

Sustav za udaljeni nadzor temperature i vlage zraka u poduzeću

Štimac, Alexander

Master's thesis / Diplomski rad

2014

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Graphic Arts / Sveučilište u Zagrebu, Grafički fakultet**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:216:128449>

Rights / Prava: [In copyright](#) / [Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2024-10-01**



Repository / Repozitorij:

[Faculty of Graphic Arts Repository](#)



**SVEUČILIŠTE U ZAGREBU
GRAFIČKI FAKULTET**

ALEXANDER ŠTIMAC

**SUSTAV ZA UDALJENI NADZOR
TEMPERATURE I VLAGE ZRAKA U
PODUZEĆU**

DIPLOMSKI RAD

Zagreb, 2014.



Sveučilište u Zagrebu
Grafički fakultet

ALEXANDER ŠTIMAC

**SUSTAV ZA UDALJENI NADZOR
TEMPERATURE I VLAGE ZRAKA U
PODUZEĆU**

DIPLOMSKI RAD

Mentor:
prof. dr. sc. Lidija Mandić

Student:
Alexander Štimac

Zagreb, 2014.

Rješenje

SAŽETAK

Ovaj diplomski rad se zasniva na integraciji danas dostupnih tehnologija u svrhu udaljenog nadzora temperature i vlage zrake na određenoj lokaciji. Tehnologije su objedinjene u funkcionalan sustav koji pruža usluge za praćenje i manipulaciju izmjerenih vrijednosti u svrhu preventivnog održavanja i analize podataka. Sustav se sastoji od više međusobno povezanih podsustava od kojih svaki ima određeni zadatak. Mjerni podsustav se sastoji od senzora povezanog na mikrokontroler, a zadatak mu mjerenje vrijednosti temperature i vlage zraka. Mrežni podsustav je zapravo udaljeni poslužitelj koji prima izmjerene vrijednosti i pohranjuje ih u bazu podataka. Također je zadužen za obradu podataka i slanje obavijesti i upozorenja u slučaju odstupanja izvan dozvoljenih granica tolerancije. Korisnički podsustav obuhvaća uređaje koje posjeduje sam korisnik (npr. mobilni uređaj), a koriste se za praćenje izmjerenih, trenutnih vrijednosti. Osim praćenja vrijednosti, korisnik ima uvid u grafički prikaz podataka te statističku analizu tijekom dnevnog ili tjednog mjerenja. Nadalje, mobilni uređaj prima obavijesti o rezultatima u realnom vremenu, što znači da korisnik uvijek može saznati trenutno stanje mjerenja. Rezultat rada je funkcionalni prototip temeljen isključivo na tehnologijama otvorenog koda što implicira na pristupačnu tržišnu cijenu, modularnost i daljnju nadogradnju.

Ključne riječi:

Arduino, mrežne tehnologije, udaljeni nadzor, obrada podataka

ABSTRACT

The basis of this work is the integration of the available open source technologies for the purpose of remote monitoring of temperature and humidity at the particular location. The technologies are integrated into a functional system that provides services for real time monitoring and manipulation of the measured values for preventative maintenance and data analysis. The system consists of several interconnected subsystems and each subsystem has a specific task. Measurement subsystem consists of temperature and humidity sensor, which is connected to the Arduino microcontroller. Network subsystem is a remote server that receives the measured values and stores them to the database. It is also responsible for processing the data and sending the notifications and alerts in case of a deviations which are beyond allowed limits of tolerance. User subsystem includes mobile devices which are used to monitor the measured values. In addition to monitoring the values, user has access to the graphical representation of data and statistical analysis during daily or weekly measurements. Mobile device also receives a notification which contains the results in real time, which means that the user can always check the current state of measurement system. The result of this work is a functional prototype based on a open source technologies, which implies at the affordable market price, modularity and further upgrade.

Key words:

Arduino, network technologies, remote monitoring, data processing

SADRŽAJ

1. Uvod.....	1
2. Sustav za nadzor temperature i vlage zraka.....	2
2.1. Temperatura	4
2.2. Vlažnost zraka.....	6
2.3. Fizičke komponente sustava	8
2.3.1. Arduino.....	8
2.3.2. Senzor za mjerenje temperature i relativne vlage zrake DHT22.....	10
2.3.3. Arduino Ethernet Shield	11
2.3.4. Usmjernik	12
2.4. Programska podrška sustava	13
2.4.1. PHP.....	13
2.4.2. Baza podataka.....	15
2.4.3. Operacijski sustav Android	17
2.4.4. XML	20
3. Praktični dio.....	21
3.1. Opći i specifični ciljevi rada	21
3.2. Plan rada.....	22
3.2.1. Povezivanje fizičkih komponenata u mjernom podsustavu	23
3.2.2. Izrada programskog rješenja za Arduino mikrokontroler	24
3.2.4. Izrada baze podataka	25
3.2.3. Izrada programske podrške na strani poslužitelja	26
3.2.4. Izrada Android mobilne aplikacije	30

4. Eksperimentalni dio	33
5. Rezultati	35
5.1. Ispitivanje pouzdanost sustava	35
5.1.1. Apsolutna pogreška	36
5.1.2. Relativna pogreška	39
5.1.3. Standardna devijacija	40
5.1.4. Izračun standardne pogreške	41
5.1.5. Izračun relativne nepouzdanosti mjerenja	42
5.2. Ispitivanje funkcionalnosti sustava	42
6. Rasprava.....	45
7. Zaključak	46
8. Literatura.....	47
Popis oznaka i kratica	47
Popis slika	50
Popis tablica.....	51
Popis programskih isječaka	52
Popis jednadžbi	53
Prilozi.....	54

1. UVOD

Utjecaj temperature i vlage zraka ima značajnu ulogu u industriji. Ti čimbenici neposredno utječu na rentabilnost čitave proizvodnje. Primjerice, proces otiskivanja i skladištenje papira u grafičkoj industriji zahtijeva klimatizirane prostore u kojima su uvjeti strogo kontrolirani. Budući da je papir kao sirovina vrlo osjetljiv na promjene stanja okoline, važno je da su temperatura i vlaga zraka regulirane i u svim prostorijama jednake. Prema standardu, preporučuje se da relativna vlaga iznosi 55%, a temperatura 23°C. Kako bi se sačuvali optimalni uvjeti, potrebno ih je redovito kontrolirati. Kontrola uvjeta vrši se u svrhu preventivnog održavanja, a time i sprečavanje dodatnih financijskih troškova te nemogućnosti proizvodnje u određenom vremenskom roku.

Na tržištu su dostupni razni uređaji koji omogućuju mjerenje i kontrolu stanja okoline, međutim, razlikuju se po svojoj funkcionalnosti. Velik udio uređaja zahtijeva manualno korištenje, odnosno, kod svakog očitavanja mjerenja operater mora biti fizički prisutan na mjestu mjerenja i zapisati rezultate kako bi kasnije mogao izraditi analizu podataka. U današnje vrijeme takav se način čini apsurdan jer postoje tehnologije koje omogućuju potpuno automatizirano i udaljeno praćenje rezultata mjerenja s bilo kojeg mjesta. Time se isključuje potreba manualnog zapisa jer su rezultati evidentirani i pohranjeni u sustav. S druge strane, uređaji s takvim sustavima su poprilično skupi što dovodi u pitanje isplativost kod manjih poduzeća.

Ovaj rad ima cilj objediniti dostupne i besplatne tehnologije, sa svrhom izgradnje sustava za udaljeni nadzor i automatizirane analize izmjerenih vrijednosti temperature i vlage zrake, a pritom uzeti u obzir ekonomsku isplativost i pouzdanost rezultata mjerenja.

2. SUSTAV ZA NADZOR TEMPERATURE I VLAGE ZRAKA

Nadzor temperature i vlage zraka ima značajnu ulogu u proizvodnim procesima koji zahtijevaju proizvodnju u strogo kontroliranim uvjetima. Takvi se uvjeti provode u mnogim granama industrije. Neke od njih su: proizvodnja lijekova u farmaceutskoj industriji, prerada sirovina u prehrambenoj industriji te čitav proces otiskivanja u grafičkoj industriji. Navedeni se procesi izvode u okviru propisanih vrijednosti temperature i vlage, a odstupanje izvan tog okvira dovodi do mnogih negativnih posljedica koje utječu na čitavu proizvodnju.

Vrijednosti temperature i vlage zrake moguće je pratiti na dva načina. Prvi, jednostavniji način se oslanja na ručno zapisivanje vrijednosti koje su izmjerene nekim mjernim uređajem, dok drugi, napredniji način predstavlja praćenje putem automatiziranog sustava. Primjer jednostavnog takvog sustava je *HWG-STE: Ethernet Termometer* koji se sastoji se od senzora, uređaja s *Ethernet* priključkom i adaptera za napajanje. Aplikacijsko rješenje temelji se na *Windows* aplikaciji koja prikuplja i obrađuje podatke, a izvještaje generira u obliku *Excel* tablica. Na uređaj je moguće priključiti do dva temperaturna senzora u rasponu od -10 do 80°C uz odstupanje od 0.1°C. *Ethernet* priključak služi za povezivanje sustava s računalom na kojem je instalirana *Windows* aplikacija. Aplikacija u pozadini prikuplja podatke, obrađuje ih i izvještaje šaljem putem *e-maila*.

Prednost sustava jest jednostavna ugradnja i instalacija, a glavni nedostatak je pregled informacija na samo jednom računalu s kojim sustav mora biti povezan. Ne postoji mogućnost udaljenog nadzora kao ni izračun srednje kinetičke temperature. U slučaju prekoračenja dozvoljene temperature, sustav ima samo jednu mogućnost, a to je slanje obavijesti putem elektronske pošte. Osim toga, aplikacijsko rješenje temelji se na *Windows* okruženju što znači da neće raditi na drugim operacijskim sustavima.

Naprednije rješenje predstavlja *Cool Chain*. To je zapravo sustav za neprekidno i visoko pouzdano laboratorijsko mjerenje temperature i vlage zraka prilikom čuvanja farmaceutskih proizvoda. Namijenjen je ugradnji u sve postojeće prostore u kojima se proizvode i čuvaju farmaceutski proizvodi, a sastoji se od umjerenih i certificiranih

senzora koji vrše mjerenje temperature i vlage uz maksimalno odstupanje senzora do 0.5°C i 5% RH. Podaci su pohranjeni na *web* poslužitelju koji omogućuje uvid u trenutne uvjete okoline putem Internet veze. Osim sofisticiranih senzora, sustav ima osiguran kontinuitet mjerenja, što znači da je potpuno samostalan i nezavisan od postojećih računala, usmjernika i lokalne mreže u objektu. Programska podrška omogućuje automatsku izradu izvještaja za ispis, izračun srednje kinetičke temperature, prikaz minimalne, maksimalne i prosječne vrijednosti temperature te SMS i *E-mail* alarmiranje u slučajevima izlaska temperature van granica. Sustav doista odlikuje izvanrednim specifikacija i prema tome je namijenjen samo velikim poduzećima u farmaceutskoj industriji (specifično je dizajniran za prostore u kojima se proizvode farmaceutski proizvodi). Nadalje, za ugradnju i održavanje sustava potrebno je izdvojiti financijska sredstva koja manjim poduzećima nisu toliko isplativa.

Kombinacijom funkcionalnosti navedenih sustava nastala je ideja za izradu vlastitog sustava koji bi se koristio u grafičkoj industriji, a služio bi za udaljeni nadzor temperature i vlage zraka budući da upravo ti parametri utječu na kvalitetu papira, a i na čitav proces otiskivanja. Promatrani parametri i korištene tehnologije opisani su u teoretskom dijelu ovog diplomskog rada.

2.1. Temperatura

Temperatura je fizikalna veličina koja pokazuje stupanj zagrijanosti neke tvari. Ovisi o energiji koju sadrži neko tijelo određene mase i tlaka. Gledano s termodinamičkog stajališta, temperatura je termička veličina koja ima istu vrijednost u dva sustava ili tijela koja su u termalnom kontaktu i pusti ih se da dođu u termalnu ravnotežu. Ako su dva tijela ili sustava u termalnoj ravnoteži onda imaju istu temperaturu. Drugim riječima, u termalnoj ravnoteži nema prolaza topline s jednog tijela na drugo. [1]

Zbog nemogućnosti izravnog mjerenja temperature, odabrane su dvije fizikalne pojave (voda koja se smrzava i voda koja isparava) koje se odvijaju uvijek pri točno određenim temperaturama i njima su pridružene brojčane vrijednosti. Brojčane vrijednosti su prikazane na temperaturnim skalama među kojima se u SI - mjernom sustavu primjenjuju samo dvije - Celsiusova (dopuštena) i Kelvinova (obavezna). [1]

Celsiusova skala (jedinica °C) je stara i najraširenija skala koja se održala jer je spretno definirana - pri tlaku 1,01325 bara ima vrijednost 0 na ledištu vode i vrijednost 100 na vrelištu vode. Naziva se "relativnom" skalom jer su obje točke proizvoljno odabrane. Preračunavanje iz Kelvinove skale izvodi se tako da se od zadane vrijednosti oduzme 273,25 (Jednadžba 1). Definirana je izrazom:

$$\vartheta^{\circ\text{C}} = [T(K) - 273,15(K)] \frac{1^{\circ\text{C}}}{1\text{K}} \quad (1)$$

gdje je:

$\vartheta^{\circ\text{C}}$ – vrijednost u Celsiusovim stupnjevima

$T(K)$ – parametar temperature u Kelvinima

Kelvinova skala (jedinica K) je temeljna temperaturna skala SI - mjernog sustava. Nastala je na jednostavnim pomicanjem (translacijom) Celsiusove skale, a definira se kao skala koja ima ishodište na apsolutnoj nuli, a pri trojnoj točki vode (+0,01°C) ima vrijednost 273,16 K.

