

**ŠOLSKI CENTER CELJE**



**Srednja šola za strojništvo, mehatroniko in medije**

**Raziskovalna naloga**

# **AVTOMATIZIRANA SUŠILNICA SUHOMESNATIH IZDELKOV**

**Avtorja:**

**Žiga VRABIČ, M-4. c**

**Miha KRANJC, M-4. c**

**Mentorji:**

**mag. Andro GLAMNIK, univ. dipl. inž. meh.**

**mag. Matej VEBER, univ. dipl. inž. meh.**

**Celje, januar 2023**

### IZJAVA\*

Mentor/-ica \_\_\_\_\_ v skladu z 20. členom Pravilnika o organizaciji mladinske raziskovalne dejavnosti »Mladi za Celje« Mestne občine Celje, zagotavljam, da je v raziskovalni nalogi z naslovom \_\_\_\_\_, katere avtor/-ica je \_\_\_\_\_:

- besedilo v tiskani in elektronski obliki istovetno,
- pri raziskovanju uporabljeno gradivo navedeno v seznamu uporabljene literature,
- da je za objavo fotografij v nalogi pridobljeno avtorjevo dovoljenje in je hranjeno v šolskem arhivu,
- da sme Osrednja knjižnica Celje objaviti raziskovalno nalogo v polnem besedilu na knjižničnih portalih z navedbo, da je raziskovalna naloga nastala v okviru projekta Mladi za Celje,
- da je raziskovalno nalogo dovoljeno uporabiti za izobraževalne in raziskovalne namene s povzemanjem misli, idej, konceptov oziroma besedil iz naloge ob upoštevanju avtorstva in korektnem citiranju,
- da smo seznanjeni z razpisni pogoji projekta Mladi za Celje.

Celje, \_\_\_\_\_

žig šole

Podpis mentorja

Podpis odgovorne osebe

\*

### POJASNILO

V skladu z 20. členom Pravilnika raziskovalne dejavnosti »Mladi za Celje« Mestne občine Celje je potrebno podpisano izjavo mentorja (-ice) in odgovorne osebe šole vključiti v izvod za knjižnico, dovoljenje za objavo avtorja (-ice) fotografskega gradiva, katerega ni avtor (-ica) raziskovalne naloge, pa hrani šola v svojem arhivu.

## **ZAHVALA**

Rada bi se zahvalila vsem, ki so nama nudili pomoč in nasvete med izdelavo najine raziskovalne naloge. Hvala predvsem najinima mentorjema mag. Andru Glamniku, univ. dipl. inž. in mag. Mateju Vebru, univ. dipl. inž., ki sta nama svetovala pravo pot do iskanih rešitev. Posebna zahvala pa gre tudi Tadeji Kolman, prof., za lektoriranje naloge.

## **POVZETEK**

Podnebne spremembe so vse večje, te pa privedejo do tega, da se vremenske razmere iz dneva v dan drastično spreminjajo. Zaradi spremenljivih vremenskih razmer je sušenje suhomesnatih izdelkov vsako leto zahtevnejše. V ta namen smo se odločil, da bomo ustvarili sistem za uravnavanje razmer pri sušenju suhomesnatih izdelkov. Ta bo uravnaval razmere v prostoru, kjer se izdelki sušijo. Spremljanje razmer pa smo olajšali z mobilno aplikacijo, ki smo jo ustvarili. V raziskovalni nalogi smo raziskovali, kako bo ta sistem vplival na razmere v prostoru in na sušenje mesa. Rezultati sistema, pa so vzpodbudni. Pri raziskovanju smo usvojili nova znanja, ki jih bomo potrebovali pri nadaljnjem šolanju.

Ključne besede: DHT22 senzor, HC-05 modul, Arduino UNO, mobilna aplikacija.

## **SUMMARY**

Climate change is increasing, and this leads to the weather conditions changing drastically from day to day. Due to changing weather conditions, drying of dried products is more complex every year. Because of this, we decided to create a system that regulates the conditions for drying dried meat products. This system regulates the conditions in the room where the products are being dried. However, we made it easier to monitor what the conditions are with the mobile app we created. In our research mission, we investigated how this system would affect the conditions in the room and the drying of meat. The results of the system are encouraging. In our research, we have gained new knowledge that we will need in further schooling.

Keywords: DHT22 sensor, HC-05 module, Arduino UNO, mobile application.

# KAZALO VSEBINE

<b>1</b>	<b>UVOD.....</b>	<b>1</b>
1.1	PREDSTAVITEV PROBLEMA .....	3
1.2	HIPOTEZE .....	3
1.3	METODE RAZISKOVANJA.....	3
<b>2</b>	<b>PREDSTAVITEV POTEKA RAZISKOVALNE NALOGE.....</b>	<b>4</b>
2.1	PROSTOR.....	5
<b>3</b>	<b>ARDUINO .....</b>	<b>8</b>
3.1	PROGRAMSKI JEZIK .....	10
3.2	SENZOR TOPLOTE IN VLAGE DHT22 .....	10
<b>4</b>	<b>HC-05 BLUETOOTH MODUL .....</b>	<b>13</b>
4.1	ZNAČILNOSTI MODULA BLUETOOTH HC-05 .....	13
4.2	UPORABA HC-05:.....	14
4.3	KAKO UPORABLJATI HC-05 BLUETOOTH SENZOR .....	14
4.4	HC-05 KONFIGURACIJA PRIKLJUČKOV .....	15
4.5	SHEMATSKI PRIKAZ PRIKLJUČKOV .....	15
<b>5</b>	<b>PROGRAM ZA IZDELOVANJE MOBILNIH APLIKACIJ.....</b>	<b>17</b>
5.1	KAKO NAREDITI APLIKACIJO V PROGRAMU APP INVERTOR? .....	17
5.2	ZAVIHEK »DESIGNER« .....	19
5.3	ZAVIHEK »BLOCKS« .....	21
6.1	NASTAJANJE APLIKACIJE .....	25
6.1.1	<i>Vizualni del</i> .....	25
6.1.2	<i>Blokovni del</i> .....	26
<b>7</b>	<b>RELE.....</b>	<b>31</b>
<b>8</b>	<b>DELOVANJE SISTEMA .....</b>	<b>32</b>
8.1	VEZJE ZA URAVNAVANJE TEMPERATURE IN VLAGE.....	32
8.2	VEZJE ZA BLUETOOTH MODUL .....	35
8.3	SKUPNO VEZJE .....	38

8.3.1	<i>Skupna koda</i> .....	42
<b>9</b>	<b>PREDSTAVITEV REZULTATOV .....</b>	<b>47</b>
9.1	PRIMERJAVA STROŠKOV IZDELAVE Z CENO TRGA .....	47
9.2	PREDSTAVITEV HIPOTEZ .....	47
<b>10</b>	<b>MOŽNOSTI NADALJNEGA RAZISKOVANJA.....</b>	<b>49</b>
<b>11</b>	<b>VIRI IN LITERATURA.....</b>	<b>50</b>

## KAZALO SLIK

Slika 1: Zgornjesavinjski želodec .....	2
Slika 2: Hitro spreminjanje temperatur .....	2
Slika 3: Radiator za višanje temperature .....	4
Slika 4: Vlažilnik zraka.....	5
Slika 5: Prostor za sušenje .....	5
Slika 6: Prostor od zunaj .....	6
Slika 7: Kozolec, v katerem bo sistem obratoval.....	6
Slika 8: Želodci med sušenjem .....	6
Slika 9: Sestavni deli Arduina.....	8
Slika 10: Logotip Arduina .....	9
Slika 11: Logotip programskega jezika C++ .....	10
Slika 12: DHT22 senzor.....	11
Slika 13: DHT22 Solidworks konstrukcija .....	12
Slika 14: HC-O5 modul .....	13
Slika 15: Priključki modula .....	15
Slika 16: Priključki .....	15
Slika 17: Logotip spletnega portala .....	17
Slika 18: Stran za izdelavo vizualnega izgleda aplikacije .....	18
Slika 19: Stran za programiranje aplikacije .....	18
Slika 20: Različne skupine blokov za programiranje .....	19
Slika 21: Prikaz delovanja aplikacije pred povezavo z Bluetooth senzorjem .....	23
Slika 22: Ob pritisku na gumb Connect nam prikaže razpoložljivi senzor za povezavo.....	24
Slika 23: Prikaz, ko pritisnemo tipko za povezavo.....	24
Slika 24: Logotip aplikacije .....	25
Slika 25: Nastajanje vizualnega izgleda 1. ....	25
Slika 26: Nastajanje vizualnega izgleda 2. ....	26
Slika 27: Pogled na vse programirljive bloke, ki omogočajo delovanje.....	26
Slika 28: Program za povezavo z Bluetooth modulom.....	27
Slika 29: Programirljivi bloki za izpis temperature in vlage na zaslonu .....	28
Slika 30: Prvi del programa .....	29
Slika 31: Drugi del programa.....	30
Slika 32: Tretji del programa .....	30



Slika 33: Konstrukcija releja.....	31
Slika 34: Testno vezje za DHT22 senzor.....	32
Slika 35: Risba vezja.....	33
Slika 36: Koda testnega vezja 1 .....	33
Slika 37: Ali funkcija za vlago.....	34
Slika 38: Ali funkcija za temperaturo .....	35
Slika 39: Vezje HC-05 modula.....	35
Slika 40: Risba vezja.....	36
Slika 41: Prvi del programa .....	37
Slika 42: Drugi del programa.....	37
Slika 43: Tretji del programa .....	38
Slika 44: Skupno vezje v programu Fritzing .....	39
Slika 45: Skupno vezje skica .....	40
Slika 46: Vezje s priloženo aplikacijo .....	41
Slika 47: Diagram poteka sistema.....	41
Slika 48: Priprava skupne kode.....	42
Slika 49: Void setup skupne kode.....	43
Slika 50: Del kode za prikazovanje na serijskem vmesniku.....	44
Slika 51: Del kode za prenos podatkov na aplikacijo in uravnavanje vlage v prostoru .....	44
Slika 52: Del kode za uravnavanje temperature .....	45
Slika 53: Del kode za odpiranje okna .....	46

## KAZALO TABEL

Tabela 1: Vreme januarja leta 1950 .....	1
Tabela 2: Vreme januarja leta 2022 .....	1
Tabela 3: Razmere pred uporabo sistema .....	7
Tabela 4: Razmere po vzpostavitvi sistema.....	7
Tabela 5: Primerjava senzorjev.....	11
Tabela 6: Potrjevanje in zavrnitev hipotez.....	47

AVTOMATIZIRANA SUŠILNICA SUHOMESNATIH IZDELKOV  
Raziskovalna naloga

---

## 1 UVOD

V današnjih časih je vse več nenadnih sprememb vremena in nastajajo vse večja nihanja. Danes je lahko 1 °C, jutri pa že 10 °C. Pred 73 leti je bilo vreme bolj konstantno, kot je dandanes, kar je razvidno v tabeli povprečne temperature in vlage meseca januarja leta 1950 in januarja leta 2022. [2]

(Vir: <https://meteo.arso.gov.si/>)

