

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond
Arvutisüsteemide instituut

Marta Aarlaid 155700
Piret Näppi 154859

KLIIMA MONITOORING

Tarkvara projekt

Juhendajad: Jürgen Soom
Joonas Tamm
Jan Toodre
Ekke Tõiv Uustalu

Tallinn 2017

Autorideklaratsioon

Kinnitame, et oleme koostanud antud projekti iseseisvalt ning seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on töös viidatud.

Autor: Marta Aarlaud

Piret Näppi

31.05.2017

Annotatsioon

Käesoleva töö eesmärgiks on luua toa kliima, täpsemalt temperatuuri ja niiskuse jälgimise süsteem, mis põhineb Raspberry Pi mikrokontrolleril. Lõpptulemuseks on süsteem, mis koosneb Raspberry Pi 2 mikrokontrollerist, DHT22 andurist ning andmete jälgimiseks on loodud veebikeskkond.

Projekt on kirjutatud eesti keeles ning sisaldab teksti 11 leheküljel, 6 peatükki, 2 joonist.

Sisukord

Autorideklaratsioon.....	2
Annotatsioon.....	3
Jooniste loetelu.....	5
1 Sissejuhatus.....	6
2 Ülesande püstitus.....	7
3 Tööjaotus.....	8
4 Riistvara.....	9
4.1 DHT22.....	9
4.2 Raspberry Pi 2.....	9
5 Tarkvara.....	10
5.1 Andurilt info saamine.....	10
5.2 Koduleht.....	10
5.2.1 Kujundus.....	10
5.2.2 Graafikud.....	10
5.2.3 API.....	11
5.3 Andmebaas.....	12
6 Kokkuvõte.....	13

Jooniste loetelu

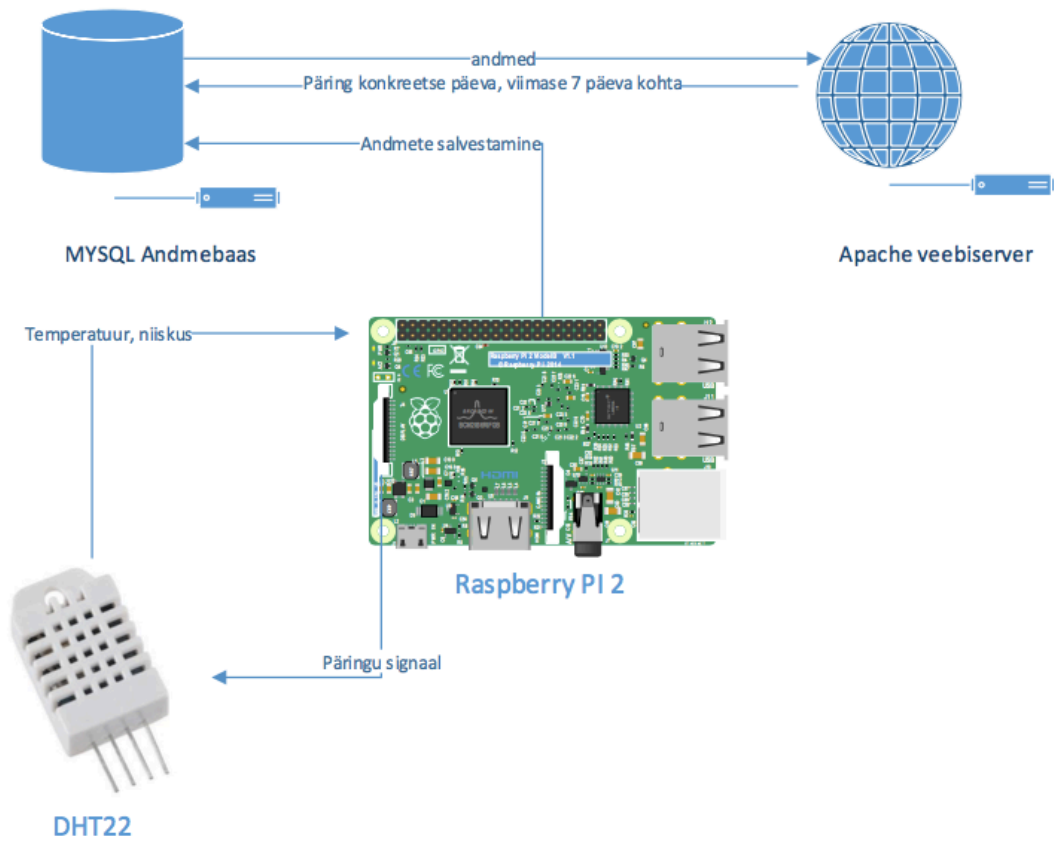
Joonis 1 Projekti mudel.....	7
Joonis 2 Andmebaasi struktuur.....	12

1 Sissejuhatus

Toa temperatuur ja niiskus on asjad, millele tavainimene pöörab tähelepanu võib-olla vaid kord päevas. Kuid kui tegemist on allergikuga, kelle allergiat võib süvendada kuiv ja külm õhk või kui tegemist on toaga, kus kasvatatakse troopilisi taimi, võib toa temperatuur ja niiskus olla väga olulise tähtsusega. Sellistest isiklikest huvidest ajendatuna valisimegi oma teema. Meie sooviks oli luua süsteem koos veebilehega, mis sisaldaks lihtsaid ja jälgitavaid graafikuid, et anda kasutajale hea ülevaade vastavalt päeva või nädala trendidest.