Preračunavanje iz Celsiusove skale (*Jednadžba 2*) se izvodi tako da se zadanoj vrijednosti doda 273,15 a definirano je izrazom:

$$T(K) = [\vartheta(^{\circ}\text{C}) + 273,15(^{\circ}\text{C})] \frac{1K}{1^{\circ}\text{C}} \quad (2)$$

gdje je:

$T(K)$ – vrijednost u Kelvinima

$\vartheta^{\circ}\text{C}$ – parametar temperature u Celsiusovim stupnjevima

Mjerenje temperatura vrši se osjetnicima za mjerenje temperature koji se mogu svrstati u dvije osnovne skupine:

Kontaktni (provođenje topline):

- Otpornički elementi
- Termistori
- Termopar
- Poluvodički pn-spoj

Beskontakti (toplinsko zračenje):

- Infracrveni termometar
- Piroelektrički termometar

U radu je korišten senzor s ugrađenim termistorom. Termistor može biti negativan (NTC) ili pozitivan (PTC), odnosno, otpor osjetnika može opadati ili rasti s porastom temperature. NTC termistori se izrađuju od metalnih oksida (Fe_2O_3 , MgCr_2O_4). Spadaju u najosjetljivije senzore temperature, ali imaju nelinearnu ovisnost otpora o temperaturi $R(T)$, a osjetljivost i temperaturni koeficijent ovise im o temperaturi. S druge strane, PTC termistorima otpor raste s porastom temperature, izrađuju se od smjese barij-karbonata, stroncij-oksida i titan-oksida.

2.2. Vlažnost zraka

Vlažnost zraka čine molekule vode koje se, uz ostale molekule, nalaze u zraku. Voda se u zraku nalazi u obliku vodene pare. Ako je njena koncentracija u zraku veća, zrak je vlažniji i obratno. Koncentracija vlage u zraku (pri određenoj stalnoj temperaturi) ne može neograničeno rasti, nego može doseći određenu maksimalnu koncentraciju (zasićenje). [2]

Vlaga zraka se može izražavati u sljedećim fizikalnim veličinama:

- Apsolutna vlaga zraka
- Parcijalni tlak vodene pare u zraku
- Relativna vlaga zraka

Apsolutna vlaga zraka

Apsolutna vlaga zraka je masena koncentracija vode (u obliku pare) u zraku. Ona je omjer mase vodene pare (u gramima) i volumena zraka (u m^3). Označava se s a , a jedinica je g/m^3 . Definira se izrazom:

$$a = \frac{m}{V} \left[\frac{g}{m^3} \right] \quad (3)$$

gdje je:

a – apsolutna vlaga zraka

m – masa vodene pare

V – volumen zraka

Apsolutna vlaga zraka pri zasićenju je konstantna pri stalnoj temperaturi, a označava se s A .

Parcijalni tlak vodene pare u zraku

Parcijalni tlak vodene pare u zraku jest tlak koji bi nastao kada bi se iz zraka odstranile sve ostale tvari, odnosno, tlak kojeg bi razvijale molekule vode kada bi bile "same" u atmosferi (bez molekula dušika, kisika, ugljičnog dioksida, plemenitih plinova).

Označava se s e , a SI jedinice je Paskal (Pa). Parcijalni tlak zasićenih vodenih para u zraku je konstantan pri stalnoj temperaturi, a označava se s E . [2]

Relativna vlaga zraka

Relativna vlaga zraka je stupanj zasićenosti zraka vodenom parom. Izražava se u postotcima (%), a predstavlja omjer stvarne (izmjerene) apsolutne vlage zraka i apsolutne vlage zraka pri zasićenju na nekoj temperaturi zraka. Definirana je izrazom:

$$r = \frac{a}{A} 100\% [\%] \quad (4)$$

gdje je:

r – relativna vlaga zraka

a – apsolutna vlaga zraka

A – apsolutna vlaga zraka pri zasićenju

Relativna vlaga zraka može se izraziti i omjerom stvarnog (izmjerenog) parcijalnog tlaka vodene pare u zraku (e) i parcijalnog tlaka zasićenih vodenih para (E) na nekoj temperaturi zraka:

$$r = \frac{e}{E} 100\% [\%] \quad (5)$$

gdje je:

r – relativna vlaga zraka

e – parcijalni tlak vodene pare u zraku

E – parcijalni tlak zasićenih vodenih para

2.3. Fizičke komponente sustava

Za izradu funkcionalnog sustava, u radu su korištene sljedeće komponente:

- *Arduino mikrokontroler*
- *Senzor za mjerenje temperature i relativne vlage zrake DHT22*
- *Arduino Ethernet Shield*
- *Usmjernik*

U nastavku je svaka komponenta sustava detaljnije opisana, te je uz opis priložena tablica sa specifikacijama.

2.3.1. Arduino

Arduino (slika 1) je niskobudžetni mikrokontroler razvijen kao edukacijsko-eksperimentalna platforma za upravljanje interaktivnim objektima iz okoline. Platforma je popularna među populacijom koja je tek počela eksperimentirati s elektronikom i elektroničkim sklopovima zbog svoje jednostavne primjene i širokog spektra mogućnosti koje se mogu s njom ostvariti. Nove mogućnosti stvara mnoštvo različitih komponentata, modula i senzora koji su u potpunosti kompatibilni s Arduino platformom. Važno je spomenuti da je Arduino programska podrška otvorenog koda, a sklopovlje vrlo jeftino što platformu čini idealnim kandidatom za izradu prototipa. Programska podrška sastoji se od integriranog razvojnog okruženja i pojednostavljenog C++ programskog jezika. Arduino je dostupan u nekoliko različitih verzija, a najpopularniji model je Arduino Uno. [3]

Arduino Uno je pločica s mikrokontrolerom ATmega328 i 32KB memorije. Sastoji se od 14 digitalnih ulazno/izlaznih pinova, od kojih se 6 mogu koristiti kao PWM izlazi te 6 analognih ulaznih pinova. Napajanje je moguće putem USB priključka ili standardnog DC adaptera. U radu je korišten ovakav tip mikrokontrolera, a odabran je zato što je njegova upotreba relativno jednostavna i cijena mu je vrlo prihvatljiva. Programira se putem USB priključka povezanog na računalo. Na računalo se nalazi razvojno okruženje koje otvara serijsku vezu prema mikrokontroleru.



Slika 1. Mikrokontroler Arduino Uno

Izvor: <http://arduino.cc/en/Main/arduinoBoardUno>

U priloženoj tablici (tablica 1) navedene su opće specifikacije modula:

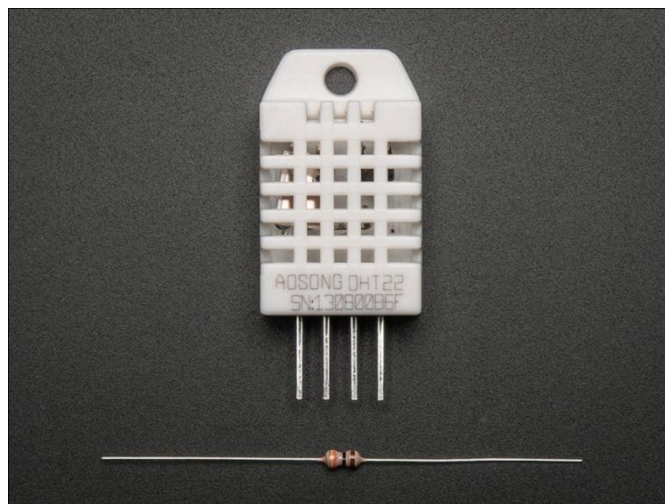
Tablica 1. Specifikacije Arduino mikrokontrolera

Mikrokontroler	ATmega328
Radni napon	5V
Ulazni napon (preporučeno)	7-12V
Ulazni napon (limitirano)	6-20V
Digital ulazno/izlazni pinovi	14
Analogni ulazni pinovi	6
DC struja po I/O pinu	40 mA
DC struja za 3.3V pin	50 mA
Flash memorija	32 KB (ATmega328)
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz

2.3.2. Senzor za mjerenje temperature i relativne vlage zrake DHT22

DHT22 (slika 3) je jednostavni senzor za mjerenje temperature i vlage zraka. Koristi polimerni kondenzator za mjerenje vlage i termistor za mjerenje temperature zraka iz okoline. Podatke šalje u obliku digitalnog signala na pin od mikrokontrolera. Prednost mu je cijena i jednostavna uporaba, a nedostatak predstavlja vrijeme očitavanja podataka koje iznosi 2 sekunde.

Senzor je kalibriran u posebnoj komori, a njegov kalibracijski koeficijent je pohranjen u OTP (*One-time programmable*) memoriju. Kalibracijski senzor poziva se svaki put prilikom mjerenja kako bi vrijednosti bile pouzdane. Osim niske potrošnje energije, senzor obuhvaća transmisijski domet do 20 metara. Na slici 2 je prikazan senzor koji je korišten u radu.



Slika 2. Senzor DHT22

Izvor: <https://www.sparkfun.com/products/10167>

Opće specifikacije DHT22 senzora su:

- Raspon mjerenja relativne vlage od 0 do 100%.
- Raspon mjerenja temperature od -40 do +125°C.
- Moguća odstupanja +-2% za vlagu, +-0.2°C za temperaturu.
- Vrijeme očitavanja svake 2 sekunde.
- Napajanje 3.3 - 6V DC
- Digitalni izlazni signal

2.3.3. *Arduino Ethernet Shield*

Arduino Ethernet Shield je komponenta koja mikrokontroleru omogućuje povezivanje na internet. Temelji se na *Wiznet W5100 ethernet* čipu koji pruža mrežni pristup za TCP i UDP i podržava do četiri istovremenih priključaka. Komponenta sadrži i utor za *Micro SD* karticu koji omogućuje čitanje i pisanje podataka. Vrlo je malih dimenzija (73mm x 54mm x 17mm) i teži 26 grama. Povezivanje s mikrokontrolerom vrši se priključivanjem pinova na Arduino pločicu. Na taj način osiguran je priključak ostalim komponentama na mikrokontroler ukoliko se one žele koristiti. Na slici 3 je prikazan *Ethernet Shield* koji se koristi u radu.



Slika 3. *Ethernet Shield*

Izvor: <http://arduino.cc/en/Main/ArduinoEthernetShield>

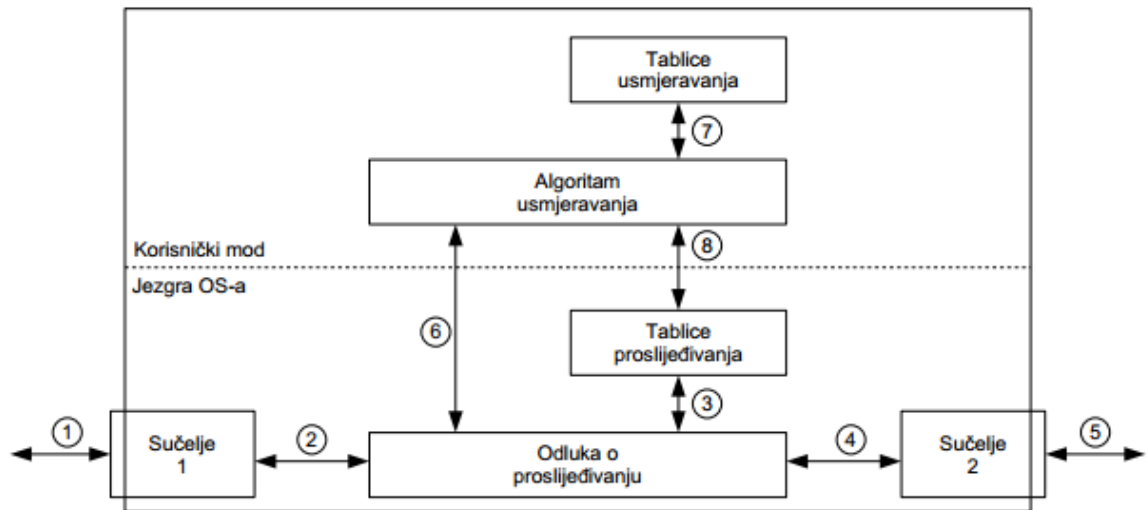
Specifikacije *Ethernet Shielda* su:

- Ne radi samostalno, potreban je priključak na Arduino pločicu
- Radni napon 5V koji pruža Arduino pločica
- Ethernet čip W5100
- Brzina do 10/100Mb-a

Arduino Ethernet Shield predstavlja jednu od ključnih komponenata u radu jer omogućuje slanje izmjerenih vrijednosti na udaljeni server. Za povezivanje sa serverom je ustvari zaslužan usmjernik koji je povezan s komponentom putem standardnog *RJ45 ethernet* priključka.

