



Univerza v Mariboru

Fakulteta za energetiko

Tomaž Hozjan

Načrtovanje povezave mikrokrmilnika s spletno stranjo

Diplomsko delo

Velenje, september 2023

Načrtovanje povezave mikrokrmilnika s spletno stranjo

Diplomsko delo

Študent: Tomaž Hozjan

Študijski program: Visokošolski strokovni študijski program 1. stopnje Energetike

Mentor: red. prof. dr. Peter Virtič

Somentor: doc. dr. Franjo Pranjic

Lektor: Maksimiljan Kobal, uni. dipl. teol.



ZAHVALA

*Zahvaljujem se mentorju in somentorju,
ki sta mi pomagala pri diplomskem delu
in mi vedno dajala nove ideje.*

NAČRTOVANJE POVEZAVE MIKROKRMILNIKA S SPLETNO STRANJO

Ključne besede: mikrokrmilnik, senzorji, povezave, spletna stran, 3D oblikovanje

UDK: 004.42:528.835(043.2)

Povzetek

V diplomski nalogi smo prikazali povezavo med senzorji in spletno stranjo. Njen namen je bil prikazati vidno in nevidno (digitalno) povezavo med samim začetkom (senzorjem), krmilnikom, programom, podatkovno bazo (lokalno in na daljavo povezano podatkovno bazo) ter spletno stranjo. V diplomski nalogi smo opisali tudi vezja, programe in 3d modele.

Na začetku diplomske naloge smo kratko opisali osnovo: krmilnike, senzorje in jezike, ki smo jih uporabil pri izvedbi diplomske naloge. Sledi opis vezja, osnovnega delovanja, natančnega delovanja in programov (na primer VirtualStudio, Atom, ...) ter opis 3d oblikovanja in tiskanja, prikaz podatkov na spletni strani ter priprav za začetek izvedbe te diplomske naloge.

DESIGN OF MICROCONTROLLER CONNECTION TO THE WEBSITE

Key words: microcontroller, sensors, connections, website, 3D design.

UDC: 004.42:528.835(043.2)

Abstract

The thesis itself shows the connection between the sensors and the website. The purpose of the thesis is the visible and invisible (digital) connection between the beginning itself (sensor) connected to the controller, the program, the database (local and remote database) and finally the website. In the thesis itself you can find circuits, programs and 3d models.

At the beginning of the thesis, the basis is briefly described (controller, sensors and languages that I used in the actual execution of the thesis). Then there's the circuit, basic operation, precise operation, programs (for example VirtualStudio, Atom). Finally, 3d design and printing, how to display the data on the website, and preparations for the actual implementation of this thesis.

IZJAVA O AVTORSTVU



Fakulteta za energetiko
(ime članice UM)

IZJAVA O AVTORSTVU ZAKLJUČNEGA DELA

Ime in priimek študenta/-ke: Tomaž Hozjan

Študijski program: ENERGETIKA

Naslov zaključnega dela: Načrtovanje povezave mikrokrmilnika s spletno stranjo

Mentor/-ica: red. prof. dr. Peter Virtič

Somentor/-ica: doc. dr. Franjo Pranjic

Podpisani/-a študent/-ka Tomaž Hozjan

- izjavljam, da je zaključno delo rezultat mojega samostojnega dela, ki sem ga izdelal/-a ob pomoči mentorja/-ice oz. somentorja/-ice;
- izjavljam, da sem pridobil/-a vsa potrebna soglasja za uporabo podatkov in avtorskih del v zaključnem delu in jih v zaključnem delu jasno in ustrezno označil/-a;
- na Univerzo v Mariboru neodplačno, neizključno, prostorsko in časovno neomejeno prenašam pravico shranitve avtorskega dela v elektronski obliki, pravico reproduciranja ter pravico ponuditi zaključno delo javnosti na svetovnem spletu preko DKUM; sem seznanjen/-a, da bodo dela, deponirana/objavljena v DKUM, dostopna široki javnosti pod pogoji licence Creative Commons BY-NC-ND, kar vključuje tudi avtomatizirano indeksiranje preko spleta in obdelavo besedil za potrebe tekstovnega in podatkovnega rudarjenja in ekstrakcije znanja iz vsebin; uporabnikom se dovoli reproduciranje brez predelave avtorskega dela, distribuiranje, dajanje v najem in priobčitev javnosti samega izvirnega avtorskega dela, in sicer pod pogojem, da navedejo avtorja in da ne gre za komercialno uporabo;
- dovoljujem objavo svojih osebnih podatkov, ki so navedeni v zaključnem delu in tej izjavi, skupaj z objavo zaključnega dela.

Uveljavljam permisivnejšo obliko licence Creative Commons: _____
(navedite obliko)

Kraj in datum:
Velika Počana, 11.9.2023

Podpis študenta/-ke:

IZJAVA O ISTOVETNOSTI TISKANE IN ELEKTRONSKE OBLIKE ZAKLJUČNEGA DELA



Fakulteta za energetiko
(ime članice UM)

IZJAVA O ISTOVETNOSTI TISKANE IN ELEKTRONSKE OBLIKE ZAKLJUČNEGA DELA

Ime in priimek študenta/-ke: Tomaž Hozjan

Študijski program: ENERGETIKA

Naslov zaključnega dela: Načrtovanje povezave mikrokrmilnika s spletno stranjo

Mentor/-ica: red. prof. dr. Peter Vrtič

Somentor/-ica: doc. dr. Franjo Pranjic

Podpisani/-a študent/-ka Tomaž Hozjan
izjavljam, da je tiskana oblika zaključnega dela istovetna elektronski obliki zaključnega dela, ki sem jo oddal/-a za objavo v DKUM.

Kraj in datum:

Velika Polana, 11.9.2023

Podpis študenta/-ke:

Hozjan

KAZALO VSEBINE

ZAHVALA	III
NAČRTOVANJE POVEZAVE MIKROKRMILNIKA S SPLETNO STRANJO	IV
DESIGN OF MICROCONTROLLER CONNECTION TO THE WEBSITE	V
IZJAVA O AVTORSTVU	VI
IZJAVA O ISTOVETNOSTI TISKANE IN ELEKTRONSKE OBLIKE ZAKLJUČNEGA DELA	VII
KAZALO VSEBINE	VIII
KAZALO SLIK	XI
UPORABLJENI SIMBOLI	XIII
UPORABLJENE KRATICE	XIV
1 UVOD	1
2 TEORETIČNI DEL (OSNOVE PROGRAMSKIH JEZIKOV, PREDSTAVITEV SENZORJEV, KRMILNIKA, MOTORJA).....	2
2.1 OSNOVE PROGRAMSKIH JEZIKOV	2
2.1.1 HTML	2
2.1.2 CSS.....	3
2.1.3 PHP.....	4
2.1.4 SQL	5
2.1.5 JAVASCRIPT	6
2.1.6 C++ ALI Arduino IDE.....	7
2.2 SENZORJI.....	8
2.2.1 SENZOR TDS	9
2.2.2 FOTUOPOR LDR.....	10
2.2.3 SENZOR VLAGE ZEMLJE.....	11

2.2.4	SENZOR DHT 11	12
2.2.5	SENZOR VODE	12
2.3	MIKROKRMILNIK	13
2.4	MOTOR	14
2.5	VEZJE ARDUINO ETHERNET SHIELD	15
3	VEZJE.....	18
3.1	VEZJE ZA SENZORJE OZIROMA VEZJE ZA PRENAŠANJE PODATKOV.....	18
3.2	VEZJE ZA MOTOR	20
4	OSNOVNO DELOVANJE	22
5	NATANČNO DELOVANJE.....	24
5.1	PODATKI	24
5.2	SPLETNA STRAN	26
6	PROGRAMI ALI ORODJA	28
6.1	ARDUINO	28
6.2	ATOM.....	29
6.3	VISUAL STUDIO CODE	29
6.4	SOLIDWORKS	30
6.5	CREALITY SLICER	30
6.6	GITHUB DESKTOP.....	31
6.7	HEROKU	31
6.8	XAMPP	32
7	3D OBLIKOVANJE IN TISKANJE	33
7.1	OSNOVNO OHIŠJE.....	34
8	PRIKAZ PODATKOV NA SPLETNI STRANI.....	40
9	PRIPRAVE.....	43
10	ZAKLJUČEK	45
	VIRI IN LITERATURA.....	46
	PRILOGE	48

PRILOGA A: PROGRAMI48

KAZALO SLIK

Slika 2.1: Prikaz programa CSS s spletne strani [Lastni vir]	4
Slika 2.2: Prikaz programa PHP s spletne strani [Lastni vir].....	5
Slika 2.3: Prikaz tabele in stolpcev pri lokalni podatkovni bazi (phpmyadmin) [Lastni vir]	5
Slika 2.4: Prikaz orodja Arduino IDE [Lastni vir].....	8
Slika 2.5: Senzor TDS [Lastni vir]	9
Slika 2.6: Fotoupor LDR [Lastni vir]	10
Slika 2.7: Senzor vlage zemlje [Lastni vir]	11
Slika 2.8: Senzor DHT 11 [Lastni vir]	12
Slika 2.9: Senzor vode [Lastni vir]	13
Slika 2.10: Mikrokrmilnik Arduino UNO [Lastni vir].....	14
Slika 2.11: Motor DC RS-755 [Lastni vir]	15
Slika 2.12: Prikaz lokalnega IP-ja na nastavitvah [Lastni vir]	16
Slika 2.13: Prikaz lokalnega IP-ja na usmerjevalniku [Lastni vir]	16
Slika 2.14: Prikaz lokalnega IP-ja na CMD [Lastni vir]	17
Slika 2.15: Vežje Arduino Ethernet Shield [Lastni vir].....	17
Slika 3.1: Vodni senzor, baterija in motor [Lastni vir].....	18
Slika 3.2: Skica vezja za senzorjev [Lastni vir]	19
Slika 3.3: Skica vezja za motor [Lastni vir]	21
Slika 4.1: Skica prenosa podatkov od senzorja do spletne strani [Lastni vir].....	22
Slika 4.2: Skica prenosa spletne strani od Visual Studio code do serverja Heroku [Lastni vir]	23
Slika 5.1: Prikaz programa za mikrokrmilnik Arduino [Lastni vir].....	25
Slika 5.2: Prikaz programa med Arduino in lokalno podatkovno bazo [Lastni vir].....	25
Slika 5.3: Prikaz tabele s podatki [Lastni vir].....	26

Slika 5.4: Skica prenosa spletne strani od Visual Studio code do serverja Heroku [Lastni vir]	27
Slika 6.1: Ikona programskega orodja Arduino [9]	28
Slika 6.2: Ikona programskega orodja Atom [10]	29
Slika 6.3: Ikona programskega orodja Visual Studio code [11]	29
Slika 6.4: Ikona programskega orodja SOLIDWORKS [12]	30
Slika 6.5: Ikona programskega orodja CREALITY SLICER [13]	30
Slika 6.6: Ikona programskega orodja GitHub [14]	31
Slika 6.7: Ikona strežnika Heroku [15]	32
Slika 6.8: Ikona strežnika XAMPP [16]	32
Slika 7.1: Tiskalnika Ender 3 in Creality CR-6 SE [Lastni vir].....	33
Slika 7.2: Prikaz osnovnega ohišja [Lastni vir].....	34
Slika 7.3: Prikaz osnovnega ohišja motorja [Lastni vir].....	35
Slika 7.4: Prikaz osnovnega ohišja motorja [Lastni vir].....	35
Slika 7.5: Prikaz osnovnega premikajočega se dela motorja [Lastni vir]	36
Slika 7.6: Prikaz osnovnega premikajočega se dela motorja [Lastni vir]	36
Slika 7.7: Ohišja baterije [Lastni vir].....	37
Slika 7.8: Prikaz zaščite za noge [Lastni vir]	37
Slika 7.9: Prikaz noge [Lastni vir].....	38
Slika 7.10: Prikaz noge in njene zaščite [Lastni vir].....	39
Slika 8.1: Prikaz izpisa tabele in grafa na spletni strani [Lastni vir]	40
Slika 8.2: Prikaz izpisa tabele in grafa na spletni strani [Lastni vir]	41
Slika 8.3: Prikaz script kode za graf [Lastni vir]	41
Slika 8.4: Prikaz kode php za tabelo [Lastni vir].....	42
Slika 9.1: Prikaz osnovnih orodij za pripravo diplomske naloge [Lastni vir].....	44

UPORABLJENI SIMBOLI

<i>ppm</i>	-	število delov na milijon
V	-	volt
Ω	-	om
mg/L	-	miligrami na liter

UPORABLJENE KRATICE

CSS	-	Cascading Style Sheets
C++	-	C plus plus
CPP	-	C plus plus
CMD	-	Command Prompt
HTML	-	HyperText Markup Language
IP	-	Internet Protocol
JS	-	JAVASCRIPT
PHP	-	Hypertext preprocessor
PLA	-	Polimlečna kislina
SQL	-	Structured Query Language

1 UVOD

Takojšen prikaz informacij (v živo) je zelo pomemben, saj se le tako lahko v primeru kakršne koli napake nanje ustrezno in hitro odzivamo. Navadno se informacije prikažejo s pomočjo naprave, ki jo sestavljajo senzor, vezje in zaslon. Poskrbeti je treba, da se podatki oziroma informacije ne prikazujejo samo na določeni lokaciji oziroma samo na zaslonu naprave. Če se namreč prikazujejo samo na zaslonu, je lahko reakcija na problem zakasnjena. Zato je zelo pomembno, da je naprava povezana tudi na lokalno oziroma javno (podatkovno) bazo ali omrežje.

Lahko rečemo, da je največje javno omrežje internet, saj na njem najdemo skoraj vse informacije.

Internet navadno povezuje spletne strani, ki nam omogočajo dostop do potrebnih informacij. Na primer, če gremo na spletno stran trgovine, mora njena spletna stran imeti v ozadju podatkovno bazo; če je nima, je taka spletna stran nezanesljiva.

Prenos podatkov prek interneta se čedalje bolj povečuje, zato je pomembno, da se zavarujemo pred morebitnimi napadi. Če je le mogoče, naj se občutljivi podatki prikazujejo samo določenim osebam oziroma samo na določenih lokacijah. Zaradi tega se podatki po navadi prikazujejo samo v lokalnih omrežjih. Ta omrežja imajo tudi svoje lokalne podatkovne baze. Podatkovna baza je baza, v kateri so shranjeni in »prikazani« podatki oziroma informacije. Podatki so navadno prikazani v obliki števil, pridobljeni pa so na podlagi izmerjenih podatkov. Informacije oziroma podatki so izračunani, saj senzorji sami ne prikazujejo »čistih podatkov«. Po navadi senzorji izkoriščajo električno vezje ali reakcije, ki potem te izmerijo. Zato je pomembno, da so senzorji in motorji povezani s krmilnikom, da lahko z njim manipuliramo.

2 TEORETIČNI DEL (OSNOVE PROGRAMSKIH JEZIKOV, PREDSTAVITEV SENZORJEV, KRMILNIKA, MOTORJA)

2.1 OSNOVE PROGRAMSKIH JEZIKOV

Pomemben del diplomske naloge je opis programov. Ti so sestavljeni iz programskih ukazov, na podlagi katerih izvedejo določeno funkcijo. Pri izvedbi diplomske naloge smo uporabili naslednje vrste programskih jezikov:

- HTML - HyperText Markup Language
- CSS - Cascading Style Sheets
- PHP - Hypertext preprocessor
- SQL - Structured Query Language
- JS - JAVASCRIPT
- C++ - C plus plus

Programske jezike HTML, CSS, PHP, SQL in JAVASCRIPT smo uporabili pri izdelavi spletne strani. PHP in SQL smo potrebovali za vzpostavitev povezave med krmilnikom in podatkovno bazo, programski jezik C++ pa smo uporabili pri mikrokrmilniku.

2.1.1 HTML

HTML je kratica za *Hyper Text Markup Language*. Glavni del tega programskega jezika je osnovna zgradba same spletne strani. [1] HTML se uporablja po navadi za besedilo,

tabelo in naslove, zato lahko rečemo, da je to osnovni jezik za splet. Osnovni del izgleda tako:

```
<!DOCTYPE html>
<html>
<head>
<title>Ime strani </title>
</head>
<body>
<h1>Naslov1</h1>
<h2>Naslov2</h2>
<h3>Naslov3</h3>
<p>navadno besedilo</p>
</body>
</html>
```

V osnovni obliki je napisano samo besedilo, ki ni oblikovano. Za oblikovanje potrebujemo programski jezik CSS. [2]

2.1.2 CSS

Namen programskega jezika CSS je oblikovanje samega HTML-ja in izboljšanje izgleda spletne strani. Z njim lahko oblikujemo barve, velikost, obliko besedila in drugo. [3] Ker je CSS namenjen za oblikovanje spletne strani, se lahko izkorišča tudi za »risanje« na spletni strani. Z uporabo in določitvijo barv se lahko na spletni strani narišejo figure. Ker pa je »risanje« na spletni strani zelo zahtevno, se temu izogibamo.

CSS se lahko piše tudi v datoteko HTML, in to tako, da se jo označi: <style></style> ali <h1 style="color:green;">Naslov1</h1> [4]. Ker pa to ni prav pregledno, se datoteka HTML poveže z datoteko CSS. Ukaz za povezavo med njima je: <link href="index.css"

rel="stylesheet" type="text/css"> [5]. V CSS ločimo dva različna tipa, in sicer identifikator (*id*) in razred (*class*). Glavna razlika med njima je, da se razred lahko uporablja večkrat (slika 2.1), identifikator pa samo enkrat.

```
.circle{
  height: 100px;
  width: 100px;
  background: #466aed;
  border: 10px solid #2b46a5;
  border-radius: 50%;
  margin-left: 1340px;
  margin-top: -300px;
}

.gradient{
  background: -webkit-radial-gradient(circle, #233c95, #466aed);
}

.Arduino{
  margin-top: 38px;
  margin-left: 15px;
  font-size: 20px;
  text-decoration: none;
  display: inline-block;
  text-align: center;
  color: rgb(255, 255, 255);
}

Arduino: the open f
```

Slika 2.1: Prikaz programa CSS s spletne strani [Lastni vir]

2.1.3 PHP

PHP je kratica za *Hypertext preprocessor* (slika 2.2). To je osnovni skriptni jezik za spletne strani. PHP smo uporabili za povezavo do podatkovne baze in tudi za povezavo med krmilnikom in podatkovno bazo ter med podatkovno bazo in spletno stranjo. Prav tako smo ga uporabili tudi pri izrisu tabel. PHP je eden izmed težjih jezikov, pri njegovi uporabi pa je treba biti zelo natančen.

Namesto datoteke HTML smo v nalogi rajši uporabljali datoteko PHP, saj se v njej lahko uporabljata tudi HTML, CSS. [6]

```

<?php
$cleardb_url = parse_url(getenv("CLEARDB_DATABASE_URL"));
$cleardb_server = $cleardb_url["host"];
$cleardb_username = $cleardb_url["user"];
$cleardb_password = $cleardb_url["pass"];
$cleardb_db = substr($cleardb_url["path"],1);
$active_group = 'default';
$query_builder = TRUE;

$conn = mysqli_connect($cleardb_server, $cleardb_username, $cleardb_password, $cleardb_db);

mysqli_select_db($conn,$cleardb_db);
$stmt=array();

$res=mysqli_query($conn,"select vlagazemlje from temperatura_senzor");
$count=0;
while($row = mysqli_fetch_array($res))
{
$stmt[$count] ["label"]=$count;
$stmt[$count] ["y"]=$row["vlagazemlje"];
$count=$count+1;
}

?>

```

Slika 2.2: Prikaz programa PHP s spletne strani [Lastni vir]

2.1.4 SQL

SQL je kratica za *Structured Query Language*. Ta programski jezik je namenjen za oblikovanje podatkov, tabel in stolpcev. [7]

Osnovna struktura slika 2.3:

The screenshot shows the phpMyAdmin interface for a table named 'temperatura_senzor'. The table structure is displayed as follows:

#	Ime	Vrsta	Pravilo za razvrščanje znakov	Atributi	Null	Privzeto	Pripombe	Dodatno	Dejanje
1	ID	bigint(20)		Ne	Brez			AUTO_INCREMENT	Spremeni Zavrzi Več
2	vlaga	float		Da	NULL				Spremeni Zavrzi Več
3	temperatura	float		Da	NULL				Spremeni Zavrzi Več
4	vlagazemlje	float		Da	NULL				Spremeni Zavrzi Več
5	voda	float		Da	NULL				Spremeni Zavrzi Več
6	sonce	float		Da	NULL				Spremeni Zavrzi Več
7	cas	timestamp		Ne	CURRENT_TIMESTAMP				Spremeni Zavrzi Več

Slika 2.3: Prikaz tabele in stolpcev pri lokalni podatkovni bazi (phpmyadmin) [Lastni vir]

Največja je tabela, potem so stolpci (ID, vlaga in tako naprej), v stolpcih pa so podatki. [8]

S programskim jezikom SQL lahko v tabeli dodajamo in spreminjamo podatke, spreminjamo tabelo, dodajamo v tabelo in tako naprej.

Stolpce je treba definirati:

- ime,
- vrsta (kaj bomo uporabili številke, čas, besedilo),
- null (za nič),
- prevzeto.

Dodatno (*AUTO_INCREMENT* za avtomatsko povečanje).

2.1.5 JAVASCRIPT

Javascript je poleg HTML-ja in CSS-ja eden izmed pomembnih spletnih programskih jezikov.

Java in Javascript imata sicer podobno ime, vendar sta si zelo različna. Programskega jezika Java sploh nismo uporabili; namenjen je predvsem za povezave. Javascript se uporablja za spletne aplikacije, razvoj, igrice in strežnik.