Tabela 1: Vreme januarja leta 1950

Podatki za mesec januar 1950		
Datum	Temperatura (v °C)	Vlaga (v %)
1. 1. 1950	−3,9	76
2. 1. 1950	−2,9	71
3. 1. 1950	0,7	65
4. 1. 1950	0,8	82
5. 1. 1950	−1,4	87
6. 1. 1950	1,3	82
7. 1. 1950	0,1	83
8. 1. 1950	−0,2	79
9. 1. 1950	3	74
10. 1. 1950	1,3	81
11. 1. 1950	−1,6	95
12. 1. 1950	−4,1	82
13. 1. 1950	−3,4	92
14. 1. 1950	−1,5	81
15. 1. 1950	−1,6	84
16. 1. 1950	3,9	84
17. 1. 1950	1,2	98
18. 1. 1950	−0,2	97
19. 1. 1950	−4,4	89
20. 1. 1950	−3,1	94
21. 1. 1950	−8,4	91
22. 1. 1950	−7,4	85
23. 1. 1950	−7,1	84
24. 1. 1950	−8,5	85
25. 1. 1950	−9,4	78
26. 1. 1950	−11,1	79
27. 1. 1950	−8	81
28. 1. 1950	−5,3	89
29. 1. 1950	−7,9	75
30. 1. 1950	−12,5	75
31. 1. 1950	−13,3	80

(Vir: <https://meteo.arso.gov.si/>)

Tabela 2: Vreme januarja leta 2022

Podatki za mesec januar 2022		
Datum	Temperatura (v °C)	Vlaga (v %)
1. 1. 2022	4,6	80
2. 1. 2022	4,9	78
3. 1. 2022	4,7	82
4. 1. 2022	10,7	72
5. 1. 2022	5,3	85
6. 1. 2022	1	93
7. 1. 2022	−3,4	92
8. 1. 2022	−2,9	78
9. 1. 2022	−1,5	86
10. 1. 2022	−0,6	88
11. 1. 2022	−2,4	80
12. 1. 2022	−4	80
13. 1. 2022	−1,8	71
14. 1. 2022	3	67
15. 1. 2022	−0,7	84
16. 1. 2022	−1,6	86
17. 1. 2022	−0,3	85
18. 1. 2022	0,7	81
19. 1. 2022	−2,3	92
20. 1. 2022	−0,1	89
21. 1. 2022	−1,6	78
22. 1. 2022	−0,3	76
23. 1. 2022	0	88
24. 1. 2022	−0,1	83
25. 1. 2022	−0,5	80
26. 1. 2022	−1,2	85
27. 1. 2022	−0,3	82
28. 1. 2022	3,5	57
29. 1. 2022	−0,6	70
30. 1. 2022	2,7	60
31. 1. 2022	−1,9	90

## AVTOMATIZIRANA SUŠILNICA SUHOMESNATIH IZDELKOV

### Raziskovalna naloga

---

Razvidno je tudi, da so dnevi v povprečju vse toplejši, kar predstavlja dodatno težavo za sušenje mesa. Želodci se namreč v preveč toplih razmerah prehitro posušijo in so zaradi tega bistveno slabše kakovosti, občasno se tudi pokvarijo. Posledično je sušenje suhomesnatih izdelkov vse težje, saj zahtevajo ti izdelki za kakovostno sušenje čim bolj uravnovešene razmere. Zato je treba vsak dan razmere nadzorovati in jih s pripomočki uravnavati. To zahteva veliko časa, ki bi ga lahko porabili drugače. S tem sistemom bi tako prihranili čas, saj bi temperaturo in vlago uravnavali avtomatsko in kar preko mobilne aplikacije. Na sliki ena vidimo izgled zgornjesavinjskega želodca.



*Slika 1: Zgornjesavinjski želodec*

(Vir: <https://www.tasteslovenia.si/sl/okusi-slovenijo/slovenska-gastronomija/znacilne-jedi/>)



*Slika 2: Hitro spreminjanje temperatur*

## 1.1 PREDSTAVITEV PROBLEMA

Zadali smo si cilj, da avtomatiziramo sušilni proces in ustvarimo brezžični nadzor. Pri ustvarjanju sistema smo naleteli na kar nekaj težav. Potrebno je bilo poiskati Bluetooth modul, ki bo ustrezal danim zahtevam. Ker smo sprva pridobili napačnega, je ta predstavljal veliko težavo. Težavo nam je predstavljalo tudi pisanje kode za Arduino UNO. Potrebno je bilo tudi poiskati program, ki bi omogočil enostavno ustvarjanje mobilne aplikacije. Izdelovanje aplikacije je bilo zahtevno, saj smo se morali naučiti celotno programiranje z logičnimi bloki v aplikaciji MIT.

## 1.2 HIPOTEZE

Cilj naše naloge je bil, da raziščemo, kako bi vplivale idealne razmere, ki bi jih uravnaval sistem za uravnavanje vlage in temperature, na sušenje suhomesnatih izdelkov. Za raziskovanje smo si zadali naslednje hipoteze:

- HIPOTEZA 1 – S sistemom bomo prihranili čas, saj ne bo potrebno vsakodnevno ročno uravnavanje pogojev.
- HIPOTEZA 2 – Spremljanje toplote in vlage bo lažje in hitrejše, saj bo vse mogoče spremljati preko mobilnega telefona.
- HIPOTEZA 3 – Sistem je cenovno ugodnejši, kot jih ponuja trg.

## 1.3 METODE RAZISKOVANJA

Med ustvarjanjem raziskovalne naloge smo uporabljali naslednje metode:

- Primerjalna metoda za določitev najboljšega temperaturnega senzorja in Bluetooth modula.
- Metoda razčlenitve, s katero smo razdelili delo na različne dele, kot so programiranje, izdelovanje aplikacije, konstruiranje in praktično delo.

## 2 PREDSTAVITEV POTEKA RAZISKOVALNE NALOGE

Pri raziskovanju smo si zadali cilj, da bomo ustvarili sistem, ki bo čim bolj avtonomno uravnaval temperaturo in vlago. Namen je bil, da bi sistem deloval brez nujnosti stalnega človeškega nadzora in kljub temu ohranjal pristen okus suhega želodca ter drugih izdelkov. Poleg tega pa smo želeli nadzorovati sušenje preko mobilne aplikacije. Zanimalo nas je, kako bo ta sistem vplival na razmere sušenja in če bo nato izdelek kakovostnejši od tistega, ki je sušen v nenadzorovanem prostoru. Če bi bili rezultati vzpodbudni, bi lahko sistem vzpostavili še pri kakšnem kmetu, ki suši večje količine suhomesnatih izdelkov in mu tako prikrajšali kar nekaj dela. Sistem preko preklapljanja relejev avtomatsko prižiga in ugaša naprave za uravnavanje temperature in vlažnosti. Radiator je viden na sliki tri, vlažilnik zraka pa na sliki štiri. Dodali pa še bomo motorček, ki bo ob zadostnih zunanjih pogojih zunanjega zraka odprl okno in ga ob nezadostnih zaprl.



*Slika 3: Radiator za višanje temperature*

*(Vir: Osebni vir)*

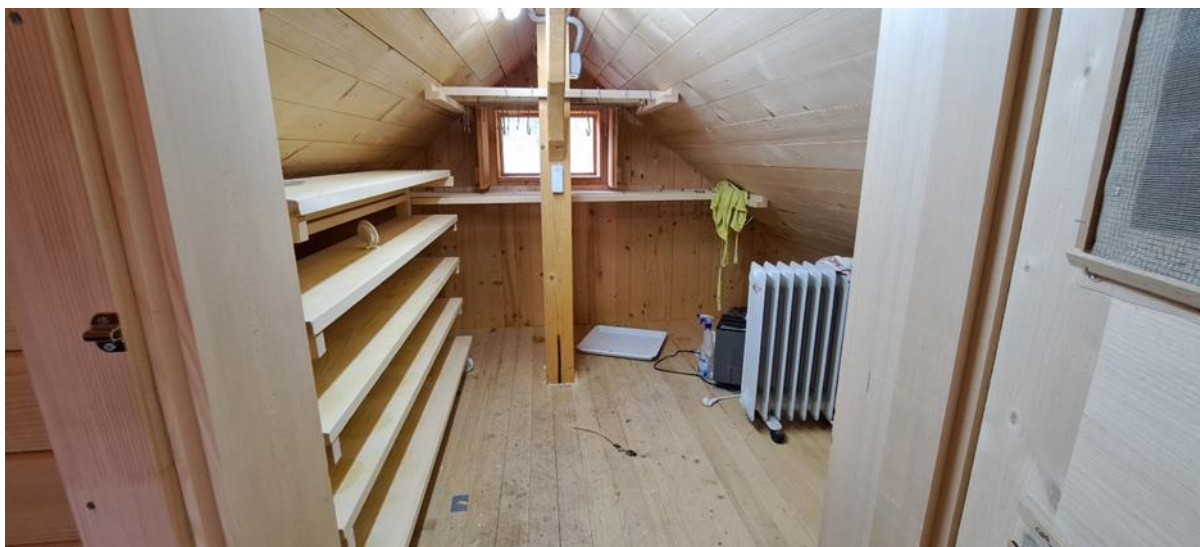


*Slika 4: Vlažilnik zraka*

*(Vir: Osebni vir)*

## 2.1 PROSTOR

Pri nas doma že vrsto let sušimo zgornjesavinjske želodce. Pred nekaj leti smo zgradili za hišo novi kozolec. V njem smo naredili prostor za sušenje mesa. Prostor v katerem se želodci sušijo je viden v sliki pet, kozolec pa v slikah šest in sedem. Ta prostor smo skrbno zatesnili, da sta temperatura in vlaga v njem čim bolj enakomerni. Ker je prostor majhen in je iz skoraj vseh strani obdan z zunanjim zrakom, le-ta ohranja velik vpliv na razmere, ki so v prostoru. Ko zunaj dežuje, vlažnost prehitro naraste na previsoke vsote. Enako se dogaja, če je mraz, suša, prevelika toplota itd. Posledično samo dobro tesnjenje ni dovolj in je potreben kakovosten sistem.



