Garden watering and monitoring(GardenWAM)

Tehnička dokumentacija

Author: Tomislav Mikić

Mentor: Mladen Savić

Sadržaj

[1. Uvod 4](#_Toc31984595)

[2. Hardverski dio 4](#_Toc31984596)

[2.1. Modul za automatizaciju 5](#_Toc31984597)

[2.1.1. Opis i funkcije 5](#_Toc31984598)

[2.1.2. Dizajn shielda 6](#_Toc31984599)

[2.1.3. Senzori 7](#_Toc31984600)

[2.1.4. Aktuatori 9](#_Toc31984601)

[2.2. Poslužitelj 11](#_Toc31984602)

[2.2.1. Uvod 11](#_Toc31984603)

[2.2.2. Specifikacije 11](#_Toc31984604)

[3. Softverski dio 11](#_Toc31984605)

[3.1. Softverski dio poslužitelja 11](#_Toc31984606)

[3.1.1. Tehnologije 11](#_Toc31984607)

[3.2. Baza podataka 12](#_Toc31984608)

[3.2.1. Tehnologije 12](#_Toc31984609)

[3.2.2. Struktura 12](#_Toc31984610)

[3.3. API(Application programming interface) 12](#_Toc31984611)

[3.3.1. Tehnologije 12](#_Toc31984612)

[3.3.2. Struktura 13](#_Toc31984613)

[3.3.3. Config 13](#_Toc31984614)

[3.3.4. MySQL dio 13](#_Toc31984615)

[3.3.5. Website 13](#_Toc31984616)

[3.4. Program za očitanje senzora i upravljanja aktuatorima 13](#_Toc31984617)

[3.4.1. Tehnologije 14](#_Toc31984618)

[3.4.2. Struktura 14](#_Toc31984619)

[3.5. Web stranica 15](#_Toc31984620)

[3.5.1. Tehnologije 15](#_Toc31984621)

[3.5.2. Struktura 15](#_Toc31984622)

[3.5.3. Početna web stranica 15](#_Toc31984623)

[3.5.4. Web stranica s biljkom 16](#_Toc31984624)

[3.5.5. Moduli 16](#_Toc31984625)

[3.5.6. Head modul 16](#_Toc31984626)

[3.5.7. NavBar modul 16](#_Toc31984627)

[3.5.8. Graph modul 17](#_Toc31984628)

[3.5.9. PlantRemover modul 17](#_Toc31984629)

[3.5.10. PlantSummary modul 18](#_Toc31984630)

[3.5.11. Settings modul 18](#_Toc31984631)

[4. Zaključak 19](#_Toc31984632)

[5. Literatura 19](#_Toc31984633)

Popis slika:

[Slika 1 Raspberry Pi 0w 4](#_Toc31984661)

[Slika 2 Custom shield 5](#_Toc31984662)

[Slika 3 Shema shielda 6](#_Toc31984663)

[Slika 4 Štampana pločica gornji dio 7](#_Toc31984664)

[Slika 5 Štampana pločica donji dio 7](#_Toc31984665)

[Slika 6 Senzor temperature i vlage zraka 8](#_Toc31984666)

[Slika 7 LM355 8](#_Toc31984667)

[Slika 8 Kapacitivni mjerač vlage 9](#_Toc31984668)

[Slika 9 Tranzistor kao sklopka 9](#_Toc31984669)

[Slika 10 Pumpa za vodu 10](#_Toc31984670)

[Slika 11 shema spajanja 10](file:///C:\Users\Tomislav\Downloads\Dokumentacija_final_noimages.docx#_Toc31984671)

[Slika 12 Raspberry GPIO 11](#_Toc31984672)

[Slika 14 Početna web stranica 15](#_Toc31984673)

[Slika 15 Web stranica s biljkom 16](#_Toc31984674)

[Slika 16 Bar za navigaciju 16](#_Toc31984675)

[Slika 17 Grafički modul 17](#_Toc31984676)

[Slika 18 Modul za micanje biljaka 17](#_Toc31984677)

[Slika 19 Općenito o biljkama 18](#_Toc31984678)

[Slika 20 Opcije 18](#_Toc31984679)

# Uvod

GardenWAM je projekt koji služi za automatizaciju zalijevanja vrta ili kućnih biljaka. Projekt se dijeli na dva djela hardverski i softverski. Uređaj omogućava automatsko zalijevanje biljaka ovisno o vlazi tla. Također omogućava i praćenje temperature i vlažnosti tla, kao i temperature i vlažnosti zraka.

Hardverski dio se sastoji od posužitelja i modula za automatizaciju. Poslužitelj je glavni uređaj. Na njemu se hosta web stranica i sql baza podataka. Dok je modul za automatizaciju dio na kojeg se spajaju svi senzori i aktuatori.

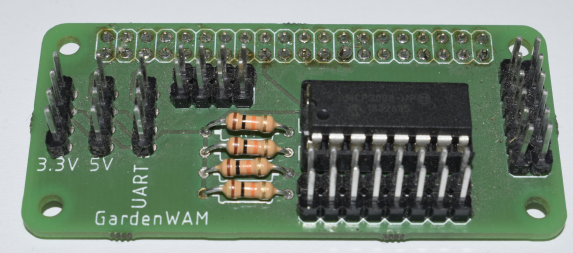
Softverski dio se sastoji od baze podataka, API-a za komunikaciju između dijelova softvera, programa za očitanje senzora i upravljanja aktuatorima i web stranice.

# Hardverski dio

Baza hardverskog dijela je Raspberry pi-a 0w(wireless) i custom shielda. Raspberry pi 0w obavlja funkciju poslužitelja, dok je custom shield modul za automatizaciju.