Kodo lahko vnašamo v datoteko HTML ali PHP z ukazom: `<script></script>`. V *script* po navadi damo funkcijo. Osnovna zgradba izgleda približno tako:

```
<!DOCTYPE html>
<html>
<head>
<script>
function besedilo() {
  document.getElementById("poskus").innerHTML = "2";
```

```
}  
</script>  
</head>  
<body>  
<p id="poskus ">1</p>  
<button type="button" onclick="besedilo()">gumb</button>  
</body>  
</html>
```

S pomočjo gumba lahko zamenjamo številko 1 z 2. Lahko bi spremenili tudi barvo, vrsto pisave in tako naprej. Datoteko Javascript lahko povežemo tudi z datoteko HTML (ali PHP), in sicer z ukazom: `<script src="scripts.js"></script>`.

2.1.6 C++ ALI Arduino IDE

Arduino IDE je programsko okolje, v katerem lahko programiramo z jezikom C++ (slika 2.4).

S programskim jezikom C++ v okolju Arduino IDE lahko programiramo in prikazujem podatke, ki jih dobi mikrokontroler Arduino UNO od senzorjev (ali drugih vezij, naprav). C++ spada med visokonivojske programske jezike in je eden izmed »starejših«, saj se je prvič pojavil leta 1985, tako da je star že 38 let.

Arduino IDE in C++ imata pri kodiranju ima zelo podobno zgradbo, v nekaterih zadevah pa se razlikujeta.

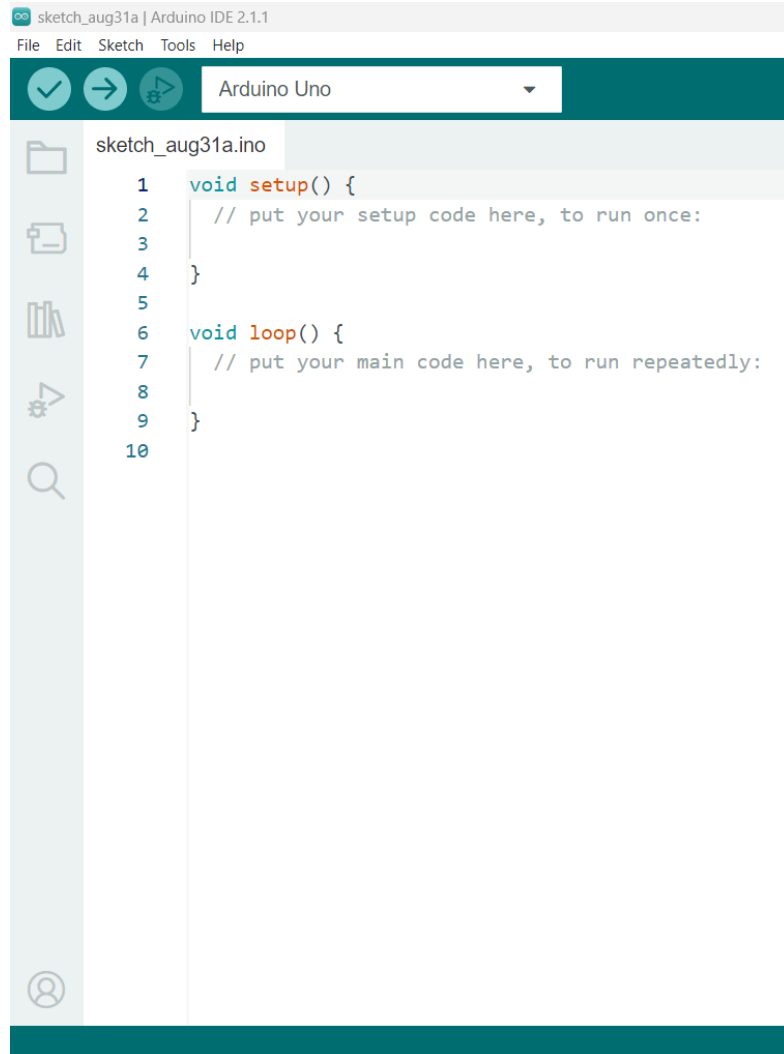
Na primer, pri izpisu ima C++ zgradbo:

```
std :: cout << "dobro";
```

pri Arduinu pa:

```
Serial.write("dobro");
```

Zaradi teh razlik se pri programiranju v programskem okolju Arduino uporablja različica C++.



Slika 2.4: Prikaz orodja Arduino IDE [Lastni vir]

2.2 SENZORJI

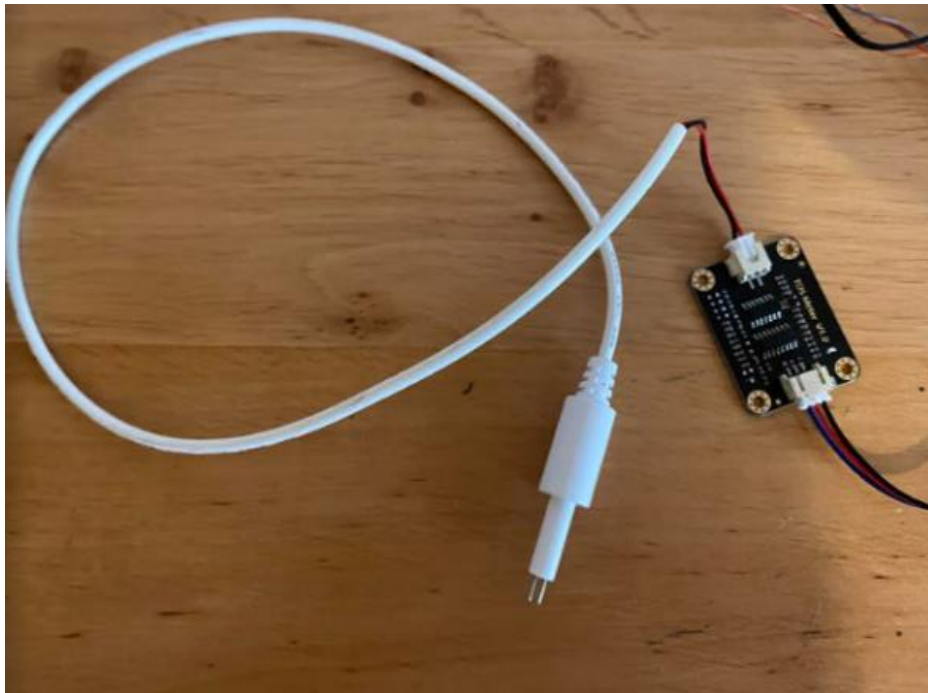
Senzorji so naprave, ki reagirajo v različnih reakcijah in nam podajajo potrebne informacije. Iz fizikalne reakcije spremenijo informacije v električno količino in iz električne količine v razumljivo informacijo, ki jo potrebujemo.

2.2.1 SENZOR TDS

Senzor TDS meri delčke oziroma trdne snovi, ki so raztopljeni v vodi; na primer sol, kovine, minerale in tako naprej. Če se količina trdnih snovi v vodi poveča, se poveča tudi prevodnost. Oznaka, ki jo pri tem uporabljamo, je ppm (parts per milions) ali mg/L. Ni pa to najboljši pokazatelj, ali je voda užitna ali ne. Senzorji TDS se največ uporabljajo v čistilcih vode in bazenih. Spodaj je prikaz senzorja Senzor TDS (slika 2.5).

Senzor ima naslednje tri povezave:

- 5V (napetost),
- GND (ozemljitev) in
- A (analogni signal).



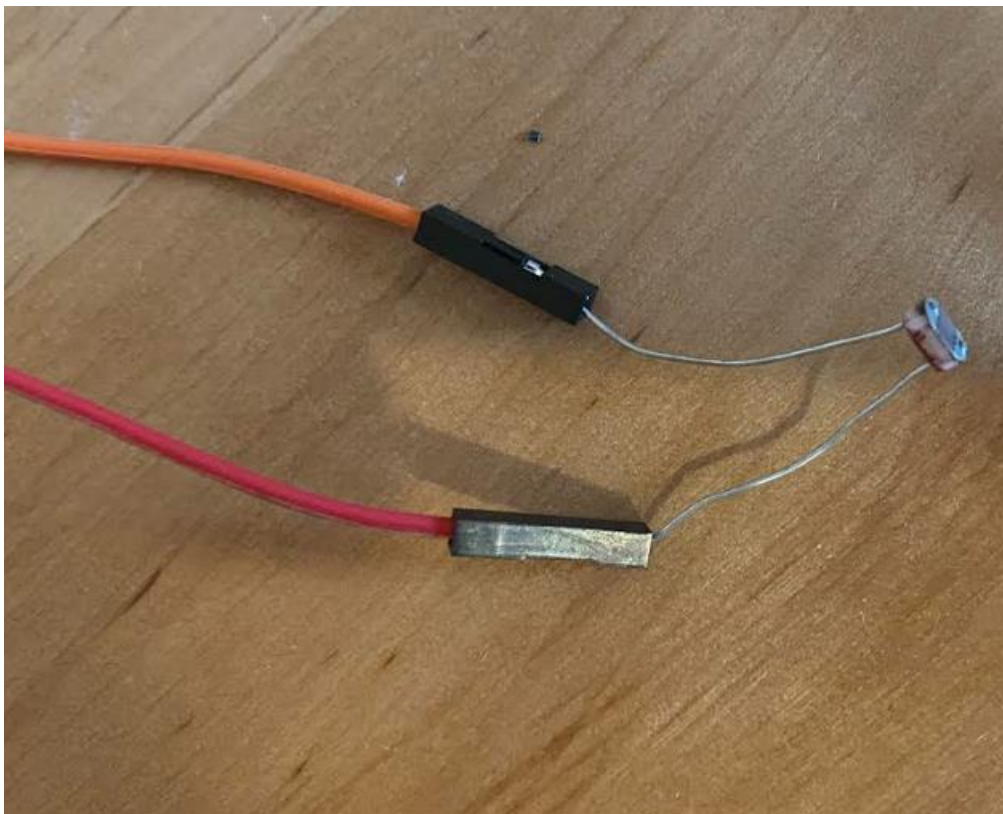
Slika 2.5: Senzor TDS [Lastni vir]

2.2.2 FOTUOPOR LDR

Fotoupor LDR spreminja upor glede na vpad svetlobe (slika 2.6). Če se upornost povečuje ali zmanjšuje, se spreminja moč svetlobe. Pri fotouporu je treba dodati še dodatni upor (10 K Ω).

Fotoupor LDR ima naslednje tri povezave:

- 5V (napetost),
- GND (ozemljitev) in
- A (analogni signal).



Slika 2.6: Fotoupor LDR [Lastni vir]

2.2.3 SENZOR VLAGE ZEMLJE

Ta senzor deluje glede na vlago, prevodnost in upornost zemlje. Kadar se upornost poveča, pomeni, da se zemlja suši; prevodnost pa je zaradi tega manjša. Pri zmanjšanju upornosti se zemlja vlaži in prevodnost poveča.

Senzor je sestavljen iz dveh delov:

- sonde in
- modula.

Modul je del med sondo in programskim okoljem Arduino. Iz sonde imamo povezave za ozemljitev in napetost. Arduino dobiva iz modula analogne podatke. Spodaj je prikaz senzor vlage zemlje (slika 2.7).

Senzor vlage zemlje ima naslednje tri povezave:

- 5V (napetost),
- GND (ozemljitev) in
- A (analogni signal).

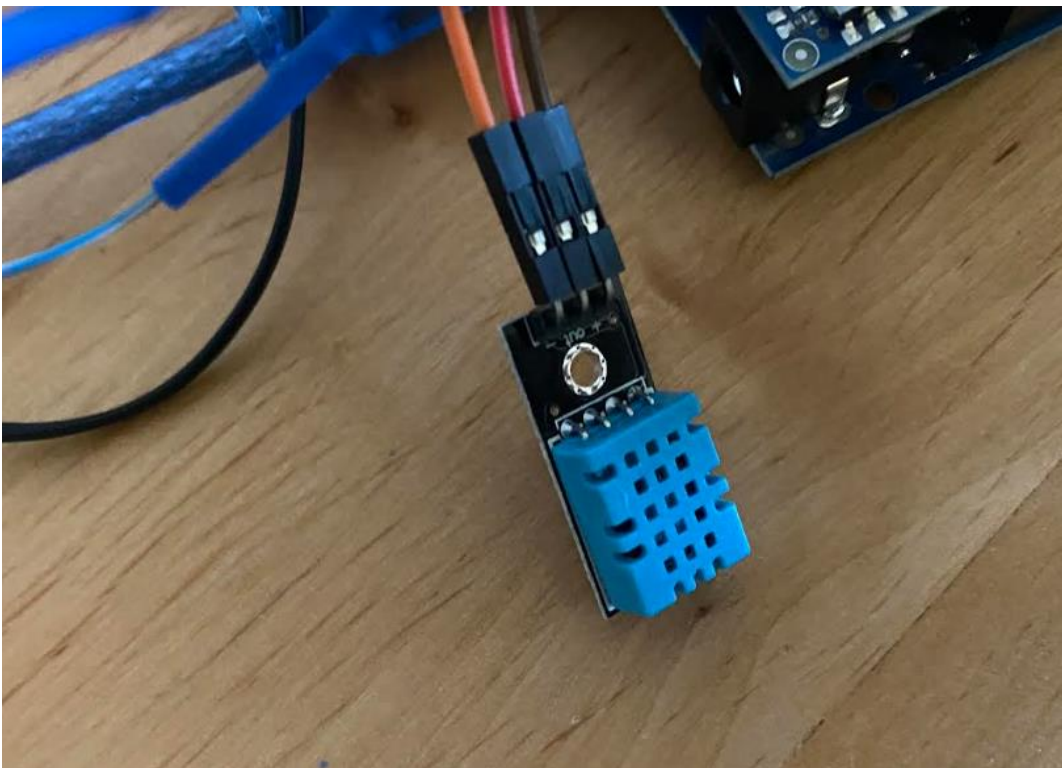


Slika 2.7: Senzor vlage zemlje [Lastni vir]

2.2.4 SENZOR DHT 11

Senzor DHT 11 je zelo poceni senzor, namenjen za merjenje vlage in temperature (slika 2.8). Vlaga je količina vode v zraku. Ta zelo vpliva na različne procese (npr. razgradnjo bioloških snovi). Za merjenje se uporablja kapacitivni senzor vlage, za temperaturo pa termistor.

S tem dobimo informacijo, ki komunicira z Arduinom.



Slika 2.8: Senzor DHT 11 [Lastni vir]

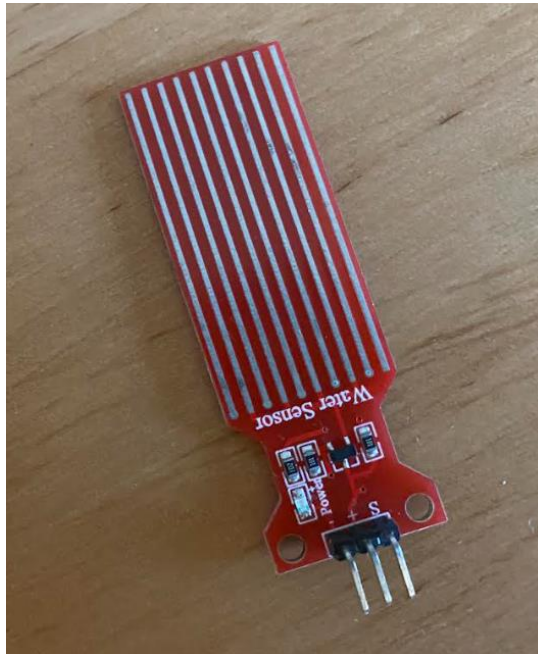
2.2.5 SENZOR VODE

Senzor vode (slika 2.9) prikazuje nivo vode in se uporablja za merjenje padavin, preliivanja. Kadar senzor zazna spremembo nivoja vode, se, glede na upor, spremeni prevodnost.

Če se poveča upornosti, pomeni, da je vode manj in s tem je tudi prevodnost manjša. Pri zmanjšanju upornosti se povečata tako nivo vode kot prevodnost.

Senzor ima naslednje tri povezave:

- P (digitalni signal),
- GND (ozemljitev) in
- A (analogni signal).



Slika 2.9: Senzor vode [Lastni vir]

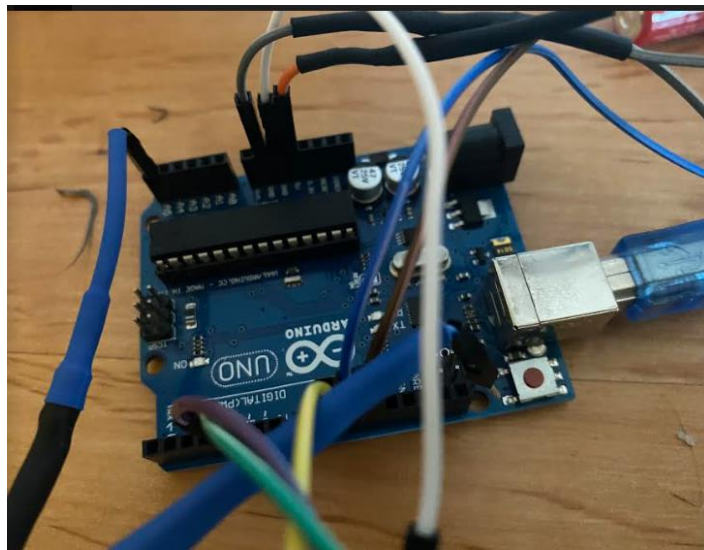
2.3 MIKROKRMILNIK

Mikrokrmilnik je majhen računalnik, ki ima vhodne in izhodne povezave. Izhodne povezave oddajajo informacije ali storitve, preko vhodnih pa informacije ali storitve dobimo.

V diplomski nalogi smo uporabili mikrokrmilnik Arduino UNO. Ta ima pet pomembnih vhodnih in izhodnih povezav. Te so:

- USB port,
- digitalni vhodi/ izhodi (P0-13),
- analogni vhodi/ izhodi(A0-5),
- ozemljitev (GND),
- napetost (5, 3.3, 9 V).

Mikrokrmilnik se programira v programskem okolju Arduino IDE, pri katerem programiramo različico C++. Slika 2.10 prikazuje mikrokrmilnik Arduino UNO.



Slika 2.10: Mikrokrmilnik Arduino UNO [Lastni vir]

2.4 MOTOR

Motor je naprava, ki s pomočjo magnetnega polja pretvarja električno energijo v mehansko.

V nalogi smo uporabili enosmerni motor DC RS-755. Njegovi sestavni deli so:

- laminirano rotorsko jedro,
- ščetke,
- kolektor,
- izolacijska plast,
- rotorsko navitje,
- vzbujalno navitje in
- laminirano statorsko jedro.

Vsi ti deli pomagajo pri pretvarjanju električne energije v mehansko. Motor, ki smo ga uporabili, ni bil primeren za visoke napetosti, zato smo morali napetost prilagoditi (zmanjšati). Napetost smo zmanjšali tako, da smo povezali baterije med seboj. Obstajajo tudi naprave za zmanjšanje napetosti, vendar so v primerjavi z baterijami precej dražje. Slika 2.11 prikazuje motor DC RS-755.



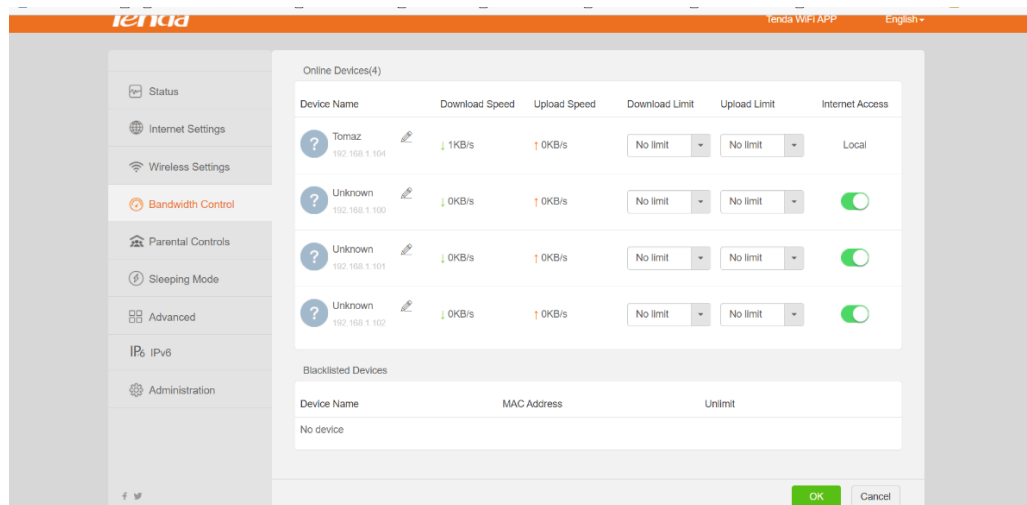
Slika 2.11: Motor DC RS-755 [Lastni vir]

2.5 VEZJE ARDUINO ETHERNET SHIELD

Arduino Ethernet Shield je vezje, ki pomaga s knjižnico prenašati podatke prek interneta. Vezje je povezano s kartico SD, na katero se lahko shranijo podatki. Kartica SD ima tudi svojo knjižnico, ki pomaga pri mikrokrmilniku. Pri tem moremo v Arduino določiti še IP in lokalni IP, ki ga moramo poiskati prek CMD (slika 2.14), rotorja (slika 2.13) ali nastavitve (slika 2.12). Slika 2.15 prikazuje Arduino Ethernet Shield.