2.3.4. Usmjernik

Da bi mikrokontroler poslao podatke na udaljeni poslužitelj, potrebno je povezati lokalnu (LAN) mrežu sa svjetskom (WAN). Uređaj koji obavlja taj proces naziva se usmjernik (engl. *router*). Usmjernik prosljeđuje informacije u druge mreže putem algoritma za usmjerenje, odnosno određivanja najboljih putova paketa kroz mrežu. Karakteristika svakog usmjernika je da određenom paketu podataka osigura put od izvorišta do odredišta. U radu izvorište predstavlja Arduino mikrokontroler, a odredište je udaljeni poslužitelj. Pobljez gledano, usmjernik je specijalizirano računalo bez vanjskih periferija, ali s barem dva sučelja. Na slici 4 je prikazan pojednostavljen model usmjernika koji posjeduje dva sučelja. [4]



Slika 4. Shematski prikaz usmjernika

Izvor: Groš, Stjepan, 2006, *Mrežni Sloj Interneta*, Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva

Na usmjerniku se paralelno odvijaju dva procesa. Prvi je proces prosljeđivanja (engl. *forwarding*), a drugi je proces izgradnje tablice prosljeđivanja (engl. *forwarding information base - FIB*).

Proces prosljeđivanja paketa podataka iz mikrokontrolera na udaljeni poslužitelj opisan je u sljedećim koracima:

1. Paket pristiže na prvo sučelja koje ga prihvaća i smješta njegov sadržaj u privremenu memoriju.
2. Slijedi prihvat paketa od strane sustava prosljeđivanja.
3. Sustav prosljeđivanja pretražuje tablicu prosljeđivanja kako bi odredio što učiniti s paketom.
4. Po tablici usmjeravanja paket treba poslati kroz sučelje 2, te se paket prebacuje u privremenu memoriju drugog sučelja.
5. Drugo sučelje šalje paket na mrežu.

2.4. Programaska podrška sustava

Sustav koristi sljedeće tehnologije kao programsku podršku:

- *PHP*
- *MySQL Bazu podataka*
- *Operacijski sustav Android*

2.4.1. PHP

PHP je skriptni programski jezik koji se izvodi na poslužiteljskoj strani (engl. *server-side*), a nastao je kao ideja za izradu dinamičkih i interaktivnih *web* stranica. Razvio ga je Rasmus Lerdorf 1995. godine kako bi unaprijedio vlastitu web stranicu, te mu je iz tog razloga dao naziv: "Alat za izradu osobne stranice (engl. *Personal Home Page Tools*)". Alat je ubrzo postao projekt otvorenog koda (engl. *open source*) i time privukao pozornost drugih programera koji su ga počeli postepeno unaprjeđivati. 1997. godine mijenja naziv u *Hypertext Preprocessor* koji se koristi i danas. [5]

PHP je jedna od najnaprednijih i najkorištenijih poslužiteljskih skriptnih tehnologija danas u upotrebi. Opisuje ju mnoštvo odlika koje ju stavljaju ispred ostalih. Važno je spomenuti njezinu mogućnost povezivanja i rad s raznim bazama podataka, poput *MySQL*, *PostgreSQL*, *Sybase*, *Oracle* i dr. Potpuno je besplatna i neovisna o operacijskom sustavu. Podržava ISAPI protokol koji omogućuje razvoj aplikacijskih

sučelja. Nadalje, na PHP-u se temelji mnogo aplikacijskih okvira (engl. *Application framework*) koji omogućuju rapidni razvoj projekta.

Aplikacijski okvir koji je korišten u radu nosi naziv *SlimPHP*, a služi kao vrlo dobar temelj za izradu REST (engl. *Representational state transfer*) aplikacijskog sučelja. REST je zapravo stil programske arhitekture za izgradnju raspodijeljenih sustava. Takvi sustavi su skalabilni, nadogradivi i jednostavni za korištenje, a temelje se na jedinstvenoj uporabi univerzalne sintakse koja se koristi u hipermedijskim poveznicama zajedno s HTTP metodama kao što su *PUT*, *GET*, *POST* i *DELETE*. U navedenom isječku prikazan je primjer dohvaćanja rezultata mjerenja putem REST metode:

```
HTTP GET http://diplomski.odin.hr/v1/readings/today
```

Primjena PHP-a u radu temelji se na izradi aplikacijskog sučelja koje služi kao veza između mikrokontrolera i korisničkih uređaja. Mikrokontroler šalje podatke na poslužitelj koji ih pohranjuje u bazu podataka. Pohranjeni podaci se zatim distribuiraju na udaljene korisničke uređaje u obliku posebno formatiranog tekstualnog formata, zvanog JSON (engl. *JavaScript Object Notation*).

JSON je tekstualni format koji koristi dvije univerzalne strukture podataka (objekte i nizove) i neovisan je o programskom jeziku. Podaci su formatirani tako da svaki podatak sadrži set s ključem i vrijednosti.

```
{ "error": false,
  "readings": [
    {
      "id": "761",
      "temperature": "20.1",
      "humidity": "66.6",
      "date": "2014-04-29T23:23:58+02:00"
    }
  ]
}
```

Programski isječak 1 - Primjer JSON formata

2.4.2. Baza podataka

Baza podataka je skup međusobno povezanih podataka, pohranjenih zajedno bez štetne ili nepotrebne zalihosti, s ciljem da ih koriste različite aplikacije. Podaci su pohranjeni u obliku neovisnom od programa koji ih koriste. Unos, izmjena i dohvat podataka obavlja se isključivo kroz zajedničko i kontrolirano sučelje. [6]

O upravljanju bazom podataka brine se sustav za upravljanje bazama podataka - SUBP. SUBP provjerava, analizira i transformira zahtjeve korisničkog programa u niz operacija koje je potrebno obaviti na fizičkoj razini.

Najvažnije zadaće SUBP-a su:

- Zaštita baze podataka od neovlaštenog korištenja (engl. *Data security*).
- Osiguranje obnove podataka u slučaju uništenja (engl. *Data recovery*).
- Sprječavanje narušavanje pravila integriteta (engl. *Data integrity*).
- Mogućnost korištenja rječnika podataka

SQL predstavlja akronim za *Structured Query Language*. To je standardizirani, referentni jezik za relacijske baze podataka. Standardiziran je prema ANSI (*American National Standards Institute*), ISO (*Organization for International Standardization*), UNIX(X(O/Open), IBM Standard i FIPS (*Federal Information Processing Standard*). Temelji se na relacijskoj algebri i predikatnom računu, a značajka jezika je neproceduralnost (opisuje se što se želi dobiti kao rezultat, ali ne i kako se do tog rezultata dolazi). SQL omogućuje definiranje podataka (*Data Definition Language - DDL*), upravljanje podacima (*Data Manipulation Language - DML*) te provođenje kontrole nad podacima u relacijskoj bazi podataka. [6]

Prva verzija SQL standarda je SQL-86 koji je obuhvaćao osnovne operacije s relacijama. Standard je proširen 1989. i 1992. godine kada je ugrađen veliki broj novih elemenata jezika. SQL-99 sadrži mogućnosti poput rekurzivnih operacija, okidača i ugradnje korisnički definiranih tipova podataka. Najnovija verzija standarda jest SQL-2011.

Osnovni objekti u SQL-u su:

- Baza podataka (engl. *Database*)
- Tablica (engl. *Table*)
- Stupac (engl. *Column*);
- Pogled ili virtualna tablica (engl. *View*)
- Sinonim (engl. *Synonym*)
- Pravili integriteta (engl. *Constraint*)
- Index (engl. *Index*)
- Pohranjena procedura (engl. *Stored Procedure*)
- Okidač (engl. *Trigger*)

Primjer SQL naredbe iz DDL dijela jezika (kreiranje relacije "readings"):

```
CREATE TABLE readings(  
    id INTEGER NOT NULL,  
    temperature FLOAT,  
    humidity FLOAT,  
    date DATETIME  
);
```

Programski isječak 2 - Primjer SQL naredbe DDL dijela jezika

Primjer SQL naredbe iz DML dijela jezika (iz relacije "readings" dohvaća sve n-torke kojima je datum zapisa jednak današnjem datumu):

```
SELECT * FROM readings WHERE DATE(datetime) = CURDATE();
```

Programski isječak 3 - Primjer SQL naredbe iz DML dijela jezika

2.4.3. Operacijski sustav Android

Android je najpopularnija platforma za mobilne uređaje. Temelji se na skupini međusobno povezanih podsustava koji zajedno čine funkcionalno rješenje. Skupinu čine operacijski sustav temeljen na modificiranoj verziji Linux operacijskog sustava, programska međuoprema (eng. *middleware*) koja povezuje operacijski sustav s aplikacijama, te aplikacijski okvir (eng. *application framework*). [7]

Razvoj Androida započela je tvrtka *Android, Inc.* u listopadu 2003. godine. Nakon dvije godine, tvrtka Google je odlučila kupiti Android i preuzeti daljnji razvoj. Cilj je bio razviti mobilnu platformu koja će u potpunosti biti prilagodljiva zahtjevima korisnika. Prva stabilna verzija izlazi 2009. godine i nosi naziv "*Cupcake*". Od tada svaka nova verzija Androida donosi sve veću zastupljenost na tržištu, a prema tradiciji nose nazive raznih desertnih jela. Danas najnovija verzija 4.4 nosi ime "*KitKat*" što implicira na *Google-ovo* partnerstvo sa švicarskom tvrtkom *Nestle*.

Važnije karakteristike Android platforme su:

- Otvoreni kod (eng. *Open source*) - omogućuje proizvođaču mobilnih uređaja slobodno korištenje i prilagodbu sustava.
- Multimedia - kvalitetni prikaz optimizirane 3D grafike temeljene na *OpenGL ES* specifikacijama, ugrađeni kodeci za reproduciranje audio i video zapisa.
- Kompatibilnost s različitim uređajima -
- Podrška za kameru, *GPS*, kompas, akcelerometar, *Bluetooth*, *EDGE*, *3G*, *WiFi* i *GSM*.
- Virtualni stroj u kojem se izvršavaju aplikacije, dizajniran posebno za mobilne uređaje.
- Upravljanje memorijom - ako se aplikacija izvršava u pozadini, ali se ne koristi, sustav će sam završiti njezin proces i tako osloboditi memoriju.

Android aplikacije su pisane u programskom jeziku Java i izvezene u jedinstveni paket koji sadrži datoteke potrebne za izvršavanje na mobilnom uređaju. Svaka aplikacija ima jedinstveni broj i pokreće se u vlastitom Linux procesu. Operacijski sustav pokreće proces kada se neka komponenta aplikacija treba izvršiti, a isključuje proces kada izvršavanje komponente više nije potrebno ili je potrebno osloboditi memoriju. [7]

Arhitektura Androida (slika 5) se sastoji od pet zasebnih cjelina od kojih svaka ima određenu funkciju unutar operacijskog sustava.



Slika 5. Arhitektura operacijskog sustava Android

Izvor: http://elinux.org/Android_Architecture

Na dnu se nalazi Linux 2.6 jezgra (engl. *kernel*) koja služi kao apstraktni sloj između sklopovlja i programske podrške. Jezgra sadrži razne *drivere* među kojima je najvažniji *driver* za međuprocesnu komunikaciju. Sloj iznad jezgre sadrži izvorne biblioteke (engl. *native libraries*) koje su pisane u programskim jezicima C i C++.

Važnije biblioteke su:

- *System C library* – implementacija standardne C systemske biblioteke
- *Media Framework* – biblioteka koja omogućuje reprodukciju audio i video zapisa.
- *Surface Manager* – biblioteka koja omogućuje prikaz više različitih 2D i 3D grafičkih slojeva na ekranu.
- *LibWebCore* – jezgra web-preglednika.
- SGL (engl. *Scalable Graphics Library*) – 2D biblioteka korištena za većinu aplikacija.
- *FreeType* – biblioteka koja omogućuje renderiranje različitih fontova.
- *SQLite* – biblioteka za upravljanje bazama podataka.
- Upravitelj aktivnosti (engl. *Activity Manager*) - održava životni ciklus aplikacije.
- Pružatelj sadržaja - komponente za enkapsulaciju podataka unutar aplikacija.
- Upravitelj resursima (engl. *Resource Manager*) - brine o podacima koji se nalaze unutar paketa aplikacije.
- Sustav elemenata prikaza (engl. *View System*) - komponente koje se koriste za izradu vizualnog grafičkog sučelja.
- Upravitelj obavijestima (engl. *Notification Manager*) - komponente koje se koriste za obavješćavanje korisnika o određenim događajima u mobilnom uređaju.