*Slika 5: Prostor za sušenje*



# AVTOMATIZIRANA SUŠILNICA SUHOMESNATIH IZDELKOV

## Raziskovalna naloga

---

*(Vir: Osebni vir)*



*Slika 6: Prostor od zunaj*

*(Vir: Osebni vir)*



*Slika 7: Kozolec, v katerem bo sistem obratoval*

*(Vir: Osebni vir)*



*Slika 8: Želodci med sušenjem*

*(Vir: Osebni vir)*

AVTOMATIZIRANA SUŠILNICA SUHOMESNATIH IZDELKOV  
Raziskovalna naloga

---

Pred vzpostavitev sistema smo v prostoru izvajali desetdnevne meritve vlage in temperature. Rezultati so razvidni v naslednji tabeli:

(Vir: Osebni vir)

Tabela 3: Razmere pred uporabo sistema

Razmere pred uporabo sistema		
Datum.	Temperatura	Vlaga
23. jan.	0,5°C	82%
24. jan.	−1°C	90%
25. jan.	−1°C	80%
26. jan.	−1°C	78%
27. jan.	−2°C	78%
28. jan.	−4°C	70%
29. jan.	−3°C	63%
30. jan.	−4°C	68%
31. jan.	−4°C	65%
1. feb.	−1°C	74%

(Vir: Osebni vir)

Tabela 4: Razmere po vzpostavitvi sistema

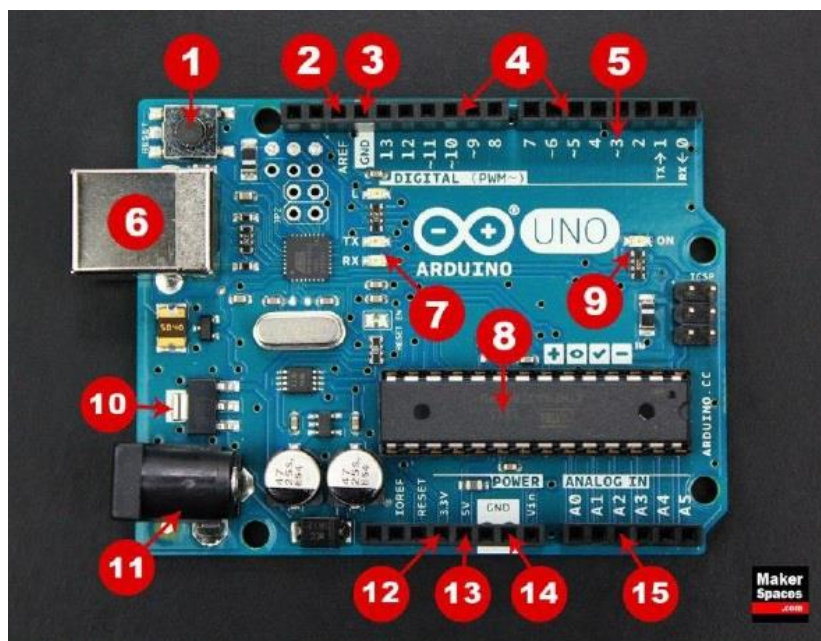
Razmere po uporabi sistema		
Datum	Temperatura	Vlaga
15. feb.	−2°C	70%
16. feb.	−1°C	85%
17. feb.	−1°C	82%
18. feb.	2°C	79%
19. feb.	3°C	73%
20. feb.	5°C	74%
21. feb.	4°C	74%
22. feb.	3°C	72%
23. feb.	4°C	70%
24. feb.	3°C	71%

Vidno je, da je predvsem nihanje vlage iz dneva v dan kar veliko, saj ima vreme kljub izolaciji velik vpliv. Po vzpostavitvi sistema pa se je to nihanje zmanjšalo. Sistem je tako nadzoroval, da vlaga ni padla pod določeno mejo, kar se je pred tem ob suhem vremenu pogosto dogajalo.



### 3 ARDUINO

Arduino je odprtokodna elektronska naprava (mikrokrmilnik na matični plošči), ki je zasnovana tako, da je postopek uporabe le-te čim bolj preprost. Strojno opremo sestavljajo 8-bitni mikrokontroler Atmel AVR ali 32-bitni Atmel ARM. Programska oprema je sestavljena iz standardnega programskega jezika, prevajalnika in zagonskega nalagalnika, ki se izvaja na mikrokrmilniku. Razvojne plošče Arduino so praviloma že sestavljene in pripravljene za uporabo. Slika devet prikazuje vse sestavne dele mikrokrmilnika Arduino UNO.[1]



Slika 9: Sestavni deli Arduina

(Vir: <https://www.3dsvet.eu/kaj-je-arduino/>)

1. Reset tipka – S pritiskom na tipko resetiramo mikrokrmilnik. Program se začne izvajati od začetka. Če imamo v programu neko vrsto inicializacije, ki se izvede samo ob zagonu in recimo resetira komunikacijo in vse postavi na začetno stanje, je reset tipka lahko zelo uporabna.
2. AREF – Okrajšava za “Analog Reference”. Na ta priključek pripeljemo zunanjo napetost, ki bo služila kot referenčna napetost za AD-pretvornik. Z njo definiramo zgornjo mejo meritve napetosti analognega vhoda.
3. GND – Priključek, na katerega je pripeljan najnižji napetostni nivo 0Vdc. Na ploščici jih imamo več.
4. Digitalni vhodi/izhodi – Priključke 0–13 lahko uporabimo za vhode ali izhode. Izbiro definiramo v programu.

5. PWM – Priključek z dodatno oznako (~) nam s pomočjo programa lahko proizvedejo PWM signal. Torej impulze spremenljive dolžine, kar je za porabnike lahko videti kot analogni izhod. Na tak način lahko simuliramo analogni izhod, ki ga Arduino UNO nima.
6. USB konektor – USB povezavo uporabljamo za napajanje in nalaganje programa. Med delovanjem lahko služi tudi kot prenos podatkov med računalnikom in mikrokrmilnikom, če smo tako sprogramirali.
7. TX/RX – LED-ici, ki utripata, ko imamo prenos podatkov. Utripata recimo, ko nalagamo program.
8. ATmega mikrokrmilnik – Mikrokrmilnik. Glavni del platforme. V njem je program.
9. LED za indikacijo napajanja.
10. Napetostni regulator – Kot že samo ime pove, regulator skrbi, da ima platforma stabilno napetost 5Vdc ne glede na vhodno napetost. Seveda ima regulator svoje omejitve in na njega lahko praviloma priklopimo 7–12Vdc. Prav tako je omejitev toka, ki lahko teče skozi regulator.
11. DC napajalni konektor – Preko njega na platformo pripeljemo zunanje napajanje (7–12Vdc)
12. 3.3V Priključek – Na tem priključku lahko dostopamo do 3.3Vdc napetosti, ki jo platforma prav tako preko napetostnega regulatorja proizvede sama. Običajno jo uporabimo za napajanje senzorjev ali naprav, ki delujejo samo na tem napetostnem nivoju in bi jih klasičnih 5Vdc uničilo.
13. 5V Priključek – Na tem priključku lahko dostopamo do 5Vdc napetosti.
14. GND – Priključek, na katerega je pripeljan najnižji napetostni nivo 0Vdc. Na ploščici jih imamo več.
15. Analogni priključek – Sem lahko pripeljemo zunanje napetosti, ki jih želimo meriti. To so običajno napetosti, ki jih dobimo iz raznih senzorjev. [9]



*Slika 10: Logotip Arduina*

*(Vir: [https://commons.wikimedia.org/wiki/File:Arduino\\_Logo.svg](https://commons.wikimedia.org/wiki/File:Arduino_Logo.svg))*

### 3.1 PROGRAMSKI JEZIK

Programski jezik C++ je splošno namenski računalniški programski jezik. C++ so podatkovni tipi statični, zapis kode je prost. C++ omogoča različne programerske pristope, in sicer proceduralno, objektno usmerjeno, generično in funkcionalno.

C++ je razvil danski računalničar Bjarne Stroustrup, v Bellovih laboratorijih. Razvijati je začel leta 1979 in takrat se je jezik imenoval »C with Classes« (C z razredi). Od 90. let je eden najbolj priljubljenih komercialnih programskih jezikov. Najprej so C-ju dodali razrede, nato med drugim virtualne funkcije, preobložitev operatorjev, večkratno dedovanje, predloge in rokovanje z izjemami.



Slika 11: Logotip programskega jezika C++

(Vir: <https://sl.wikipedia.org/wiki/C%2B%2B>)

C++ se prevede v strojni jezik, kar mu omogoča, da enak program izvaja bistveno hitreje od jezikov, ki se interpretirajo ali pa se prevedejo v vmesni jezik (Java, C#, Python, javascript, PHP ...). Poleg tega, da za isto opravilo program porabi manj časa, porabi tudi manj energije, kar je pomembno za naprave z baterijskim napajanjem. C++ se je razvil iz jezika C in programe, napisane v jeziku C, je razen nekaterih redkih izjem mogoče brez sprememb prevesti s prevajalnikom za C++. [7]

### 3.2 SENZOR TOPLOTE IN VLAŽE DHT22

DHT22 je senzor, ki meri vlago in toploto v njegovem okolišu. Uporablja kapacitivni senzor za merjenje vlažnosti in termistor za merjenje temperature zraka. Senzor nato pošlje digitalni signal na podatkovni priključek. S tem senzorjem bo sistem deloval, saj bo le-ta pošiljal Arduinou podatke o vlagi in temperaturi v in izven prostora. DHT22 senzor smo sicer izbrali,

AVTOMATIZIRANA SUŠILNICA SUHOMESNATIH IZDELKOV  
Raziskovalna naloga

---

ker ima širšo merilno območje in je natančnejši od njegove starejše verzije DHT11. Razlike med tema dvema senzorjema so razvidne v spodnji tabeli. Na sliki dvanajst, je fotografija DHT22 senzorja. [3]

(Vir: <https://www.hnhcart.com/blogs/sensors-modules/dht11-vs-dht12-vs-dht22>)

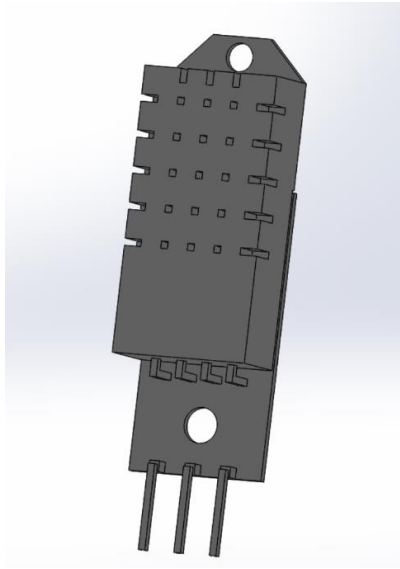
Tabela 5: Primerjava senzorjev

	DHT11	DHT22
Delovna napetost	3 do 5V	3 do 5V
Maks. operativni tok	2.5 mA max	2.5 mA max
Obseg vlažnosti	20–80% / 5%	0–100% / 2–5%
Obseg temperature	0–50 °C / $\pm 2$ °C	–40 do 80 °C / $\pm 0.5$ °C
Pogostost merjenja	1 Hz (bere vsako sekundo)	0.5 Hz (bere vsaki dve sekundi)
Velikost senzorja	15.5 mm x 12 mm x 5.5 mm	15.1 mm x 25 mm x 7.7 mm
Prednosti	Nizka cena	Natančnejši



Slika 12: DHT22 senzor

(Vir: Osebni vir)



*Slika 13: DHT22 Solidworks konstrukcija*

*(Vir: Osebni vir)*

## 4 HC-05 BLUETOOTH MODUL

Za prikaz temperature in vlage na mobilni aplikaciji je potrebno te podatke pridobiti preko HC-05 modula. V aplikaciji se ob prižigu podatki še ne bodo prikazali, vendar bodo prišli preko HC-05 modula. Ob povezavi z Bluetooth senzorjem HC-05 bo začel mobilni telefon prejemati podatke o temperaturi in vlagi v sušilnici. S tem bomo lahko v živo pridobivali podatke. Modul Bluetooth HC-05 je poceni, lahko dostopen in enostaven za uporabo, namenjen podatkovni komunikaciji na kratke razdalje. Gre za napravo združljivo s standardom Bluetooth v2.0, ki lahko deluje v podrejenem in glavnem načinu, kar mu omogoča komunikacijo s številnimi drugimi napravami, ki podpirajo Bluetooth, kot so pametni telefoni, tablični računalniki, prenosni računalniki in drugi. Modul HC-05 je zasnovan za povezavo z mikrokontrolerji, kot so tisti v ploščah Arduino, in zagotavlja serijski komunikacijski vmesnik z gostiteljsko napravo. Spodnja slika prikazuje izgled HC-05 modula.