Slika 2.12: Prikaz lokalnega IP-ja na nastavitvah [Lastni vir]



Slika 2.13: Prikaz lokalnega IP-ja na usmerjevalniku [Lastni vir]


```
Ukazni poziv
Unknown adapter Povezava z lokalnim omrežjem:
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . . . . . :

Wireless LAN adapter Povezava z lokalnim omrežjem* 1:
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . . . . . :

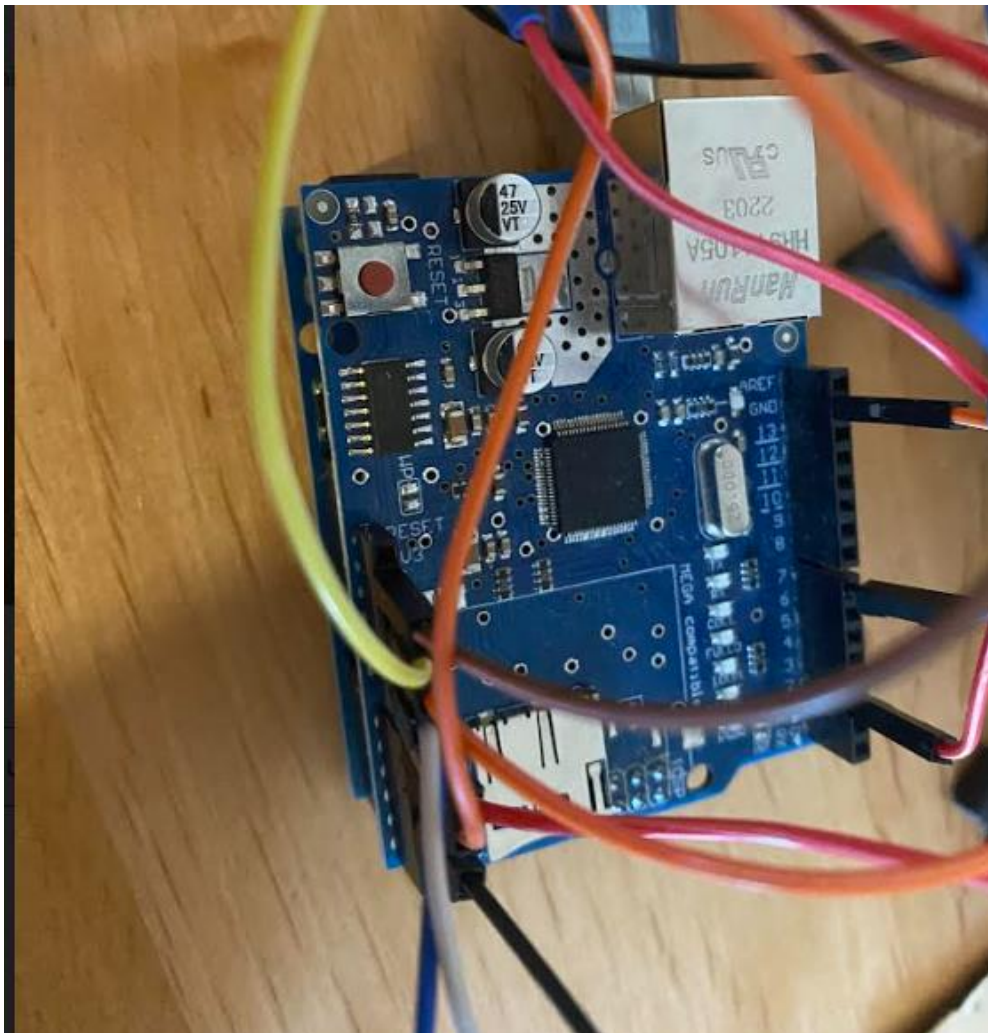
Wireless LAN adapter Povezava z lokalnim omrežjem* 2:
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . . . . . :

Wireless LAN adapter Wi-Fi:
Connection-specific DNS Suffix . . . . . : www.tendawifi.com
Link-local IPv6 Address . . . . . : fe80::6633:3f8:c720:1eb6%10
IPv4 Address. . . . . : 192.168.1.104
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.1.1

Ethernet adapter Omrežna povezava Bluetooth:
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . . . . . :

C:\Users\hozja>
```

Slika 2.14: Prikaz lokalnega IP-ja na CMD [Lastni vir]



Slika 2.15: Vezje Arduino Ethernet Shield [Lastni vir]

3 VEZJE

Vežje zagotavlja povezavo med senzorjem, motorjem in drugimi priključki, ki so v našem primeru povezani v mikrokrmilniku. V nalogi smo za povezavo uporabili kable DIY KIT in bakrene kable iz internetnega kabla. Slika 3.1 prikazuje vodni senzor, baterijo in motor.



Slika 3.1: Vodni senzor, baterija in motor [Lastni vir]

3.1 VEZJE ZA SENZORJE OZIROMA VEZJE ZA PRENAŠANJE PODATKOV

Na začetku smo povezali vse vhode in izhode iz mikrokrmilnika Arduino UNO v Arduino Ethernet Shield (namenjen za povezavo prek interneta). Po priklopu na Arduino UNO in Arduino Ethernet Shield smo dodali senzor DHT 11. Senzor DHT 11 smo povezali v napajanje 3,3 V, na ozemljitev GND in na digitalni signal. Pri povezavi smo uporabili kable iz DIY KIT-a. Po dodanem temperaturnem senzorju smo zvezali še senzor TDS.

Ta senzor smo povezali z ozemljitvijo, ki je enaka z ozemljitvijo senzorja DHT 11. Povezana napetost je bila 9 V, signal pa analogni. Za ta senzor smo uporabili bakrene

3.2 VEZJE ZA MOTOR

Motor je povezan s senzorjem vode za primer, če voda preseže določeno mejo. S tem preprečimo uničenje samega senzorja TDS. Sestava tega vezja je zelo enostavna. Zanj potrebujemo:

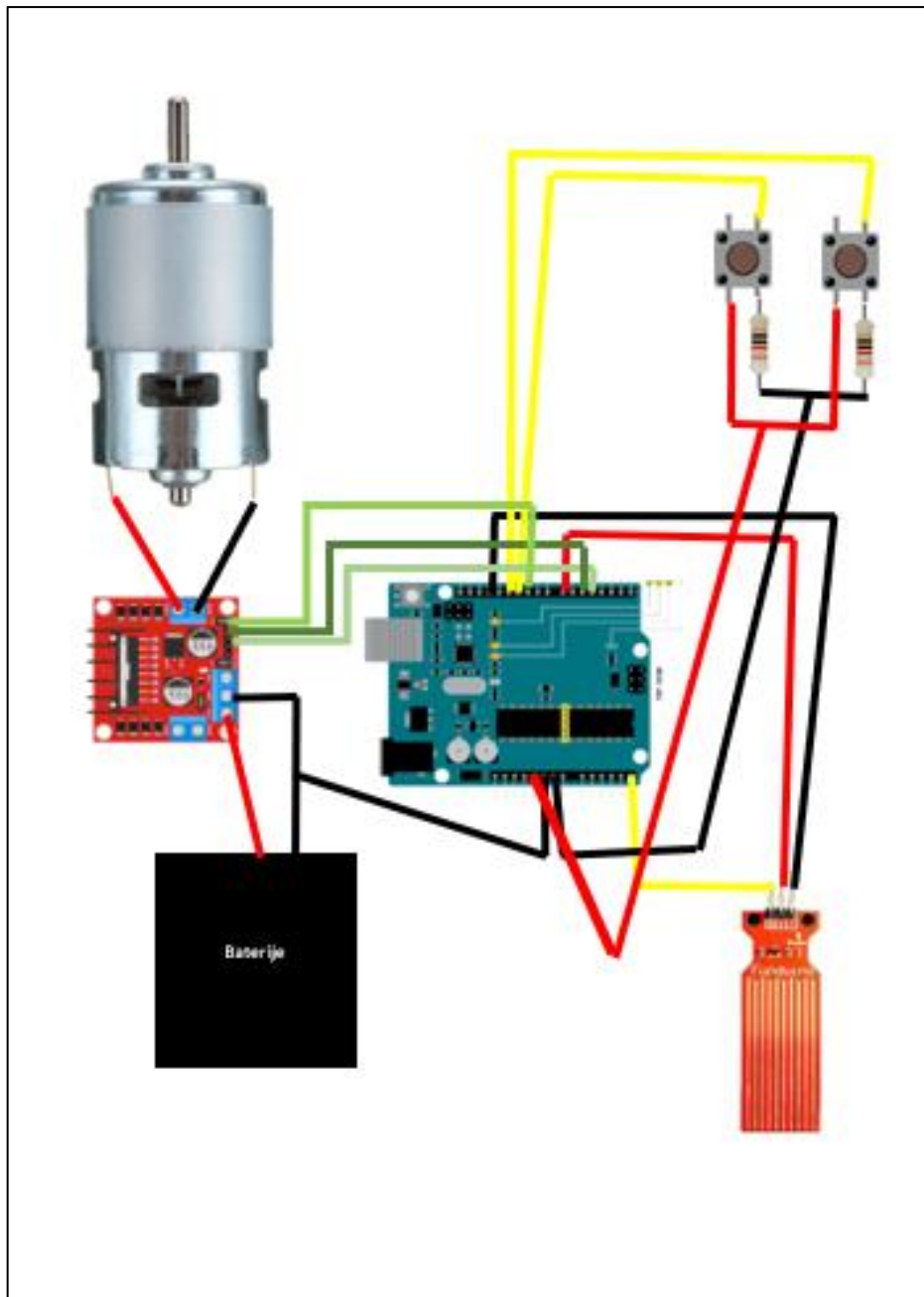
- dva gumba,
- dva upora ($10\text{ K}\Omega$),
- motor,
- L298N in
- vodni senzor.

Motor smo povezali na napetost L298N in na ozemljitev. Potem smo od vezja L298N povezali baterije na napetost in ozemljitev. Tudi mikrokrmilnik smo povezali na ozemljitev, iz L298N pa smo povezali tri signalne žice na mikrokrmilnik.

Dva gumba smo povezali z uporom ($10\text{ K}\Omega$), upor pa z ozemljitvijo. Gumba smo povezali še z napetostjo in ozemljitvijo. Na koncu smo povezali še vodni senzor na mikrokrmilnik, napetost in ozemljitev.

L298N je vezje, ki je že kreirano in ima enako funkcijo kot gumbi. To vezje je namenjeno za obračanje (na levo ali desno stran) motorja. Nanj smo priklopili dodatni dovod napetosti. Gumbi pa so namenjeni za vklop in izklop motorja, če voda presega določeno mejo.

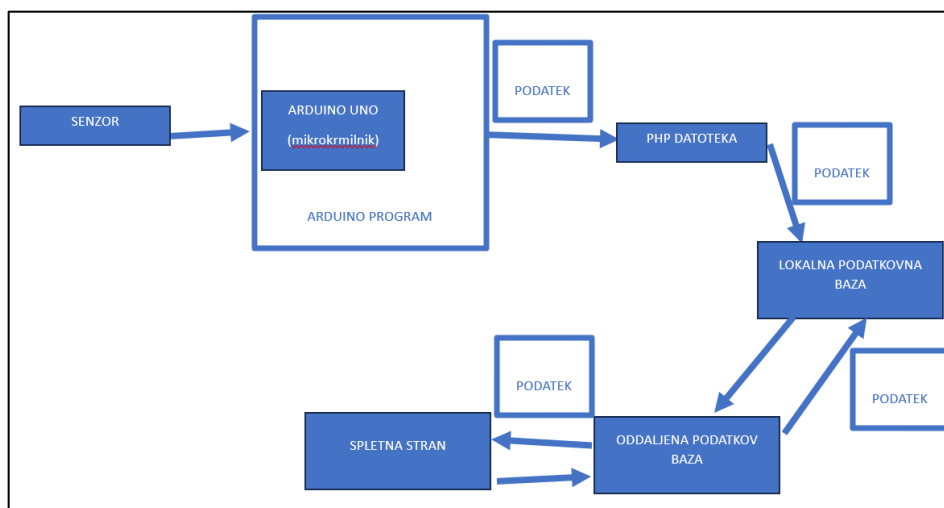
Slika 3.3 prikazuje vezje motorja.



Slika 3.3: Skica vezja za motor [Lastni vir]

4 OSNOVNO DELOVANJE

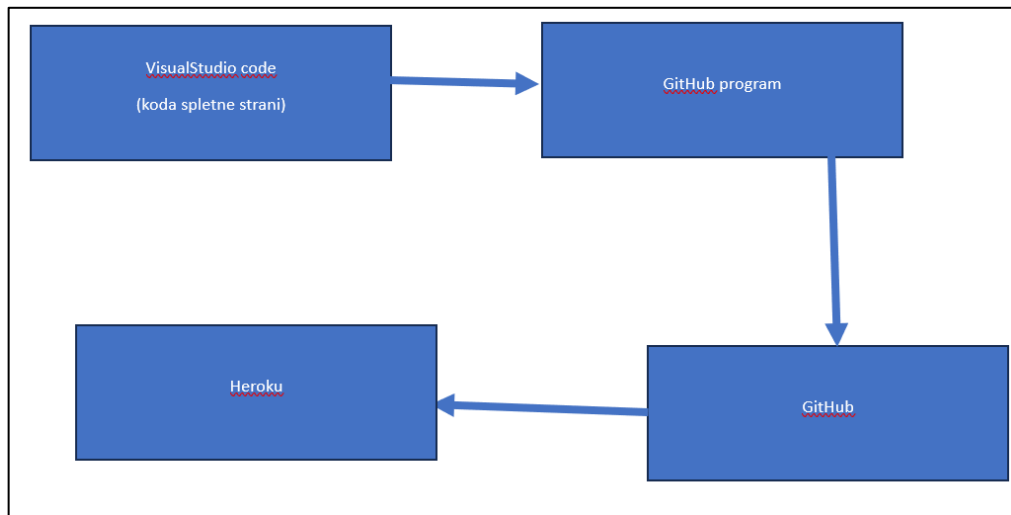
Za bolj razločno predstavitev osnovnega delovanja je opisano delovanje samo z enim senzorjem, in to na primeru DHT 11. Na začetku smo naredili vezje med senzorjem in mikrokrmilnikom. V mikrokrmilnik je vgrajen program Arduino, ki iz sensorja dobiva informacije. PHP zaprosi informacije oziroma podatke od programa Arduino in jih pošlje v lokalno podatkovno bazo. Ta je povezana z oddaljeno podatkovno bazo. Iz te baze vnovič zaprosimo za podatke iz spletnega programa. Če bi bila povezava med lokalno in oddaljeno podatkovno bazo prekinjena, bi se podatki vseeno shranili v lokalni podatkovni bazi. Oddaljena podatkovna baza bi v tem primeru kazala samo stare podatke, ob vnovični povezavi pa bi pokazala tudi nove. Slika 4.1 prikazuje prenos podatkov.



Slika 4.1: Skica prenosa podatkov od senzorja do spletne strani [Lastni vir]

Ko smo razložili povezave med podatki, se je postavilo vprašanje glede povezave s spletno stranjo. Spletna stran oziroma koda je na začetku napisana v programskem jeziku *Visual Studio code*.

Če smo program spremenili, je bilo to vidno na programu GitHub, ki je bil naložen na računalniku. Na GitHubu se je program shranil. Potem je Heroku (server, na katerem je spletna stran) zaprosil program iz GitHuba in zamenjal staro spletno stran z novo. Slika 4.2 prikazuje prenos spletne strani.



Slika 4.2: Skica prenosa spletne strani od Visual Studio code do serverja Heroku [Lastni vir]

5 NATANČNO DELOVANJE

5.1 PODATKI

Zaradi jasnosti smo opisali delovanje samo enega senzorja, in sicer senzorja DHT 11.

Senzorji imajo enak postopek, razlikujejo pa se glede vezja in programa za pridobivanje podatkov. Na začetku smo definirali različne vhodne in izhodne podatke in dodali določeno knjižnico senzorjev. Pri definiranju smo dodali vse kazalce za pridobivanje informacij (int, float, const int, ...). S funkcijo `void setup()` smo definirali vhode in izhode, na koncu pa smo podatke izpisali. V našem primeru smo dodali v novi `void` za povezavo z datoteko `php`. Pri tem se podatki senzorjev razlikujejo. Sledi razlaga samo enega senzorja.

Senzor smo povezali s mikrokrmilnikom, mikrokrmilnik pa z računalnikom in internetom. Pri tem smo zagnali program Arduino. Nato smo dodali knjižnico (za senzor in za internetno povezavo) ter dva `float-a` (za vlago in temperaturo). Potem smo dodali lokalni IP in novi IP (novi IP je namenjen za prenašanje informacij). V `void setup()` smo definirali vhode. Če povezava z internetom ni bila vzpostavljena, se nam je izpisalo, da se ni povežalo. Pri `void loop()` smo definirali, da so vhodni podatki enaki `float` podatkom. Potem smo v `void loop()` zagnali novi `void`. Ta je bil namenjen za vzpostavljanje povezave z datoteko `php` in z lokalnim IP-jem. Če povezava ni bila vzpostavljena, se nam je izpisalo, da ni povežano.

Slika 5.1 prikazuje program za mikrokrmilnik Arduino.

```

v5e_110.ino
117 // Serial.println( "vseperilnica" );
118 client.print(temperatureData);
119 Serial.println(temperatureData);
120 client.print("vlagaZeniJe");
121 Serial.println("vlagaZeniJe");
122 client.print(vlagaZeniJe);
123 Serial.println(vlagaZeniJe);
124 client.print("&voda=");
125 Serial.println("&voda=");
126 client.print(vodaValue);
127 Serial.println(vodaValue);
128 client.print("&sonce=");
129 Serial.println("&sonce=");
130 client.print(idrValue);
131 Serial.println(idrValue);
132 client.print(" ");
133 client.print("HTTP/1.1");
134 client.println();
135 client.print("Host: 192.168.1.104");
136 client.print("Connection: close");
137 client.println();
138 } else {
139
140 Serial.println("Ni povezano");
141 }

```

Slika 5.1: Prikaz programa za mikrokrmilnik Arduino [Lastni vir]

Podatki so se prenašali iz mikrokrmilnika na datoteko php (slika 5.2). Ta datoteka je shranjena na serverju XAMPP. Datoteka php najprej zaprosi za podatke, nato pa se poveže z lokalno podatkovno bazo in vanjo pošlje podatke.

```

v5e.php
1 class Temperatura_senzor{
2     public $link="";
3     function __construct($temperatura, $vlaga, $vlagaZeniJe, $voda, $sonce){
4         $this->connect();
5         $this->storeInDB($temperatura, $vlaga, $vlagaZeniJe, $voda, $sonce);
6     }
7
8     function connect(){
9         $this->link = mysqli_connect("ou.cdbw-west-03.cleardb.net","bf303ea28424","e978bb12") or die("Nemore povezati PB");
10        mysqli_select_db($this->link, "heroku_e17778c5d8ff122") or die("Nemore izbrati PB");
11    }
12
13    function storeInDB($temperatura, $vlaga, $vlagaZeniJe, $voda, $sonce){
14        $query = "insert into temperatura_senzor set vlaga='".$vlaga."', temperatura='".$temperatura."', vlagaZeniJe='".$vlagaZeniJe."', voda='".$voda."', sonce='".$sonce.'";
15        $result = mysqli_query($this->link,$query) or die("Napaka: ".$query);
16    }
17
18    }
19
20    if($_GET['temperatura'] != "" and $_GET['vlaga'] != "" and $_GET['vlagaZeniJe'] != "" and $_GET['voda'] != "" and $_GET['sonce'] != ""){
21        $temperatura_senzor=new Temperatura_senzor($_GET['temperatura'],$_GET['vlaga'],$_GET['vlagaZeniJe'],$_GET['voda'],$_GET['sonce']);
22    }
23
24

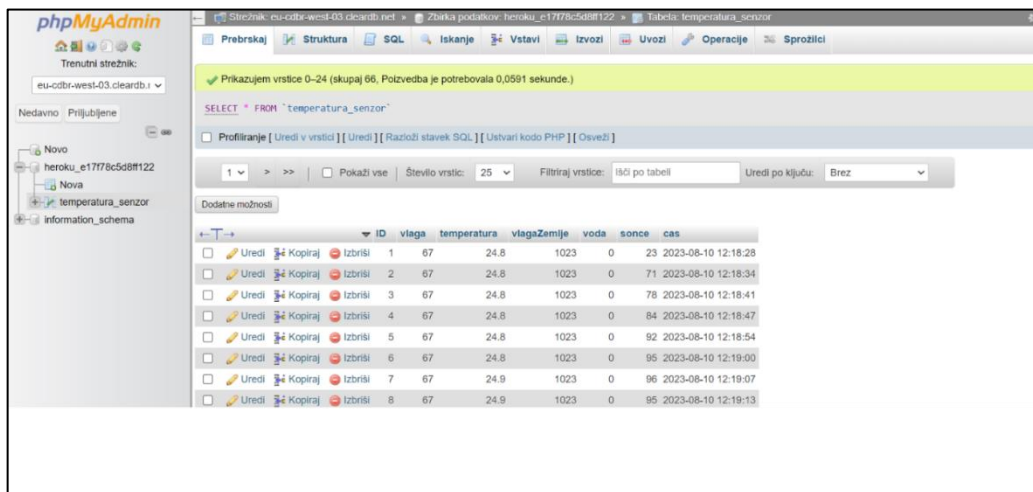
```

Slika 5.2: Prikaz programa med Arduino in lokalno podatkovno bazo [Lastni vir]

Ko so bili podatki v lokalni bazi, smo morali vzpostaviti povezavo še z oddaljeno podatkovno bazo. To podatkovno bazo smo morali dodati v server, in sicer tako, da smo v datoteko config.inc dodali (podatke smo dobili s serverja Heroku):

```
$i++;
$config["Servers"][$i]["host"] = "eu-cdbw-west-03.cleardb.net";
$config["Servers"][$i]["user"] = "bcf303ea2042d4";
$config["Servers"][$i]["password"] = "e979bb12";
$config["Servers"][$i]["auth_type"] = "config";
```

Ko so bile povezave narejene, smo lahko v lokalni in oddaljeni podatkovni bazi odprli in pogledali tabelo, v katero smo pošiljali podatke. Slika 5.3 prikazuje tabelo s podatki.