2.4.4. XML

Extensible Markup Language je tekstualni format prvobitno namijenjen za pohranu i prijenos podataka. Prva radna verzija izrađena je 1996. godine, a svoju široku primjenu pronašla je u povezivanju s raznim tehnologijama i programskim jezicima. Iako je po strukturi gotovo identičan HTML-u, funkcija XML-a je u potpunosti različita (HTML je dizajniran za prikaz, a XML isključivo za pohranu i prijenos podataka). Struktura se sastoji od hijerarhijskog skupa elemenata u kojem svaki element sadrži određenu vrijednost i dodatne attribute, čime se pojednostavljuje obrada i procesiranje na strani tehnologije koja je zaprimila datoteku. Osnovnu strukturu podataka čine liste i stabla podataka. U priloženom programskom isječku prikazan je primjer prijena podataka mjerenja u XML formatu.

```
<mjerenje>
  <id>761</id>
  <temperatura>20.1</temperatura>
  <vlaga>66.5</vlaga>
  <datum>2014-04-29T23:23:58+02:00</datum>
</mjerenje>
```

Programski isječak 4. Primjer XML formata

XML ima važnu ulogu u izradi Android aplikacija. U njemu su pisane datoteke za lokalizaciju i pohranu podataka, razne konfiguracijske datoteke, a najistaknutija primjena je izrada korisničkog sučelja. Za izradu sučelja koriste se unaprijed definirani elementi s različitim atributima. Raspored elemenata organiziran je u određene formate (engl. *layouts*) u kojima svaki element odgovara određenom pogledu (engl. *view*) odnosno objektu, čija podatkovna struktura u sebi sadrži zapis izgleda određenog pravokutnog područja na zaslonu. Primjer takvog objekta prikazan je u programskom isječku 5.

```
<TextView
  android:id="@+id/min_num_txt"
  android:layout_width="match_parent"
  android:layout_height="wrap_content"
  android:textSize="15dp"
/>
```

Programski isječak 5. Primjer dijela korisničkog sustava u Android aplikaciji

3. PRAKTIČNI DIO

3.1. Opći i specifični ciljevi rada

Cilj ovog rada jest osmisliti i fizički ostvariti sustav koji će od udaljenog mjesta mjerenja korisniku omogućiti pregled rezultata mjerenja s napomenom da korisnik može u bilo kojem trenutku i na bilo kojem uređaju s mrežnom komunikacijom (računalo, *tablet*, *smartphone*) pristupiti podacima. Sustav za udaljeni nadzor je samostalan i neovisan o korisničkom uređaju, a za postavljanje na određeno mjesto mjerenja mora imati osiguran izvor napajanja putem standardne naponske mreže te pristup internetu (usmjernik s pristupom globalnoj mreži). Postavljanje sustava je iznimno jednostavno; uređaj za mjerenje se postavi na željeno mjesto te se na njega poveže usmjernik i izvor napajanja. Nakon toga sustav počinje s mjerenjem temperature i vlage zraka, a svakih pet minuta podatke šalje poslužitelju.

Pobliže gledano, sustav objedinjuje tri manja podsustava od kojih svaki ima određenu funkciju:

- Mjerni sustav - mjeri temperaturu i vlažnost zraka i rezultate šalje na poslužitelj.
- Mrežni sustav - prikuplja i obrađuje podatke te ih distribuira na korisničke uređaje.
- Korisnički sustav - aplikacije za prikaz rezultata mjerenja.

Prema spomenutim navodima deklarirani su sljedeći ciljevi rada:

- Opći cilj rada - osmisliti sustav za udaljeni nadzor temperature i vlage zraka, te ostvariti fizički prototip takvog sustava.
- Konačni specifični cilj rada - izraditi programsku podršku za obradu i pohranu rezultata mjerenja, izraditi korisničke aplikacije za pregled rezultata te ispitati pouzdanost sustava putem eksperimenta koji se temelji na uspoređivanju dobivenih rezultata s referentnim vrijednostima prikupljenim drugim sustavom.

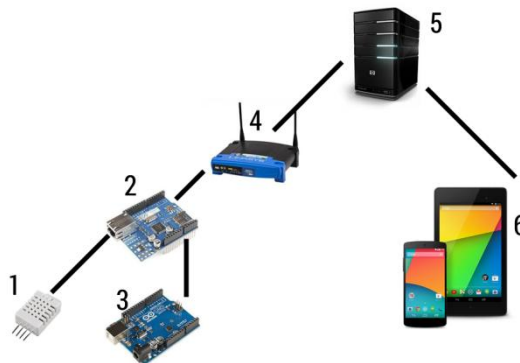
Važno je spomenuti i da, osim navedenih ciljeva, postoji još jedan cilj u pozadini koji se odnosi na povezivanje više različitih tehnologija u jedinstvenu, funkcionalnu cjelinu. Zanimljiva je činjenica da su sve korištene tehnologije otvorenog koda (engl. *open source*) što u krajnjem slučaju implicira na modularnost i proširivost čitavog sustava.

3.2. PLAN RADA

Kako bi se dobio detaljni pregled i poboljšala efikasnost i organiziranost tijekom izrade sustava, izrađen je plan rada koji sadrži sve potrebne stavke za izradu istog. Plan rada je sljedeći:

- Povezivanje fizičkih komponenata u mjernom podsustavu.
- Povezivanje mjernog podsustava s usmjernikom.
- Izrada programskog rješenja za Arduino mikrokontroler koje prikuplja podatke o temperaturi i vlazi iz okoline.
- Izrada baze podataka u kojoj će se rezultati pohranjivati.
- Izrada programske podrške na strani poslužitelja.
- Izrada korisničke web aplikacije.
- Izrada korisničke Android aplikacije.
- Povezivanje aplikacija s poslužiteljem.
- Testiranje funkcionalnosti sustava, traženje i ispravljanje pogrešaka.
- Ispitivanje pouzdanosti sustava - eksperimentalni dio.

Slika 6 prikazuje shematski prikaz sustava u kojemu su povezani DHT22 senzor (1) s *Ethernet Shieldom* (2) i Arduino mikrokontrolerom (3). Mikrokontroler šalje rezultate mjerenja preko usmjernika (4) do udaljenog poslužitelja (5) koji pohranjuje i obrađuje podatke i prosljeđuje ih korisničkim uređajima (6).



Slika 6. Shematski prikaz sustava

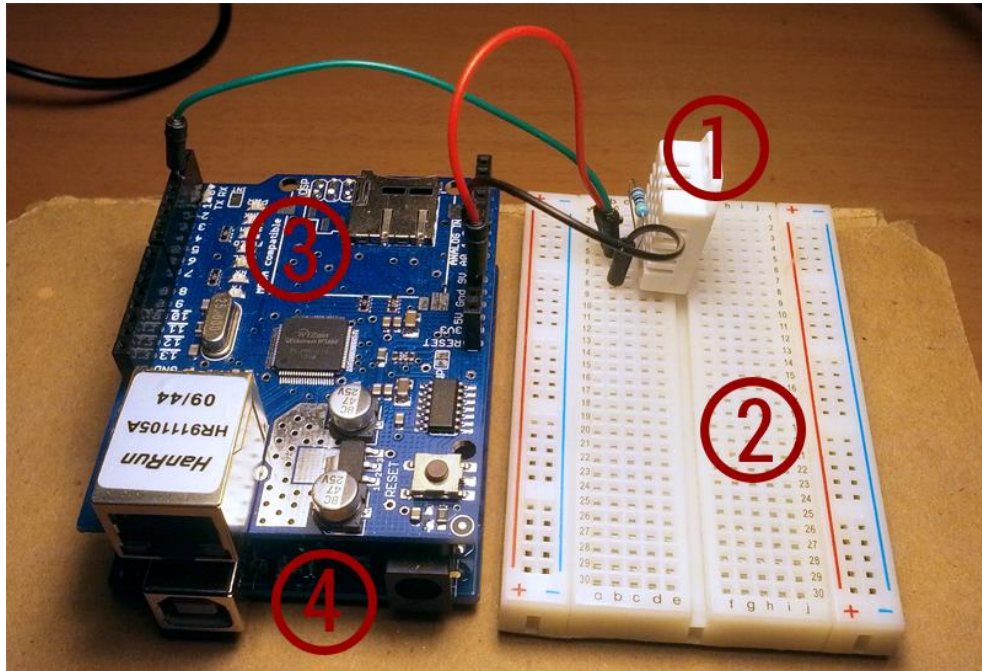
Izradio: Alexander Štimac

3.2.1. Povezivanje fizičkih komponenata u mjernom podsustavu

Mjerni podsustav sastoji se od senzora za mjerenje temperature i vlage, Arduino mikrokontrolera i mrežne komponente *Ethernet Shield*. Senzoru je potreban radni napon od 3 do 5 V kojeg dobiva iz mikrokontrolera, a uz njega je spojen otpornik od 4.7 k Ω . Izmjereni podaci šalju se mikrokontroleru putem Data Pin žice. *Ethernet shield* komponenta povezana je s mikrokontrolerom preko priključnih pinova. Komponenta je povezana sa usmjernikom putem *crossover* kabela, a mikrokontroler je spojen na standardnu naponsku mrežu putem USB kabela i priključnog adapter kojemu je izlazni napon 5V i jakost struje 1A.

Slika 7 prikazuje mjerni podsustav u kojem su:

1. DHT Senzor za mjerenje temperature i vlage zraka.
2. Proto pločica
3. *Ethernet Shield* komponenta
4. Arduino mikrokontroler



Slika 7. Mjerni podsustav

Izradio: Alexander Štimac

3.2.2. Izrada programskog rješenja za Arduino mikrokontroler

Programsko rješenje temelji se na DHT biblioteci koja omogućuje očitavanje izmjerenih podataka s priključenog senzora na vrlo jednostavan način. Suština rješenja nalazi se u funkciji `loop()` koja se, kako joj i samo ime nalaže, izvršava uzastopno.

```
void loop() {
    float h = dht.readHumidity();
    float t = dht.readTemperature();

    if (isnan(t) || isnan(h)) {
        Serial.println("Greška prilikom očitavanja DHT");
    } else {
        postData(t, h);
    }

    delay(300000);
}
```

Programski isječak 6. Očitavanje i slanje izmjerenih vrijednosti

U funkciji (Programski isječak 6) se uzimaju očitane vrijednosti temperature i relativne vlažnosti sa senzora. Vrijednosti se dodatno provjeravaju, te se u slučaju pogreške ispisuje određena poruka. U suprotnom se vrijednosti šalju na poslužitelj. Nakon što su vrijednosti poslane, cijeli sustav miruje 300000 milisekundi, odnosno 5 minuta i nakon toga funkcija kreće iz početka.

Za slanje podataka na udaljeni poslužitelj zaslužna je *Ethernet Shield* komponenta i njoj pripadajuća biblioteka. U nastavku je priložen programski isječak 7 koji prikazuje povezivanje i slanje vrijednosti na server. Provjerava se da li je uspostavljena konekcija sa serverom. Ukoliko postoji, funkcija generira odgovarajuću HTTP POST metodu koja sadrži odgovarajuće vrijednosti i dodatne *Header* informacije. U slučaju da veza nije uspostavljena, ispisuje se odgovarajuća poruka.


```


if (client.connect(server, 80)) {
    client.println("POST /v1/sensor.php HTTP/1.1");
    client.println("Host: diplomski.odin.hr");
    client.println("User-Agent: Arduino/1.0");
    client.println("Connection: close");
    client.println("Content-Type: application/x-www-form-
urlencoded;");
    client.print("Content-Length: ");
    client.println(yourdata.length());
    client.println();
    client.println(yourdata);
    client.stop();
}
else {
    Serial.println("Connection failed");
    client.stop();
}

```

Programski isječak 7. Slanje izmjerenih vrijednosti na poslužitelj

3.2.4. Izrada baze podataka

Bazu podataka čini tablica koja sadrži četiri atributa. Identifikacijski broj se automatski inkrementira prilikom svakog zapisa, a zapis sadrži informacije o temperaturi, relativnoj vlažnosti zraka i vremenu u kojem su vrijednosti izmjerene (slika 8).

readings	
 id	int
temperature	float
humidity	float
reading_datetime	datetime

Slika 8. Shematski prikaz baze podataka

Izradio: Alexander Štimac

3.2.3. Izrada programske podrške na strani poslužitelja

Programska podrška na poslužitelju predstavlja jedan od najvažnijih čimbenika u cjelokupnom radu. Većina funkcionalnosti odrađuje se upravo u tom dijelu. Osim što se brinu o pohrani rezultata mjerenja, skripte na serveru su također zadužene za manipulaciju i obradu podataka, te za kompliciranije funkcionalnosti kao što su slanje *Push* notifikacija na korisničke uređaje i generiranje izvještaja o mjerenju. Kao što je ranije spomenuto, skripte su pisane u PHP programskom jeziku unutar *SlimPHP* aplikacijskog okvira.

Prema dosadašnjem radu implementirane su sljedeće funkcionalnosti:

- Dohvaćanje trenutnih rezultata s mjesta mjerenja.
- Pohrana rezultata mjerenja u bazu podataka.
- Izračun minimalne, maksimalne i srednje vrijednosti temperature i vlage zrake za određeni datum.
- Ispis svih rezultata mjerenja za određeni datum.
- Slanje GCM obavijesti na mobilne uređaje.
- Slanje E-mail obavijesti.
- Izrada izvještaja mjerenja te izvoz u Excel datoteku.