2 Ülesande püstitus

Eesmärgiks on saada informatsiooni toa temperatuuri ja niiskustaseme trendidest. Ülevaate saamiseks genereeritakse graafikud veebilehele vastavalt jooksva päeva ja nädala kohta (viimased 7 päeva). Lisaks sellele kuvatakse ka kogu perioodi maksimumid ja miinimumid.



Joonis 1 Projekti mudel

3 Tööjaotus

- Koos – Raspberry Pi konfigureerimine, veebiserveri (Apache) ülesseadmine, andmebaasi ülesseadmine ja dokumentatsioon
- Marta – Andmebaasi põhjal veebi sh graafikute loomine
- Piret – Andurilt info kogumine ning selle postitamine andmebaasi

4 Riistvara

4.1 DHT22

Andur sai valitud seetõttu, et see võimaldas meil pärida niiskust ja temperatuuri samaaegselt. Vastasel juhul oleksime pidanud soetama kaks eraldi komponenti, mis ei oleks olnud otstarbekas projekti hinna seisukohalt. Seda soetades selgus, et andur oli niivõrd populaarne, et saime Oomipoest viimase toote. See oli meile kinnituseks, et tegime hea valiku.

4.2 Raspberry Pi 2

Esimesel ettekandel tuli meie suunas etteheited, et miks just Raspberry, kui sama tööd suudaks ka väiksema võimekusega kontrolleri pakkuda. Meie otsustasime Raspberry kasuks just selle tõttu, et sellele sai paigaldada veebiserverit ja andmebaasi. Vastasel juhul oleksime pidanud teenuse kasutamiseks lisariistvara paigaldama. Samuti oli meil Raspberry Pi 2 ka juba olemas ning teise kontrolleri ostmine poleks olnud otstarbekas.

Esimestel katsetusel proovisime internetiühenduse saamiseks rakendada wifimoodulit (WiPi), kuid pikemal kasutamisel selgus, et see moodul ei olnud alati töökindel ning seetõttu võtsime kasutusele Ethernet kaabli, millega pole tõrkeid veel siia maani esinenud.

5 Tarkvara

5.1 Andurilt info saamine

Info küsimine on realiseeritud programmeerimiskeeles Python. Kuna tootja oli GitHub'is väljastanud anduriga suhtlemiseks mooduli Adafruit_DHT, siis parema töökindluse tagamiseks saigi see kasutusele võetud. Programm ise on failis `measure.py`. Mõõtmine toimub iga tunni aja tagant ning, kui mõõtmist ei suudetud kolmel järjestikkusel korral sooritada, siis tulemust andmebaasi ei kanta. Vahepealsel ajavahemikul programm ei tööta. Seda käivitab operatsioonisüsteemi tööriist *crontab*, mis on loodud automaattööde loomiseks. Meie projektis oli automaattööks käivitada mõõtmisprogrammi iga tunni aja tagant. See aeg sai valitud just seetõttu, et meie projekti töökeskkonnas ei toimu väiksema ajavahemiku vältel suuri temperatuuri ega niiskuse muutusi.

5.2 Koduleht

5.2.1 Kujundus

Kodulehe kujundamisel on kasutatud struktuuri loomiseks HTML'i ning kujunduskeelena CSS'i. Kujundus on jaotatud kaheks osaks - kujundus konkreetse päeva vaates ja kujundus viimase seitsme päeva vaates. Need asuvad failides `index.php` (tänapäev) ja `week.php` (viimane nädal). Kujunduse kirjeldamiseks kasutatud faili nimi on `style.css`.

5.2.2 Graafikud

Graafikute loomiseks proovisime kasutada PHP teeki PHPlot, kuid see ei õnnestunud, sest sellel puudus paindlikkus. Seetõttu oleksime pidanud rohkem koodi kirjutama ning ei oleks saanud samu programmeerimismustreid erinevate graafikute loomisel kasutada, vaid oleks pidanud iga graafiku jaoks eraldi faili looma, mis oli selle teegi kasutamise eelduseks.