ID	vlaga	temperatura	vlagaZamije	voda	sonce	cas
1	67	24.8	1023	0	23	2023-08-10 12:18:28
2	67	24.8	1023	0	71	2023-08-10 12:18:34
3	67	24.8	1023	0	78	2023-08-10 12:18:41
4	67	24.8	1023	0	84	2023-08-10 12:18:47
5	67	24.8	1023	0	92	2023-08-10 12:18:54
6	67	24.8	1023	0	95	2023-08-10 12:19:00
7	67	24.9	1023	0	96	2023-08-10 12:19:07
8	67	24.9	1023	0	95	2023-08-10 12:19:13

Slika 5.3: Prikaz tabele s podatki [Lastni vir]

5.2 SPLETNA STRAN

Program za spletno stran smo napisali v programskem jeziku *Visual Studio code*. Potem smo s pomočjo aplikacije GitHub določili lokacijo, na kateri smo spletno stran shranili. Ob spremembi programa na Visual Studio code je bila ta sprememba vidna v aplikaciji GitHub; to spremembo je treba vedno znova poslati na glavni GitHub. Potem smo se prek serverja Heroku prijavi na GitHub in ju povezali. Ob spremembi programa smo

morali vnovič posodobiti in zamenjati staro spletno stran z novo. Ker uporabljamo Visual Studio code, smo lahko uporabili tudi terminal in povezali Heroku z Visual Studio code.

V terminal smo se vpisali in prijavi:

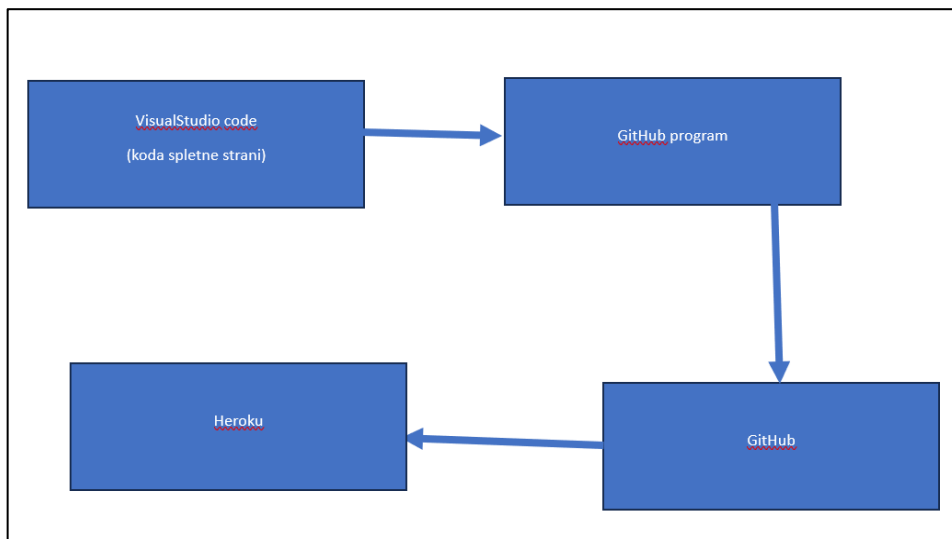
```
$ heroku login
```

```
$ git add
```

```
$ git commit -am "make it better"
```

```
$ git push heroku master
```

Slika 5.4 prikazuje prenos spletne strani.



Slika 5.4: Skica prenosa spletne strani od Visual Studio code do serverja Heroku [Lastni vir]

6 PROGRAMI ALI ORODJA

Pri izvedbi diplomske naloge smo uporabili različne strani, programe in serverje.

Na začetku izdelave diplomskega dela smo že nekaj vedeli o programskih orodjih, kot so Arduino IDE, Atom, Visual Studio code, SOLIDWORKS in Creality Slicer, nismo pa poznali programskih orodij GitHub, Heroku in XAMPP; te smo spoznali med pisanjem diplomskega dela.

6.1 ARDUINO

Arduino IDE je programsko orodje, v katero se lahko vpišemo in z njim manipuliramo mikrokontroler s senzorji in drugimi dodatki. Naložitev je enostavna, saj lahko gremo na njihovo spletno stran in ga brezplačno naložimo. Slika 6.1 prikazuje ikono programskega orodja Arduinoja.



Slika 6.1: Ikona programskega orodja Arduino [9]

6.2 ATOM

Atom je programsko orodje za oblikovanje datotek. Gre za boljšo verzijo, kot je navadna beležnica. V Atomu lahko tudi programiramo spletne strani, vendar za nas to ni bilo najbolj primerno. Naložitev programskega orodja je brezplačna. Slika 6.2 prikazuje ikono programskega orodja Atoma.



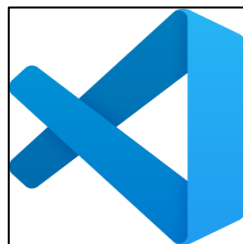
Slika 6.2: Ikona programskega orodja Atom [10]

6.3 VISUAL STUDIO CODE

Visual Studio code je programsko orodje za programiranje spletnih strani. Je boljše kot Atom, saj ima nekatere funkcije, ki jih Atom nima, na primer: prikaz problemov, output, debug console in terminal.

Čeprav se to programsko orodje uporablja v glavnem za programiranje spletnih strani, lahko z njim programiramo tudi aplikacije. Naložitev programskega orodja je brezplačna.

Slika 6.3 prikazuje ikono programskega orodja Visual Studio code.



Slika 6.3: Ikona programskega orodja Visual Studio code [11]

6.4 SOLIDWORKS

SOLIDWORKS je programsko orodje za oblikovanje raznih delov in risanje tehničnih risb. Sicer ima tudi druge funkcije, a mi smo ga uporabili le za oblikovanje delov za 3d tiskanje. Slika 6.4 prikazuje ikono programskega orodja SOLIDWORKS.



Slika 6.4: Ikona programskega orodja SOLIDWORKS [12]

6.5 CREALITY SLICER

Programsko orodje Creality Slicer je namenjeno za pripravo 3d tiskanja. Z njim oblikovani del pripravimo za 3d tiskanje. To je zadnje orodje preden se del natisne. V tem programu smo določili tiskalnik in kvaliteto tiskanega dela. Naložitev programskega orodja je brezplačna.

Slika 6.5 prikazuje ikono programskega orodja CREALITY SLICER.

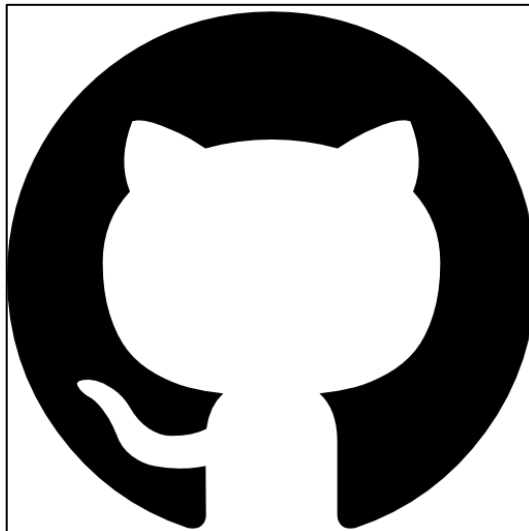


Slika 6.5: Ikona programskega orodja CREALITY SLICER [13]

6.6 GITHUB DESKTOP

GitHub Desktop je programsko orodje, namenjeno za spremljanje programov, saj nam točno prikaže, kaj smo spremenili. Je zelo uporabno, saj nam prikaže vrstico, v kateri smo spremenili staro in novo verzijo. Pri tem smo se morali povezati z datoteko, v kateri je bil program shranjen. Pri zagonu se program pošlje na glavni GitHub. Na GitHub smo se morali prijaviti. Naložitev programskega orodja je brezplačna.

Slika 6.6 prikazuje ikono programskega orodja GitHub.



Slika 6.6: Ikona programskega orodja GitHub [14]

6.7 HEROKU

Heroku je strežnik, na katerem smo objavili svojo spletno stran. Na začetku smo se prijavili in povezali terminal ali GitHub s strežnikom Heroku. V našem primeru smo uporabili dodatno funkcijo, in sicer oddaljeno podatkovno bazo, ki je plačljiva. Cena je odvisna od tega, kako dolgo je spletna stran objavljena.

Slika 6.7 prikazuje ikono strežnika Heroku.



Slika 6.7: Ikona strežnika Heroku [15]

6.8 XAMPP

XAMPP je strežnik, na katerem smo shranjevali podatke. Na njem je lokalna podatkovna baza in tudi tabela. Tabela ima vrstice, ki jih je treba dodati. Strežnik XAMPP je brezplačen in ga lahko naložimo s spletne strani.

Slika 6.8 prikazuje ikono strežnika XAMPP.



Slika 6.8: Ikona strežnika XAMPP [16]

7 3D OBLIKOVANJE IN TISKANJE

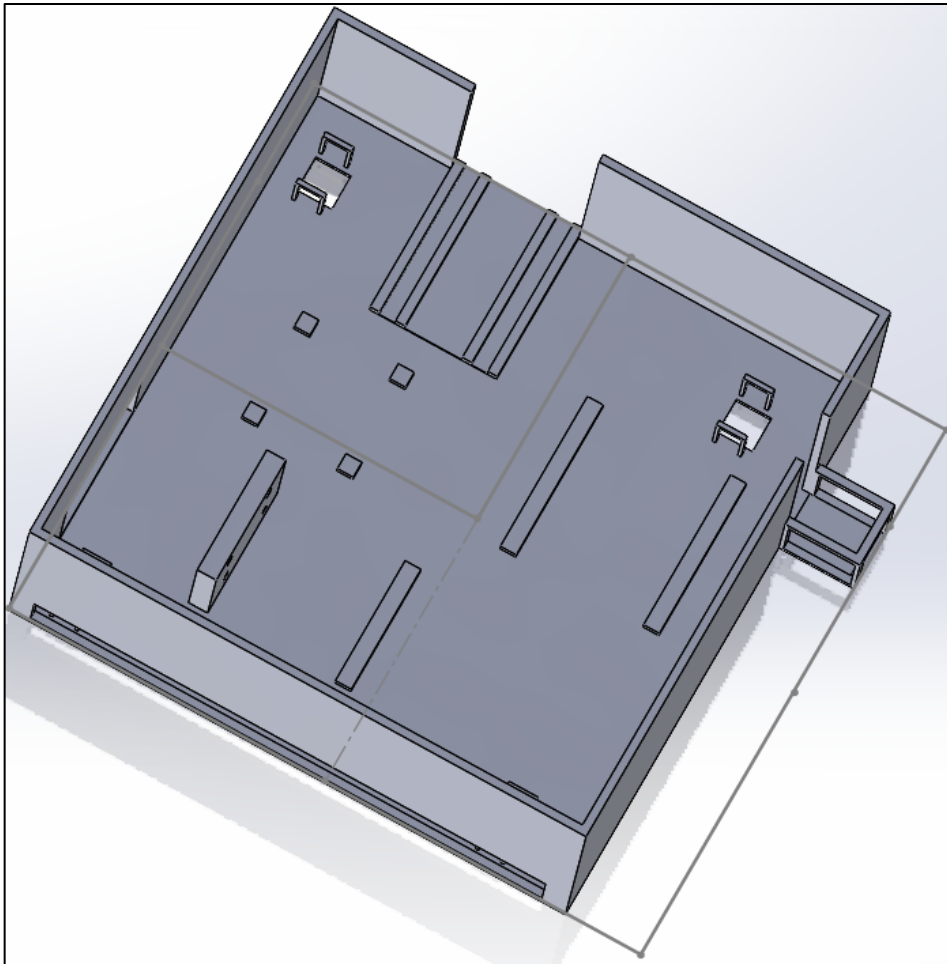
Vsi deli, ki smo jih naredili, so bili prototipi, kar pomeni, da jih je še vedno možno nadgraditi. Na začetku smo morali narediti ohišje, v katerem bo vezje. Posebno oblikovanje za povezavo vezja z internetom ni bilo težko, bolj zapleteno pa je bilo oblikovanje povezave vezja z motorjem. Za 3d oblikovanje smo uporabili programsko orodje SOLIDWORKS, za pripravo 3d tiskanja pa CREALITY SLICER. Programirane dele smo natisnili s tiskalnikom Ender 3 in Creality CR-6 SE (slika 7.1), in sicer s filamentom PLA.



Slika 7.1: Tiskalnika Ender 3 in Creality CR-6 SE [Lastni vir]

7.1 OSNOVNO OHIŠJE

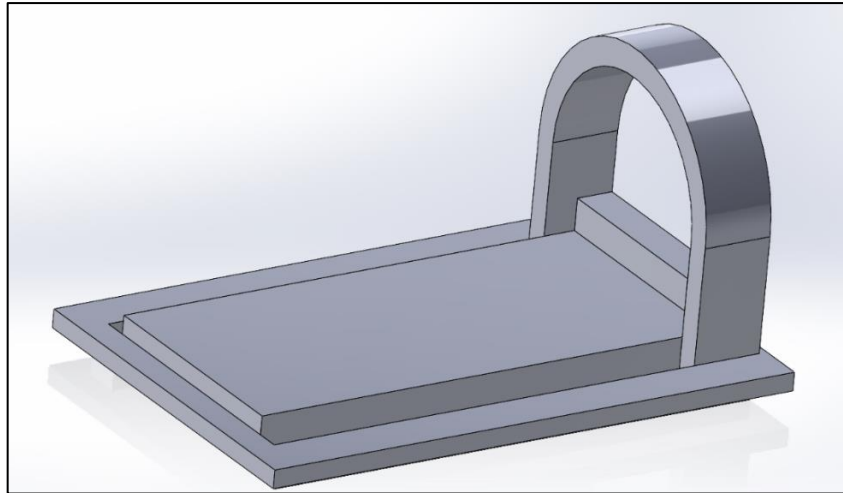
Za osnovni prikaz smo naredili osnovno ohišje (slika 7.2), v katerem so vezja, ter podnožje, ki ga sestavljajo štiri noge.



Slika 7.2: Prikaz osnovnega ohišja [Lastni vir]

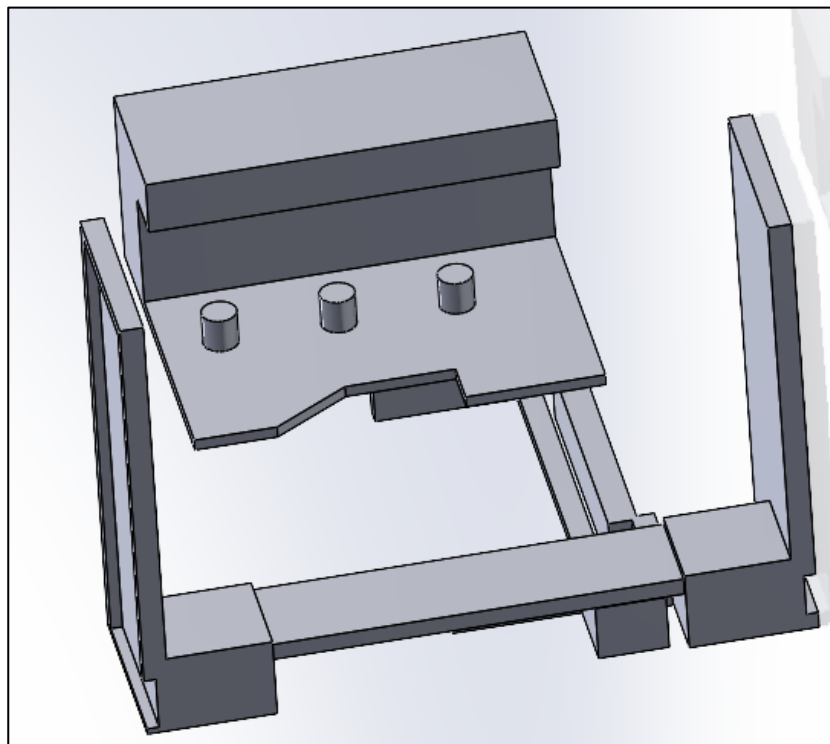
Leva zgornja stran je namenjena za motor, pod njim pa je L298N in Arduino Uno. Desna stran je namenjena za Arduino Uno in fotoupor, levo od Arduino Uno pa je prostor za baterijo. Izrezani del na spodnji strani je namenjen za kable. V vsakem kotu je utor, namenjen za nogo.

Spodnji del je namenjen za vezje za motor (slika 7.3), pri čemer je sprednji vrtljivi del na desni strani.



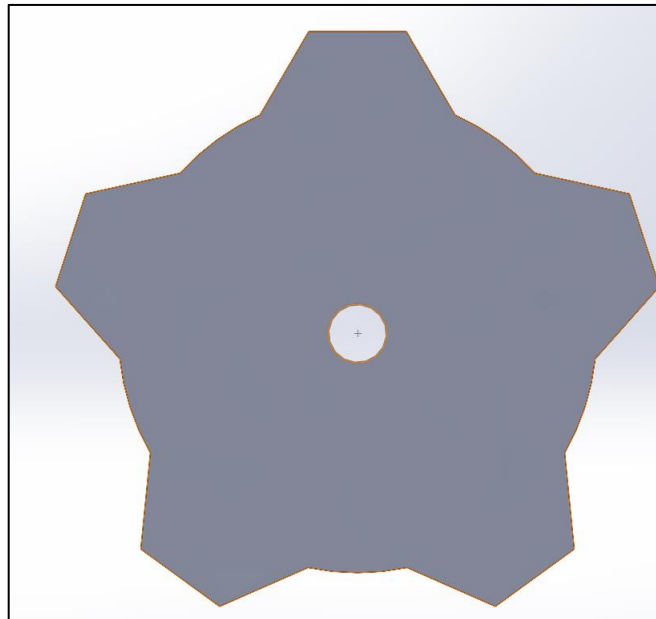
Slika 7.3: Prikaz osnovnega ohišja motorja [Lastni vir]

Slika 7.4 prikazuje ohišje, ki je nadgradnja zgornjega dela, na katerega se priključi. Namenjen je za premikajoči se del. Leva stran je namenjena za gumba, desna pa za premikanje.



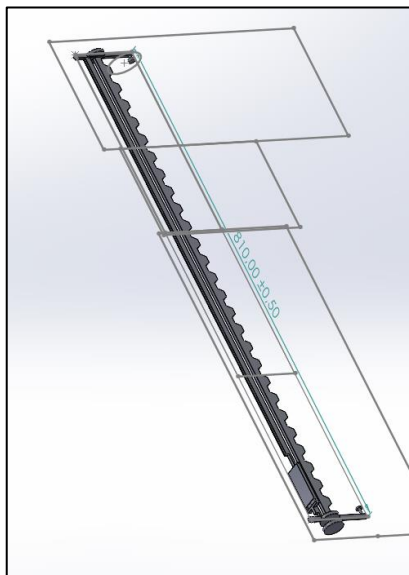
Slika 7.4: Prikaz osnovnega ohišja motorja [Lastni vir]

Spodnja slika 7.5 prikazuje del za sprednji del motorja, ki je namenjen za obračanje.



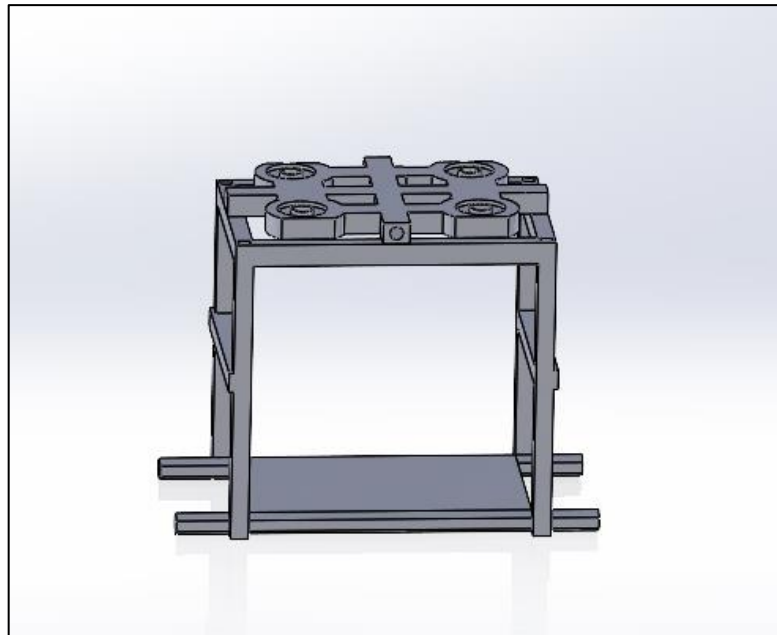
Slika 7.5: Prikaz osnovnega premikajočega se dela motorja [Lastni vir]

Slika 7.6 prikazuje del, namenjen za obračanje in je dodatek zgornjega dela.



Slika 7.6: Prikaz osnovnega premikajočega se dela motorja [Lastni vir]

Na koncu smo izdelali še ohišje baterije. Slika 7.7 prikazuje ohišje baterije.



Slika 7.7: Ohišja baterije [Lastni vir]

Noge so narejene iz lesa. Na spodnjem delu noge smo morali narediti zaščito (slika 7.8 in slika 7.10).



Slika 7.8: Prikaz zaščite za noge [Lastni vir]

Nato smo izdelali podnožje oziroma noge (slika 7.9). To so edini elementi, ki niso narejeni iz plastike, ampak iz lesa. Posamezna noga je sestavljena iz dveh delov, in sicer zato, da se lahko nastavi višina. Pri vsaki nogi smo zgoraj zvrtili luknjo za spoj z zgornjim delom ohišja. Sredinske luknje so namenjene za nastavitev višine, luknja na spodnjem delu noge pa je namenjena za pritrditev zaščite.



