-bordered table-hover'>";
    $tabela = $tabela."<thead>";
    $tabela = $tabela."<tr class='bg-info text-white-50 text-center'>";
    $tabela = $tabela."<th>#</th>";
    $tabela = $tabela."<th>Datum</th>";
    $tabela = $tabela."<th>Ura</th>";
    $tabela = $tabela."<th>Temperatura</th>";
    $tabela = $tabela."<th>Vlažnost</th>";
    $tabela = $tabela."</tr>";
    $tabela = $tabela."</thead>";
    $tabela = $tabela."<tbody>";
    $st = ($page-1)*$per_page;
    while ($row = mysqli_fetch_assoc($result))
    {
        $st++;
        $tabela = $tabela."<tr class='text-center'>";
        $tabela = $tabela."<td>".$st."</td>";
        $datum = $row['datum'];
        $tabela = $tabela."<td>".$datum."</td>";
        $ura = $row['ura'];
        $tabela = $tabela."<td>".$ura."</td>";
        $temperatura = $row['temperatura'];
        $tabela = $tabela."<td>".$temperatura."</td>";
        $vlaga = $row['vlaga'];
        $tabela = $tabela."<td>".$vlaga."</td>";
        $prostor = $row['prostor'];
        $tabela = $tabela."</tr>";
    }
    $tabela = $tabela."</tbody>";
    $tabela = $tabela."</table>";
    $conn->close();
    echo $tabela;
}
```

Zadnja meritev:

```
function zadnja_meritev()
{
    $conn = connect_mysql();
    $sql = "SELECT * FROM meritve ORDER BY id DESC LIMIT 1";
    $result = $conn->query($sql);
    $row = mysqli_fetch_assoc($result);
    $datum = $row['datum'];
    $ura = $row['ura'];
    $temperatura = $row['temperatura'];
    $vlaga = $row['vlaga'];
    $prostor = $row['prostor'];
    $conn->close();
    echo "<p class='lead'>Čas meritve: <br><span class='text-white'><strong>\".$datum.\"
<small>\".$ura.\"</small></strong></span></p>";
    echo "<p class='lead'>Temperatura: <br><span class='badge badge-info text-center'
style='font-size: x-large'>\".$temperatura.\"°C</span></p>";
    echo "<p class='lead'>Vlažnost: <br><span class='badge badge-info text-center'
style='font-size: x-large'>\".$vlaga.\"%</span></p>";
}
```

Število meritev:

```
function st_meritev()
{
    $conn = connect_mysql();
    $sql = "SELECT COUNT(*) FROM meritve";
    $result = $conn->query($sql);
    $total_rows = mysqli_fetch_array($result)[0];
    $conn->close();
    return $total_rows;
}
```

Branje podatkov s senzorja in zapis podatkov v bazo (python):

```
#!/usr/bin/env python
import Adafruit_DHT
import time
import sys
import mysql.connector
import datetime

DHT_SENSOR = Adafruit_DHT.DHT11
DHT_PIN = 4

vlaga, temperatura = Adafruit_DHT.read(DHT_SENSOR, DHT_PIN)
if vlaga is not None and temperatura is not None:
    print("Temp={0:0.1f}C Humidity={1:0.1f}%".format(temperatura, vlaga))
    i = datetime.datetime.now()
    year = i.strftime("%Y")
    month = i.strftime("%m")
    day = i.strftime("%d")
    datum = year + "-" + month + "-" + day
    hour = i.strftime("%H")
    minute = i.strftime("%M")
    second = i.strftime("%S")
    ura = hour + ":" + minute + ":" + second
    db = mysql.connector.connect(host="localhost", user="PHPvaje",
passwd="Februar2020!", db="senzor_dht")
    # create a cursor for the select
    cur = db.cursor()
    cur.execute("INSERT INTO meritve (datum, ura, temperatura, vlaga, prostor) VALUES
('" + datum + "', '" + ura + "', " + str(temperatura) + ", " + str(vlaga) + ", 'dnevna
soba')")
    db.commit()
    cur.close()
    # close the connection
    db.close()
else:
    print("Sensor failure. Check wiring.")
```

Ocenjevanje

Ocenjuje se tako oblika kot funkcionalnost. Pri **obliki** se ocenjuje uporaba ustreznih elementov in slogov ter ujemanje s predlogo:

- naslov (10 točk)
- menijska vrstica (10 točk)
- števec obiskov (10 točk)
- vabilo (10 točk)
- kontakt (10 točk)

Pri **funkcionalnosti** se ocenjuje uporaba ustreznih metod, postopkov in ukazov:

- kontaktni obrazec (10 točk)
- prikaz števca (10 točk)
- povečanje števca (10 točk)
- meni (10 točk)
- izpis datuma zadnjega obiska (10 točk)

Točkovanje:

Oblika			
št. možnih točk	element	točkovnik	točke
10	naslov	ikona, besedilo, ozadje, link	
10	opis	naslov, črta, besedilo, keyboard	
10	prikaz slik	razporeditev slik, oblika slik, caption	
10	tabela	naslovna vrstica, telo, strani, število meritev	
10	zadnja meritev	card, header, body, footer, gumb	
50	SKUPAJ		
Funkcionalnost			
št. možnih točk	element	točkovnik	točke
10	meritev	branje stanja senzorja	
10	tabela	branje podatkov iz tabele in prikaz podatkov	
10	številčenje	izdelava krmiljenja po straneh	
10	strani	prikaz trenutne številke strani in število vseh strani	
10	gumb	izvedba meritve	
50	SKUPAJ		
100	SKUPAJ		

Kriterij ocenjevanja:

Število možnih točk:		100
točke (meje)	točke (zgornja meja)	ocena
0 - 49	49 % = 49,00	nezadostno (1)
50 - 62	62 % = 62,00	zadostno (2)
63 - 75	75 % = 75,00	dobro (3)
76 - 88	88 % = 88,00	prav dobro (4)
89 - 100	100 % = 100,00	odlično (5)

Načrtovanje in razvoj spletnih aplikacij

Priimek in ime: _____

Datum: _____

Število točk: ____ / 100

Ocena: _____