Slika 1 Raspberry Pi 0w



Slika 2 Custom shield

## Modul za automatizaciju

### Opis i funkcije

Modul za automatizaciju u trenutačnoj verziji 1 koristi se samo kao breakout za sve pinove i za rad IC-a MCP3008 koji sluzi kao ADC(Analogno digitalni pretvarač) njegova funkcija je da sve analogne vrijednosti senzora pretvori u digitalne kako bi ih software mogao obraditi. Na lijevoj strani nalaze se pinovi za I2C(inter-intergrated Circuit) koji radi na naponu od 3.3v (oznaka 3.3V), zatim pinovi I2C(inter-intergrated Circuit) koji radi na naponu od 5v (oznaka 5V), te pinovi UART-a (Universal asynchronous reciver/transmitter) koji služi za serijsku komunikaciju sa drugim uređajima, na vrhu 6x2 pinovi koji služe kao pinout-ovi za 4 GPIO pina(gornji pinovi) i masu za svaki pin na njih se spajaju aktuatori. U sredini je MPC3008 koji ima 4 analogna ulaza sa pullup-om na 3.3v a ostala 4 pina su floating. On komunicira s Raspberry-om pomoću SPI(Serial Peripheral Interface). Skroz desno imamo 6x2 pinova na kojima se spaja na desnoj strani 3.3v a na lijevoj masa.

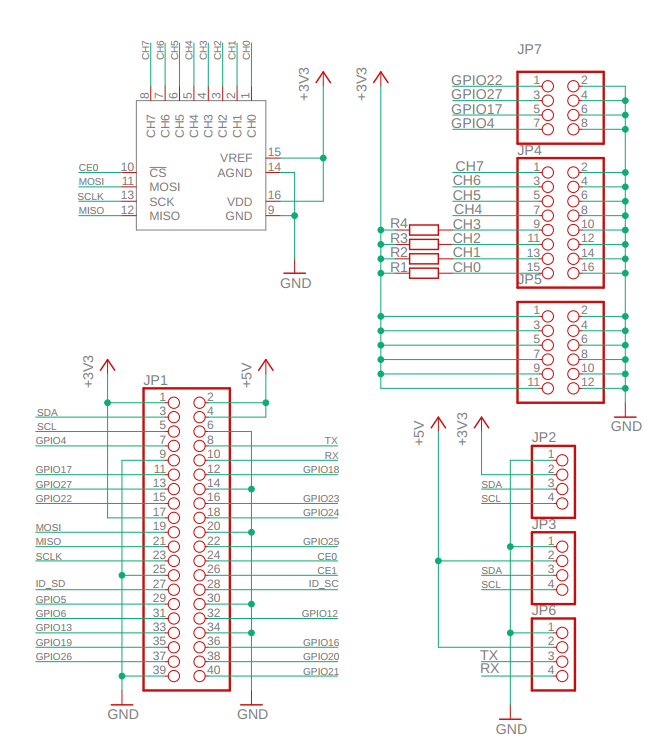
#### Protokoli

1. I2C je protokol za serijsku komunikaciju između više uređaja. Ima 2 pina SCL(Serial Clock Line) i SDA(Serial Data Line). Protokol je sinkroni što znači da ovisi o taktu, multi-master/multi-slave ovo omogućuje da komunicira vise od jednog uređaja pomoću adresa, s time da master-i su oni dobivaju podatke da ih obrade za korisnika, a slave-ovi daju ili spremaju podatke.
2. SPI je protokol isto za serijsku komunikaciju između vise uređaja. Ima 4 pina SCLK(Serial Clock), MOSI(Master Output Slave Input), MISO(Master Input Slave Output), CE(Chip Enable) ili CS(Chip Select). Za razliku od I2C-a ovaj protokol može raditi u full-duplex modu što znači da slave i master mogu komunicirati u isto vrijeme, no u SPI-ju imamo samo jedan master a više slave-ova. Isto je sinkroni protokol. Mogućnost biranja slave-ova radi se preko CE tako da CE stavimo u „1“ s slave-om s kojim želimo komunicirat.
3. UART je protokol za serijsku komunikaciju. Za razliku od SPI-a i I2C-a ovaj protokol je asinkroni tj protokol nema nikakvi takt. Ima 2 pina Rx(Recive) i Tx(Transmit). Sluzi za komunikaciju između dva uređaja najčešće između dva kompjutera/mikro upravljača

### Dizajn shielda

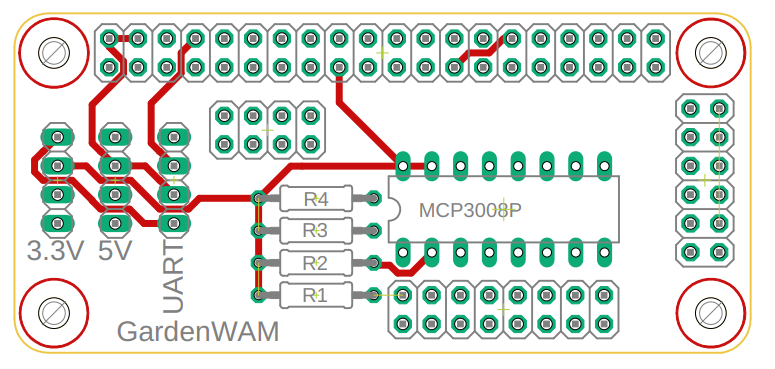
Shield je napravljen u programu Eagle.