Slika 14: HC-05 modul

(Vir: <https://microcontrollerslab.com/wp-content/uploads/2020/01/HC-05-Bluetooth-Module-rotated-e1579011159766.jpg>)

Ta modul podpira različne načine delovanja, vključno s samodejno povezavo, ročno povezavo in oddaljeno konfiguracijo. Ima vgrajeno anteno in podporo za zunanje antene ter lahko komunicira na razdalji do okoli 30 metrov, odvisno od okolja. Modul HC-05 je opremljen tudi z vgrajenimi indikatorji LED, ki lahko zagotavljajo informacije o stanju in pomoč pri odpravljanju napak. Modul Bluetooth HC-05 je vsestranska, cenovno ugodna in za uporabo enostavna rešitev za dodajanje možnosti brezžične komunikacije v številne elektronske projekte.

### 4.1 ZNAČILNOSTI MODULA BLUETOOTH HC-05

Modul Bluetooth HC-05 ponuja dva načina komunikacije za krajše razdalje s hitro hitrostjo. Ima priključek, ki omogoča preklapljanje med ukaznim in podatkovnim načinom. Naprava uporablja UART (serijska komunikacija), ki omogoča enostavno povezovanje s katerim koli mikrokrmilnikom ali sistemom. Njen doseg je do okoli 10 metrov, vendar signalizira nižje, če je na njeni poti kakršna koli ovira. Naprava za napajanje uporablja 5 V. Modul lahko deluje kot glavni ali podrejeni modul. [6]

Modul lahko uporablja naslednje hitrosti prenosa podatkov (signalne enote na sekundo):

- 9600
- 19200
- 38400
- 57600
- 115200
- 230400
- 460800

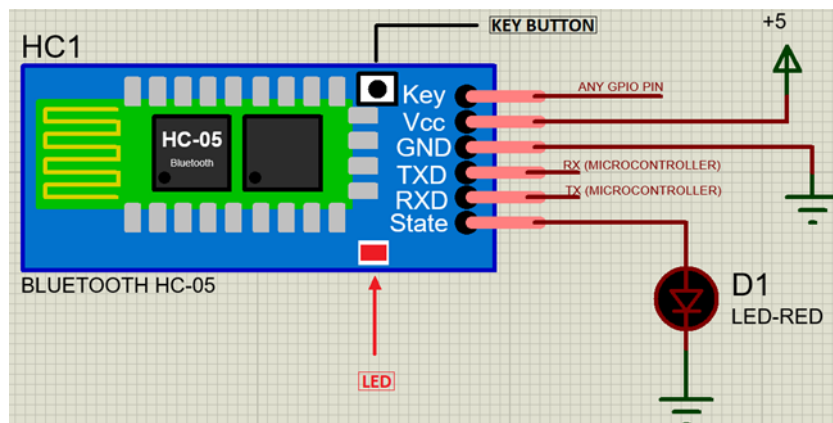
## **4.2 UPORABA HC-05:**

Modul Bluetooth HC-05 se običajno uporablja za brezžični prenos podatkov med več mikrokrmilniki. Uporablja se lahko tudi za komunikacijo med elektronskimi napravami, kot so mobilni telefoni, prenosni računalniki in računalniki za prenos podatkov. Uporablja se tudi v različnih aplikacijah za beleženje informacij in podatkov.

## **4.3 KAKO UPORABLJATI HC-05 BLUETOOTH SENZOR**

Uporaba modula Bluetooth HC-05 je preprosta za vsak mikrokrmilnik, tudi za vsak sistem, ki ima serijski komunikacijski vmesnik. Modul ima dva serijska vhodna nožiča, ki sta namenjena za podatkovno komunikacijo. TX se bo povezal z RX priključkom mikrokrmilnika, RX pa s TX priključkom mikrokrmilnika. Podatki se bodo prenašali s hitrostjo prenosa. Hitrost prenosa je določena v mikrokrmilniku. Podatki se bodo prenašali s hitrostjo 9600, ki je privzeta, vendar jo je mogoče tudi spremeniti.



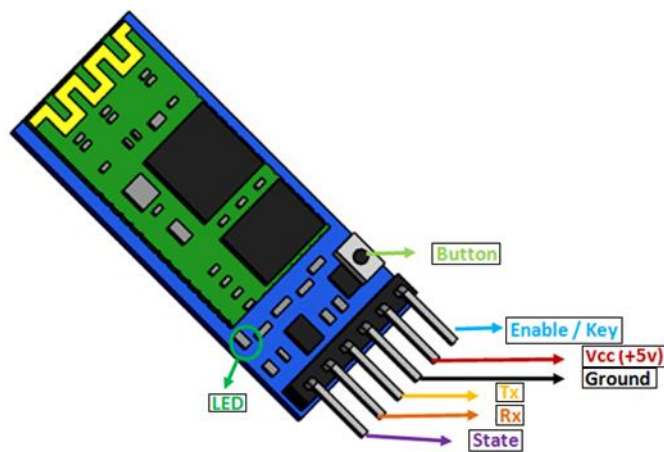


Slika 15: Priključki modula

(Vir: <https://microcontrollerslab.com/wp-content/uploads/2020/01/HC-05-Bluetooth-Module-circuit.png>)

## 4.4 HC-05 KONFIGURACIJA PRIKLJUČKOV

Naprava HC-05 je opremljena s številnimi priključki in indikatorji, ki omogočajo nadzor različnih operacij in prikaz njihovega stanja s pomočjo indikatorjev. Na spodnji sliki so prikazani vsi priključki. Prikazana je tudi funkcionalnost vsakega priključka. Priključki so vidni na spodnji sliki.



Slika 16: Priključki

(Vir: <https://microcontrollerslab.com/wp-content/uploads/2020/01/HC-05-Bluetooth-Module-Priključek-Configuration.png>)

## 4.5 SHEMATSKI PRIKAZ PRIKLJUČKOV

- VCC Priključek

Kot vsaka naprava je tudi delovanje modulov HC05 odvisno od napajanja, priključek VCC pa omogoča povezavo z zunanjim virom napajanja.



- GND Priključek

Za povezavo z mikrokrmilnikom je za vsako napravo potrebna skupna masa. Zahtevo bo rešil ozemljitveni priključek modula.

- TX Priključek

Modul Bluetooth HC-05 se z mikrokrmilnikom poveže preko komunikacije UART. TX je priključek za prenos podatkov modula v UART.

- RX Priključek

Ta priključek bo v komunikaciji UART sprejemal podatke.

- State Priključek

Stanje bo prikazovalo trenutno stanje Bluetooth. Upravljalniku daje povratne informacije o povezljivosti Bluetooth z drugo napravo.

- Enable/Key Priključek

Enable/Key Priključek je edini priključek, po katerem je HC-05 edinstven od drugih. Z njim lahko z zunanjim signalom preklomite napravo med podatkovnim in ukaznim načinom. Logično stanje HIGH bo napravo preneslo v ukazni način, logično stanje LOW pa v podatkovni način. Privzeto stanje naprave brez kakršnega koli signala je podatkovni način.

- Button Priključek

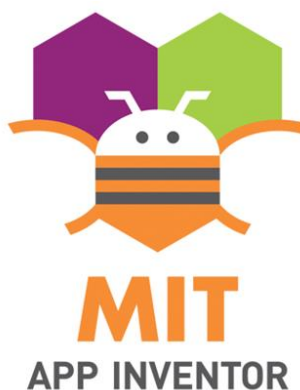
Stanje ukaznega in podatkovnega načina je mogoče spreminjati z gumbom na modulu.

Obstaja še tretji priključek (Key), ki je povezan z zunanjim priključkom GPIO ali gumbom, vendar ga je mogoče upravljati z gumbom. Osnovni namen priključka Key je preklapljanje med načini. V privzetem načinu, ko na vhod Key ne bo nobenega vhoda, bo naprava delovala v podatkovnem načinu. V podatkovnem načinu bo potekal samo prenos podatkov iz naprave. Ko bo na KEY priključek vhodni signal HIGH, bo naprava prešla v ukazni način.

## 5 PROGRAM ZA IZDELOVANJE MOBILNIH APLIKACIJ

MIT App Inventor je intuitivno, vizualno programsko okolje, ki vsakomur – tudi otrokom – omogoča izdelavo popolnoma funkcionalnih aplikacij za telefone Android, iPhone in tablične računalnike Android/iOS. Tisti, ki se z MIT App Inventorjem šele spoznava, lahko v manj kot 30 minutah pripravijo prvo preprosto aplikacijo. Še več, njihovo orodje, ki temelji na blokih, omogoča ustvarjanje zapletenih, zelo vplivnih aplikacij v bistveno krajšem času kot tradicionalna programska okolja. Cilj projekta MIT App Inventor je demokratizacija razvoja programske opreme z omogočanjem vsem ljudem, zlasti mladim, da preidejo od potrošnje tehnologije k njenemu ustvarjanju.

Majhna skupina zaposlenih in študentov MIT CSAIL, ki jo vodi profesor Hal Abelson, tvori jedro mednarodnega gibanja izumiteljev. Poleg vodenja izobraževalnega dela v zvezi s programom MIT App Inventor in izvajanja raziskav o njenih učinkih ta osrednja ekipa vzdržuje brezplačno spletno okolje za razvoj aplikacij, ki služi več kot 6 milijonom registriranih uporabnikov. [5]



Slika 17: Logotip spletnega portala

(Vir: <http://appinventor.mit.edu/>)

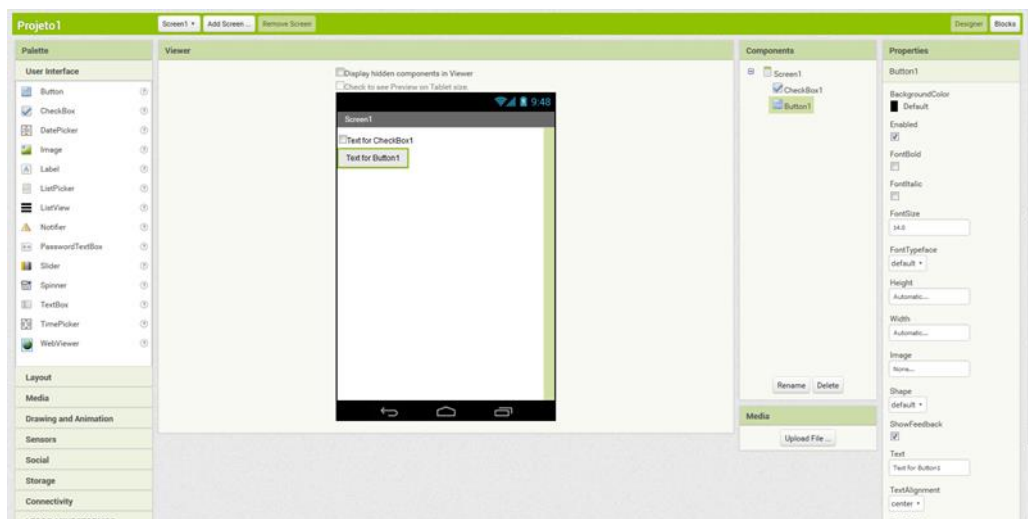
### 5.1 KAKO NAREDITI APLIKACIJO V PROGRAMU APP INVERTOR?