Slika 7.9: Prikaz noge [Lastni vir]

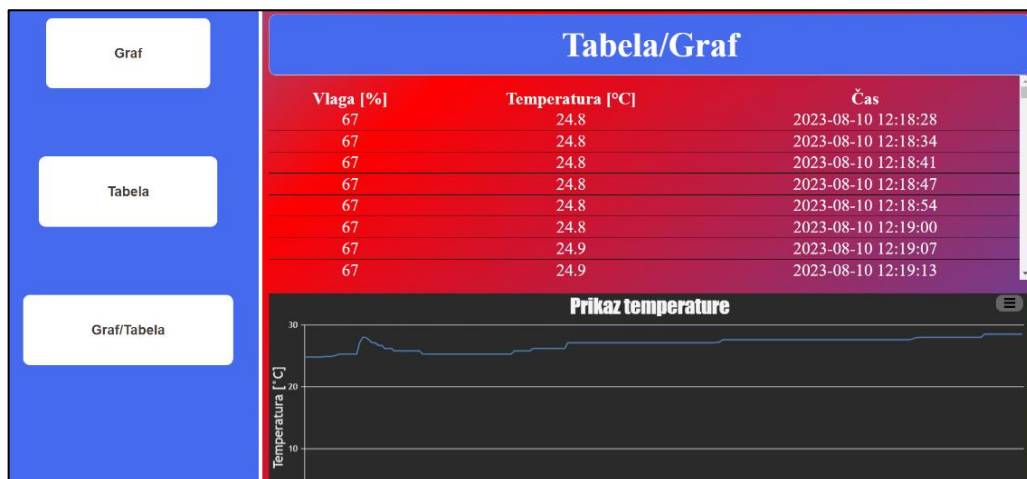


Slika 7.10: Prikaz noge in njene zaščite [Lastni vir]

8 PRIKAZ PODATKOV NA SPLETNI STRANI

Za prikaz podatkov smo na spletno stran dali tabelo in graf (slika 8.1). Tabela je na levi strani prikazovala izmerjeno številko, na desni pa čas meritve. Spodaj je bil na podlagi teh podatkov izrisan graf, ki je na levi strani prikazoval skalo s številkami.

Za izbiro so bili na desni strani zaslona gumbi, s pomočjo katerih smo lahko izbirali, da se izriše samo graf ali samo tabela, zadnji gumb pa je bil namenjen za oboje; tako za izpis tabele kot grafa istočasno.



Slika 8.1: Prikaz izpisa tabele in grafa na spletni strani [Lastni vir]

Na spletni strani smo najprej zaprosili za povezavo do tabele in za podatke za graf (slika 8.2).

```

<?php
$cleardb_url = parse_url(getenv("CLEARDB_DATABASE_URL"));
$cleardb_server = $cleardb_url["host"];
$cleardb_username = $cleardb_url["user"];
$cleardb_password = $cleardb_url["pass"];
$cleardb_db = substr($cleardb_url["path"],1);
$active_group = 'default';
$query_builder = TRUE;

$conn = mysqli_connect($cleardb_server, $cleardb_username, $cleardb_password, $cleardb_db);

mysqli_select_db($conn,$cleardb_db);
$test=array();

$res=mysqli_query($conn,"select temperatura,vlaga from temperatura_senzor");
$count=0;
while($row = mysqli_fetch_array($res))
{
    $test[$count] ["label"]=$count;
    $test[$count] ["y"]=$row["temperatura"];
    $count=$count+1;
}

```

Slika 8.2: Prikaz izpisa tabele in grafa na spletni strani [Lastni vir]

Ko smo definirali podatke, smo s funkcijo uredili prikaz grafa (slika 8.3).

```

<script>
window.onload = function () {
var chart = new CanvasJS.Chart("chartContainer", {
    animationEnabled: true,
    exportEnabled: true,
    theme: "dark1", // "light1", "light2", "dark1", "dark2"
    title: {
        text: "Prikaz temperature"
    },
    axisY: {
        title: "Temperatura [°C]",
        includeZero: true
    },axisX: {
        title: "Število meritve",
        includeZero: true
    },
    data: [{
        type: "line", //change type to bar, line, area, pie, etc
        //indexLabel: "{y}", //Shows y value on all Data Points
        indexLabelFontColor: "#8e5939",
        indexLabelPlacement: "outside",
        dataPoints: <?php echo json_encode($test, JSON_NUMERIC_CHECK); ?>
    ]
});
chart.render();

```

Slika 8.3: Prikaz script kode za graf [Lastni vir]

Nato smo zaprosili za podatke iz povezave; prikazala se je tabela (slika 8.4).

```
PHP
$sql = " SELECT vlaga,temperatura,cas FROM temperatura_senzor";
$result = mysqli_query($conn, $sql);

echo "<br>";
echo "
<style>
tr:hover {background-color: black;color:white}
</style>
";

echo "<table style=' border-collapse:collapse; width: 100%; margin-left: auto;margin-right: auto; '>";
echo "<tr>";
echo "<th style='color:white; font-size: 25px;'>Vlaga [%]</th>";
echo "<th style='color:white; font-size: 25px;'>Temperatura [°C]</th>";
echo "<th style='color:white; font-size: 25px;'>Cas</th>";
echo "</tr>";
while ($row = mysqli_fetch_assoc($result)) {

echo "<tr style='padding: 8px; text-align: left; border-bottom: 1px solid black;text-align: center;font-size: 25px; '>";
foreach ($row as $field => $value) {
echo "<td style='color:white;'>" . $value . "</td>";
}
echo "</tr>";
}
echo "</table>";

?>
```

Slika 8.4: Prikaz kode php za tabelo [Lastni vir]

9 PRIPRAVE

Pred pisanjem diplomske naloge smo imeli veliko priprav. Med izvedbo so se pojavili problemi, ki smo jih sprti reševali. Opisani so v tem poglavju.

Pri pripravi smo se morali prijaviti na programsko orodje GitHub in dodati nov projekt, v katerem se je shranjeval program spletne strani. Prijaviti smo se morali tudi na strežnik Heroku in na njem kreirati novi projekt za prikaz spletne strani. Na Heroku smo dodali tudi oddaljeno podatkovno bazo.

Za naložitev smo potrebovali naslednja programska orodja: GitHub desktop, XAMPP, Visual Studio code, SolidWorks, Arduino IDE in Atom; glede teh nismo potrebovali posebnih priprav. Problemi so se pojavili v zvezi s povezavo med programskim orodjem XAMPP in oddaljeno spletno stranjo, saj je bilo zelo težko pridobiti podatke za povezavo.

Med pomembnejše dele diplomskega dela spada vezje. Kupiti smo morali vse senzorje, motor in baterijo. Kjer kabli niso bili primerni, smo za senzor naredili podaljšek za kabel. Problemi so bili tudi s senzorji, ki so potrebovali podaljšane kable, saj je obstajala verjetnost, da se podatki predvsem zaradi prevelike razdalje ne bodo prikazali.

Sami smo naredili tudi kabel za internet, ki smo ga potrebovali za povezavo s podatkovno bazo.

Za pripravo 3d tiskanja smo potrebovali nekaj kilogramov plastike PLA.

Za zaključek pa smo potrebovali še orodje za obdelovanje nog.

Slika 9.1 prikazuje osnovna orodja za pripravo diplomske naloge.



Slika 9.1: Prikaz osnovnih orodij za pripravo diplomske naloge [Lastni vir]

10 ZAKLJUČEK

Diplomska naloga prikazuje, kako s pomočjo senzorja pridobimo informacije in kako te informacije v živo prikažemo na spletni strani. Prikazuje tudi programsko in ne programsko povezavo. Če pride pri povezavah do kakršne koli napake v katerem koli delu, se lahko vse poruši oziroma ne deluje. Narediti spletno stran in pridobiti informacije preko senzorjev je dokaj preprosto, zaplete pa se pri prenašanju teh informacij v živo na spletno stran. Problemi so se pojavili že na začetku pri povezavi na lokalno podatkovno bazo. Lokalno podatkovno bazo in informacije iz senzorja je bilo treba narediti, pri tem pa se je pojavil IP (številka, ki označuje računalnik). Ker nismo vedeli, da potrebujemo lokalni IP, smo vpisovali naključne IP-je. Ko smo ta problem rešili, so se spet pojavile težave s senzorji. Druge težave glede povezav med lokalno podatkovno bazo in oddaljeno podatkovno bazo ter med njo in spletno stranjo smo reševali sproti. Spletna stran ima svojo povezavo: <https://diplomski-del-php.herokuapp.com/>, preko katere je dostopna. Povezovanje s spletno stranjo ni bilo težko, saj je šlo za majhno količino podatkov. Povezana je s programskimi orodji Visual Studio code, GitHub in na koncu s strežnikom Heroku. Vezja nam niso delala težav, le glede senzorjev se je bilo težje odločiti, katerega uporabiti in katerega ne. Pri 3d tiskanju in oblikovanju je bila najbolj problematična hitrost tiskanja, saj je 3d tiskanje zelo počasno.

Na koncu se še enkrat zahvaljujem mentorju in somentorju, ki sta mi s svojimi idejami pomagala pri diplomski nalogi. Zahvaljujem se tudi svoji družini za neverjetno spodbudo pri pisanju in izvedbi diplomske naloge.

VIRI IN LITERATURA

- [1] M. Mcgrath, HTML in easy steps, 16 Hamilton Terrace, Holly Walk, Leamington Spa CV32 4LY, Združeno kraljestvo: In Easy Steps Limited, 2020.
- [2] „L. Pan and J. S. Ma, "HTML+CSS Implementation based on Image Intelligent Scene Recognition Algorithm," 2022 International Conference on Augmented Intelligence and Sustainable Systems (ICAISS), Trichy, India, 2022, pp. 532-536, doi: 10.1109/ICAISS55157.2022“.
- [3] M. Mcgrath, CSS in easy steps, 16 Hamilton Terrace, Holly Walk, Leamington Spa CV32 4LY, Združeno kraljestvo, 2020 : In Easy Steps Limited, 2020.
- [4] „D. Sorn and S. Rimcharoen, "Web page template design using interactive genetic algorithm," 2013 International Computer Science and Engineering Conference (ICSEC), Nakhonpathom, Thailand, 2013, pp. 201-206, doi: 10.1109/ICSEC.2013.6694779.“.
- [5] F. Dongyan, „"An elastic - Fluid mixing Web layout design based on css," 2010 2nd International Conference on Education Technology and Computer, Shanghai, China pp. V1-518-V1-520, doi: 10.1109/ICETC.2010.5529195., 2010.
- [6] M. Mcgrath, PHP in easy steps - Updated for PHP 8, 16 Hamilton Terrace, Holly Walk, Leamington Spa CV32 4LY, Združeno kraljestvo: In Easy Steps Limited, 2020.
- [7] M. Mcgrath, SQL in easy steps, 16 Hamilton Terrace, Holly Walk, Leamington Spa CV32 4LY, Združeno kraljestvo: In Easy Steps Limited, 2020.
- [8] „E. Merlo, D. Letarte and G. Antoniol, "SQL-Injection Security Evolution Analysis in PHP," 2007 9th IEEE International Workshop on Web Site Evolution, Paris, France, 2007, pp. 45-49, doi: 10.1109/WSE.2007.4380243.“.

- [9] „freeicons.io,“ arduino icon, [Elektronski]. Available: Dostopno na:<https://freeicons.io/vector-and-svg-logos-02/arduino-icon-2452>. [Poskus dostopa 8.9.2023].
- [10] „freeicons.io,“ atom icon, [Elektronski]. Available: Dostopno na:<https://freeicons.io/icon/vector-and-svg-logos-02/atom-icon-2457>. [Poskus dostopa 8.9.2023].

PRILOGE

PRILOGA A: PROGRAMI

Spletna stran cas.css:

```
body{
  background: linear-gradient(141deg, #0b7870 30%, #17e6d8 51%, #0fa399
75%);
  margin: 0px;
}
.gumb_home{

  width:80px;
  height:80px;
}
.home{

  margin-top: 10px;
  position: absolute;
  left: -10px;
  width: 100px;
  height: 120px;

}
#tabela_graf{
  display: none;
}
.glava {

  background-color: linear-gradient(141deg, #17e6d85d 10% , #0fa399
65%);
  padding-bottom: 2px;
  padding-top: 2px;
  text-align: center;
  font-size: 45px;
  color: rgb(255, 255, 255);
  border-bottom: 6px solid white;

}
```

```
.glavna-beseda::before {
  transform: scaleX(0);
  transform-origin: bottom right;
}

.glavna-beseda:hover::before {
  transform: scaleX(1);
  transform-origin: bottom left;
}

.glavna-beseda::before {
  content: " ";
  display: block;
  position: absolute;
  top: 0; right: 0; bottom: 0; left: 0;
  inset: 0 0 0 0;
  background: #17e6d85d;
  z-index: -1;
  transition: transform 4s ease;
}

.glavna-beseda {
  position: relative;
  padding-top: 10px;
  padding-bottom: 10px;
}

.barva-gumb{
  border-top: 6px solid white;
  background-color: linear-gradient(141deg, #17e6d85d 10% , #0fa399
65%);
  padding: 30px;
  margin-bottom: -10px;
}

.gumb_glava{
  border-radius: 10px;
  border: 1px solid white;
  padding: 10px;
  background-color:#17e6d85d;
  color:white;
  text-decoration: none;
  display: inline-block;
  text-align: center;
}
```

```
.gumb_glava:hover{
  color:white;
  background-color: rgb(68, 64, 64);
  text-align: center;
}

.leva-stran-section {
  display: flex;
}

.leva-stran-nav {
  flex: 1;
  background-color: #0fa399;
  padding-bottom: 900px;
  border-right: 6px solid white;
  border-radius: 10px;
}

.leva-stran-nav-ul {
  list-style-type: none;
  text-align: center;
}

.leva-stran-nav-ul-li{
  padding-bottom: 100px;
}

.leva-stran-article {
  flex: 3;
  padding: 10px;
}

button{
  padding: 40px;
  padding-left: 100px;
  padding-right: 100px;
  margin-left: -50px;
  background-color: white;
  color: #0fa993;
  font-weight: bold;
  border: none;
  border-radius: 10px;
  letter-spacing: 10 px;
  overflow: hidden;
  transition: 0.5s;
  cursor: pointer;
```



```
    font-size: 20px;
  }

  button:hover{
    background: #14d5c8;
    color: white;
    box-shadow: 0 0 5px #14d5c8,
                0 0 25px #14d5c8,
                0 0 50px #14d5c8,
                0 0 200px #14d5c8;
  }

  #panel, #flip {
    padding: 5px;
    text-align: center;
    background-color: #0fa399;
    border: solid 1px #ffffff;
    border-radius: 10px;
  }

  #panel {
    padding: 50px;
    display: none;
    color:white;
    font-size: 20px;
  }

  .izpis{
    font-size: 50px;
    text-align: center;
    color: white;
    background-color: linear-gradient(141deg, #17e6d85d 10% , #0fa399
65%);
    padding: 20px;
    margin: -10px;
  }

  footer {
    border-top: 6px solid white;
    background-color: #3c4140;
    padding: 40px;
    color: rgb(255, 255, 255);
    font-size: 200%;
    font-family: 'Source Sans Pro', sans-serif;
```

```
}

.circle{
  height: 100px;
  width: 100px;
  background: #00A8AF;
  border: 10px solid #02787F;
  border-radius: 50%;
  margin-left: 1340px;
  margin-top: -250px;
}

.gradient{
  background: -webkit-radial-gradient(circle, #02AEB8, #04868B);
}

.Arduino{
  margin-top: 38px;
  margin-left: 15px;
  font-size: 20px;
  text-decoration: none;
  display: inline-block;
  text-align: center;
  color:BLACK;
}

.Arduino:hover{
  color:#02787F;
}

.gradient:hover{
  background: -webkit-radial-gradient(circle, white, #04868B);
}

.circle:hover{
  border: 10px solid #ffffff;
}
```

Spletna stran cas.php:

```
<?php

$cleardb_url = parse_url(getenv("CLEARDB_DATABASE_URL"));
$cleardb_server = $cleardb_url["host"];
$cleardb_username = $cleardb_url["user"];
$cleardb_password = $cleardb_url["pass"];
$cleardb_db = substr($cleardb_url["path"],1);
```

```
$active_group = 'default';
$query_builder = TRUE;

$conn = mysqli_connect($cleardb_server, $cleardb_username,
$cleardb_password, $cleardb_db);

mysqli_select_db($conn,$cleardb_db);
$test=array();

$res=mysqli_query($conn,"select temperatura from temperatura_senzor");
$count=0;
while($row = mysqli_fetch_array($res))
{
$test[$count] ["label"]=$count;
$test[$count] ["y"]=$row["temperatura"];
$count=$count+1;
}

?>

<!DOCTYPE html>
<html lang="en" style="margin-top: -20px;">

<head>
<link rel="stylesheet" href="cas.css">
<title>Arduino</title>
<link rel = "icon" href =
"https://static.thenounproject.com/png/34403-200.png"
type = "image/x-icon">
<meta http-equiv="Content-Type" content="text/html";
charset="utf-8" />
<meta name="viewport" content="width=device-width, initial-scale=1">
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.4/jquery.min.js"></
script>

<script>
window.onload = function () {

var chart = new CanvasJS.Chart("chartContainer", {
animationEnabled: true,
exportEnabled: true,
theme: "dark1", // "light1", "light2", "dark1", "dark2"
title:{
```

```
        text: "Prikaz temperature"
    },
    axisY:{
        includeZero: true
    },axisX:{
        includeZero: true
    },
    data: [{
        type: "line", //change type to bar, line, area, pie, etc
        //indexLabel: "{y}", //Shows y value on all Data Points
        indexLabelFontColor: "#8e5939",
        indexLabelPlacement: "outside",
        dataPoints: <?php echo json_encode($test, JSON_NUMERIC_CHECK); ?>
    }]
});
chart.render();
}
</script>

</head>
<body>
<header class="glava" >
<div class="glavna-beseda">
<div class="home">
<a href="index.php" ></a>
</div>
<h1>Čas</h1>
</div>

<div class="barva-gumb">

<a class="gumb_glava" href="vlaga_temperatura.php" >Vlaga in
temperatura</a>
<a class="gumb_glava" href="kvaliteta_vode.php" >Kvaliteta vode</a>
<a class="gumb_glava" href="vlaga_zemlje.php" >Vlaga zemlje</a>
<a class="gumb_glava" href="cas.php" style="background-color: rgb(68,
64, 64);">Čas</a>
<a class="gumb_glava" href="sonce.php" >Sonce</a>
</div>
</header>
<section class="leva-stran-section">
<nav class="leva-stran-nav">
<ul class="leva-stran-nav-ul">
```

```
<script>

function Tabela() {
document.getElementById("demo").style.fontSize = "50px";
document.getElementById("demo").style.color = "white";
document.getElementById("demo").style.backgroundColor = "linear-
gradient(141deg, #17e6d85d 10% , #0fa399 65%)";
demo.innerHTML = "Tabela";
}

</script>
<li class="leva-stran-nav-ul-li"><button class="gumb-levo-tabela"
onclick="Tabela();">Tabela</button></li>
</ul>
</nav>
  <script>
    $(document).ready(function(){
      $("#flip").click(function(){
        $("#panel").slideToggle("slow");
      });
    });
  </script>

  <article class="leva-stran-article">

    <div id="flip">
  <h1 class="izpis" id="demo">Izpis</h1>
  </div >
  <div id="panel">
  Spodaj se izriše graf ali tabela.
  </div>

<?php
$sql = " SELECT vlaga,temperatura,vlagaZemlje,voda,sonce,cas FROM
temperatura_senzor";
$result = mysqli_query($conn, $sql);

echo "<br>";
echo "
<style>
tr:hover {background-color: black;color:white}
</style>
```

```
";

echo "<table style=' border-collapse:collapse; width: 100%; margin-left:
auto;margin-right: auto; '>";
echo "<tr>";
echo "<th style='color:white; font-size: 25px;'>Vlaga [%]</th>";
echo "<th style='color:white; font-size: 25px;'>Temperatura [°C]</th>";
echo "<th style='color:white; font-size: 25px;'>Vlaga zemlje</th>";
echo "<th style='color:white; font-size: 25px;'>Število delcev na milijon
[ppm]</th>";
echo "<th style='color:white; font-size: 25px;'>Fotoupor</th>";
echo "<th style='color:white; font-size: 25px;'>Čas</th>";
echo "</tr>";
while ($row = mysqli_fetch_assoc($result)) {

echo "<tr style='padding: 8px; text-align: left; border-bottom: 1px
solid black;text-align: center;font-size: 25px; '>";
foreach ($row as $field => $value) {
echo "<td style='color:white;'>" . $value . "</td>";
}
echo "</tr>";
}
echo "</table>";

?>

</article>
</section>
</div>
<footer>

<p>Oseba: Tomaž</p>
<p><a href="mailto:diplomski.del@gmail.com">gmail</a></p>
</footer>

</script>
<div>
<div class="circle gradient">
```

```
<a href="https://www.arduino.cc/" class="Arduino">Arduino</a>
</div>
</div>
</body>

</html>
```

Spletna stran index.css:

```
body{
  background-color: #5b47f4;
  margin: 0px;
}

.glava {
  padding-bottom: 2px;
  padding-top: 2px;
  text-align: center;
  font-size: 40px;
  color: rgb(255, 255, 255);
  border-bottom: 6px solid white;
}

.glavna-beseda::before {
  transform: scaleX(0);
  transform-origin: bottom right;
}

.glavna-beseda:hover::before {
  transform: scaleX(1);
  transform-origin: bottom left;
}

.glavna-beseda::before {
  content: " ";
  display: block;
  position: absolute;
  top: 0; right: 0; bottom: 0; left: 0;
  inset: 0 0 0 0;
  background: #4632e05d;
  z-index: -1;
  transition: transform 4s ease;
}

.glavna-beseda {
```

```
position: relative;
padding-top: 10px;
padding-bottom: 10px;
}

.barva-gumb{
border-top: 6px solid white;
background-color: #1d00f95d;
padding: 30px;
margin-bottom: -10px;
}

.gumb_glava{
border-radius: 10px;
border: 1px solid white;
padding: 10px;
margin-right: 3%;
margin-left: 0.5%;

color:white;
text-decoration: none;
display: inline-block;
text-align: center;
}

.gumb_glava:hover{
color:white;
background-color: rgb(68, 64, 64);
text-align: center;
}

.besedilo_sredina{
font-size: 30px;
text-align: center;
color:white;
}

.glava2{
padding-bottom: 2px;
padding-top: 2px;
text-align: center;
font-size: 45px;
color: rgb(255, 255, 255);
}
```



```
.glavna-beseda2::before {
  transform: scaleX(0);
  transform-origin: bottom right;
}

.glavna-beseda2: hover::before {
  transform: scaleX(1);
  transform-origin: bottom left;
}

.glavna-beseda2::before {
  content: " ";
  display: block;
  position: absolute;
  top: 0; right: 0; bottom: 0; left: 0;
  inset: 0 0 0 0;
  z-index: -1;
  transition: transform 4s ease;
}

.glavna-beseda2 {
  position: relative;
  padding-top: 10px;
  padding-bottom: 10px;
}

.barva-gumb2{

  padding: 30px;
  margin-bottom: -10px;
}

.gumb_glava2{
  border-radius: 10px;
  border: 1px solid white;
  padding: 10px;
  margin-right: 3%;
  margin-left: 0.5%;

  color:white;
  text-decoration: none;
  display: inline-block;
  text-align: center;
}
```

```
.gumb_glava2:hover{
  color:white;
  background-color: rgb(68, 64, 64);
  text-align: center;
}

footer {
  border-top: 6px solid white;
  background-color: #3c4140;
  padding: 40px;
  color: rgb(255, 255, 255);
  font-size: 200%;
  font-family: 'Source Sans Pro', sans-serif;
}

.circle{
  height: 100px;
  width: 100px;
  background:#466aed;
  border: 10px solid #2b46a5;
  border-radius: 50%;
  margin-left: 1340px;
  margin-top: -250px;
}

.gradient{
  background: -webkit-radial-gradient(circle, #233c95, #466aed);
}

.Arduino{
  margin-top: 38px;
  margin-left: 15px;
  font-size: 20px;
  text-decoration: none;
  display: inline-block;
  text-align: center;
  color:rgb(255, 255, 255);
}

.Arduino:hover{
  color:#000000;
}

.gradient:hover{
```

```
background: -webkit-radial-gradient(circle, white, #2c4290);
}

.circle:hover{
border: 10px solid #ffffff;
}
```