#### Shema



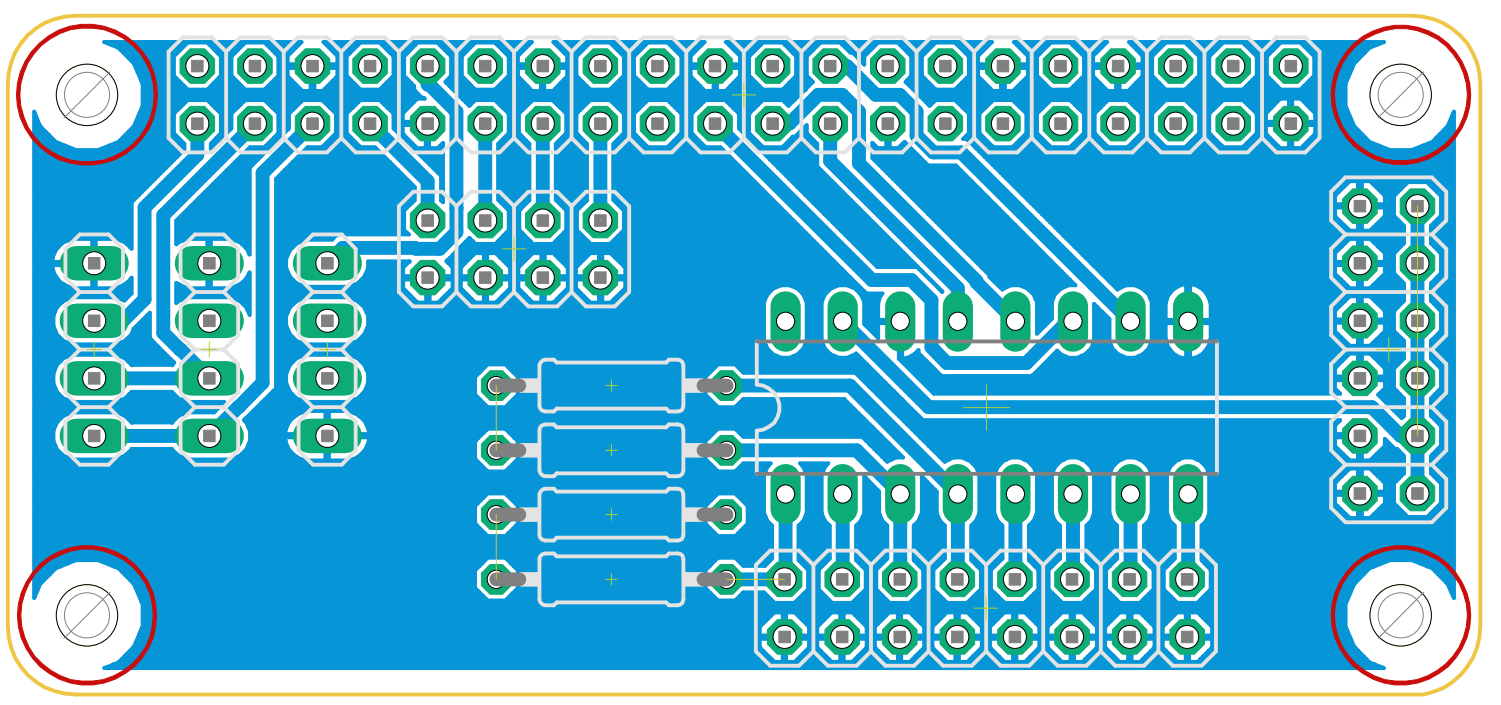
Slika 3 Shema shielda

#### PCB top



Slika 4 Štampana pločica gornji dio

#### PCB Bottom



Slika 5 Štampana pločica donji dio

### Senzori

Osnovni senzori koji se koriste za vrt trebali bi biti senzor vlage i temperature tla i zraka, senzor svjetlosti nije potreban za kućne biljke). Za temperaturu i vlagu zraka koristio sam SHT21. Temperatura tla mjerim pomoću LM335z, a vlaga tla se mjeri pomoću kapacitivnog senzora za vlagu tla.

#### SHT21

SHT21 je senzor za temperaturu i vlagu zraka. Za komunikaciju koristi I2C protokol. Ima 4 pina +Vcc napajanje, GND masa, SCL i SDA pinovi za I2C protokol. Za njega sam morao programirati „library“ da se mogu očitavati podatci.

|  |  |  |
| --- | --- | --- |
| Vrijednost | Temperatura | Relativna vlaga |
| Raspon mjerenja | -40°C - 125°C | 0-100% RH |
| Preciznost | 0.01°C u 14 bitnom modu | 0.04% RH |
| Tolerancija | +- 0.3°C | +- 2% |

Tablica 1



Slika 6 Senzor temperature i vlage zraka

#### LM355

LM355 je senzor za temperaturu koji radi na principu zener diode koja ima proboj napona tako da svakih 10mV odgovara 1°K pa pomoću formule dobijamo vrijednost temperature u °C s time da je Uz napon na zener diodi izražen u voltima(V). Kako trebam mjeriti temperaturu tla ovaj senzor sam stavio u termo bužir kako bi ga izolirao od vode. Ima 3 pina 2 su izvod zener diode a treći je za kalibraciju senzora.

. 

Slika 7 LM355

#### Kapacitivni seznor vlage

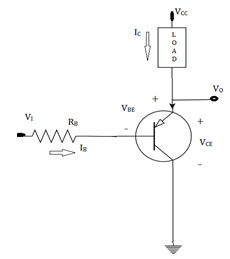
Kapacitivni seznor vlage radi na principu da zemlja je izolator a senzor se sastoji od 2 terminala i to čini kapacitet, a kako zalijevamo biljku mijenjamo svojstva tla s vise ili manje vode a voda ima u sebi ion-e koji rade zemlju lošijim izolatorom i tako kapacitet pada. Obzirom kako se kapacitet mijenja tako se mijenja i napon. Ima 3 pina Vcc za napajanje GND za masu i AOut za analogni izlaz seznora.