Tänu juhendajatele leidsime internetist ühe vabavaralise graafikugeneraatori Chart.js, mis tegi just seda, mida vaja. Mooduli kasutamiseks on loodud suhtlusfunktsioonid failis `customJavaScript.js`. Graafiku loomine näeb välja järgnevalt:

```

var lineData = {
    labels: x_labels, // lisame x teljele pealkirjad
    datasets:[
        {
            label: label, //graafiku pealkiri
            backgroundColor: '#008000', // graafiku värv
            data: y_data, // y telg on andmete telg
        }
    ]
};
var ctx = document.getElementById(canvas).getContext('2d'); //koht
kuhu graafikut ehitada laseme
var myChart = new Chart(ctx, {
    type: 'line', //graafiku tüüp
    height: 260, // graafiku kõrgus
    data: lineData, //andmed
    options: {
        scales: {
            yAxes: [{
                ticks: {
                    min: Math.round(min_y_val) - 1, // paneme paika endale
                    //sobiva graafiku miinimumi
                }
            }]
        }
    }
});

myChart.render(); //joonistame graafiku

```

Lisaks sellele oli ülesandeks kätte saada ka miinimum, keskmine ja maksimum. Nende arvutamiseks käisime kõik Y-telje väärtused (temperatuurid ja niiskustulemused) käsitsi *for* tsükli läbi.

5.2.3 API

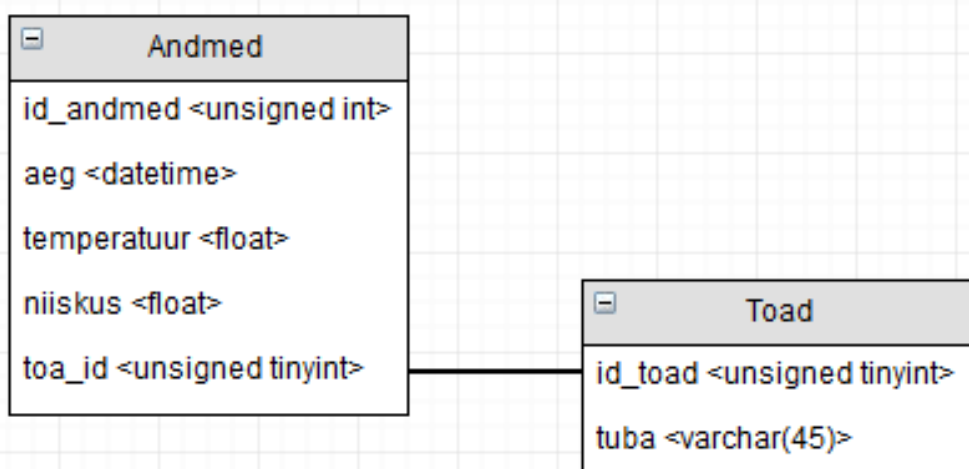
APIks võib nimetada teenust, mis väljastab teatavate parameetrite alusel andmeid. Alternatiivselt võib selle ka välja jätta, kuid proovisime antud töös õppida uusi meetodikaid.

Meie koostatud API tööpõhimõte on järgnev: Javascript teeb päringu failis meie veebiserveris, mille nimi on `getData.php`, kuhu paneb sisse parameetri (`temp_today`, `temp_week`, `humi_today`, `humi_week`). PHP fail tuvastab selle parameetri ning selle alusel teeb andmebaasile (failiga `db.php`) teatava päringu. Kui päring tehtud, kodeeritakse tulem JSON formaati, mis on eelkõige kasulik just selleks, et hiljem päringu teinud JavaScript funktsioon oskaks neid andmeid kõige lihtsamalt töödelda.

Kui PHP fail saab tundmatu parameetri, siis tagastab ta väärtuse “404”, mis on internetimaailmas kokkulepitud tähistus viga päringu puhul.

5.3 Andmebaas

Andmebaasiks võtsime kasutusele tasuta tarkvara MySQL, mille puhul saime kasutada mugavat tööriista MySQL Workbench, mis tundus kasutatavuse poolest meile kõige lihtsam. Andmebaasis hoiame järgmisi andmeid: mõõtmise kuupäev, niiskuse mõõtetulemus, temperatuuri mõõtetulemus ja tuba.



Joonis 2 Andmebaasi struktuur

Kuna mõõtmistulemuste kellaegade näitamine oli meie jaoks oluline, siis tuli hoolitseda selle eest, et alati oleks näidu juures kuvatud õige kuupäev. Selle probleemi lahendasime sellega, et seadistasime Raspberry operatsioonisüsteemi nii, et see küsiks kellaega pidevalt internetist. Andmebaasi seadistasime jällegi nii, et see küsiks pidevalt kuupäeva operatsioonisüsteemilt. Nii viisi saime õige kellaaja ka siis, kui Raspberry oli pikemaks ajaks vooluvõrgust eemaldatud. Kogu lahenduse miinuseks on siiski see, et eeldatakse pidevat internetiühenduse olemasolu.

6 Kokkuvõte

Selliseid süsteeme saaks osta poest ka juba valmiskujul, kuid huvitav on näha ise seda protsessi ja kõike, mis selle valmimisse läheb. Projektist saime palju uusi teadmisi ja arendasime oma koostöö oskusi. Samuti on meil võimalus soovi korral seda projekti tulevikus edasi arendada, lisades sinna erinevaid andureid ning komponente, mille abil oleks võimalik toa kliimat ka kontrollida, mitte ainult jälgida. Võimalus on ka kogu süsteemi suuremaks teha ning rakendada kogu maja või korteri peal.