Program App Inventor je sestavljen iz oblikovalca in urejevalnika blokov.

Z oblikovalcem lahko ustvarimo vmesnik aplikacije:

# AVTOMATIZIRANA SUŠILNICA SUHOMESNATIH IZDELKOV

## Raziskovalna naloga



Slika 18: Stran za izdelavo vizualnega izgleda aplikacije

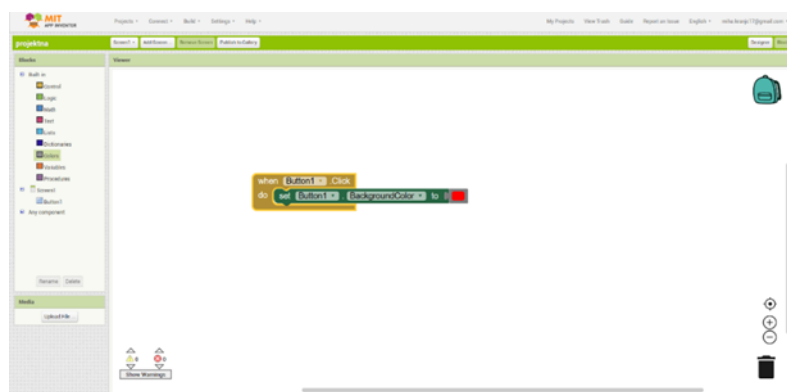
(Vir: [https://en.wikipedia.org/wiki/MIT\\_App\\_Inventor](https://en.wikipedia.org/wiki/MIT_App_Inventor))

**Paleta (Palette):** Poiščite svoje komponente in jih povlecite v pregledovalnik, da jih dodate v aplikacijo.

**Gumb oblikovalca (Designer button):** Kliknite na katerem koli zavihku, da preklopite na zavihek Oblikovalec.

**Lastnosti (Properties):** Izberite komponento na seznamu komponent in tukaj spremenite njene lastnosti (barvo, velikost, obnašanje).

**Pregledovalnik (Viewer):** Povlecite komponente s palete v pregledovalnik in si oglejte, kako bo videti vaša aplikacija. [4]



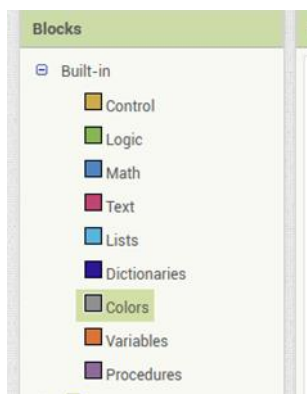
Slika 19: Stran za programiranje aplikacije

(Vir: Osebni vir)

V urejevalniku blokov lahko programirate obnašanje aplikacije s sestavljanjem blokov:

- Vgrajeni predali (Built-in Drawers): Poiščite bloke za splošna vedenja, ki jih boste morda želeli dodati v aplikacijo, in jih povlecite v pregledovalnik blokov.
- Predali za posamezne komponente (Component-Specific Drawers): Poiščite bloke za obnašanja za določene komponente in jih povlecite v pregledovalnik blokov.
- Gumb Bloki (Blocks Button): Kliknite na katerem koli zavihku, da preidete na zavihek Bloki.
- Blok (Block): Blokov za nastavitve obnašanja aplikacije: pritrdite bloke skupaj.
- Pregledovalnik (Viewer): Bloke iz predalov povlecite v pregledovalnik blokov, da vzpostavite razmerja in obnašanje.

Vgrajeni bloki so na voljo ne glede na to, katere komponente so v vašem projektu. Poleg teh jezikovnih blokov ima vsaka komponenta v projektu svoj nabor blokov, ki so značilni za njene dogodke, metode in lastnosti. To je pregled vseh vgrajenih blokov, ki so na voljo v urejevalniku blokov. Vse skupine blokov, so vidne na sliki 20. [4]



Slika 20: Različne skupine blokov za programiranje

(Vir: Osebni vir)

## 5.2 ZAVIHEK »DESIGNER«

Za izdelavo vizualnega izgleda je bilo iz palete potrebno vzeti naslednje komponente:

- Vodoravni razpored (Horizontal Arrangement) – Uporabimo komponento za vodoravno razporeditev za prikaz skupine komponent, ki so razporejene od leve proti desni. Ta komponenta je element oblikovanja, v katerega postavite komponente, ki morajo biti prikazane od leve proti desni. Če želite, da so komponente prikazane ena nad drugo, namesto tega uporabite navpično razporeditev. V vodoravnem razpored so komponente razporejene vzdolž

vodoravne osi, navpično pa so poravnane po sredini. Če je lastnost višina (Height) pri vodoravnem razporedu nastavljena na avtomatsko (Automatic), je dejanska višina ureditve določena z najvišjo komponento v ureditvi, katere lastnost višine ni nastavljena na napolni nadrejeno (Fill Parent). Če je lastnost višina aranžmaja vodoravni razpored nastavljena na avtomatsko in vsebuje samo komponente, katerih lastnost višina je nastavljena na napolni nadrejeno, se dejanska višina aranžmaja izračuna z uporabo samodejnih višin komponent. Če je lastnost višina elementa vodoravni razpored nastavljena na avtomatsko in je prazna, bo višina enaka 100. Če je lastnost širina (Width) aranžmaja vodoravni razpored nastavljena na avtomatsko, je dejanska širina aranžmaja določena z vsoto širin komponent. Če je lastnost širina aranžmaja vodoravni razpored nastavljena na samodejno, se bodo vse komponente, katerih lastnost širina je nastavljena na napolni nadrejeno, obnašale, kot da bi bile nastavljene na samodejno. Če je lastnost širina pri sestavinah vodoravni razpored nastavljena napolni nadrejeno, bodo enako zavzele širino, ki je ne zasedajo druge komponente.

- Izbirnik seznama (List Picker) – Gumb, ki ob kliku prikaže seznam besedil, med katerimi lahko uporabnik izbira. Besedila lahko določite prek oblikovalca ali urejevalnika blokov z nastavitvijo lastnosti Elementi iz niza (ElementsFromString) na njihov niz, ločen med seboj (na primer izbira 1, izbira 2, izbira 3), ali z nastavitvijo lastnosti elementov iz niza v urejevalniku blokov. Z nastavitvijo lastnosti pokaži vrstico filtrov (ShowFilterBar) na da (true) bo seznam omogočal iskanje.
- Oznaka (Label) – Oznake so komponente, ki se uporabljajo za prikaz besedila. Oznaka prikazuje besedilo, ki je določeno z lastnostjo besedilo (Text). Z drugimi lastnostmi, ki jih lahko vse nastavimo v oblikovalcu ali urejevalniku blokov, upravljamo videz in postavitev besedila.
- Gumb (Button) – Gumb z zmožnostjo zaznavanja klikov. Spreminjamo lahko številne vidike njegovega videza in tudi to, ali je klik vklopljen. Njegove lastnosti lahko spremenite v oblikovalcu ali v urejevalniku blokov.
- Odjemalec Bluetooth (BluetoothClient) – S programom odjemalec Bluetooth lahko svojo napravo povežemo z drugimi napravami prek povezave Bluetooth. Ta komponenta za komunikacijo uporablja profil Serial Port Profile (SPP).
- Obveščevalnik (Notifier) – Komponenta obveščevalnik prikazuje opozorilna sporočila in z različnimi metodami ustvarja vnose v dnevnik sistema Android.
- Ura (Clock) – Nevidna komponenta, ki zagotavlja trenutni čas z uporabo notranje ure v telefonu. V redno nastavljenih časovnih intervalih lahko sproži časovnik in izvaja časovne izračune, manipulacije ter pretvorbe. Operacije z datumi in časi, na primer iz datum izbiralnika

(DatePickerja) in časovnega izbiralnika (TimePickerja), se izvajajo z metodami `ura`. Datum in čas sta predstavljena kot `Trenutna v času` (`InstantInTime`) in `Trajanje` (`Duration`). Funkcija – `Trenutna v času` je sestavljena iz `leto` (`Year`), `mesec` (`Month`), `dan v mesecu` (`DayOfMnoth`), `ura` (`Hour`), `minuta` (`Minute`) in `sekunda` (`Second`). `Trenutek` lahko ustvarimo z metodami `naredi takojšnje` (`MakeInstant`), `naredi takojšnje iz milisekund` (`MakeInstantFromMillis`) in `naredi takojšnje iz delov` (`MakeInstantFromParts`). `Trajanje`: čas v milisekundah, ki preteče med posameznimi trenutki. `Trajanje` lahko dobimo z metodo `trajanje`. Predpostavlja se, da so podatki v lokalnem časovnem pasu. Pri pretvorbi v milisekunde ali iz njih se milisekunde za določeno instanco izračunajo od 1. januarja 1970 v UTC (greenwiški srednji čas). Na voljo so tudi metode za pretvorbo trenutka v besedilo. Sprejemljivi vzorci so prazen niz, `MM/dd/YYYY HH:mm:ss` a ali `MMM d, yyyy HH:mm`. Prazen niz bo zagotovil privzeto obliko, ki je `"MMM d, yyyy HH:mm:ss a"` za `FormatDateTime`, `"MMM d, yyyy"` za `Datum` oblikovanja (`FormatDate`).