Slika 8 Kapacitivni mjerač vlage

### Aktuatori

Osnovni aktuatori za vrt je ventil, no kako sam trebao napraviti projekt portabilnim odlučio sam se koristiti mini pumpu i bocu vode kao izvor za zalijevanje, s time da se u trenutačnoj verziji shielda trebao koristiti relej za upravljanje pumpom. To sam izbjegao tako da sam ugradio u žicu tranzistor kao sklopku koja upravlja uređajima koji uzimaju više struje.



Slika 9 Tranzistor kao sklopka

#### Mini pumpa

Mini pumpa je mala pumpa koja može raditi na 3.3v i zato je korištena u projektu. 

Slika 10 Pumpa za vodu

Shema spajanja

Slika 11 shema spajanja

Raspberry Pi 0w

MCP3008

GPIO

SHT21

Kapacitivni senzor vlage tla

LM335z

I2C

Pumpa

GPIO 4

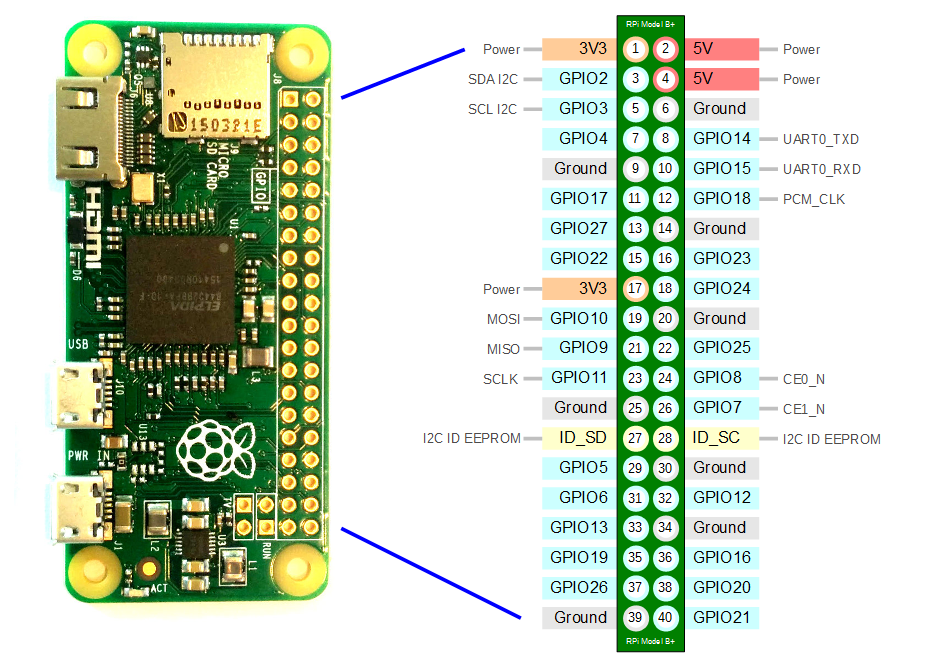
CH 7

CH 8

## Poslužitelj

### Uvod

Baza poslužitelja je Raspberry pi 0w(wireless).Raspberry pi radi Britanska kompanija Raspberry pi Foundation za promociju osnova informatike. Raspberry pi 0w je u familiji „Zero“ koja se sastoji još od Raspberry pi 0-a koji je po svemu isti kao 0w no nema wifi i bluetooth. Izasao je 2017 godine. Svi Raspberry-ji imaju GPIO port ili General purpose input output.



Slika 12 Raspberry GPIO

### Specifikacije

Raspberry pi 0w ima Broadcom BCM2835 SoC(System on chip) koji u sebi ima jednojezgreni procesor od 1GHz i Boradcom VideoCore IV GPU, 512Mb rama koji dijeli sa GPU-om, ima mreznu karticu sa 802.11 b/g/n WLAN-om i 4.1 BLE. Za napajanje koristi micro usb port, a kao memoriju koristi MicroSD karticu, 16Gb u mom slucaju.

# Softverski dio

## Softverski dio poslužitelja

### Tehnologije

Poslužitelj ima na sebi operativni sustav Raspberrian „Buster“ Lite, dodatno je installirano LAMP(Linux Apache MySQL Php) koji omogućuje hostanje webstranice. Apache je program za hostanje webstranice sa samo HTML-om i CSS-om.

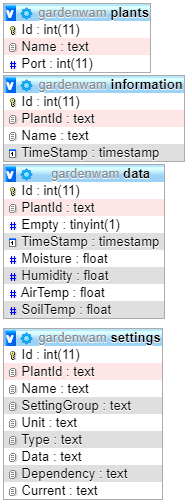
## Baza podataka

Baza podataka je glavni dio softvera koji sprema sve podatke na jednom mjestu.

### Tehnologije

Baza podataka napravljena pomoću MarinaDB inačica 10.4.12. koja je u svojoj pazi MySQL.