Dohvaćanje trenutnih rezultata s mjesta mjerenja

Za trenutni rezultat smatra se onaj koji je zadnji unesen u bazu podataka. Budući da se identifikacijski broj automatski inkrementira tijekom svakog zapisa, zadnji zapis je onaj s najvećim ID-em. Prema tome, moguće je napraviti upit na bazu tako da se dohvati redak s najvećim identifikacijskim brojem (Programski isječak 8):

```
SELECT * FROM sensor ORDER BY id DESC LIMIT 1;

$app->get('/readings/new', function() {
    $db = new DbHandler();
    $result = $db->getLastReading();
    echoResponse(200, $result);
});
```

Programski isječak 8. Upit na bazu kojim se dohvaća zadnje unesena vrijednost

Pohrana rezultata mjerenja u bazu podataka

Nakon što mikrokontroler pošalje podatke na server, poziva se priložena funkcija (*Programski isječak 9*) pod pretpostavkom da postoji validirana konekcija s bazom podataka. Funkcija kao parametre prima vrijednosti temperature i vlage, a trenutni se datum generira automatski. Ukoliko postoji konekcija s bazom, podaci se zapisuju u nju, a u suprotnom se ispisuje poruke po pogrešci.

```
public function insertSensorReadings($temp, $humidity) {
    $stmt = $this->conn->prepare("INSERT INTO sensor
    (temperature, humidity, date) values(?, ?, ?)");
    $datetime = date('Y-m-d H:i:s');
    $stmt->bind_param("dds", $temp, $humidity, $datetime);
    $result = $stmt->execute();
    if (false === $result) {
        die('execute() failed: ' . htmlspecialchars($stmt->
        error));
    }
    $stmt->close();
    return $result;
}
```

Programski isječak 9. Pohrana izmjerenih vrijednosti u bazu podataka

Minimalne, maksimalne i srednje vrijednosti

Navedene vrijednosti moguće je dohvatiti putem SQL upita na tablicu s atributima u bazi podataka. U priloženom programskom isječku (10) prikazan je primjer takvog upita koji dohvaća minimalne, maksimalne i srednju vrijednosti očitavanja tijekom proteklih tjedan dana.

```
SELECT date(datetime) as reading_date, MIN(temperature) AS
min_temp, MIN(humidity) AS min_hum, MAX(temperature) AS
max_temp, MAX(humidity) AS max_hum, AVG(temperature) AS avg_temp,
AVG(humidity) AS avg_hum
FROM sensor_readings
WHERE date(datetime) >= subdate(CURDATE(), INTERVAL 1 WEEK)
GROUP BY reading_date;
```

Programski isječak 8. Upit koji dohvaća minimalnu, maksimalnu i srednju vrijednost

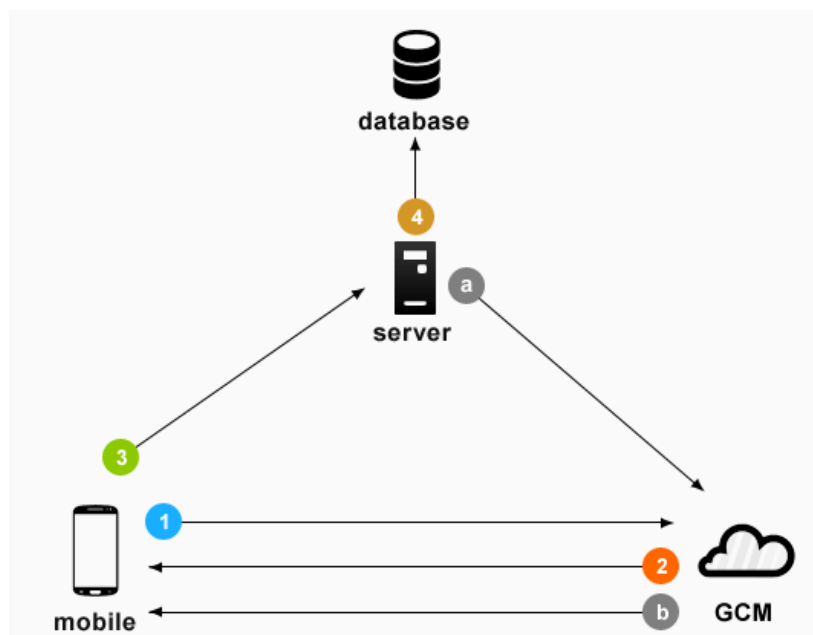
Slanje E-mail Obavijesti

Slanje obavijesti putem elektronske pošte predstavlja jedan od načina za isporuku rezultata. Omogućuje korisniku uvid u trenutne rezultate mjerenja i zaprimanje izvještaja na dnevnoj ili tjednoj bazi.

Slanje GCM Obavijesti

Google Cloud Messaging je usluga koja omogućuje slanje informacija s poslužitelja na korisničke mobilne uređaje. Potpuno je besplatna i nema ograničeni broj poruka koje se mogu poslati. Upravo su to čimbenici koji uslugu čine idealnim rješenjem za obavješavanje korisnika. Poruke se šalje u realnom vremenu, što znači da će korisnik uvijek biti u toku s izmjerenim vrijednostima temperature i vlage zraka.

Arhitektura usluge (slika 9) sastoji se od poslužitelja, korisničkog uređaja i GCM Servera. Da bi se pretplatio na primanje GCM obavijesti, korisnički uređaj treba registrirati svoj jedinstveni broj uređaja na GCM Server (1, 2), a zatim taj broj sinkronizirati s udaljenim poslužiteljem (3) koji ga pohranjuje u bazu podataka (4). Poslužitelj generira određenu obavijest i šalje ju GCM serveru (a) koji ju prosljeđuje registriranom uređaju (b).



Slika 9. Arhitektura GCM usluge

Izvor: <http://www.androidhive.info/2012/10/android-push-notifications-using-google-cloud-messaging-gcm-php-and-mysql/>

Generiranje izvještaja mjerenja

Za generiranje izvještaja i izvoz u datoteku korištena je biblioteka *PHPExcel* koju je bilo potrebno uključiti u skriptu na poslužitelju. Biblioteka sadrži sve funkcije koje su potrebne za izradu izvještaja. Primjer generiranja prikazan je u programskom isječku (11), a temelji se na iteriranju kroz niz dohvaćenih vrijednosti. Niz sadrži podatke u obliku ključ - vrijednost te se tijekom iteracije provjerava ključ, a vrijednost se zapisuje u određenu ćeliju u Excel datoteku. Generirani izvještaj je zatim moguće pohraniti na računalo i krenuti s daljnjom obradom podataka (računanje vrijednosti, izrada grafičkog prikaza i drugo).

```
$arr = $db->getTodaysReadingsForExcel();
foreach ($arr as $key => $val) {
    foreach ($val as $k => $v) {
        if(strcmp($k,"temperature") == 0){
            $objPHPExcel->getActiveSheet()->SetCellValue('A'.$i, $v);
        }
        else if(strcmp($k,"humidity") == 0){
            $objPHPExcel->getActiveSheet()->SetCellValue('B'.$i, $v);
        }
        else if(strcmp($k,"date") == 0){
            $objPHPExcel->getActiveSheet()->SetCellValue('C'.$i, $v);
        }
        else{
            //no op
        }
    }
    $i++;
}
```

Programski isječak 9. Generiranje izvještaja

Priložena slika 10 prikazuje primjer generiranog izvještaja koji sadrži izmjerene vrijednosti temperature, vlage zrake te vrijeme zapisa.

A	B	C
TEMPERATURA	VLAGA	DATUM
23,2	53,7	2014-05-30T00:00:03+02:00
23,1	54,4	2014-05-30T00:05:05+02:00
23,1	54,2	2014-05-30T00:10:07+02:00
23,2	53,8	2014-05-30T00:15:08+02:00
23,2	53,9	2014-05-30T00:20:10+02:00
23,1	54,2	2014-05-30T00:25:13+02:00
22,9	54,8	2014-05-30T00:30:15+02:00
22,7	55,6	2014-05-30T00:35:17+02:00
22,6	55,9	2014-05-30T00:40:18+02:00
22,4	56,2	2014-05-30T00:45:22+02:00

Slika 10. Primjer izvještaja u Excel-u

3.2.4. Izrada Android mobilne aplikacije

Android mobilna aplikacija predstavlja komponentu koja omogućuje udaljeni nadzor temperature i vlage zraka. Sadrži osnovne funkcionalnosti koje pružaju uvid u trenutne rezultate mjerenja. Budući da je aplikacija povezana s poslužiteljem, korisnik se može nalaziti na bilo kojoj lokaciji, a svejedno će dobivati nove informacije u realnom vremenu pod uvjetom da je mobilni uređaj povezan na Internet. Takav pristup zapravo objašnjava pojam "udaljeni nadzor", a ujedno i olakšava posao djelatnika koji je zadužen za praćenje rezultata mjerenja. Aplikacija je osmišljena da radi kao podsjetnik ili alarm koji će automatski obavijestiti korisnika kada dođe do promijene temperature ili vlage zraka. Princip rada navedene funkcionalnosti temelji se GCM (engl. *Google Cloud Messaging*) obavijestima koje šalje poslužitelj. Princip je sljedeći: Senzor mjeri vrijednosti i šalje ih mikrokontroleru koji ih prosljeđuje poslužitelju. Poslužitelj provjerava podatke i, ukoliko je došlo do promjene, šalje obavijest mobilnom uređaju da osvježi prikaz podataka u aplikaciji. Osim spomenutih obavijesti, aplikacija nudi razne funkcionalnosti za detaljniji pregled rezultata. Neke od njih su prikaz minimalne, maksimalne i prosječne vrijednosti tijekom dana, pretvorba temperature skale (npr. iz Celsiusove skale u Kelvinovu) te grafički prikaz podataka. Na slici 11 je prikazan trenutni izgled sučelja na mobilnom uređaju.



Slika 11. Prikaz mobilne aplikacije

Dohvaćanje podataka sa poslužitelja

Za dohvaćanje i ažuriranje podataka s udaljenog poslužitelja koristi se biblioteka za mrežnu komunikaciju pod nazivom *Volley*. *Volley* je osmišljen u Google-u, a omogućuje brzu i jednostavnu razmjenu informaciju između mobilnog uređaja i poslužitelja. Informacije su formatirane u JSON tekstualnom obliku. Primjer jedne od funkcija za dohvaćanje podataka je priložen u programskog isječku 12.

```
getJSON("/readings/new", ApiRequest.class, new Listener
<ApiRequest>() {
@Override
public void onResponse(ApiRequest response) {
    if(response != null && ! response.getReadings().isEmpty()){
        listener.onSuccess(response.getReadings().get(0));
    } else {
        listener.onFailure();
    }
}, new ErrorListener() {
@Override
public void onErrorResponse(VolleyError error) {
    listener.onFailure();
}
});
```

Programski isječak 10. Dohvat JSON formata

Razlika između native i web aplikacije

Izrada aplikacije uključuje planiranje i osmišljavanje strukture koja objedinjuje procese izrade u jednu funkcionalnu cjelinu. Prije same izrade aplikacije važno je razmotriti koje će sve mogućnosti i funkcionalnosti biti uključene u razvoj aplikacije. Funkcionalnosti predstavljaju važne čimbenike koji određuju odabir ciljanih tehnologija i implementaciju istih. Native aplikacije su namijenjene specifičnoj platformi za koju su predviđene odabrane tehnologije. Prednost takvih aplikacija je bolje korisničko iskustvo, efikasnost i korištenje svih dostupnih mogućnosti unutar platforme. Glavni nedostatak predstavlja utrošak vremena budući da se razvoj aplikacije procesira zasebno za svaku platformu. S druge strane, hibridne aplikacije su osmišljene da se jednoliko izvršavaju na svim platformama, čime se štedi vrijeme i novac. To su ustvari web

aplikacije koje je moguće pokretati direktno iz *web* preglednika na mobilnom uređaju. Međutim, takve aplikacije imaju ograničenu funkcionalnost što dovodi u pitanje njihovu korisnost i svrhu.

U ovom radu je nativna aplikacija stavljena u prvi plan. Glavni razlog tome jest potreba za funkcionalnim sustavom obavještanja korisnika na daljinu. Kao najbolje rješenje pokazala se GCM usluga koja je implementirana u aplikaciju, a veže se direktno za korisnički uređaj. Time je omogućeno slanje obavijesti o trenutnim rezultatima mjerenja temperature i vlage zraka svim korisnicima koji su registrirali svoj uređaj na GCM uslugu. S druge strane, *web* aplikacija ne pruža takvu mogućnost, već je kao kompenzacijsko rješenje implementirana funkcionalnost slanja izmjerenih rezultata putem elektronske pošte.