- `Besedilo v govor` (`TextToSpeech`) – Komponenta `besedilo v govor` glasno govori določeno besedilo. Nastavimo lahko višino in hitrost govora. Nastavimo lahko tudi jezik z vnosom kode jezika. To spremeni izgovorjavo besed in ne dejanskega jezika, ki se govori. Če na primer nastavimo jezik na francoski in govorimo angleško besedilo, bo zvenelo, kot da nekdo govori angleško (`en`) s francoskim naglasom. Državo lahko določim tudi z vnosom kode države. To lahko vpliva na izgovorjavo. Na primer, britanska angleščina (`GBR`) bo zvenela drugače kot ameriška angleščina (`USA`). Vsaka koda države ne vpliva na vsak jezik. Jeziki in države, ki so na voljo, so odvisni od posamezne naprave in jih lahko navedemo z lastnostmi `razpoložljivi jeziki` (`AvailableLanguages`) in `razpoložljive države` (`AvailableCountries`). [4]

### 5.3 ZAVIHEK »BLOCKS«

Vgrajeni bloki so na voljo ne glede na to, katere komponente so v vašem projektu. Poleg teh jezikovnih blokov ima vsaka komponenta v projektu svoj nabor blokov, ki so značilni za njene dogodke, metode in lastnosti. To je pregled vseh vgrajenih blokov, ki so na voljo v urejevalniku blokov. Poznamo:

- `Control blocks` (Bloki za nadzor),
- `Logic blocks` (Logični bloki),
- `Math blocks` (Matematični bloki),

- Text blocks (Besedilni bloki),
- Lists blocks (Blok seznamov),
- Dictionaries blocks (Blok slovarjev),
- Colors blocks (Blok barv),
- Variables blocks (Blok spremenljivk),
- Procedures blocks (Blok postopkov). [4]

## 6 APLIKACIJA ŽELODEC

Aplikacija za prikazovanje stanja temperature in vlage znotraj prostora ter stanja temperature, je programirana v spletnem portalu Mit App Inventor.

Vsebuje:

- gumb za povezavo s HC-05 Bluetooth modulom, ki poveže aplikacijo s sistemom Arduino za prikaz podatkov;
- ali je modul povezan ali ne;
- prikaz temperature in vlage v prostoru;
- trenutni čas in kdaj so bili podatki nazadnje posodobljeni;
- gumb za prekinitev povezave s sistemom.

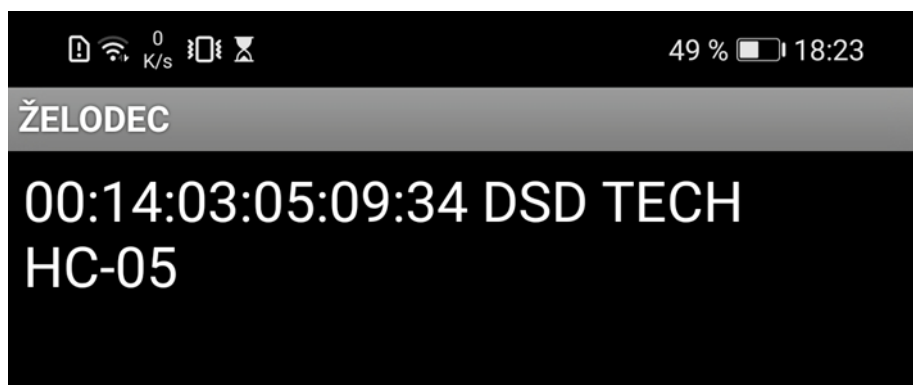
V zavihku »Designer« je izdelana vizualna podoba aplikacije, zavihek »Blocks« pa vsebuje programirljive bloke, ki so programirani po določenih korakih. Izgled aplikacije in njeno delovanje je vidno na spodnjih slikah.



Slika 21: Prikaz delovanja aplikacije pred povezavo z Bluetooth senzorjem

(Vir: Osebni vir)





Slika 22: Ob pritisku na gumb Connect nam prikaže razpoložljivi senzor za povezavo.

(Vir: Osebni vir)



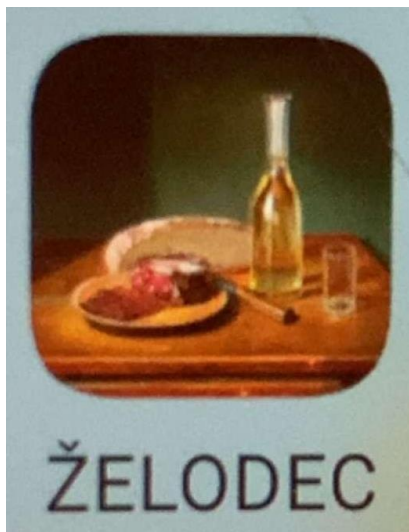
Slika 23: Prikaz, ko pritisnemo tipko za povezavo

(Vir: Osebni vir)

# AVTOMATIZIRANA SUŠILNICA SUHOMESNATIH IZDELKOV

## Raziskovalna naloga

---

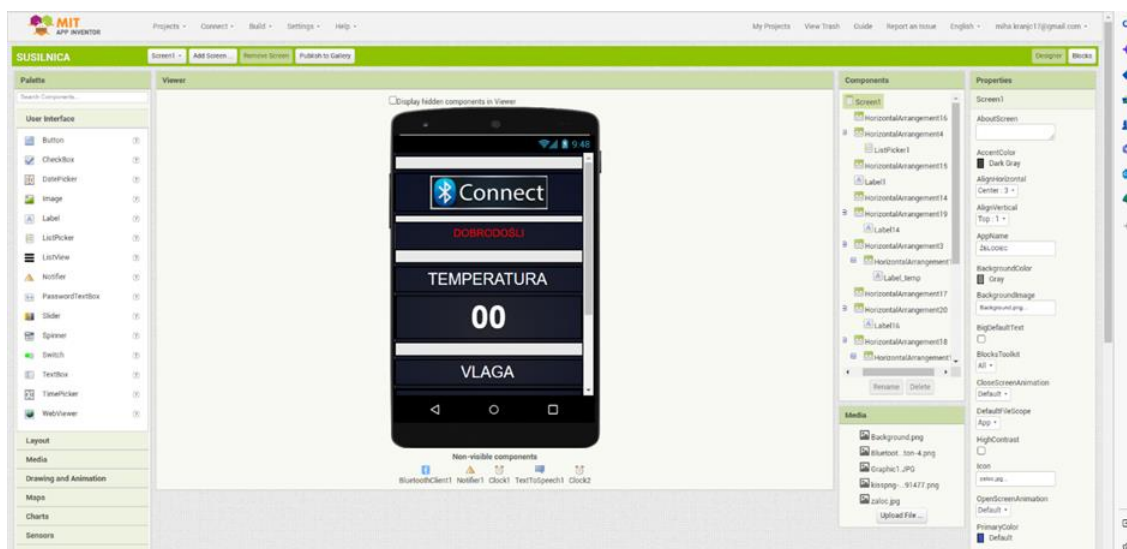


Slika 24: Logotip aplikacije

(Vir: Osebni vir)

## 6.1 NASTAJANJE APLIKACIJE

### 6.1.1 Vizualni del

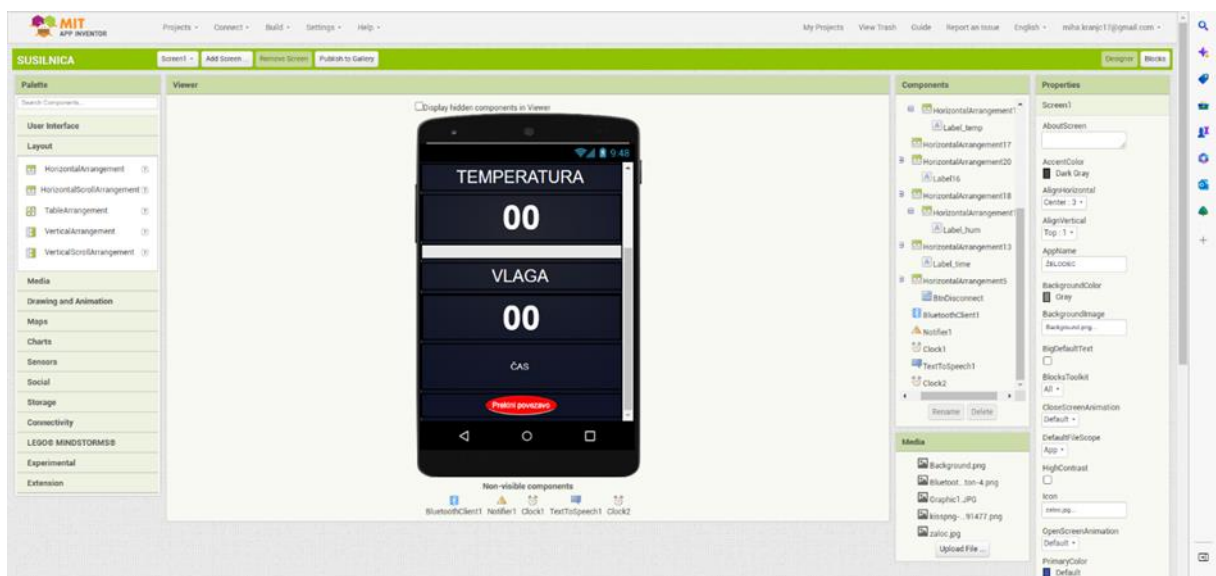


Slika 25: Nastajanje vizualnega izgleda 1.

(Vir: Osebni vir)

# AVTOMATIZIRANA SUŠILNICA SUHOMESNATIH IZDELKOV

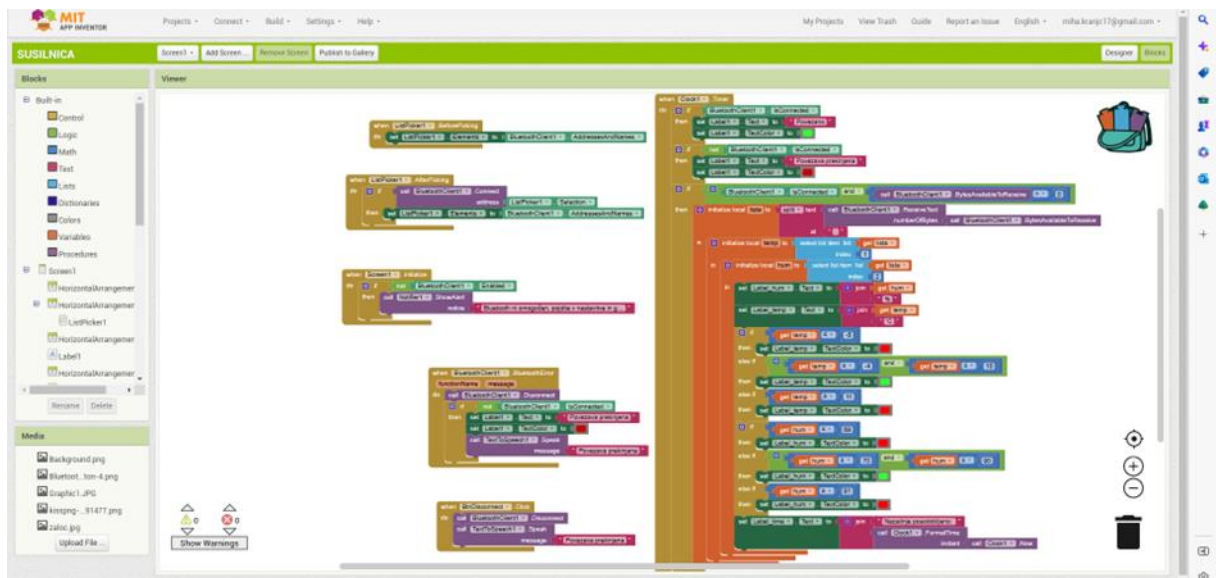
## Raziskovalna naloga



Slika 26: Nastajanje vizualnega izgleda 2.