### Struktura



Definira podatke koje se mogu poslati i kada su poslane opcija empty sluzi kao nulta vrijednost za novo napravljene biljke

Definira sve glavne vrijednost kako bi mogli implementirati custom opcije SettingsGroup je vrsta podatka koja se mjenja a Depedency je ovisnost o nekoj drugoj vjednosti

Definira sve vrijednost koje se mijenjaju a potrebno je praćenje

Definira sve biljke i njihove portove

## API(Application programming interface)

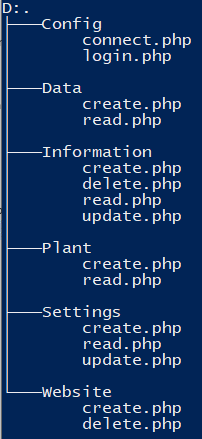
API je glavni dio softverskog djela koji omogućuje komunikaciju između svakog djela softvera. API je baziran na CRUD(Create, read, update, delete) API-u . CRUD je jednostavan API koji se koristi za kreiranje, čitanje, osvježavanje i brisanje. Ja nisam koristio sve te opcije jer za neke dijelove API-a nije bilo potrebe.

### Tehnologije

Baza API-a je PHP inačica 7.4.2. PHP služi kao backend jezik preko njega omogućujem komunikaciju između različitih dijelova sustava preko MySQL data baze.

### Struktura

Struktura mog API-a ima 3 glavne cjeline:



3) Website

1) Config

2) MySQL dio

### Config

On sluzi za konfiguriranje API-a pod connect.php je kako se spajamo na MySQL databazu. Login.php je kako bi se dio softvera prijavilo da koristi API pomoću basic AUTH-a, sto koristi samo Ime i lozinku.

### MySQL dio

Ovdje se nalaze svi djelovi API-a koji se spajaju na MySQL bazu podataka. U svim dijelovima svaki file radi praktično isto pa ću objasniti svaki file pojedinačno. Create.php radi novi redak. Read.php čita redak ili retke ovisno kako je definiran. Update.php koji osvježuje određeni redak ovisno o ulaznim podacima. Delete.php brise određeni redak ovisno o ulaznim podacima.

### Website

Ovaj dio služi da se dinamično rade webstranice. Ovisno o danim podacima. Glavni dio prikazivanja web stranice je moj modularni sistem koji ću kasnije objasniti

## Program za očitanje senzora i upravljanja aktuatorima

Program za očitanje senzora i upravljanja aktuatorima je glavni dio koji povezuje hardver sa softverom točnije modul za automatizaciju sa web stranicom.

### Tehnologije

Raspberry pi out the box koristi za upravljanjemGPIO-om python, tako da sam ja koristio za ovaj dio Python 3.7 sa raznim bibliotekama. Najvažnije su biblioteke bile adafruit mcp3xxx, requests i biblioteku koju sam programirao mysht21. Da omogućim da sve radi trebao sam koristiti još i json, digitalio, board, time, threading, datetime. Adafruit mcp3xxx i mysht21 se koriste za očitanje podataka sa senzora. Requests služi da šaljem http request-ove na svoj API kako bi mogao spremiti podatke u bazu podataka. Json sluzi za pisanje i čitanje JSON formata. DigitalIO sluzi za kako bi mogli koristi Raspberry-jeve GPIO pinove. Board služi za lakše uzimanje GPIO pinova. Time služi da znam koliko dugo traje neki process. Theading za rađenje vise threadova u programu. DataTime za pretvaranje MySQL timestamp-a u broj.

### Struktura

Struktura koda je jednostavna prvi dio koda uzima trenutačne vrijednosti senzora svih biljaka napravi si objekte sa svim vrijednostima i pinovima koji koristi svaki port, drugi dio koda koji se zapravo dijeli na dva threada, prvi thread takozvani Checker koristi da pregledava ima li promjene u opcijama i ako se dogodi neka promjena upiše nove vrijednosti u objekt sa svim biljkama i pregledava da li je došlo do promjene svakih x sekundi, gdje je x broj koji se može postaviti, drugi thread takozvani AddData on ima tri definirana intervala prvi koji se vrti svake sekunde je takozvani DataTime to čita sve vrijednosti senzora. Drugi interval je onaj koji možemo definirati preko web stranice je koliki će biti vrijeme koliko često upisujemo podatke u MySQL bazu. Treći je interval koji traje x sekundi, on određuje koliko dugo nakon zalijevanja možemo ponovno zaliti biljku.

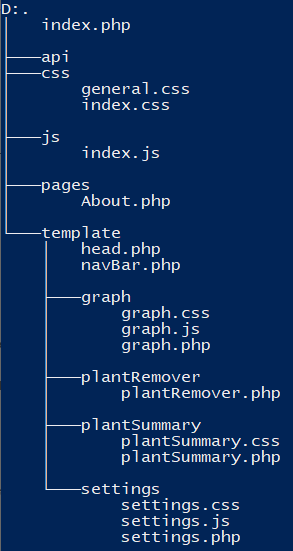
## Web stranica

Web stranica je napravljen na malo drugačiji način od uobičajenog. Ovu metodu izrade stranica sam nazvao modularnim sistemom, jer se web stranica sastoji od malih modula.

### Tehnologije

Glavni dio za modularnosti stranice je PHP inačica 7.4.2 , za izgled web stranicu koristim HTML inačica 5 i CSS inačica 3. Funkcionalnost stranice je napravljen pomoću JavaScripta(JS). U JS-u sam koristio biblioteke JQuery za jednostavnije AJAX pozive i lakše uzimanje elementa stranice i Flot biblioteku koji koristim za crtanje grafova.