4. EKSPERIMENTALNI DIO

U sklopu ovog rada provedeno je kvantitativno istraživanje u kojem se ispitivala funkcionalnost i pouzdanost izrađenog sustava. Tijekom istraživanja prikupljali su se primarni podaci, odnosno, fizikalne veličine prikupljene metodom izravnog mjerenja. Mjerenje vrijednosti provedeno je tijekom sedam dana, pri čemu su se vrijednosti očitavale svakih pet minuta. Takvim pristupom dobiven je konačan skup uzoraka od ukupno 2016 elemenata, gdje svaki element sadrži izmjerene vrijednosti temperature i relativne vlage zraka. Prikupljeni podaci podvrgnuti su sistemskom uzorkovanjem, odnosno, uzimanjem svakog n-tog elementa iz konačnog skupa. Sistemsko uzorkovanje omogućuje sortiranje elemenata po satima ili danima. Budući da su vrijednosti mjerene svakih pet minuta, moguće je izračunati vrijednosti za određene sate ili dane. Svaki 12-ti element sadrži vrijednosti za pojedini sat, a svaki 288-ti element sadrži vrijednosti za jedan dan. Mjerenje je izvršeno u prostoriji u kojoj su temperatura i vlaga zraka relativno stabilne, odnosno, ne postoji uzrok koji omogućuje njihovo drastično povećanje ili smanjenje. Iz tog razloga važno je provesti istraživanje o funkcionalnosti sustava, kako bi se utvrdilo da ne postoje velike promjene i odstupanja između dva mjerenja.

Drugi razlog takvog istraživanja je provjera vremena ispitivanja stanja. Sustav je zamišljen da radi besprekidno, međutim, on ovisi o vanjskim čimbenicima koji utječu na njegovu funkcionalnost. Neki od uzroka nepravilnog funkcioniranja sustava mogu biti nestanak struje u prostoriji ili nemogućnost povezivanja na usmjernik. Nakon ispitivanja funkcionalnosti sustava, proveden je drugi dio istraživanja koji se temelji na ispitivanju pouzdanosti izmjerenih vrijednosti.

U prostoriji je uz sustav bio postavljen uređaj koji je paralelno mjerio i zapisivao vrijednosti. Navedene vrijednosti predstavljaju točne, pouzdane reference s kojima su uspoređeni prikupljeni uzorci.

Iz navedenih činjenica formirane su dvije hipoteze:

- Sustav će svoju funkciju u smislu mjerenja i pohrane vrijednosti obnašati bespriječno i besprekidno, a rezultati mjerenja biti će dostupni korisnicima s bilo kojeg nadzornog mjesta.
- Sustav će mjeriti točne, pouzdane vrijednosti koje u usporedbi s referentnim vrijednostima imaju moguća odstupanja 0.5°C za temperaturu i 5% za relativnu vlažnost zraka.

5. REZULTATI

5.1. Ispitivanje pouzdanost sustava

U eksperimentu je prikupljeno ukupno 288 uzoraka vrijednosti temperature i vlage zraka koji su paralelno uzimani s mjernog podsustava i referentnog uređaja. Broj uzoraka odgovara broju mjerenju u jednom danu. Izmjerene vrijednosti podvrgnute su statističkoj analizi kako bi se utvrdilo odstupanje, a time i pouzdanost izrađenog sustava. Odstupanje je izračunato proračunom slučajnih pogrešaka prema zakonima vjerojatnosti i statistike.

Kako bi se izračunala apsolutna i relativna pogreška, potrebno je najprije izračunati srednju vrijednost (Jednadžba 6) niza mjerenja izrađenog sustava i referentnog uređaja. Aritmetička srednja vrijednost izražava se kvocijentom zbroja članova i brojem broja članova nekog niza, definira se izrazom:

$$\bar{x} = \frac{\sum_{i=1}^n x_n}{n} \quad (6)$$

gdje je:

\bar{x} – srednja vrijest

n – broj članova nekog niza

Prema navedenoj definiciji dobivene su sljedeće vrijednosti:

Tablica 2. Aritmetička sredina izmjerenih vrijednosti

Aritmetička sredina	DHT22 senzor (sustav)	Aria G25 (referentni uređaj)
Temperatura	22.96°C	23.66°C
Relativna vlaga zraka	60.9%	53.85%

Iz priložene tablice može se zaključiti da su odstupanja prilično velika, iako su se oba uređaja nalazila u istoj prostoriji, a vrijednosti su mjerene paralelno u isto vrijeme. *Aria*

G25 predstavlja referentni uređaj, što znači da su njegove vrijednosti uzete u obzir kao točne i precizne. Prema tome, potrebno je izračunati apsolutnu i relativnu pogrešku izrađenog sustava, te pokušati odrediti korekciju.

5.1.1. Apsolutna pogreška

Apsolutnom pogreškom se nazivaju sva odstupanja koja su vezana uz nesavršenost opreme, mjernog postupka te pogreške osobe koja mjeri. Definiira se izrazom:

$$\text{apsolutna pogreška} = \text{izmjerena vrijednost} - \text{prava vrijednost} \quad (7)$$

Za pravu vrijednost uzeta je aritmetička srednja vrijednost niza mjerenja referentnog uređaja (temperatura 23.66°C i vlaga 53.85%), dok je za izmjerenu vrijednost uzeta srednja vrijednost niza mjerenja izrađenog sustava (temperatura 22.96°C, vlaga 60.9%).

Prema definicija izračunata je apsolutna pogreška, koja iznosi:

Apsolutna pogreška temperature:

$$p_a = T_{iz} - T_{pr} = 22.96^\circ\text{C} - 23.66^\circ\text{C} = -0.7^\circ\text{C}$$

Apsolutna pogreška relativne vlage zraka:

$$p_a = H_{iz} - H_{pr} = 60.9\% - 53.85\% = 7.05\%$$

Iz apsolutne vrijednosti je izvedena korekcija. Korekcija je vrijednost koju je potrebno dodati izmjerenoj vrijednosti kako bi se dobila prava vrijednost mjerene veličine, a ima istu vrijednost kao apsolutna pogreška sa suprotnim predznakom. Prema tome, korekcija temperature iznosi:

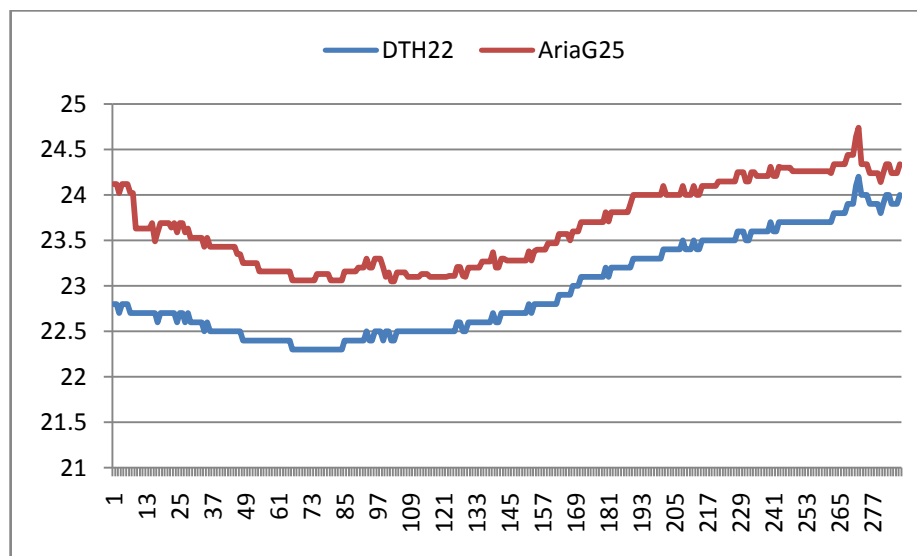
$$k_T = 0.7^\circ\text{C}$$

Vidljivo je da izračunata vrijednost odstupa od dozvoljene vrijednosti koja je pretpostavljena na maksimalno 0.5°C.

Korekcija relativne vlage zraka iznosi:

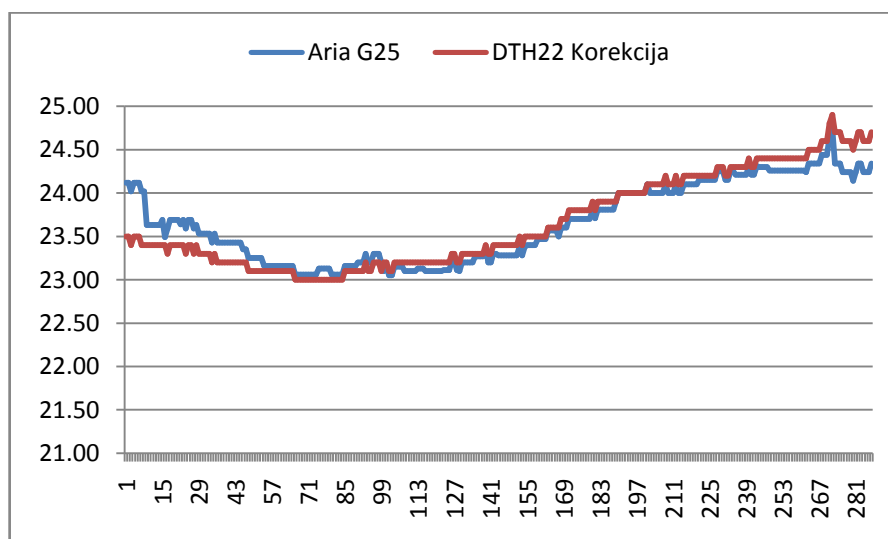
$$k_H = 7.05\%$$

Slika 12 prikazuje grafički prikaz vrijednosti temperature izmjerenih izrađenim sustavom i referentnim uređajem. Os X prikazuje broj mjerenja, a os Y prikazuje temperaturu izraženu u °C. Vidljivo je prilično veliko odstupanje vrijednosti temperatura pa se iz tog razloga ne može reći da je sustav dovoljno pouzdan.



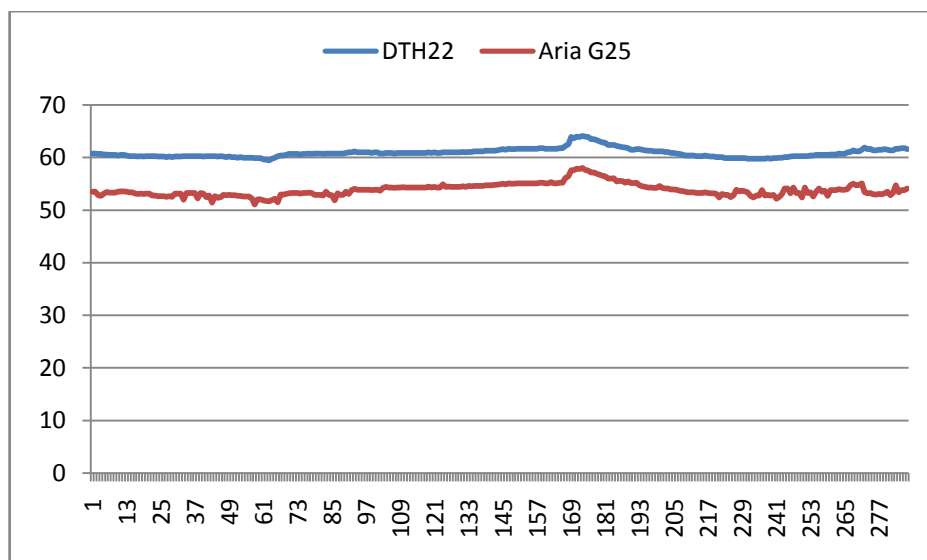
Slika 12. Prikaz temperature i vlage zrake - stvarne vrijednosti

Slika 13 prikazuje grafički prikaz vrijednosti temperature nakon korekcije. Vrijednostima izmjerenim DTH22 senzorom dodana je izračunata korekcija od 0.7°C čime se odstupanje smanjilo, a pouzdanost povećala.



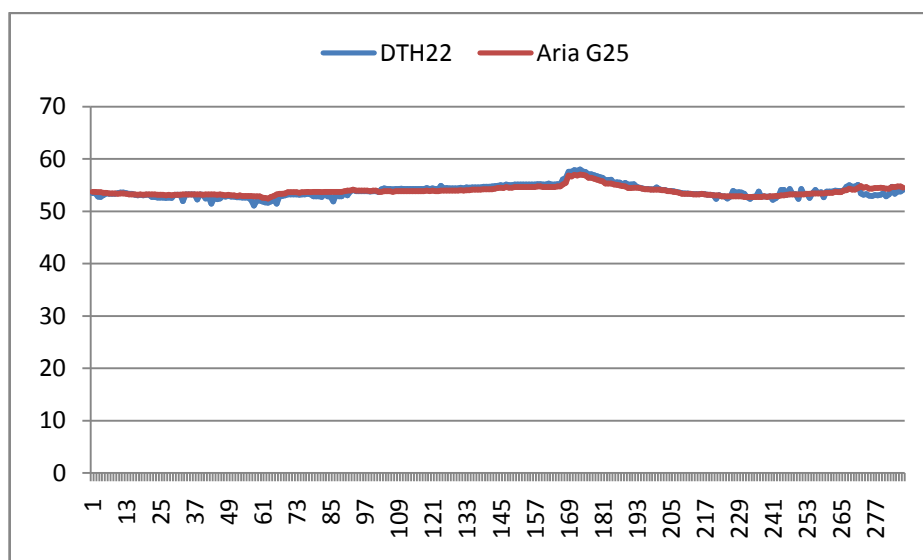
Slika 13. Temperatura - apsolutna pogreška - korekcija

Slika 14 prikazuje grafički prikaz vrijednosti vlage zrake izmjerene izrađenim sustavom i referentnim uređajem. Kao i kod temperature, vidljivo je odstupanje koje prelazi dopuštenu granicu od 5%, stoga se ne može reći da je mjerenje vlage pouzdano.