Spletna stran index.php:

```
<!DOCTYPE html>
<html lang="sl">
<head>
  <title>Diplomska naloga</title>
  <link rel = "icon" href =
"https://static.thenounproject.com/png/34403-200.png"
  type = "image/x-icon">
  <meta http-equiv="Content-Type" content="text/html"; charset="utf-8" />
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" type="text/css" href="index.css">
</head>
<body>
<header class="glava" >
<div class="glavna-beseda">

  <h1 > Diplomska naloga</h1>
  </div>
  <div class="barva-gumb">

    <a class="gumb_glava" href="index.php" style="
background-color: rgb(68, 64, 64);">Domača stran</a>

    <a class="gumb_glava"
href="senzorji.php">Senzorji</a>

    <a class="gumb_glava" href="info.php">Info </a>
  </div>
</header>

  <h1 class="besedilo_sredina" >Dobrodošli na spletno stran </h1>
  <p style="text-align: center;color:white;">Spodaj so prikazani
podatki oziroma informacije od senzorjev.</p>

  <div class="glava2">
  <div class="barva-gumb2">
```

```

        <a class="gumb_glava2"
href="vlaga_temperatura.php">Vlaga/temperatura</a>
        <a class="gumb_glava2"
href="kvaliteta_vode.php">Kvalitete vode</a>
        <a class="gumb_glava2" href="vlaga_zemlje.php">Vlaga
zemlje</a>
        <a class="gumb_glava2" href="cas.php">Čas</a>
        <a class="gumb_glava2" href="sonce.php" >Sonce</a>
    </div>
</div>

    <footer>

<p>Oseba: Tomaž</p>
<p><a href="mailto:diplomski.del@gmail.com">gmail</a></p>
</footer>

</script>
<div>
<div class="circle gradient">
<a href="https://www.arduino.cc/" class="Arduino">Arduino</a>
</div>
</div>
</body>
</html>

```

Spletna stran info.php:

```

<!DOCTYPE html>
<html lang="sl">
<head>
    <title>Diplomska naloga</title>
    <link rel = "icon" href =
"https://static.thenounproject.com/png/34403-200.png"
    type = "image/x-icon">
    <meta http-equiv="Content-Type" content="text/html";
charset="utf-8" />
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" type="text/css" href="index.css">
</head>
<body>
<header class="glava" >
<div class="glavna-beseda">

        <h1 > Diplomska naloga</h1>

```

```

        </div>
        <div class="barva-gumb">

            <a class="gumb_glava" href="index.php" >Domača
stran</a>

            <a class="gumb_glava"
href="senzorji.php">Senzorji</a>

            <a class="gumb_glava" href="info.php" style="
background-color: rgb(68, 64, 64);">Info </a>
        </div>
</header>

        <h1 class="besedilo_sredina" >Info</h1>
        <p style="text-align: center;color:white;">Spletna stran je
narajena za diplomsko nalogo. Za več informacij lahko kontaktirate. </p>

        <footer>

<p>Oseba: Tomaž</p>
<p><a href="mailto:diplomski.del@gmail.com">gmail</a></p>
</footer>

</script>
<div>
<div class="circle gradient">
<a href="https://www.arduino.cc/" class="Arduino">Arduino</a>
</div>
</div>
</body>
</html>

```

Spletna stran kvaliteta_vode.css:

```

body{
    background: linear-gradient(141deg, #4e70ed 30%, #617ee7 51%, #3958c9
75%);
    margin: 0px;
}
.gumb_home{

    width:80px;
    height:80px;

```

```
}
.home{

  margin-top: 10px;
  position: absolute;
left: -10px;
width: 100px;
height: 120px;

}

.glava {

  background-color: linear-gradient(141deg, #466aed 10% , #466aed 65%);
padding-bottom: 2px;
padding-top: 2px;
text-align: center;
font-size: 45px;
color: rgb(255, 255, 255);
border-bottom: 6px solid white;

}

}

.glavna-beseda::before {
  transform: scaleX(0);
  transform-origin: bottom right;
}

.glavna-beseda:hover::before {
  transform: scaleX(1);
  transform-origin: bottom left;
}

.glavna-beseda::before {
  content: " ";
  display: block;
  position: absolute;
  top: 0; right: 0; bottom: 0; left: 0;
  inset: 0 0 0 0;
  background: #3259e4;
  z-index: -1;
  transition: transform 4s ease;
}

}

.glavna-beseda {
  position: relative;
padding-top: 10px;
```

```
padding-bottom: 10px;
}

.barva-gumb{
border-top: 6px solid white;
background-color: linear-gradient(141deg, #466aed 10% , #617ee7 65%);
padding: 30px;
margin-bottom: -10px;
}

.gumb_glava{
border-radius: 10px;
border: 1px solid white;
padding: 10px;

background-color:#466aed;
color:white;
text-decoration: none;
display: inline-block;
text-align: center;
}

.gumb_glava:hover{
color:white;
background-color: rgb(68, 64, 64);
text-align: center;
}

.leva-stran-section {
display: flex;
}

.leva-stran-nav {
flex: 1;
background-color: #466aed;
padding-bottom: 900px;
border-right: 6px solid white;
border-radius: 10px;
}

.leva-stran-nav-ul {
list-style-type: none;
text-align: center;
}

.leva-stran-nav-ul-li{
```

```
padding-bottom: 100px;
}

.leva-stran-article {
  flex: 3;
  padding: 10px;
}

button{
  padding: 40px;
  padding-left: 100px;
  padding-right: 100px;
  margin-left: -50px;
  background-color: white;
  color: #433831;
  font-weight: bold;
  border: none;
  border-radius: 10px;
  letter-spacing: 10 px;
  overflow: hidden;
  transition: 0.5s;
  cursor: pointer;
  font-size: 20px;
}

button:hover{
  background: #5d7ced;
  color: white;
  box-shadow: 0 0 5px #5d7ced,
             0 0 25px #5d7ced,
             0 0 50px #5d7ced,
             0 0 200px #5d7ced;
}

#panel, #flip {
  padding: 5px;
  text-align: center;
  background-color: #466aed;
  border: solid 1px #c3c3c3;
  border-radius: 10px;
}

#panel {
  padding: 50px;
  display: none;
  color:white;
}
```



```
    font-size: 20px;
  }
  #tabela_graf{
    display: none;
  }

  .izpis{
    font-size: 50px;
    text-align: center;
    color: white;
    background-color: linear-gradient(141deg, #466aed 10% , #5e7ae0 65%);
    padding: 20px;
    margin: -10px;

  }

  footer {
    border-top: 6px solid white;
    background-color: #3c4140;
    padding: 40px;
    color: rgb(255, 255, 255);
    font-size: 200%;
    font-family: 'Source Sans Pro', sans-serif;
  }

  .circle{
    height: 100px;
    width: 100px;
    background:#466aed;
    border: 10px solid #2b46a5;
    border-radius: 50%;
    margin-left: 1340px;
    margin-top: -300px;

  }

  .gradient{
    background: -webkit-radial-gradient(circle, #233c95, #466aed);
  }

  .Arduino{
    margin-top: 38px;
    margin-left: 15px;
    font-size: 20px;
    text-decoration: none;
    display: inline-block;
    text-align: center;
```

```
    color:rgb(255, 255, 255);
  }
  .Arduino:hover{
    color:#000000;
  }

  .gradient:hover{
    background: -webkit-radial-gradient(circle, white, #2c4290);
  }

  .circle:hover{
    border: 10px solid #ffffff;
  }
}
```

Spletna stran kvaliteta_vode.php:

```
<?php

$cleardb_url = parse_url(getenv("CLEARDB_DATABASE_URL"));
$cleardb_server = $cleardb_url["host"];
$cleardb_username = $cleardb_url["user"];
$cleardb_password = $cleardb_url["pass"];
$cleardb_db = substr($cleardb_url["path"],1);
$active_group = 'default';
$query_builder = TRUE;

$conn = mysqli_connect($cleardb_server, $cleardb_username,
$cleardb_password, $cleardb_db);

mysqli_select_db($conn,$cleardb_db);
$test=array();

$res=mysqli_query($conn,"select voda from temperatura_senzor");
$count=0;
while($row = mysqli_fetch_array($res))
{
    $test[$count] ["label"]=$count;
    $test[$count] ["y"]=$row["voda"];
    $count=$count+1;
}

?>

<!DOCTYPE html>
<html lang="en" style="margin-top: -20px;">
```

```
<head>
<link rel="stylesheet" href="kvaliteta_vode.css">
<title>Arduino</title>
<link rel = "icon" href =
"https://static.thenounproject.com/png/34403-200.png"
    type = "image/x-icon">
    <meta http-equiv="Content-Type" content="text/html";
charset="utf-8" />
<meta name="viewport" content="width=device-width, initial-scale=1">
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.4/jquery.min.js"></
script>

<script>
window.onload = function () {

var chart = new CanvasJS.Chart("chartContainer", {
    animationEnabled: true,
    exportEnabled: true,
    theme: "dark1", // "light1", "light2", "dark1", "dark2"
    title:{
        text: "Prikaz kvalitete vode"
    },
    axisY:{
        title: "Število delcev na milijon [ppm]",
        includeZero: true
    },axisX:{
        title: "Število meritve",
        includeZero: true
    },
    data: [{
        type: "line", //change type to bar, line, area, pie, etc
        //indexLabel: "{y}", //Shows y value on all Data Points
        indexLabelFontColor: "#8e5939",
        indexLabelPlacement: "outside",
        dataPoints: <?php echo json_encode($test, JSON_NUMERIC_CHECK); ?>
    }]
    });
chart.render();

}
</script>

</head>
```

```
<body>
<header class="glava" >
<div class="glavna-beseda">
<div class="home">
<a href="index.php" ></a>
</div>
<h1>Kvaliteta vode</h1>
</div>

<div class="barva-gumb">

<a class="gumb_glava" href="vlaga_temperatura.php" >Vlaga in
temperatura</a>
<a class="gumb_glava" href="kvaliteta_vode.php" style="background-color:
rgb(68, 64, 64);" >Kvaliteta vode</a>
<a class="gumb_glava" href="vlaga_zemlje.php" >Vlaga zemlje</a>
<a class="gumb_glava" href="cas.php">Čas</a>
<a class="gumb_glava" href="sonce.php" >Sonce</a>
</div>
</header>
<section class="leva-stran-section">
<nav class="leva-stran-nav">
<ul class="leva-stran-nav-ul">
<script>
function Graf() {
document.getElementById("demo").style.fontSize = "50px";
document.getElementById("demo").style.color = "white";
document.getElementById("demo").style.backgroundColor = "linear-
gradient(141deg, #17e6d85d 10% , #0fa399 65%)";
demo.innerHTML = "Graf";}

function Tabela() {
document.getElementById("demo").style.fontSize = "50px";
document.getElementById("demo").style.color = "white";
document.getElementById("demo").style.backgroundColor = "linear-
gradient(141deg, #17e6d85d 10% , #0fa399 65%)";
demo.innerHTML = "Tabela";
}
function Graf_Tabela() {
document.getElementById("demo").style.fontSize = "50px";
document.getElementById("demo").style.color = "white";
document.getElementById("demo").style.backgroundColor = "linear-
gradient(141deg, #17e6d85d 10% , #0fa399 65%)";
demo.innerHTML = "Graf in tabela";
}
}
```

```
</script>
<li class="leva-stran-nav-ul-li"><button class="gumb-levo-graf"
onclick="Graf(); funkcija_graf();">Graf</button></li>
<li class="leva-stran-nav-ul-li"><button class="gumb-levo-tabela"
onclick="Tabela();funkcija_tabela();">Tabela</button></li>
<li class="leva-stran-nav-ul-li"><button class="gumb-levo-graf-tabela"
onclick="Graf_Tabela();funkcija_tabela_graf();">Graf/Tabela</button></li>
</ul>
</nav>

<script>
$(document).ready(function(){
  $("#flip").click(function(){
    $("#panel").slideToggle("slow");
  });
});
</script>

<article class="leva-stran-article">

  <div id="flip">
<h1 class="izpis" id="demo">Izpis</h1>
</div >
<div id="panel">
Spodaj se izriše graf ali tabela.
</div>

  <div id="tabela__">
<?php
$sql = "SELECT voda,cas FROM temperatura_senzor";
$result = mysqli_query($conn, $sql);

echo "<br>";
echo "
<style>
tr:hover {background-color: black;color:white}
</style>
";

echo "<table style=' border-collapse:collapse; width: 100%; margin-left:
auto;margin-right: auto; '>";
echo "<tr>";
echo "<th style='color:white; font-size: 25px;'>Število delcev na milijon
[ppm]</th>";
echo "<th style='color:white; font-size: 25px;'>Čas</th>";
```

```
echo "</tr>";
while ($row = mysqli_fetch_assoc($result)) {

echo "<tr style='padding: 8px; text-align: left; border-bottom: 1px
solid black;text-align: center;font-size: 25px; '>";
foreach ($row as $field => $value) {
echo "<td style='color:white;'" . $value . "</td>";
}
echo "</tr>";
}
echo "</table>";

?>
</div>
<script>
function funkcija_tabela() {
    var x = document.getElementById("tabela__");
    if (x.style.display === "none") {
        x.style.display = "block";
        demo.innerHTML = "Tabela";
    } else {
        x.style.display = "none";
        demo.innerHTML = "Izberi";
    }

    var y = document.getElementById("graf__");
    var z = document.getElementById("tabela_graf");
    z.style.display = "none";
    y.style.display = "none";

}

</script>

<div style="height: 300px; overflow: auto" id="tabela_graf">
<?php
$sql = "SELECT voda,cas FROM temperatura_senzor";
$result = mysqli_query($conn, $sql);

echo "<br>";
echo "
<style>
tr:hover {background-color: black;color:white}
</style>
";
```

```
echo "<table style=' border-collapse:collapse; width: 100%; margin-left:
auto;margin-right: auto; '>";
echo "<tr>";
echo "<th style='color:white; font-size: 25px;'>Število delcev na milijon
[ppm]</th>";
echo "<th style='color:white; font-size: 25px;'>Čas</th>";
echo "</tr>";
while ($row = mysqli_fetch_assoc($result)) {

echo "<tr style='padding: 8px; text-align: left; border-bottom: 1px
solid black;text-align: center;font-size: 25px; '>";
foreach ($row as $field => $value) {
echo "<td style='color:white;'>" . $value . "</td>";
}
echo "</tr>";
}
echo "</table>";

?>
</div>

<div id="graf__">
<div id="chartContainer" style="height: 370px; width: 100%; margin-
top:20px;"></div>
<script src="https://cdn.canvasjs.com/canvasjs.min.js"></script>

</article>
</section>
</div>
<script>
function funkcija_tabela_graf() {
  var x = document.getElementById("tabela_graf");
  var z = document.getElementById("graf__");
  if (x.style.display === "none" && z.style.display === "none") {
    x.style.display = "block";
    z.style.display = "block";
    demo.innerHTML = "Tabela/Graf";
  }
  if(x.style.display === "block" && z.style.display === "none"){
    x.style.display = "none";
```

```
z.style.display = "none";
demo.innerHTML = "Izberi";
}
if(x.style.display === "none" && z.style.display === "block"){
x.style.display = "block";
z.style.display = "block";
demo.innerHTML = "Tabela/Graf";
}

var y = document.getElementById("tabela__");
y.style.display = "none";
}
</script>

<script>
function funkcija_graf() {
var x = document.getElementById("graf__");
if (x.style.display === "none") {
x.style.display = "block";
demo.innerHTML = "Graf";
} else {
x.style.display = "none";
demo.innerHTML = "Izberi";
}

var y = document.getElementById("tabela__");
y.style.display = "none";
var z = document.getElementById("tabela_graf");
z.style.display = "none";
}
</script>

<footer>

<p>Oseba: Tomaž</p>
<p><a href="mailto:diplomski.del@gmail.com">gmail</a></p>
</footer>

</script>

<div>
<div class="circle gradient">
<a href="https://www.arduino.cc/" class="Arduino">Arduino</a>
```



```
</div>
</div>

</body>

</html>
```

Spletna stran senzorji.php:

```
<!DOCTYPE html>
<html lang="sl">
<head>
  <title>Diplomska naloga</title>
  <link rel = "icon" href =
  "https://static.thenounproject.com/png/34403-200.png"
  type = "image/x-icon">
  <meta http-equiv="Content-Type" content="text/html";
  charset="utf-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" type="text/css" href="index.css">
</head>
<body>
<header class="glava" >
<div class="glavna-beseda">

  <h1 > Diplomska naloga</h1>
  </div>
  <div class="barva-gumb">

    <a class="gumb_glava" href="index.php" >Domača
stran</a>

    <a class="gumb_glava" href="senzorji.php" style="
background-color: rgb(68, 64, 64);">Senzorji</a>

    <a class="gumb_glava" href="info.php">Info </a>
  </div>
</header>

  <h1 class="besedilo_sredina" >Senzorji</h1>
  <p style="text-align: center;color:white;">Senzorji so od
arduinoja.</p>
```

```
        <footer>

<p>Oseba: Tomaž</p>
<p><a href="mailto:diplomski.del@gmail.com">gmail</a></p>
</footer>

</script>
<div>
<div class="circle gradient">
<a href="https://www.arduino.cc/" class="Arduino">Arduino</a>
</div>
</div>
</body>
</html>
```

Spletna stran sonce.css:

```
body{
    background: linear-gradient(141deg, #cded4e 30%, #617ee7 51%, #cded4e
75%);
    margin: 0px;
}
.gumb_home{

    width:80px;
    height:80px;
}
.home{

    margin-top: 10px;
    position: absolute;
    left: -10px;
    width: 100px;
    height: 120px;

}
.glava {

    background-color: linear-gradient(141deg, #cded4e 10% , #466aed 65%);
    padding-bottom: 2px;
    padding-top: 2px;
    text-align: center;
    font-size: 45px;
    color: rgb(255, 255, 255);
    border-bottom: 6px solid white;

}
}
```

```
.glavna-beseda::before {
  transform: scaleX(0);
  transform-origin: bottom right;
}

.glavna-beseda:hover::before {
  transform: scaleX(1);
  transform-origin: bottom left;
}

.glavna-beseda::before {
  content: " ";
  display: block;
  position: absolute;
  top: 0; right: 0; bottom: 0; left: 0;
  inset: 0 0 0 0;
  background: #3259e4;
  z-index: -1;
  transition: transform 4s ease;
}

.glavna-beseda {
  position: relative;
  padding-top: 10px;
  padding-bottom: 10px;
}

.barva-gumb{
  border-top: 6px solid white;
  background-color: linear-gradient(141deg, #cded4e 10% , #617ee7 65%);
  padding: 30px;
  margin-bottom: -10px;
}

.gumb_glava{
  border-radius: 10px;
  border: 1px solid white;
  padding: 10px;

  background-color:#466aed;
  color:white;
  text-decoration: none;
  display: inline-block;
  text-align: center;
}
```

```
.gumb_glava:hover{
  color:white;
  background-color: rgb(68, 64, 64);
  text-align: center;
}

.leva-stran-section {
  display: flex;
}

.leva-stran-nav {
  flex: 1;
  background-color: #466aed;
  padding-bottom: 900px;
  border-right: 6px solid white;
  border-radius: 10px;
}

.leva-stran-nav-ul {
  list-style-type: none;
  text-align: center;
}

}

.leva-stran-nav-ul-li{
  padding-bottom: 100px;
}

}

.leva-stran-article {
  flex: 3;
  padding: 10px;
}

}

button{
  padding: 40px;
  padding-left: 100px;
  padding-right: 100px;
  margin-left: -50px;
  background-color: white;
  color: #433831;
  font-weight: bold;
  border: none;
  border-radius: 10px;
  letter-spacing: 10 px;
  overflow: hidden;
  transition: 0.5s;
}
```

```
    cursor: pointer;
    font-size: 20px;
}

button:hover{
    background: #5d7ced;
    color: white;
    box-shadow: 0 0 5px #cded4e,
                0 0 25px #cded4e,
                0 0 50px #cded4e,
                0 0 200px #cded4e;
}

#panel, #flip {
    padding: 5px;
    text-align: center;
    background-color: #466aed;
    border: solid 1px #c3c3c3;
    border-radius: 10px;
}

#panel {
    padding: 50px;
    display: none;
    color:white;
    font-size: 20px;
}

#tabela_graf{
    display: none;
}

.izpis{
    font-size: 50px;
    text-align: center;
    color: white;
    background-color: linear-gradient(141deg, #cded4e 10% , #5e7ae0 65%);
    padding: 20px;
    margin: -10px;
}

footer {
    border-top: 6px solid white;
    background-color: #3c4140;
    padding: 40px;
    color: rgb(255, 255, 255);
```

```
font-size: 200%;
font-family: 'Source Sans Pro', sans-serif;
}

.circle{
  height: 100px;
  width: 100px;
  background:#466aed;
  border: 10px solid #cded4e;
  border-radius: 50%;
  margin-left: 1340px;
  margin-top: -250px;
}

.gradient{
  background: -webkit-radial-gradient(circle, #cded4e, #466aed);
}

.Arduino{
  margin-top: 38px;
  margin-left: 15px;
  font-size: 20px;
  text-decoration: none;
  display: inline-block;
  text-align: center;
  color:rgb(255, 255, 255);
}

.Arduino:hover{
  color:#000000;
}

.gradient:hover{
  background: -webkit-radial-gradient(circle, white, #2c4290);
}

.circle:hover{
  border: 10px solid #ffffff;
}
```