### Struktura

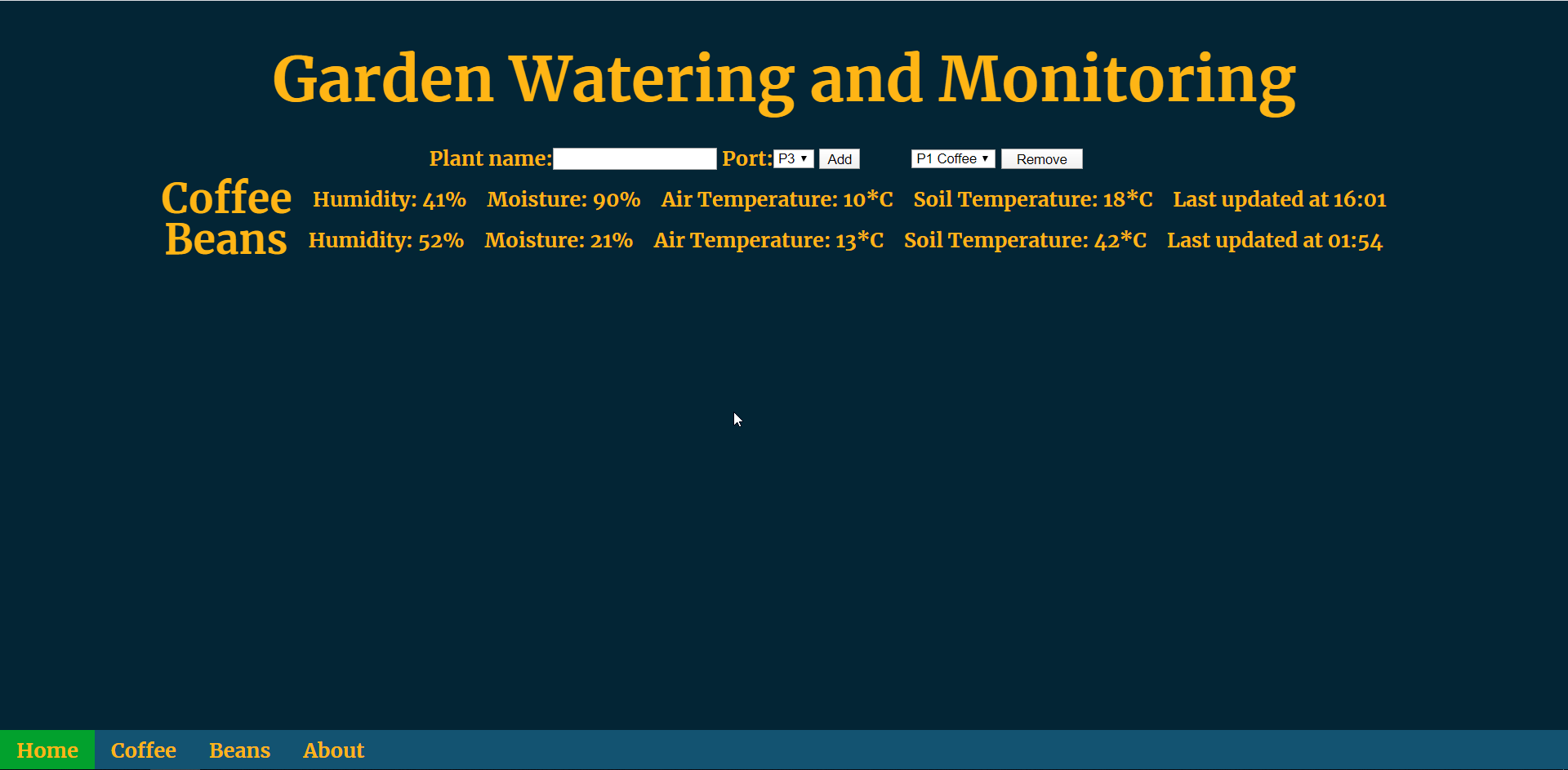


Folder sa web stranicama s biljkama

Početna web stranica

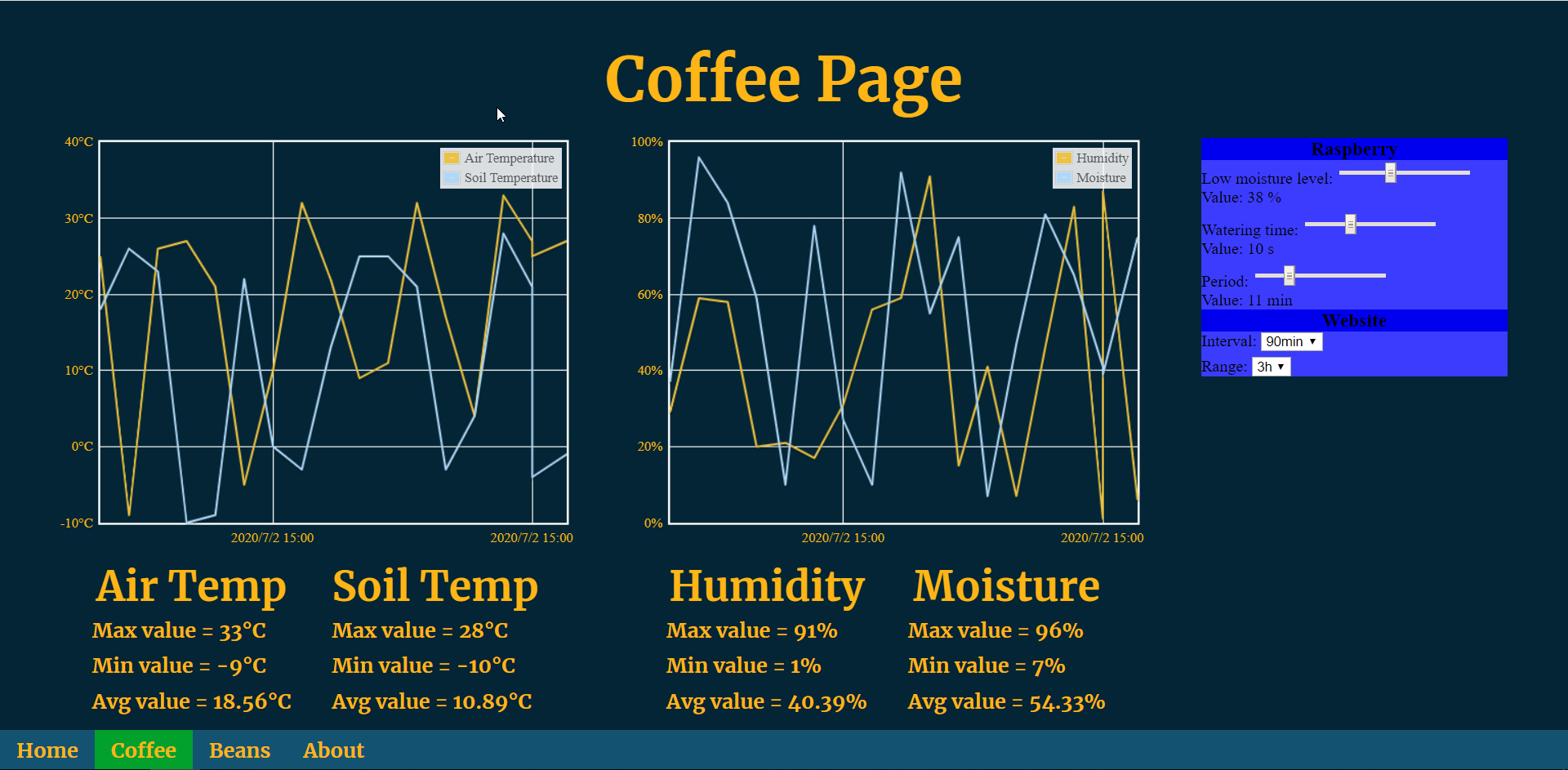
Moduli

### Početna web stranica



Slika 14 Početna web stranica

### Web stranica s biljkom



Slika 15 Web stranica s biljkom

### Moduli

Moduli su elementi stranice koji se definiraju pomoću PHP klase. Svaka klasa ima svoje private varijable koje možemo dobit pomoću get\_[varijabla]-a i set\_[varijabla]-a gdje je [varijabla] ime te varijable koju želimo postavit ili uzet, u nekim modulima ima