(Vir: Osebn vir)

### 6.1.2 Blokovni del

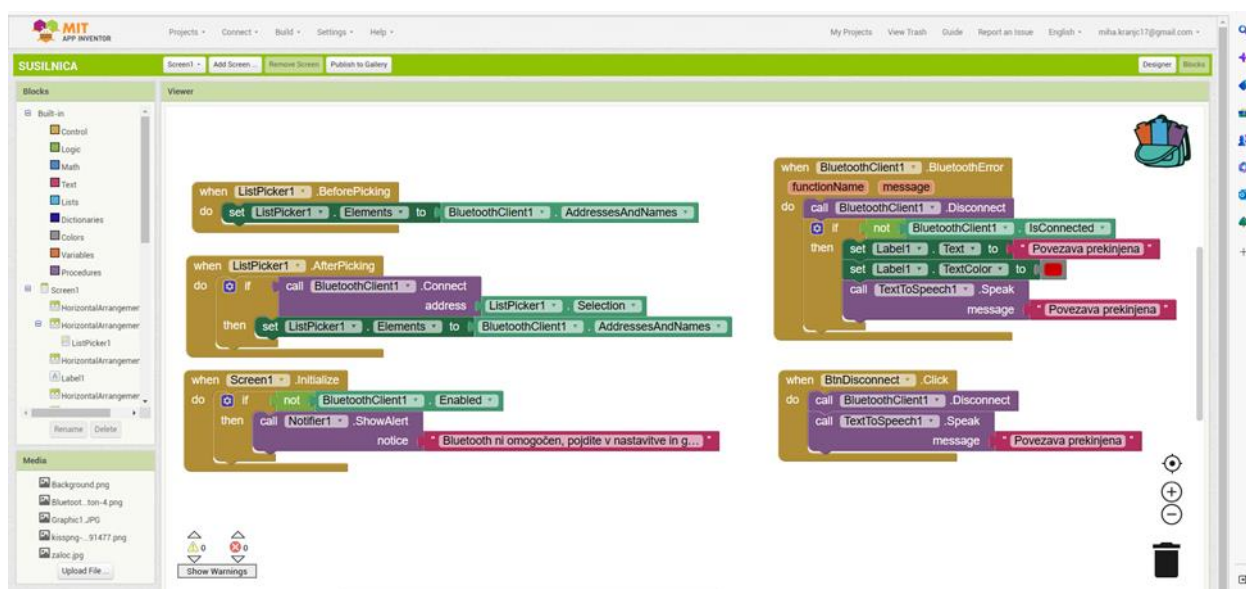


Slika 27: Pogled na vse programirljive bloke, ki omogočajo delovanje

(Vir: Osebn vir)

# AVTOMATIZIRANA SUŠILNICA SUHOMESNATIH IZDELKOV

## Raziskovalna naloga

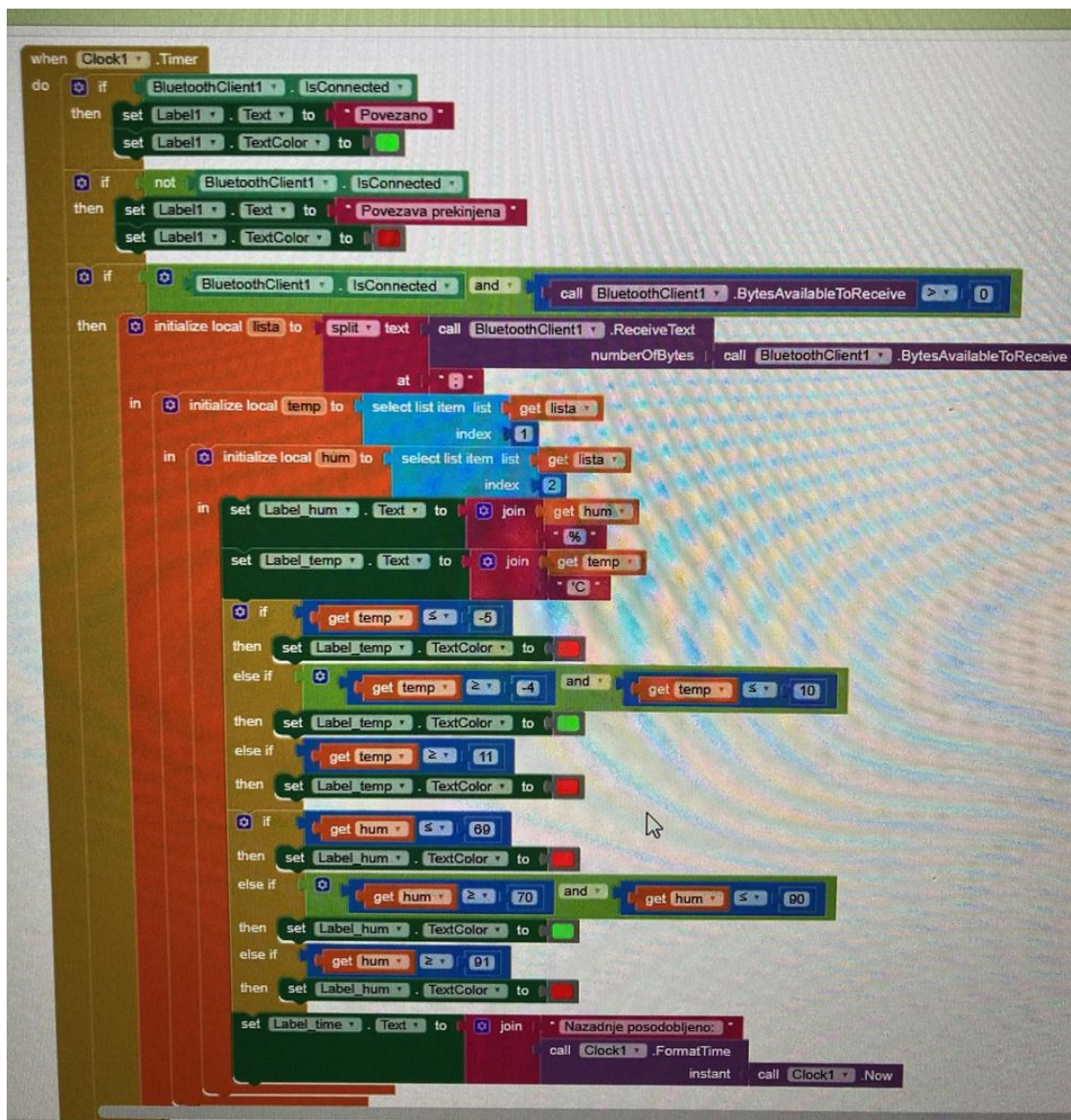


Slika 28: Program za povezavo z Bluetooth modulom

(Vir: Osebni vir)

Na sliki osemindvajset so sestavljeni programirni bloki, ki nam v aplikaciji omogočajo povezavo z Bluetooth modulom HC-05. Ta del programa vsebuje bloke za nadzor, logične bloke, besedilne bloke, bloke barv in bloke postopkov. Prikazano je, da nam ob pritisku na gumb Connect (ListPicker1), prikaže razpoložljive Bluetooth naprave. Če Bluetooth ne dela, nam izpiše, da povezava ni mogoča. Ob morebitni napaki, da pride do prekinitve povezave, se nam na zaslonu z rdečo barvo izpiše »Povezava prekinjena«. Ob pritisku na gumb »Prekinitve povezave« lahko po želji ročno prekinemo povezavo med mobilnim telefonom in sistemom Arduino.

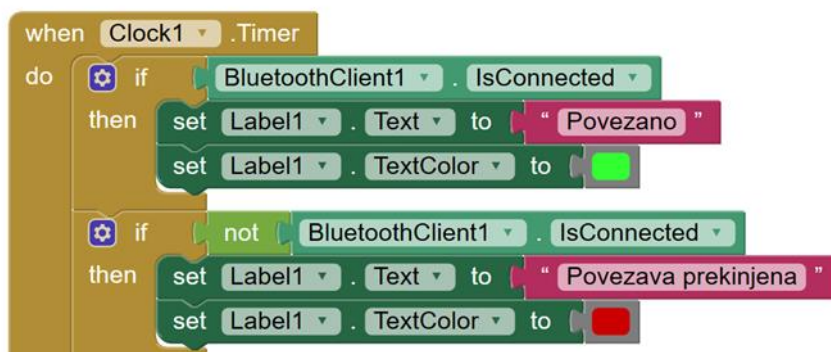




Slika 29: Programirljivi bloki za izpis temperature in vlage na zaslonu

(Vir: Osebni vir)

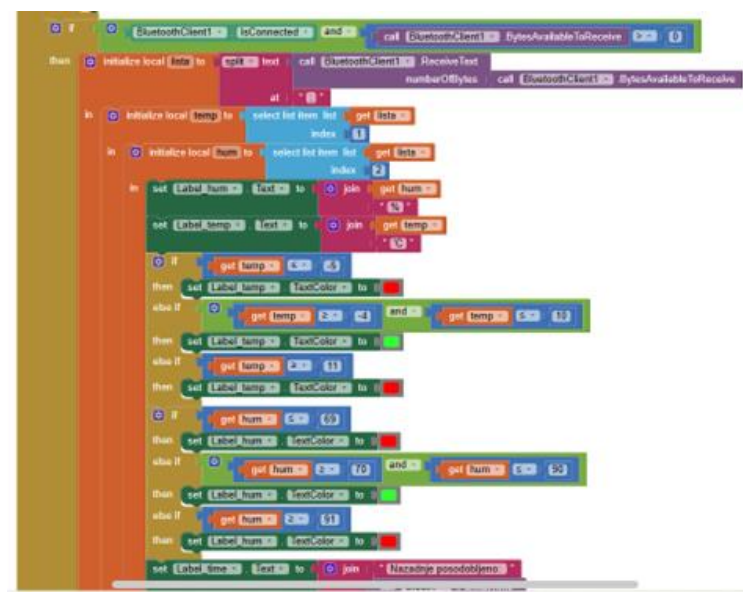
Prvi del programa, vidnega na sliki trideset, nam sporoča, ali smo z Bluetooth povezavo povezani na sistem ali ne. Če smo, se na zaslonu izpiše »Povezano«, če pa ne, pa »Povezava prekinjena«. Uporabili smo bloke za nadzor, logične bloke, besedilne bloke in bloke barv.