Spletna stran sonce.php:

```
<?php
$cleardb_url = parse_url(getenv("CLEARDB_DATABASE_URL"));
$cleardb_server = $cleardb_url["host"];
$cleardb_username = $cleardb_url["user"];
```

```
$cleardb_password = $cleardb_url["pass"];
$cleardb_db = substr($cleardb_url["path"],1);
$active_group = 'default';
$query_builder = TRUE;

$conn = mysqli_connect($cleardb_server, $cleardb_username,
$cleardb_password, $cleardb_db);

mysqli_select_db($conn,$cleardb_db);
$test=array();

$res=mysqli_query($conn,"select sonce from temperatura_senzor");
$count=0;
while($row = mysqli_fetch_array($res))
{
    $test[$count] ["label"]=$count;
    $test[$count] ["y"]=$row["sonce"];
    $count=$count+1;
}

?>

<!DOCTYPE html>
<html lang="en" style="margin-top: -20px;">

<head>
<link rel="stylesheet" href="sonce.css">
<title>Arduino</title>
<link rel = "icon" href =
"https://static.thenounproject.com/png/34403-200.png"
    type = "image/x-icon">
    <meta http-equiv="Content-Type" content="text/html";
charset="utf-8" />
<meta name="viewport" content="width=device-width, initial-scale=1">
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.4/jquery.min.js"></
script>

<script>
window.onload = function () {

var chart = new CanvasJS.Chart("chartContainer", {
    animationEnabled: true,
    exportEnabled: true,
```

```
theme: "dark1", // "light1", "light2", "dark1", "dark2"
title:{
  text: "Svetloba"
},
axisY:{
  title: "Fotoupor",
  includeZero: true
},axisX:{
  title: "Število meritve",
  includeZero: true
},
data: [{
  type: "line", //change type to bar, line, area, pie, etc
  //indexLabel: "{y}", //Shows y value on all Data Points
  indexLabelFontColor: "#8e5939",
  indexLabelPlacement: "outside",
  dataPoints: <?php echo json_encode($test, JSON_NUMERIC_CHECK); ?>
}]
});
chart.render();
}
</script>

</head>
<body>
<header class="glava" >
<div class="glavna-beseda">
<div class="home">
<a href="index.php" ></a>
</div>
<h1>Sonce</h1>
</div>
<div class="barva-gumb">
<a class="gumb_glava" href="vlaga_temperatura.php">Vlaga in
temperatura</a>
<a class="gumb_glava" href="kvaliteta_vode.php">Kvaliteta vode</a>
<a class="gumb_glava" href="vlaga_zemlje.php" >Vlaga zemlje</a>
<a class="gumb_glava" href="cas.php">Čas</a>
<a class="gumb_glava" href="sonce.php" style="background-color: rgb(68,
64, 64);" >Sonce</a>
</div>
</header>
<section class="leva-stran-section">
```



```
<nav class="leva-stran-nav">
<ul class="leva-stran-nav-ul">
<script>
function Graf() {
document.getElementById("demo").style.fontSize = "50px";
document.getElementById("demo").style.color = "white";
document.getElementById("demo").style.backgroundColor = "linear-
gradient(141deg, #17e6d85d 10% , #0fa399 65%)";
demo.innerHTML = "Graf";}

function Tabela() {
document.getElementById("demo").style.fontSize = "50px";
document.getElementById("demo").style.color = "white";
document.getElementById("demo").style.backgroundColor = "linear-
gradient(141deg, #17e6d85d 10% , #0fa399 65%)";
demo.innerHTML = "Tabela";
}
function Graf_Tabela() {
    document.getElementById("demo").style.fontSize = "50px";
    document.getElementById("demo").style.color = "white";
    document.getElementById("demo").style.backgroundColor = "linear-
gradient(141deg, #17e6d85d 10% , #0fa399 65%)";
    demo.innerHTML = "Graf in tabela";
}
</script>
<li class="leva-stran-nav-ul-li"><button class="gumb-levo-graf"
onclick="Graf(); funkcija_graf();">Graf</button></li>
<li class="leva-stran-nav-ul-li"><button class="gumb-levo-tabela"
onclick="Tabela();funkcija_tabela();">Tabela</button></li>
<li class="leva-stran-nav-ul-li"><button class="gumb-levo-graf-tabela"
onclick="Graf_Tabela();funkcija_tabela_graf();">Graf/Tabela</button></li>
</ul>
</nav>

<script>
$(document).ready(function(){
    $("#flip").click(function(){
        $("#panel").slideToggle("slow");
    });
});
</script>

<article class="leva-stran-article">

    <div id="flip">
<h1 class="izpis" id="demo">Izpis</h1>
```

```
</div >
<div id="panel">
  Spodaj se izriše graf ali tabela.
</div>

  <div id="tabela__">
<?php
$sql = " SELECT sonce,cas FROM temperatura_senzor";
$result = mysqli_query($conn, $sql);

echo "<br>";
echo "
<style>
tr:hover {background-color: black;color:white}
</style>
";

echo "<table style=' border-collapse:collapse; width: 100%; margin-left:
auto;margin-right: auto; '>";
echo "<tr>";
echo "<th style='color:white; font-size: 25px;'>Fotoupor</th>";
echo "<th style='color:white; font-size: 25px;'>Čas</th>";
echo "</tr>";
while ($row = mysqli_fetch_assoc($result)) {

echo "<tr style='padding: 8px; text-align: left; border-bottom: 1px
solid black;text-align: center;font-size: 25px; '>";
foreach ($row as $field => $value) {
echo "<td style='color:white;'>" . $value . "</td>";
}
echo "</tr>";
}
echo "</table>";

?>
</div>
<script>
function funkcija_tabela() {
  var x = document.getElementById("tabela__");
  if (x.style.display === "none") {
    x.style.display = "block";
    demo.innerHTML = "Tabela";
  } else {
    x.style.display = "none";
    demo.innerHTML = "Izberi";
  }
}
```

```
}
    var y = document.getElementById("graf__");
    var z = document.getElementById("tabela_graf");
    z.style.display = "none";
    y.style.display = "none";
}
</script>

<div style="height: 300px; overflow: auto" id="tabela_graf">
<?php
$sql = " SELECT sonce,cas FROM temperatura_senzor";
$result = mysqli_query($conn, $sql);

echo "<br>";
echo "
<style>
tr:hover {background-color: black;color:white}
</style>
";

echo "<table style=' border-collapse:collapse; width: 100%; margin-left:
auto;margin-right: auto; '>";
echo "<tr>";
echo "<th style='color:white; font-size: 25px;'>Fotoupor</th>";
echo "<th style='color:white; font-size: 25px;'>Čas</th>";
echo "</tr>";
while ($row = mysqli_fetch_assoc($result)) {

echo "<tr style='padding: 8px; text-align: left; border-bottom: 1px
solid black;text-align: center;font-size: 25px; '>";
foreach ($row as $field => $value) {
echo "<td style='color:white;'>" . $value . "</td>";
}
echo "</tr>";
}
echo "</table>";

?>
</div>

<div id="graf__">
```

```
<div id="chartContainer" style="height: 370px; width: 100%; margin-top:20px;"></div>
<script src="https://cdn.canvasjs.com/canvasjs.min.js"></script>

</article>
</section>
</div>
<script>
function funkcija_tabela_graf() {
  var x = document.getElementById("tabela_graf");
  var z = document.getElementById("graf__");
  if (x.style.display === "none" && z.style.display === "none") {
    x.style.display = "block";
    z.style.display = "block";
    demo.innerHTML = "Tabela/Graf";
  }
  if(x.style.display === "block" && z.style.display === "none"){
    x.style.display = "none";
    z.style.display = "none";
    demo.innerHTML = "Izberi";
  }
  if(x.style.display === "none" && z.style.display === "block"){
    x.style.display = "block";
    z.style.display = "block";
    demo.innerHTML = "Tabela/Graf";
  }
  }

  var y = document.getElementById("tabela__");
  y.style.display = "none";
}
</script>

<script>
function funkcija_graf() {
  var x = document.getElementById("graf__");
  if (x.style.display === "none") {
    x.style.display = "block";
    demo.innerHTML = "Graf";
  } else {
    x.style.display = "none";
    demo.innerHTML = "Izberi";
  }
}
```

```
}

var y = document.getElementById("tabela__");
y.style.display = "none";
var z = document.getElementById("tabela_graf");
z.style.display = "none";

}
</script>

<footer>

<p>Oseba: Tomaž</p>
<p><a href="mailto:diplomski.del@gmail.com">gmail</a></p>
</footer>

</script>
<div>
<div class="circle gradient">
<a href="https://www.arduino.cc/" class="Arduino">Arduino</a>
</div>
</div>
</body>

</html>
```

Spletna stran vlaga_temperatura.css:

```
body{
background: linear-gradient(141deg, #4e8bed 10%, #ff0000 31%, #3958c9
75%);
margin: 0px;
}
.gumb_home{

width:80px;
height:80px;
}
.home{

margin-top: 10px;
position: absolute;
left: -10px;
width: 100px;
height: 120px;
```

```
}
#tabela_graf{
  display: none;
}
.glava {

  background-color: linear-gradient(141deg, #4e8bed 10% , #ff0000 31%);
  padding-bottom: 2px;
  padding-top: 2px;
  text-align: center;
  font-size: 45px;
  color: rgb(255, 255, 255);
  border-bottom: 6px solid white;

}

.glavna-beseda::before {
  transform: scaleX(0);
  transform-origin: bottom right;
}

.glavna-beseda:hover::before {
  transform: scaleX(1);
  transform-origin: bottom left;
}

.glavna-beseda::before {
  content: " ";
  display: block;
  position: absolute;
  top: 0; right: 0; bottom: 0; left: 0;
  inset: 0 0 0 0;
  background: #3259e4;
  z-index: -1;
  transition: transform 4s ease;
}

.glavna-beseda {
  position: relative;
  padding-top: 10px;
  padding-bottom: 10px;
}

.barva-gumb{
  border-top: 6px solid white;
  background-color: linear-gradient(141deg, #4e8bed 10% , #ff0000 31%);
```

```
padding: 30px;
margin-bottom: -10px;
}

.gumb_glava{
border-radius: 10px;
border: 1px solid white;
padding: 10px;

background-color:#466aed;
color:white;
text-decoration: none;
display: inline-block;
text-align: center;
}

.gumb_glava:hover{
color:white;
background-color: rgb(68, 64, 64);
text-align: center;
}

.leva-stran-section {
display: flex;
}

.leva-stran-nav {
flex: 1;
background-color: #466aed;
padding-bottom: 900px;
border-right: 6px solid white;
border-radius: 10px;
}

.leva-stran-nav-ul {
list-style-type: none;
text-align: center;
}

.leva-stran-nav-ul-li{
padding-bottom: 100px;
}

.leva-stran-article {
flex: 3;
padding: 10px;
```

```
}

button{
  padding: 40px;
  padding-left: 100px;
  padding-right: 100px;
  margin-left: -50px;
  background-color: white;
  color: #433831;
  font-weight: bold;
  border: none;
  border-radius: 10px;
  letter-spacing: 10 px;
  overflow: hidden;
  transition: 0.5s;
  cursor: pointer;
  font-size: 20px;
}

button:hover{
  background: #5d7ced;
  color: white;
  box-shadow: 0 0 5px #5d7ced,
             0 0 25px #5d7ced,
             0 0 50px #5d7ced,
             0 0 200px #5d7ced;
}

#panel, #flip {
  padding: 5px;
  text-align: center;
  background-color: #466aed;
  border: solid 1px #c3c3c3;
  border-radius: 10px;
}

#panel {
  padding: 50px;
  display: none;
  color:white;
  font-size: 20px;
}

.izpis{
  font-size: 50px;
}
```



```
text-align: center;
color: white;
background-color: linear-gradient(141deg, #4e8bed 10% , #ff0000 31%);
padding: 20px;
margin: -10px;

}

footer {
border-top: 6px solid white;
background-color: #3c4140;
padding: 40px;
color: rgb(255, 255, 255);
font-size: 200%;
font-family: 'Source Sans Pro', sans-serif;
}

.circle{
height: 100px;
width: 100px;
background:#466aed;
border: 10px solid #2b46a5;
border-radius: 50%;
margin-left: 1340px;
margin-top: -250px;
}

.gradient{
background: -webkit-radial-gradient(circle, #233c95, #466aed);
}

.Arduino{
margin-top: 38px;
margin-left: 15px;
font-size: 20px;
text-decoration: none;
display: inline-block;
text-align: center;
color:rgb(255, 255, 255);
}

.Arduino:hover{
color:#000000;
}

.gradient:hover{
background: -webkit-radial-gradient(circle, white, #2c4290);
```

```
}  
  
.circle:hover{  
  border: 10px solid #ffffff;  
}
```

Spletna stran vlaga_temperatura.php:

```
<?php  
  
$cleardb_url = parse_url(getenv("CLEARDB_DATABASE_URL"));  
$cleardb_server = $cleardb_url["host"];  
$cleardb_username = $cleardb_url["user"];  
$cleardb_password = $cleardb_url["pass"];  
$cleardb_db = substr($cleardb_url["path"],1);  
$active_group = 'default';  
$query_builder = TRUE;  
  
$conn = mysqli_connect($cleardb_server, $cleardb_username,  
$cleardb_password, $cleardb_db);  
  
mysqli_select_db($conn,$cleardb_db);  
$test=array();  
  
$res=mysqli_query($conn,"select temperatura,vlaga from  
temperatura_senzor");  
$count=0;  
while($row = mysqli_fetch_array($res))  
{  
  $test[$count] ["label"]=$count;  
  $test[$count] ["y"]=$row["temperatura"];  
  $count=$count+1;  
}  
  
?>  
  
<!DOCTYPE html>  
<html lang="en" style="margin-top: -20px;">  
  
<head>  
<link rel="stylesheet" href="vlaga_temperatura.css">  
<title>Arduino</title>  
<link rel = "icon" href =  
"https://static.thenounproject.com/png/34403-200.png"  
  type = "image/x-icon">
```

```
    <meta http-equiv="Content-Type" content="text/html";
charset="utf-8" />
<meta name="viewport" content="width=device-width, initial-scale=1">
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.4/jquery.min.js"></
script>

<script>
window.onload = function () {

var chart = new CanvasJS.Chart("chartContainer", {
  animationEnabled: true,
  exportEnabled: true,
  theme: "dark1", // "light1", "light2", "dark1", "dark2"
  title:{
    text: "Prikaz temperature"
  },
  axisY:{
    title: "Temperatura [°C]",
    includeZero: true
  },axisX:{
    title: "Število meritve",
    includeZero: true
  },
  data: [{
    type: "line", //change type to bar, line, area, pie, etc
    //indexLabel: "{y}", //Shows y value on all Data Points
    indexLabelFontColor: "#8e5939",
    indexLabelPlacement: "outside",
    dataPoints: <?php echo json_encode($test, JSON_NUMERIC_CHECK); ?>
  ]
});
chart.render();

}
</script>

</head>
<body>
<header class="glava" >
<div class="glavna-beseda">
<div class="home">
<a href="index.php" ></a>
```

```
</div>
<h1>Vlaga in temperatura</h1>
</div>
<div class="barva-gumb">
<a class="gumb_glava" href="vlaga_temperatura.php" style="background-
color: rgb(68, 64, 64);" >Vlaga in temperatura</a>
<a class="gumb_glava" href="kvaliteta_vode.php" >Kvaliteta vode</a>
<a class="gumb_glava" href="vlaga_zemlje.php" >Vlaga zemlje</a>
<a class="gumb_glava" href="cas.php">Čas</a>
<a class="gumb_glava" href="sonce.php" >Sonce</a>
</div>
</header>
<section class="leva-stran-section">
<nav class="leva-stran-nav">
<ul class="leva-stran-nav-ul">
<script>
function Graf() {
document.getElementById("demo").style.fontSize = "50px";
document.getElementById("demo").style.color = "white";
document.getElementById("demo").style.backgroundColor = "linear-
gradient(141deg, #17e6d85d 10% , #0fa399 65%)";
demo.innerHTML = "Graf";}

function Tabela() {
document.getElementById("demo").style.fontSize = "50px";
document.getElementById("demo").style.color = "white";
document.getElementById("demo").style.backgroundColor = "linear-
gradient(141deg, #17e6d85d 10% , #0fa399 65%)";
demo.innerHTML = "Tabela";
}
function Graf_Tabela() {
    document.getElementById("demo").style.fontSize = "50px";
    document.getElementById("demo").style.color = "white";
    document.getElementById("demo").style.backgroundColor = "linear-
gradient(141deg, #17e6d85d 10% , #0fa399 65%)";
    demo.innerHTML = "Graf in tabela";
}
</script>
<li class="leva-stran-nav-ul-li"><button class="gumb-levo-graf"
onclick="Graf(); funkcija_graf();">Graf</button></li>
<li class="leva-stran-nav-ul-li"><button class="gumb-levo-tabela"
onclick="Tabela();funkcija_tabela();">Tabela</button></li>
<li class="leva-stran-nav-ul-li"><button class="gumb-levo-graf-tabela"
onclick="Graf_Tabela();funkcija_tabela_graf();">Graf/Tabela</button></li>
</ul>
</nav>
```

```
<script>
$(document).ready(function(){
  $("#flip").click(function(){
    $("#panel").slideToggle("slow");
  });
});
</script>

<article class="leva-stran-article">

  <div id="flip">
<h1 class="izpis" id="demo">Izpis</h1>
</div >
<div id="panel">
Spodaj se izriše graf ali tabela.
</div>

  <div id="tabela__">
<?php
$sql = " SELECT vlaga,temperatura,cas FROM temperatura_senzor";
$result = mysqli_query($conn, $sql);

echo "<br>";
echo "
<style>
tr:hover {background-color: black;color:white}
</style>
";

echo "<table style=' border-collapse:collapse; width: 100%; margin-left:
auto;margin-right: auto; '>";
echo "<tr>";
echo "<th style='color:white; font-size: 25px;'>Vlaga [%]</th>";
echo "<th style='color:white; font-size: 25px;'>Temperatura [°C]</th>";
echo "<th style='color:white; font-size: 25px;'>Čas</th>";
echo "</tr>";
while ($row = mysqli_fetch_assoc($result)) {

echo "<tr style='padding: 8px; text-align: left; border-bottom: 1px
solid black;text-align: center;font-size: 25px; '>";
foreach ($row as $field => $value) {
echo "<td style='color:white;'>" . $value . "</td>";
}
echo "</tr>";
}
}
```

```
echo "</table>";

?>
</div>
<script>
function funkcija_tabela() {
    var x = document.getElementById("tabela__");
    if (x.style.display === "none") {
        x.style.display = "block";
        demo.innerHTML = "Tabela";
    } else {
        x.style.display = "none";
        demo.innerHTML = "Izberi";
    }

    var y = document.getElementById("graf__");
    var z = document.getElementById("tabela_graf");
    z.style.display = "none";
    y.style.display = "none";
}
</script>

<div style="height: 300px; overflow: auto" id="tabela_graf">
<?php
$sql = " SELECT vlaga,temperatura,cas FROM temperatura_senzor";
$result = mysqli_query($conn, $sql);

echo "<br>";
echo "
<style>
tr:hover {background-color: black;color:white}
</style>
";

echo "<table style=' border-collapse:collapse; width: 100%; margin-left:
auto;margin-right: auto; '>";
echo "<tr>";
echo "<th style='color:white; font-size: 25px;'>Vlaga [%]</th>";
echo "<th style='color:white; font-size: 25px;'>Temperatura [°C]</th>";
echo "<th style='color:white; font-size: 25px;'>Čas</th>";
echo "</tr>";
while ($row = mysqli_fetch_assoc($result)) {
```

```
echo "<tr style='padding: 8px; text-align: left; border-bottom: 1px
solid black;text-align: center;font-size: 25px; '>";
foreach ($row as $field => $value) {
echo "<td style='color:white;'" . $value . "</td>";
}
echo "</tr>";
}
echo "</table>";

?>
</div>

<div id="graf__">
<div id="chartContainer" style="height: 370px; width: 100%; margin-
top:20px;"></div>
<script src="https://cdn.canvasjs.com/canvasjs.min.js"></script>

</article>
</section>
</div>
<script>
function funkcija_tabela_graf() {
    var x = document.getElementById("tabela_graf");
    var z = document.getElementById("graf__");
    if (x.style.display === "none" && z.style.display === "none") {
        x.style.display = "block";
        z.style.display = "block";
        demo.innerHTML = "Tabela/Graf";
    }
    if(x.style.display === "block" && z.style.display === "none"){
        x.style.display = "none";
        z.style.display = "none";
        demo.innerHTML = "Izberi";
    }
    if(x.style.display === "none" && z.style.display === "block"){
        x.style.display = "block";
        z.style.display = "block";
        demo.innerHTML = "Tabela/Graf";
    }
}

var y = document.getElementById("tabela__");
```

```
    y.style.display = "none";
}
</script>

<script>
function funkcija_graf() {
    var x = document.getElementById("graf__");
    if (x.style.display === "none") {
        x.style.display = "block";
        demo.innerHTML = "Graf";
    } else {
        x.style.display = "none";
        demo.innerHTML = "Izberi";
    }

    var y = document.getElementById("tabela__");
    y.style.display = "none";
    var z = document.getElementById("tabela_graf");
    z.style.display = "none";
}
</script>

<footer>

<p>Oseba: Tomaž</p>
<p><a href="mailto:diplomski.del@gmail.com">gmail</a></p>
</footer>

</script>
<div>
<div class="circle gradient">
<a href="https://www.arduino.cc/" class="Arduino">Arduino</a>
</div>
</div>
</body>
</html>
```