podklase koji služe kako bi jasnije napravio kako radi taj kod. Najvažnija funkcija u svim klasama je get\_HTML() koji nam vraća HTML ovisno o postavljenim vrijednostima.

### Head modul

Head modul je modul koji nam radi head dio HTML-a i importa sve važne biblioteke i css design sheetove

### NavBar modul

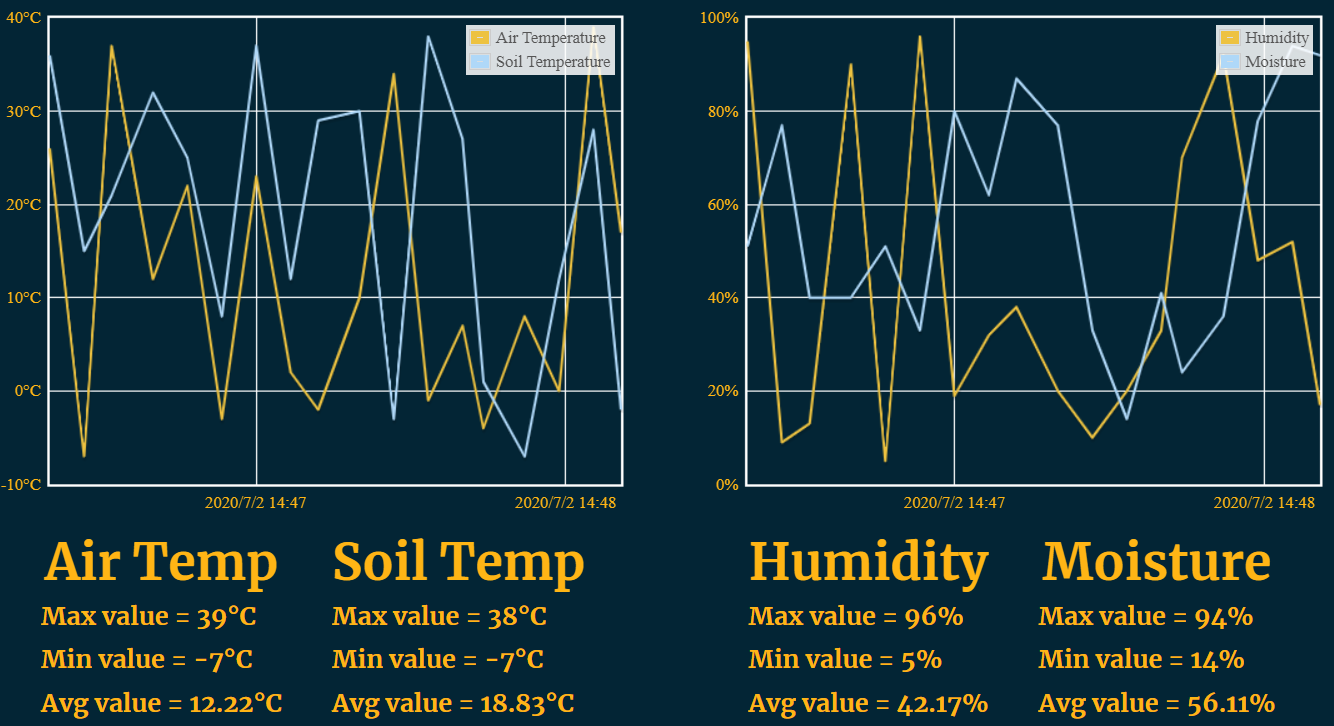
NavBar je navigacijski Bar, ovaj modul ima dodatnu funkcionalnost da uzima sva imena datoteka u Pages folder i za svaki od njih radi novi navigacijski item i to nam omogućuje navigaciju između svih stranica