Slika 14. Vлага zraka DHT22 i Aria G25

Na slici 15 prikazane su vrijednosti izmjerene vlage zraka nakon korekcije. Korekcija je u ovom slučaju iznosila 7.05% i time se odstupanje drastično smanjilo. Sustav prikazuje gotovo identične vrijednosti kao referentni uređaj, stoga se može zaključiti da su vrijednosti nakon korekcije pouzdane.



Slika 15. Relativna vlaga - apsolutna pogreška - korekcija

5.1.2. Relativna pogreška

Relativna pogreška (Jednadžba 8) se koristi za određivanje točnosti mjerenja i točnosti mjernih instrumenata. Definiira se kao odnos apsolutne pogreške i prave vrijednosti, a izražava se u postocima.

$$\text{relativna pogreška} = \frac{\text{izmjerena vrijednost} - \text{prava vrijednost}}{\text{prava vrijednost}} \cdot 100\% \quad (8)$$

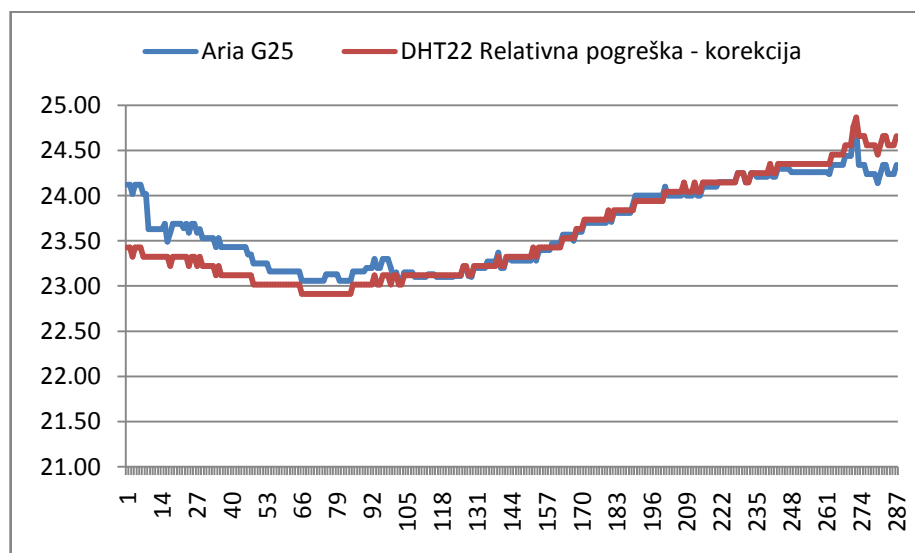
Prema navedenoj definiciji, relativna pogreška temperature iznosi:

$$p_r = \frac{T_{iz} - T_{pr}}{T_{pr}} \cdot 100\% = \frac{22.96 - 23.61}{23.61} \times 100 = -2.75\%$$

Dok je relativna pogreška vlage zrake:

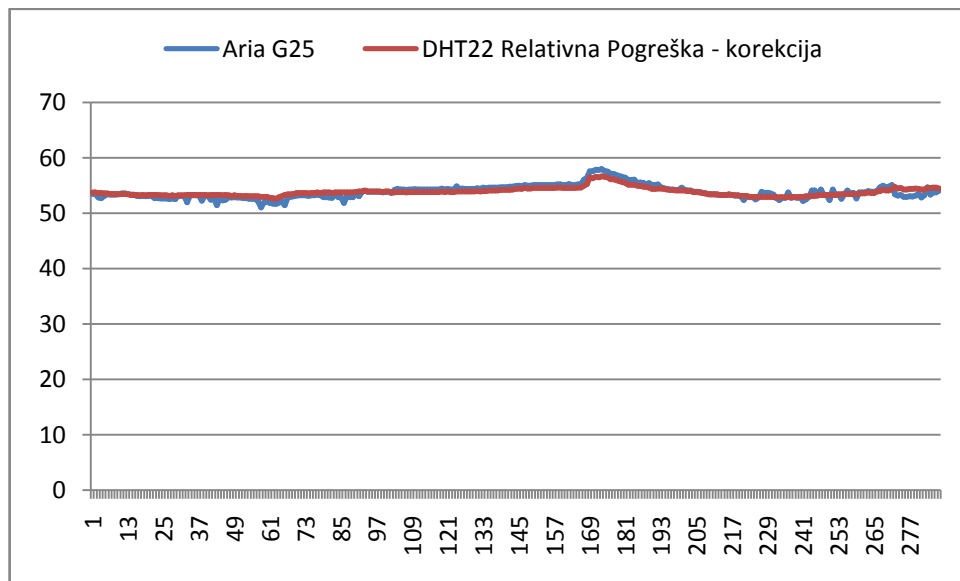
$$p_r = \frac{H_{iz} - H_{pr}}{H_{pr}} \cdot 100\% = \frac{60.9 - 53.85}{53.85} \times 100 = 13.09\%$$

Slika 16 prikazuje grafički prikaz krivulja temperature nakon primjene korekcije relativne pogreške, odnosno, sve vrijednosti izmjerene DHT22 senzorom su uvećane za 2.75%. Time se smanjilo odstupanje određenih vrijednosti. U usporedbi s korekcijom izvedenom apsolutnom pogreškom, korekcija s relativnom pogreškom daje preciznije rezultate.



Slika 16. Temperatura - relativna pogreška - korekcija

Priložena slika 17 prikazuje grafički prikaz krivulja relativne vlage zraka izmjerene referentnim uređajem i DHT22 senzorom s korekcijom relativne pogreške. Sve vrijednosti izmjerene DHT22 senzorom su uvećane za 13.09%. Tim su se postupkom izmjerene vrijednosti približile referentnim, a maksimalno odstupanje iznosi 3%. Takvo odstupanje zadovoljava kriterije budući da ne odstupa od dopuštene granice koja iznosi 5%.



Slika 17. Vlaga zraka - Relativna pogreška - korekcija

5.1.3. Standardna devijacija

Standardna devijacija je statistički pojam koji iskazuje prosječno rasipanje rezultata mjerenja koje je posljedica nepreciznosti mjernih uređaja. Računa se vađenjem drugog korijena iz varijance uzoraka, a varijanca uzoraka se definira kao prosječno kvadratno odstupanje od prosjeka izrazom:

$$s = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n - 1}} \quad (9)$$

gdje je:

s – standardna devijacija

x – član niza

n – ukupan broj članova u nizu

Standardna devijacija za dovoljan broj mjerenja poprima stalnu vrijednost, odnosno, ne mijenja se znatno s povećanjem broja mjerenja. U nastavku je priložena tablica 3 s dobivenim standardnim devijacijama temperature i vlage zraka za izrađeni sustav i referentni uređaj.

Tablica 3. Standardna devijacija temperature i vlage zraka

Standardna devijacija	DHT22	Aria G25
Temperatura	0.533750410532076	0.456150491210748
Vlaga zraka	0.855984654824545	1.21054175168388

5.1.4. Izračun standardne pogreške

Standardna pogreška je mjera nepouzdanosti rezultata mjerenja. Mjera nepouzdanosti rezultata je srednja kvadratna pogreška aritmetičke sredine koja je za faktor $\frac{1}{\sqrt{n}}$ manja od standardne devijacije pojedinog mjerenja. Računa se prema relaciji:

$$M_n = \frac{1}{\sqrt{n}} \cdot \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n - 1}} \quad (10)$$

Standardna pogreška mjerenja temperature i vlage zraka za izrađeni sustav i referentni uređaj prikazana je u tablici 4.

Tablica 4. Standardna pogreška mjerenja temperature i vlage zraka

Standardna pogreška	DHT22	Aria G25
Temperatura	0.0315062905473307	0.0269257121414987
Vlaga zraka	0.0505271765731747	0.071456020258995

5.1.5. Izračun relativne nepouzdanosti mjerenja

Relativna nepouzdanost mjerenja prikazuje se omjerom standardne pogreške i aritmetičke sredine mjerene fizikalne veličine, a izražava se u postocima, a definira izrazom:

$$R_M = \frac{M_n}{\bar{x}} \cdot 100\% \quad (11)$$

gdje je:

R_M – relativna nepouzdanost mjerenja

M_n – standardna pogreška

\bar{x} – aritmetička sredina mjerene fizikalne veličine

Prema definiciji izračunata je relativna nepouzdanost mjerenja temperature i vlage zraka za oba sustava, a rezultati su prikazani u *tablici 5*:

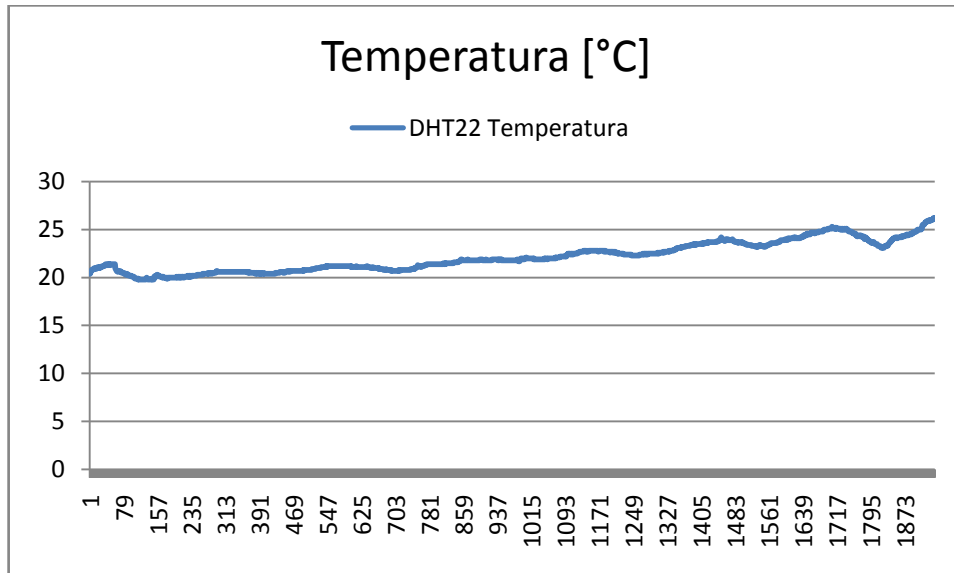
Tablica 5. Relativna nepouzdanost mjerenja temperature i vlage zraka

Relativna nepouzdanost	DHT22	Aria G25
Temperatura	0.1372%	0.1140%
Vlaga zraka	0.0829%	0.1327%

5.2. Ispitivanje funkcionalnosti sustava

Za ispitivanje funkcionalnosti sustava izvršeno je mjerenje temperature i vlage zraka tijekom sedam dana. Budući da sustav mjeri vrijednosti svakih pet minuta, pretpostavljeno je da će se za sedam dana prikupiti ukupno 2016 uzoraka. Krajnji rezultat se nije pokazao zadovoljavajući jer je sustav prikupio 1941 uzorak, odnosno, 96.28% od ukupnog broja. Razlog tome može biti nemogućnost povezivanja sa poslužiteljem ili prekinuti izvor napajanja. Izmjereni podaci podvrgnuti su statističkoj analizi i obradi kako bi se detaljnije prikazale pogreške pri mjerenju.

Slika 18 prikazuje grafički prikaz promjene temperature kroz sedam dana. Temperatura je mjerena u prostoriji u kojoj uvjeti nisu kontrolirani, a iz grafa je vidljivo da se vrijednost temperature s vremenom povećava. Razlog tome su vanjski čimbenici koji utječu na toplinske promjene.