Spletna stran vlaga_zemlje.css:

```
body{
    background: linear-gradient(141deg, #8e5939 30%, #b76b3b 51%, #8e5939 75%);
}
```



```
margin: 0px;
}
.gumb_home{

width:80px;
height:80px;
}
.home{

margin-top: 10px;
position: absolute;
left: -10px;
width: 100px;
height: 120px;

}
#tabela_graf{
display: none;
}
.glava {

background-color: linear-gradient(141deg, #8e5939 10% , #b76b3b 65%);
padding-bottom: 2px;
padding-top: 2px;
text-align: center;
font-size: 45px;
color: rgb(255, 255, 255);
border-bottom: 6px solid white;

}

.glavna-beseda::before {
transform: scaleX(0);
transform-origin: bottom right;
}

.glavna-beseda:hover::before {
transform: scaleX(1);
transform-origin: bottom left;
}

.glavna-beseda::before {
content: " ";
display: block;
position: absolute;
top: 0; right: 0; bottom: 0; left: 0;
```

```
inset: 0 0 0 0;
background: #964614;
z-index: -1;
transition: transform 4s ease;
}

.glavna-beseda {
  position: relative;
  padding-top: 10px;
  padding-bottom: 10px;
}

.barva-gumb{
  border-top: 6px solid white;
  background-color: linear-gradient(141deg, #8e5939 10% , #b46a3d 65%);
  padding: 30px;
  margin-bottom: -10px;
}

.gumb_glava{
  border-radius: 10px;
  border: 1px solid white;
  padding: 10px;

  background-color:#8e5939;
  color:white;
  text-decoration: none;
  display: inline-block;
  text-align: center;
}

.gumb_glava:hover{
  color:white;
  background-color: rgb(68, 64, 64);
  text-align: center;
}

.leva-stran-section {
  display: flex;
}

.leva-stran-nav {
  flex: 1;
  background-color: #8e5939;
  padding-bottom: 900px;
  border-right: 6px solid white;
  border-radius: 10px;
```

```
}

.leva-stran-nav-ul {
  list-style-type: none;
  text-align: center;
}

.leva-stran-nav-ul-li{
  padding-bottom: 100px;
}

.leva-stran-article {
  flex: 3;
  padding: 10px;
}

button{
  padding: 40px;
  padding-left: 100px;
  padding-right: 100px;
  margin-left: -50px;
  background-color: white;
  color: #433831;
  font-weight: bold;
  border: none;
  border-radius: 10px;
  letter-spacing: 10 px;
  overflow: hidden;
  transition: 0.5s;
  cursor: pointer;
  font-size: 20px;
}

button:hover{
  background: #ce885c;
  color: white;
  box-shadow: 0 0 5px #ce885c,
             0 0 25px #ce885c,
             0 0 50px #ce885c,
             0 0 200px #ce885c;
}

#panel, #flip {
  padding: 5px;
  text-align: center;
}
```

```
background-color: #8e5939;
border: solid 1px #c3c3c3;
border-radius: 10px;
}

#panel {
padding: 50px;
display: none;
color:white;
font-size: 20px;
}

.izpis{
font-size: 50px;
text-align: center;
color: white;
background-color: linear-gradient(141deg, #4a3223 10% , #8e5939 65%);
padding: 20px;
margin: -10px;
}

footer {
border-top: 6px solid white;
background-color: #3c4140;
padding: 40px;
color: rgb(255, 255, 255);
font-size: 200%;
font-family: 'Source Sans Pro', sans-serif;
}

.circle{
height: 100px;
width: 100px;
background: #8e5939;
border: 10px solid #4a3223;
border-radius: 50%;
margin-left: 1340px;
margin-top: -250px;
}

.gradient{
background: -webkit-radial-gradient(circle, #4a3223, #8e5939);
}
```

```
.Arduino{
  margin-top: 38px;
  margin-left: 15px;
  font-size: 20px;
  text-decoration: none;
  display: inline-block;
  text-align: center;
  color:rgb(255, 255, 255);
}
.Arduino:hover{
  color:#000000;
}

.gradient:hover{
  background: -webkit-radial-gradient(circle, white, #4a3223);
}

.circle:hover{
  border: 10px solid #ffffff;
}
```

Spletna stran vlaga_zemlja.php:

```
<?php

$cleardb_url = parse_url(getenv("CLEARDB_DATABASE_URL"));
$cleardb_server = $cleardb_url["host"];
$cleardb_username = $cleardb_url["user"];
$cleardb_password = $cleardb_url["pass"];
$cleardb_db = substr($cleardb_url["path"],1);
$active_group = 'default';
$query_builder = TRUE;

$conn = mysqli_connect($cleardb_server, $cleardb_username,
$cleardb_password, $cleardb_db);

mysqli_select_db($conn,$cleardb_db);
$test=array();

$res=mysqli_query($conn,"select vlagaZemlje from temperatura_senzor");
$count=0;
while($row = mysqli_fetch_array($res))
{
  $test[$count] ["label"]=$count;
  $test[$count] ["y"]=$row["vlagaZemlje"];
  $count=$count+1;
}
```

```
?>

<!DOCTYPE html>
<html lang="en" style="margin-top: -20px;">

<head>
<link rel="stylesheet" href="vlaga_zemlje.css">
<link rel = "icon" href =
"https://static.thenounproject.com/png/34403-200.png"
    type = "image/x-icon">
<title>Arduino</title>
<meta http-equiv="Content-Type" content="text/html"; charset="utf-8" />
<meta name="viewport" content="width=device-width, initial-scale=1">
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.4/jquery.min.js"></
script>

<script>
window.onload = function () {
var chart = new CanvasJS.Chart("chartContainer", {
    animationEnabled: true,
    exportEnabled: true,
    theme: "dark2", // "light1", "light2", "dark1", "dark2"
    title:{
        text: "Prikaz vlage zemlje"
    },
    axisY:{
        title: "Vlaga zemlje",
        includeZero: true
    },axisX:{
        title: "Število meritve",
        includeZero: true
    },
    data: [{
        type: "line", //bar, line, area, pie, etc
        //indexLabel: "{y}",
        indexLabelFontColor: "#8e5939",
        indexLabelPlacement: "outside",
        dataPoints: <?php echo json_encode($test, JSON_NUMERIC_CHECK); ?>
    }]
});
chart.render();
}
```

```
</script>

</head>
<body>
<header class="glava" >
<div class="glavna-beseda">
<div class="home">
<a href="index.php" ></a>
</div>
<h1>Vlaga zemlje</h1>
</div>

<div class="barva-gumb">
<a class="gumb_glava" href="vlaga_temperatura.php" >Vlaga in
temperatura</a>
<a class="gumb_glava" href="kvaliteta_vode.php" >Kvaliteta vode</a>
<a class="gumb_glava" href="vlaga_zemlje.php" style="background-color:
rgb(68, 64, 64);" >Vlaga zemlje</a>
<a class="gumb_glava" href="cas.php">Čas</a>
<a class="gumb_glava" href="sonce.php" >Sonce</a>
</div>
</header>
<section class="leva-stran-section">
<nav class="leva-stran-nav">
<ul class="leva-stran-nav-ul">
<script>
function Graf() {
document.getElementById("demo").style.fontSize = "50px";
document.getElementById("demo").style.color = "white";
document.getElementById("demo").style.backgroundColor = "linear-
gradient(141deg, #17e6d85d 10% , #0fa399 65%)";
demo.innerHTML = "Graf";}

function Tabela() {
document.getElementById("demo").style.fontSize = "50px";
document.getElementById("demo").style.color = "white";
document.getElementById("demo").style.backgroundColor = "linear-
gradient(141deg, #17e6d85d 10% , #0fa399 65%)";
demo.innerHTML = "Tabela";
}
function Graf_Tabela() {
document.getElementById("demo").style.fontSize = "50px";
document.getElementById("demo").style.color = "white";
```

```
document.getElementById("demo").style.backgroundColor = "linear-
gradient(141deg, #17e6d85d 10% , #0fa399 65%);
demo.innerHTML = "Graf in tabela";
}
</script>
<li class="leva-stran-nav-ul-li"><button class="gumb-levo-graf"
onclick="Graf(); funkcija_graf();">Graf</button></li>
<li class="leva-stran-nav-ul-li"><button class="gumb-levo-tabela"
onclick="Tabela();funkcija_tabela();">Tabela</button></li>
<li class="leva-stran-nav-ul-li"><button class="gumb-levo-graf-tabela"
onclick="Graf_Tabela();funkcija_tabela_graf();">Graf/Tabela</button></li>
</ul>
</nav>

<script>
$(document).ready(function(){
  $("#flip").click(function(){
    $("#panel").slideToggle("slow");
  });
});
</script>

<article class="leva-stran-article">

  <div id="flip">
<h1 class="izpis" id="demo">Izpis</h1>
</div >
<div id="panel">
Spodaj se izriše graf ali tabela.
</div>

<div id="tabela__">
<?php
$sql = " SELECT vlagaZemlje,cas FROM temperatura_senzor";
$result = mysqli_query($conn, $sql);

echo "<br>";
echo "
<style>
tr:hover {background-color: black;color:white}
</style>
";

echo "<table style=' border-collapse:collapse; width: 100%; margin-left:
auto;margin-right: auto; ' start='1'>";
```



```
echo "<tr>";
echo "<th style='color:white; font-size: 25px;'>Vlaga zemlje</th>";
echo "<th style='color:white; font-size: 25px;'>Čas</th>";
echo "</tr>";
while ($row = mysqli_fetch_assoc($result)) {

echo "<tr style='padding: 8px; text-align: left; border-bottom: 1px
solid black;text-align: center;font-size: 25px; ' start='1'>";
foreach ($row as $field => $value) {
echo "<td style='color:white;'>" . $value . "</td>";
}
echo "</tr>";
}
echo "</table>";

?>
</div>
<script>
function funkcija_tabela() {
    var x = document.getElementById("tabela__");
    if (x.style.display === "none") {
        x.style.display = "block";
        demo.innerHTML = "Tabela";
    } else {
        x.style.display = "none";
        demo.innerHTML = "Izberi";
    }

    var y = document.getElementById("graf__");
    var z = document.getElementById("tabela_graf");
    z.style.display = "none";
    y.style.display = "none";

}

</script>

<div style="height: 300px; overflow: auto" id="tabela_graf">
<?php
$sql = " SELECT vlagaZemlje,cas FROM temperatura_senzor";
$result = mysqli_query($conn, $sql);

echo "<br>";
echo "
<style>
```

```
tr:hover {background-color: black;color:white}
</style>
";

echo "<table style=' border-collapse:collapse; width: 100%; margin-left:
auto;margin-right: auto; ' start='1'>";
echo "<tr>";
echo "<th style='color:white; font-size: 25px;'>Vlaga zemlje</th>";
echo "<th style='color:white; font-size: 25px;'>Čas</th>";
echo "</tr>";
while ($row = mysqli_fetch_assoc($result)) {

echo "<tr style='padding: 8px; text-align: left; border-bottom: 1px
solid black;text-align: center;font-size: 25px;' start='1'>";
foreach ($row as $field => $value) {
echo "<td style='color:white;'>" . $value . "</td>";
}
echo "</tr>";
}
echo "</table>";

?>
</div>

<div id="graf__">
<div id="chartContainer" style="height: 370px; width: 100%; margin-
top:20px;"></div>
<script src="https://cdn.canvasjs.com/canvasjs.min.js"></script>

</article>
</section>
</div>
<script>
function funkcija_tabela_graf() {
    var x = document.getElementById("tabela_graf");
    var z = document.getElementById("graf__");
    if (x.style.display === "none" && z.style.display === "none") {
        x.style.display = "block";
        z.style.display = "block";
        demo.innerHTML = "Tabela/Graf";
    }
}
```

```
if(x.style.display === "block" && z.style.display === "none"){
  x.style.display = "none";
  z.style.display = "none";
  demo.innerHTML = "Izberi";
}
if(x.style.display === "none" && z.style.display === "block"){
  x.style.display = "block";
  z.style.display = "block";
  demo.innerHTML = "Tabela/Graf";
}

var y = document.getElementById("tabela__");
y.style.display = "none";
}
</script>

<script>
function funkcija_graf() {
  var x = document.getElementById("graf__");
  if (x.style.display === "none") {
    x.style.display = "block";
    demo.innerHTML = "Graf";
  } else {
    x.style.display = "none";
    demo.innerHTML = "Izberi";
  }
}

var y = document.getElementById("tabela__");
y.style.display = "none";
var z = document.getElementById("tabela_graf");
z.style.display = "none";
}
</script>

<footer>

<p>Oseba: Tomaž</p>
<p><a href="mailto:diplomski.del@gmail.com">gmail</a></p>
</footer>

</script>
<div>
<div class="circle gradient">
```

```

<a href="https://www.arduino.cc/" class="Arduino">Arduino</a>
</div>
</div>
</body>

</html>

```

Arduino vse.ino :

```

#include "DHT.h"
#include <SPI.h>
#include <Ethernet.h>
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
#define DHTPIN 2
#define DHTTYPE DHT11
#define sensorPower 7
#define sensorPin A0
#define TdsSensorPin A1
#define VREF 5.0
#define SCOUNT 30
int analogBuffer[SCOUNT];
int analogBufferTemp[SCOUNT];
int analogBufferIndex = 0;
int copyIndex = 0;
DHT dht(DHTPIN,DHTTYPE);
float averageVoltage = 0;
float tdsValue = 0;
float temperature = 16;
float humidityData;
float temperatureData;
float vlagaZemlje;
const int ldrPin = A3;
int ldrValue = 0;

int getMedianNum(int bArray[], int iFilterLen){
    int bTab[iFilterLen];
    for (byte i = 0; i<iFilterLen; i++)
        bTab[i] = bArray[i];
    int i, j, bTemp;
    for (j = 0; j < iFilterLen - 1; j++) {
        for (i = 0; i < iFilterLen - j - 1; i++) {
            if (bTab[i] > bTab[i + 1]) {
                bTemp = bTab[i];
                bTab[i] = bTab[i + 1];
                bTab[i + 1] = bTemp;
            }
        }
    }
}

```

```

    }
    if ((iFilterLen & 1) > 0){
        bTemp = bTab[(iFilterLen - 1) / 2];
    }
    else {
        bTemp = (bTab[iFilterLen / 2] + bTab[iFilterLen / 2 - 1]) / 2;
    }
    return bTemp;
}
char server[] = "192.168.1.104";
IPAddress ip(192,168,1,177);
EthernetClient client;
void setup() {
    Serial.begin(9600);
    dht.begin();
    pinMode(sensorPower, OUTPUT);
    digitalWrite(sensorPower, LOW);
    pinMode(TdsSensorPin,INPUT);
    if (Ethernet.begin(mac) == 0) {
        Serial.println("Failed to configure Ethernet using DHCP");
        Ethernet.begin(mac, ip);
    }
    delay(100);
}
void loop(){
    static unsigned long analogSampleTimepoint = millis();
    if(millis()-analogSampleTimepoint > 400){
        analogSampleTimepoint = millis();
        analogBuffer[analogBufferIndex] = analogRead(TdsSensorPin);
        analogBufferIndex++;
        if(analogBufferIndex == SCOUNT){
            analogBufferIndex = 0;
        }
    }
}

static unsigned long printTimepoint = millis();
if(millis()-printTimepoint > 800){
    printTimepoint = millis();
    for(copyIndex=0; copyIndex<SCOUNT; copyIndex++){
        analogBufferTemp[copyIndex] = analogBuffer[copyIndex];
        averageVoltage = getMedianNum(analogBufferTemp,SCOUNT) *
(float)VREF / 1024.0;
        float compensationCoefficient = 1.0+0.02*(temperature-25.0);
        float compensationVoltage=averageVoltage/compensationCoefficient;
        tdsValue=((133.42*compensationVoltage*compensationVoltage*compensat
ionVoltage - 255.86*compensationVoltage*compensationVoltage +
857.39*compensationVoltage)*0.5)-4981.11;

```

```

    Serial.print(tdsValue,0);
    if(tdsValue<0){
        tdsValue=0;
    }else{
        tdsValue=((133.42*compensationVoltage*compensationVoltage*compensationVoltage - 255.86*compensationVoltage*compensationVoltage + 857.39*compensationVoltage)*0.5)-4981.11;
    }
}
}
}
delay(300);
humidityData = dht.readHumidity();
temperatureData = dht.readTemperature();
vlagaZemlje=readSensor();
ldrValue = analogRead(ldrPin);
Sending_To_phpmyadmindatabase();
delay(3000);
}
int readSensor() {
    digitalWrite(sensorPower, HIGH);
    delay(3000);
    int val = analogRead(sensorPin);
    digitalWrite(sensorPower, LOW);
    return val;
}
void Sending_To_phpmyadmindatabase()
{
    if (client.connect(server, 80)) {
        Serial.println("connected");
        Serial.print("GET /php/vse.php?vlaga=");
        client.print("GET /php/vse.php?vlaga=");
        Serial.println(humidityData);
        client.print(humidityData);
        client.print("&temperatura=");
        Serial.println("&temperatura=");
        client.print(temperatureData);
        Serial.println(temperatureData);
        client.print("&vlagaZemlje=");
        Serial.println("&vlagaZemlje=");
        client.print(vlagaZemlje);
        Serial.println(vlagaZemlje);
        client.print("&voda=");
        Serial.println("&voda=");
        client.print(tdsValue);
        Serial.println(tdsValue);
        client.print("&sonce=");
    }
}

```

```
    Serial.println("&sonce=");
    client.print(ldrValue);
    Serial.println(ldrValue);
    client.print(" ");
    client.print("HTTP/1.1");
    client.println();
    client.println("Host: 192.168.1.104");
    client.println("Connection: close");
    client.println();
} else {

    Serial.println("Ni povezano");
}
}
```

Arduino MOTOR-SENZOR.ino:

```
#define POWER_PIN 7
#define SIGNAL_PIN A5

int motor2pin1 = 4;
int motor2pin2 = 5;
int value = 0;
const int buttonPin1 = 12;
const int buttonPin2 = 11;
int buttonState1 = 0;
int buttonState2 = 0;

void setup() {
    pinMode(buttonPin1, INPUT);
    pinMode(buttonPin2, INPUT);
    pinMode(motor2pin1, OUTPUT);
    pinMode(motor2pin2, OUTPUT);
    pinMode(10, OUTPUT);
    Serial.begin(9600);
    pinMode(POWER_PIN, OUTPUT);
    digitalWrite(POWER_PIN, LOW);
}

void loop() {
    buttonState1 = digitalRead(buttonPin1);
    buttonState2 = digitalRead(buttonPin2);
    digitalWrite(POWER_PIN, HIGH);
    delay(10);
    value = analogRead(SIGNAL_PIN);
    digitalWrite(POWER_PIN, LOW);
    Serial.println(value);
}
```

```

if(value>100 && buttonState1==LOW ){
    analogWrite(10, 200);
    digitalWrite(motor2pin1, LOW);
    digitalWrite(motor2pin2, HIGH);
    delay(300);
}
if(value>100 && buttonState1==HIGH ){
    analogWrite(10, 200);
    digitalWrite(motor2pin1, LOW);
    digitalWrite(motor2pin2, LOW);
    delay(300);
}

if(value<100 && buttonState2==HIGH){
    analogWrite(10, 200);
    digitalWrite(motor2pin1, LOW);
    digitalWrite(motor2pin2, LOW);
    delay(300);
}
if(value<100 && buttonState2==LOW){
    analogWrite(10, 200);
    digitalWrite(motor2pin1, HIGH);
    digitalWrite(motor2pin2, LOW);
    delay(300);
}
}
}

```

Povezava med Arduino in podatkovno bazo:

```

<?php
class temperatura_senzor{
    public $link='';
    function __construct($temperatura, $vlaga, $vlagaZemlje, $voda,
$sonce){
        $this->connect();
        $this->storeInDB($temperatura, $vlaga, $vlagaZemlje, $voda,
$sonce);
    }

    function connect(){
        $this->link = mysqli_connect('eu-cdbr-west-
03.cleardb.net','bcf303ea2042d4','e979bb12') or die('Nemore povezati
PB');
        mysqli_select_db($this->link,'heroku_e17f78c5d8ff122') or
die('Nemore izbrati PB');
    }
}

```



```
function storeInDB($temperatura, $vlaga, $vlagaZemlje, $voda,
$sonce){
    $query = "insert into temperatura_senzor set vlaga='". $vlaga."',
temperatura='". $temperatura."', vlagaZemlje='". $vlagaZemlje."',
voda='". $voda."', sonce='". $sonce.'";
    $result = mysqli_query($this->link,$query) or die('Napaka:
' . $query);
}

}
if($_GET['temperatura'] != '' and $_GET['vlaga'] != '' and
$_GET['vlagaZemlje'] != '' and $_GET['voda'] != '' and $_GET['sonce'] !=
''){
    $temperatura_senzor=new
temperatura_senzor($_GET['temperatura'], $_GET['vlaga'], $_GET['vlagaZemlje
'], $_GET['voda'], $_GET['sonce']);
}
?>
```

Povezava z oddaljeno podatkovno bazo:

```
$i = 0;
```

```
$i++;
```

```
$cfg["Servers"][$i]["host"] = "eu-cdbbr-west-03.cleardb.net";
```

```
$cfg["Servers"][$i]["user"] = "bcf303ea2042d4";
```

```
$cfg["Servers"][$i]["password"] = "e979bb12";
```

```
$cfg["Servers"][$i]["auth_type"] = "config";
```