Slika 16 Bar za navigaciju

### Graph modul

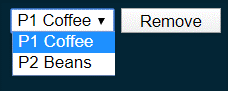
Graph modul je modul koji nam na stranici za svaku biljke radi grafički prikaz svih vrijednosti te biljke, ima svoj css i js file zato jer ima više pod elementa koji trebaju lijepo izgledat pa nam treba custom css. U javascriptu uzimam sve vrijednost definirane u MySQL bazi podataka po dobivenim vrijednostima i opcijama se napravi grafički prikaz svih izabranih podataka te biljke.



Slika 17 Grafički modul

### PlantRemover modul

PlantRemover modul koji nam napravi select element u htmlu pomoću kojeg možemo maknuti izabranu biljku.



Slika 18 Modul za micanje biljaka

### PlantSummary modul

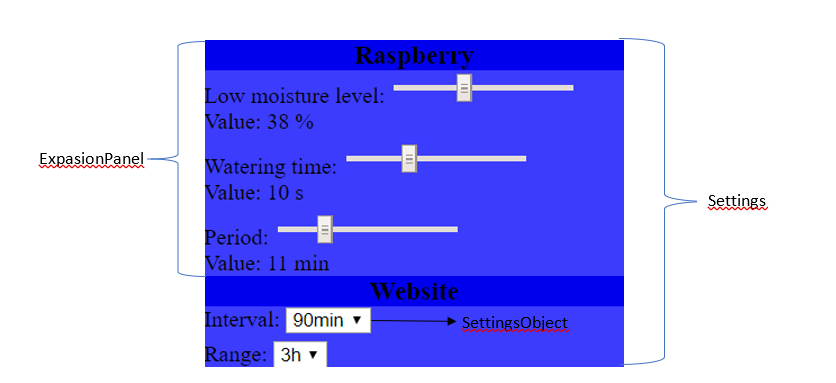
PlantSummary modul koristimo na početnoj stranici kako bi pokazali zadnje vrijednosti dobivene biljaka. Zbog više elementa potreban nam je css kako bi uljepšali sve



Slika 19 Općenito o biljkama

### Settings modul

Settings modul je najkompliciraniji modul. Njegova php klasa se sastoji od 3 pod klase koje su settings, settingsObject i expansion panel. SettingsObject je najmanji element u ovom modulu on nam radi pojedinačne opcije koje mogu biti 3 tipa RadioSelect, select i slider. Expansion panel je idući po redu koji sadrži settingsObjekte koji mijenjaju opcije istom tipu vrijednosti i na kraju imamo settings objekt koji spaja vise expasionPanela u jedno. Sve opcije uzimamo preko MySQL baze to je glavni dio ove modularnosti jer svaka opcija je definirana u MySQL bazi pa lako mogu raditi custom opcije za svaku biljke. Svaka biljka treba imat svoje opcije. Js file ovog modula je napravljen tako da ima svoj lokalni settings object koji ima sve važne definicije svakog object-a. Select opcije mogu imati definiran dependancy što nam omogućuje da jedan select objekt ovisi o drugom. Primjer ovog je Range i interval. Range je koliko podataka uzimamo tj koliko idemo u prošlost, na primjer ako imamo 2 ranga od 1sat i 1dan. Za Range od 1 dana vremenski interval na x osi biti će 3 sata a za Range od 1 sata vremenski interval na x osi biti će 20 minuta.



Slika 20 Opcije

# Zaključak

Ovaj projekt je napravljen s namjerom da se može biljka zalijevat bez prisutnosti fizičke osobe. U trenutačnoj verziji mogu zalijevat samo jedno biljku jer nema opreme za više no sa dodatnim resursima moglo bi se napraviti da zalijevam više biljki odjednom.

Planovi za budućnost će biti pojednostaviti shield tako da se ne koriste pin headovi nego konektori, shieldu treba dodati tranzistore kao sklopke. Mogućnost više od 2 senzora po biljci/vrtu, više aktuatora. Softverski treba napraviti tako da stranica bude još više modularna tako da se može definirati koje vrijednosti pratimo na kojem pin-u porta. Preraditi API tako da podržava modularne podatke.

Kroz ovaj projekt sam naučio niz programskih jezika koje sam samo djelomično poznavao. Iznenadio sam se koliko sam stvari koje sam naučio u školi da sam primijenio. Sve u svemu mislim da je projekt dobro djelovao na mene i jer sam se naučio radnoj etici i kontinuitetu rada.

# Literatura

Za literaturu korištena dokumentacija sa slijedećih web stranica:

<https://en.wikipedia.org/wiki/I%C2%B2C>

<https://en.wikipedia.org/wiki/Universal_asynchronous_receiver-transmitter>

<https://en.wikipedia.org/wiki/Serial_Peripheral_Interface>

<https://e-radionica.com/en/>

<https://www.php.net/>

<https://mariadb.org/>

<https://jquery.com/>

<https://www.adafruit.com/>

<https://www.flotcharts.org/>

<https://www.raspberrypi.org/>