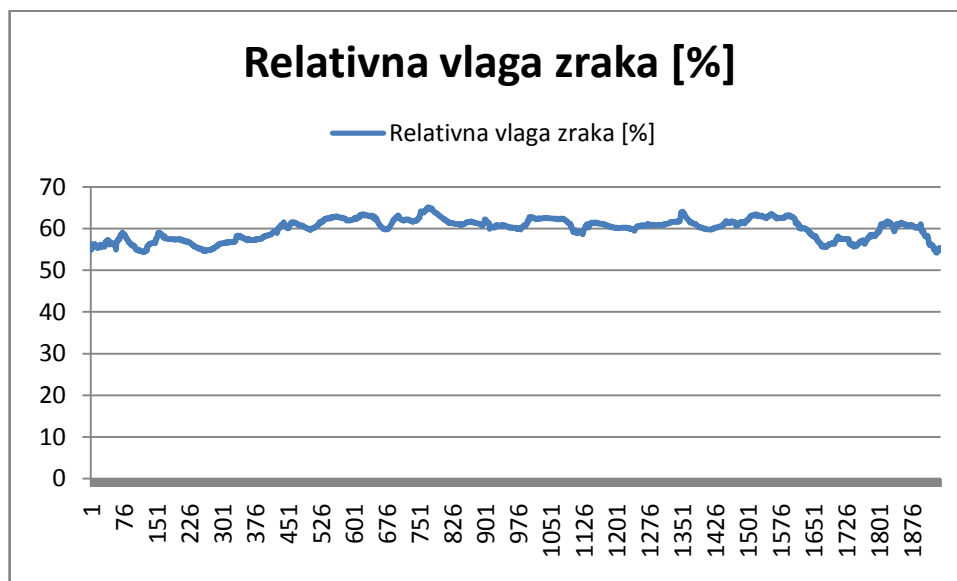
Slika 18. Temperatura - tjedno mjerenje

Prema prikupljenim vrijednostima temperature izrađena je statistička analiza koja je prikazana u priloženoj tablici 6:

Tablica 6. Analiza mjerenja temperature kroz 7 dana

Broj uzoraka	Srednja vrijednost	Medijan	Min. vrijednost	Maks. vrijednost	Varijanca	Std. devijacija	Std. pogreška
1941	22.1467	21.9	19.8	26.2	2.345	1.53154	0.03476

Slika 19 prikazuje grafički prikaz krivulje relativne vlage zraka tijekom mjerenja. Vidljiva je nekonzistentnost, a razlika između minimalne i maksimalne vrijednosti iznosi 10.9%. Takva odstupanja se u prostoriji s kontroliranim uvjetima ne smiju dogoditi, međutim, kao što je već spomenuto, tijekom istraživanja mijenjali su se vremenski uvjeti koji su utjecali na promjene vlage u prostoriji u kojoj je izvršeno mjerenje.



Slika 19. Relativna vlaga zraka - tjedno mjerenje

Prema prikupljenim podacima izrađena je statistička analiza, a njezini rezultati su prikazani u tablici 7:

Tablica 7. Analiza mjerenja relativne vlage zraka kroz 7 dana

Broj uzoraka	Srednja vrijednost	Medijan	Min. vrijednost	Maks. vrijednost	Varijanca	Std. devijacija	Std. pogreška
1941	60.0054	60.7	54.3	65.2	6.00956	2.45144	0.05564

6. RASPRAVA

Rezultati ovog istraživanja predstavljaju samo temelj za daljnju nadogradnju i optimizaciju sustava. Tijekom istraživanja uočene pogreške i nedostaci samog mjerenja, a statističkom analizom se došlo do zaključka da sustav ne prikazuje dovoljno pouzdane vrijednosti, odnosno, da dolazi do odstupanja od stvarnih vrijednosti.

Ispitivanje funkcionalnosti sustava je uspješno odrađeno. Sustav je u unutar sedam dana uzastopno mjerio podatke i prikupio 96.28% od ukupno izmjerenih vrijednosti. Podaci su podvrgnuti statističkoj analizi u kojoj su dobiveni konačni rezultati mjerenja. Iako rezultati nemaju veliki značaj budući da se mjerenje vršilo u prostoru gdje ne postoje kontrolirani uvjeti, oni predstavljaju osnovu koja se može primijeniti u skladištima ili poduzećima u kojima je važno održavati vrijednosti temperature i vlage zrake konstantnima. Time se otvara mogućnost implementacije napredne statističke analize unutar samog sustava, a osoba zadužena za nadzor bi imala uvid u trenutne rezultate u realnom vremenu.

Kod ispitivanja pouzdanosti izračunata je apsolutna pogreška koja za mjerenje temperature iznosi -0.7°C , a za mjerenje vlage zraka 7.05%. Iz navedenih pogrešaka definiran je faktor korekcije koji je primijenjen na izmjerene vrijednosti, kako bi se one približile referentnim vrijednostima. Međutim, takav pristup se ne može smatrati sigurnim budući da uvjeti nisu bili kontrolirani, a broj uzoraka je bio premalen. Pretpostavljen je razlog nepreciznog mjerenja u sustavu koji se odnosi na DHT22 senzor čiji osjetnici za temperaturu i vlagu zraka s vremenom slabe. Nadalje, prostor je čitavo vrijeme bio pod utjecajem vremenskih uvjeta što je dovelo do velike razlike između minimalne i maksimalne vrijednosti.

Navedene stavke pokazuju smjer u kojem bi se nadogradnja sustava i istraživanje trebalo nastaviti. Potrebno je ugraditi više senzora koji bi paralelno mjerili temperaturu i vlagu zraka na različitim mjestima u prostoriji, a istraživanje je potrebno provesti u kontroliranim uvjetima u kojima su vrijednosti temperature i vlage zraka konstantne.

7. ZAKLJUČAK

U današnje vrijeme ubrzanog razvoja novih tehnologija javlja se potreba za objedinjenjem istih u svrhu unaprjeđivanja i automatiziranja dugoročnih procesa. Osnovni cilj ovog diplomskog rada predstavlja integraciju dostupnih tehnologija u jedan automatizirani sustav za udaljeni nadzor temperature i vlage zraka, a prema provedenom istraživanju o funkcionalnosti može se zaključiti da je taj cilj uspješno ostvaren. Premda se radi o prototipu, izrađeni sustav sadrži temelj koji se uz određene modifikacije može primijeniti u industriji. Njegova prednost je modularnost i nezavisnost, a, budući da koristi besplatne tehnologije, ekonomski je isplativ. Trenutni nedostatak je prikazan u ispitivanju pouzdanosti, a odnosi se na neprecizno mjerenje stvarnih vrijednosti. Razlika između vrijednosti izmjerenih sustavom i referentnim uređajem iznosi 2.75% za temperaturu i 13.09% za relativnu vlagu zraku, što je u svakom slučaju previše. Drugi nedostatak je potreba za mrežnim povezivanjem s usmjernikom.

Rješenja za navedene nedostatke već postoje, a daljnjim razvojem sustava se mogu ostvariti. Za veću preciznost i bolju pouzdanost sustava potrebno je dodati više različitih senzora koji paralelno mjere vrijednosti temperature i vlage zraka, a sustav računa njihovu srednju vrijednost. Osim srednje vrijednosti, u sustav je potrebno implementirati alate za proračun pogrešaka i statističke analize u realnom vremenu. Time bi se dobio detaljan pregled trenutnog stanja sa svim mogućim pogreškama i odstupanjima.

Premda je sustav rađen sa svrhom preventivnog održavanja, njegova mu modularnost omogućuje povezivanje raznih senzora i komponenata, a time i proširivanje funkcionalnosti čitavog sustava.

8. LITERATURA

1. Pavković B. (2008). *Vježba 2: Mjerenje Temperature*, Tehnički fakultet u Rijeci, Sveučilišni studij strojarstva, Rijeka
2. ***<http://physics.mef.hr/> - *Praktikum iz fizike, Sveučilište u Zagrebu, Medicinski fakultet*, 20. 05. 2014.
3. Margolis M. (2011). *Arduino Cookbook*, O'Reilly Media, California
4. Groš S. (2006). *Mrežni Sloj Interneta*, Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva, Zagreb
5. Naramore E., Gerner J., Le Scouarnec Y., Stolz J., Glass M. (2005)., *PHP5, Apache, MySQL Web Development*, Wiley Publishing Inc, Indianapolis
6. Baranović M., Zakošek, S. (2007). *Baze Podataka*, Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva, Zagreb
7. Smith D., Friesen, J. (2011). *Android Recipes: A Problem-Solution Approach*, Apress, New York
8. Murphy M. (2011). *Android Programming Tutorials, CommonsWare*, United States of America
9. Goldberg K. (2009). *XML, Second Edition*, Peachpit Press, Berkeley
10. Krčum P. (2012). *Električna Mjerenja*, Sveučilište u Splitu, Odjel za stručne studije, Split
11. *** <http://www.phy.pmf.unizg.hr/~bosiocic/SOM/obrada.pdf> - *Obrada rezultata mjerenja*, 28. 05. 2014.
12. Koceić Bilan N. (2011). *Primijenjena statistika*, Prirodoslovno-matematički fakultet u Splitu, Split

13. *** www.pbf.unizg.hr/hr/layout/set/.../vjezba+1.pdf - *Statistička obrada mjerenih podataka*, 01. 05. 2014.
14. ***<http://www.cool-chain.com/>- *Coolchain sustav*, 01. 05. 2014.
15. ***<http://www.hw-group.com/>- *HWG-STE Ethernet thermometer* , 01. 05. 2014.
16. ***<http://racunala.ttf.unizg.hr/files/Statistika.pdf>- *Statistika*, 02. 05. 2014.
17. ***http://www.riteh.uniri.hr/zav_katd_sluz/zvd_teh_term_energ/nas/laboratorijske_vjezbe_%2007/3_temperatura.pdf - *Mjerenje temperature*, 02. 05. 2014.
18. ***http://info.grad.hr/res/odbfiles/1825/z_pogresketeoruvod.pdf - *O Pogreškama Mjerenja*, 03. 05. 2014.
19. ***<http://kolegij.fizika.unios.hr/pof1/files/2011/02/Skripta.pdf> - *Praktikum iz Osnova Fizika I*, 03. 05. 2014.
20. ***<https://developers.google.com/> - *Google Developers* , 05. 05. 2014.
21. ***<http://www.adafruit.com/datasheets/Digital%20humidity%20and%20temperature%20sensor%20AM2302.pdf> - *Adafruit datasheet*, 06. 06. 2014.
22. *** <http://www.php.net/> - *PHP NET*, 07. 05. 2014.

POPIS OZNAKA I KRATICA

Oznaka	Puni naziv
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
LAN	Local Area Network
WAN	Wide Area Network
PHP	Hypertext Preprocessor
SQL	Structured Query Language
XML	eXtensible Markup Language
JSON	Javascript Object Notation
SDK	Software Development Kit
API	Application Programming Interface
GCM	Google Cloud Messaging

POPIS SLIKA

Slika 1. Mikrokontroler Arduino Uno	9
Slika 2. Sensor DHT22	10
Slika 3. Ethernet Shield	11
Slika 4. Shematski prikaz usmjernika	12
Slika 5. Arhitektura operacijskog sustava Android	18
Slika 6. Shematski prikaz sustava	22
Slika 7. Mjerni podsustav	23
Slika 8. Shematski prikaz baze podataka	25
Slika 9. Arhitektura GCM usluge	28
Slika 10. Primjer izvještaja u Excel-u	29
Slika 11. Prikaz mobilne aplikacije	30
Slika 12. Prikaz temperature i vlage zrake - stvarne vrijednosti	37
Slika 13. Temperatura - apsolutna pogreška - korekcija	37
Slika 14. Vlaga zraka DHT22 i Aria G25	38
Slika 15. Relativna vlaga - apsolutna pogreška - korekcija	38
Slika 16. Temperatura - relativna pogreška - korekcija	39
Slika 17. Vlaga zraka - Relativna pogreška - korekcija	40
Slika 18. Temperatura - tjedno mjerenje	43
Slika 19. Relativna vlaga zraka - tjedno mjerenje	44

POPIS TABLICA

Tablica 1. Specifikacije Arduino mikrokontrolera	9
Tablica 2. Aritmetička sredina izmjerenih vrijednosti.....	35
Tablica 3. Standardna devijacija temperature i vlage zraka	41
Tablica 4. Standardna pogreška mjerenja temperature i vlage zraka	41
Tablica 5. Relativna nepouzdanost mjerenja temperature i vlage zraka.....	42

POPIS PROGRAMSKIH ISJEČAKA

Programski isječak 1 - Primjer JSON formata.....	14
Programski isječak 2 - Primjer SQL naredbe DDL dijela jezika.....	16
Programski isječak 3 - Primjer SQL naredbe iz DML dijela jezika	16
Programski isječak 4. Primjer XML formata.....	20
Programski isječak 5. Primjer dijela korisničkog sustava u Android aplikaciji	20
Programski isječak 6. Očitavanje i slanje izmjerenih vrijednosti	24
Programski isječak 7. Slanje izmjerenih vrijednosti na poslužitelj	25
Programski isječak 8. Pohrana izmjerenih vrijednosti u bazu podataka.....	27
Programski isječak 9. Upit na bazu kojim se dohvaća zadnje unesena vrijednost	26
Programski isječak 10. Upit koji dohvaća minimalnu, maksimalnu i srednju vrijednost ...	27
Programski isječak 11. Generiranje izvještaja	29
Programski isječak 12. Dohvat JSON formata	31

POPIS JEDNADŽBI

Jednadžba 1. Preračun iz Kelvinove skale u Celsiusovu	11
Jednadžba 2. Preračun iz Celsiusove skale u Kelvinovu	12
Jednadžba 3. Apsolutna vlaga zraka	13
Jednadžba 4. Relativna vlaga zraka	14
Jednadžba 5. Relativna vlaga zraka - omjer tlaka.....	14
Jednadžba 6. Aritmetička sredina	42
Jednadžba 7. Apsolutna pogreška	43
Jednadžba 8. Relativna pogreška	44
Jednadžba 9. Standardna devijacija	47
Jednadžba 10. Standardna pogreška	48
Jednadžba 11. Relativna nepouzdanost	49

PRILOZI

- PRILOG 1: Izvorni kod Android aplikacije (CD)
- PRILOG 2: Izvorni kod za Arduino mikrokontroler (CD)
- PRILOG 3: Rezultati mjerenja u Excel datoteci (CD)