Slika 30: Prvi del programa

(Vir: Osebni vir)

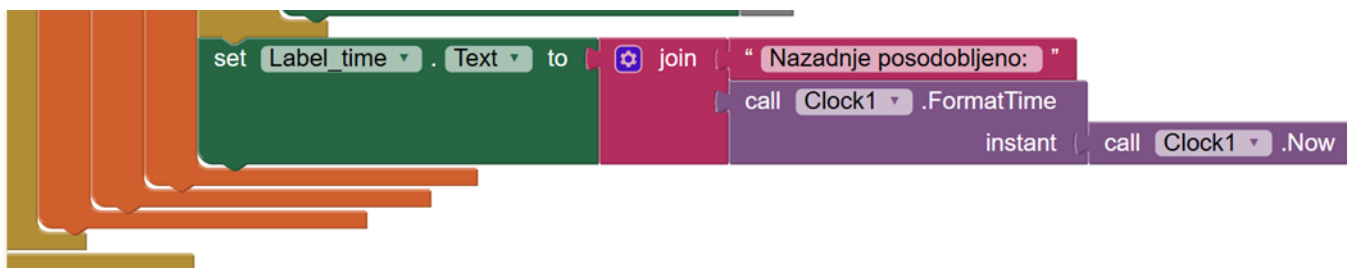
Drugi del programa nam omogoča izpis podatkov o temperaturi in vlagi na zaslonu. Uporabili smo kontrolne bloke, logične bloke, matematične bloke, besedilne bloke, bloke seznamov, bloke barv, bloke spremenljivk in bloke postopkov. Če je temperatura v prostoru manjša ali enaka  $-4^{\circ}\text{C}$ , ali večja ali enaka  $10^{\circ}\text{C}$ , so številke, ki prikazujejo temperaturo, obarvane rdeče. Če pa je temperatura med  $-4^{\circ}\text{C}$  in  $10^{\circ}\text{C}$ , pa so številke obarvane z zeleno. Če je vrednost vlage v prostoru manjša ali enaka 69 %, ali večja ali enaka 91 %, so številke, ki prikazujejo vlago v prostoru, obarvane rdeče. Če pa je vsebnost vlage med 70 % in 90 %, pa so številke obarvane z zeleno. Ti bloki so vidni na sliki enaintrideset



Slika 31: Drugi del programa

(Vir: Osebni vir)

V tretjem delu programa v aplikaciji prikazujemo čas, kdaj so bili podatki nazadnje posodobljeni. Uporabili smo kontrolne bloke, besedilne bloke in bloke postopkov. Na zaslonu bo izpisano »Nazadnje posodobljeno«, zraven tega pa bo prikazan trenutni čas. V primeru, da bo povezava prekinjena, bo čas ostal isti kot takrat, ko je bila nazadnje vzpostavljena povezava.

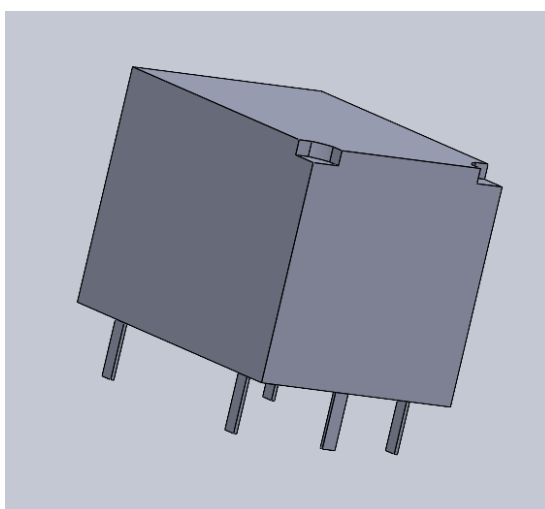


Slika 32: Tretji del programa

(Vir: Osebni vir)

## 7 RELE

Pri priklopu na izvor enosmerne napetosti steče skozi navitje releja enosmerni tok, ki ima za posledico magnetni flux ali magnetni pretok. Ko je magnetni pretok dovolj velik, magnetna sila pritegne kotvo k jedru, kontaktna peresa se sklenejo. Po izklopu releja magnetna sila popusti in kontakti se vrnejo v prvotni položaj. Releji bodo uporabljeni za preklapljanje naprav iz stanja delovanja v stanje mirovanja. Rele bo preklopil, ko bo mikrokrmilnik poslal signal, da se mora določena naprava prižgati. V stanje mirovanja bo preklopil, ko bo temperatura ali vlaga v dovoljenih merah. Spodnja slika prikazuje konstrukcijo releja. [8]



*Slika 33: Konstrukcija releja*

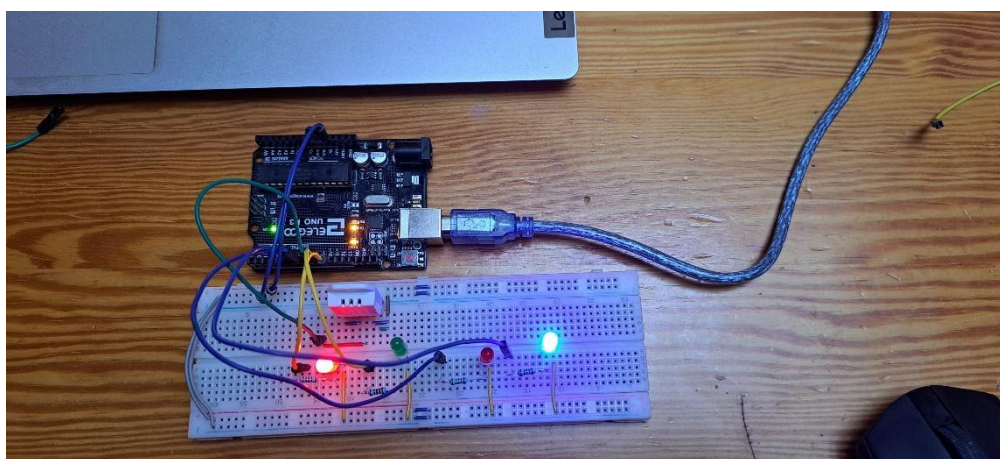
*(Vir: Osebni vir)*



## 8 DELOVANJE SISTEMA

Sistem deluje na podlagi merjenja temperature in vlage. Senzor DHT22 zaznava podatke teh dveh enot v prostoru. Podatke pošlje naprej mikrokrmilniku, ta dane podatke obdela. Po napotkih kode izvrši zadane naloge. Za delovanje sistema smo sprva naredili vezje za uravnavanje nivoja temperature in vlage ter vezje za povezavo mobilne aplikacije z Arduino. V nadaljevanju pa smo vezji zvezali skupaj in nastal je enoten sistem.

### 8.1 VEZJE ZA URAVNAVANJE TEMPERATURE IN VLAGE

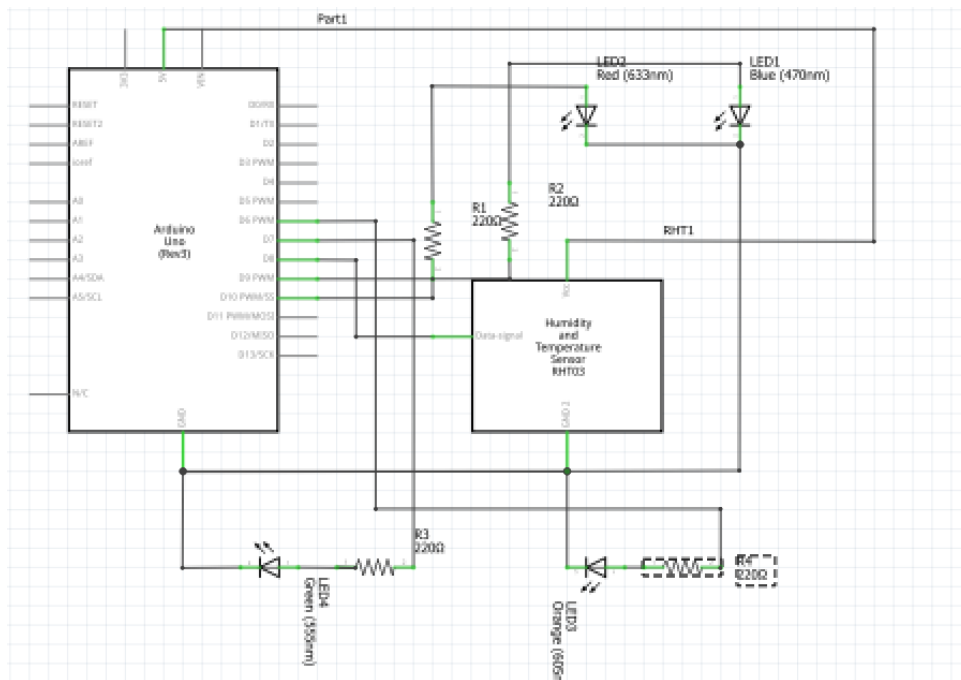


*Slika 34: Testno vezje za DHT22 senzor*

*(Vir: Osebni vir)*

Testno vezje, ki je vidno na sliki 34, je sestavljeno iz 4 LED diod, 4 uporov vrednosti 220  $\Omega$ , testne plošče, DHT22 senzorja, kablov in Arduino UNO mikrokrmilnika. S pomočjo tega vezja smo lahko napisali kodo za uravnavanje temperature in vlage. Ko sta bili ti dve vrednosti previsoki ali prenizki, so se prižgale vnaprej določene LED diode. Ko je ta sistem deloval brezhibno, smo lahko nadaljevali na Bluetooth modul.

AVTOMATIZIRANA SUŠILNICA SUHOMESNATIH IZDELKOV  
Raziskovalna naloga



(Vir: Osebni vir)

```
#include "DHT.h"

#define DHTPIN 8

#define DHTTYPE DHT22

DHT dht(DHTPIN, DHTTYPE);

void setup() {
  dht.begin();

  pinMode(7, OUTPUT);
  pinMode(6, OUTPUT);
  pinMode(10, OUTPUT);
  pinMode(9, OUTPUT);

  Serial.begin(9600);
}

void loop() {
  delay(500);

  float vlaga = dht.readHumidity();
  float temp = dht.readTemperature();

  Serial.print(temp);

  Serial.print(" ; ");

  Serial.println(vlaga);
}
```

(Vir: Osebni vir)

Začetni del kode je namenjen definiranju DHT22 senzorja in priključkov, na katerih bodo povezane naprave, v tem primeru še zaenkrat LED diode. Funkcija `Serial.begin` pa prižge serijski prikazovalnik, s pomočjo katerega lahko na računalniku spremljamo spremembo vlage in temperature. Oznaka `#include` se uporablja za vključitev zunanjih knjižnic v skico. To omogoča programerju dostop do velike zbirke standardnih C knjižnic (zbirka vnaprej izdelanih funkcij) in tudi knjižnic napisanih zlasti za Arduino. `#define` je uporabna C++ komponenta, ki omogoča programerju, da pred sestavo programa da ime stalni vrednosti. Definirane konstante v Arduinu ne zavzemajo nobenega programskega spominskega prostora na čipu. Compiler bo sklice na te konstante zamenjal z določeno vrednostjo ob času sestave. [2]

```
if ( vlaga < 70){  
    digitalWrite(10, HIGH);  
}  
  
else  
{digitalWrite(10, LOW);  
}  
  
if ( vlaga > 90){  
    digitalWrite(9,HIGH);  
}  
else {  
    digitalWrite(9,LOW);  
}
```

Slika 37: Ali funkcija za vlago

(Vir: Osebni vir)

Prvi del ali funkcij je namenjen za vlago. Če je vlaga v prostoru nižja od 70 %, potem je na Arduino vklopljen priključek 10. Če je vlaga večja od 70 %, je ta priključek brez signala. Drugi del ali stavek pa je uporabljen takrat, ko je vlaga v prostoru nad 90 %, takrat pa je vklopljen priključek 9.

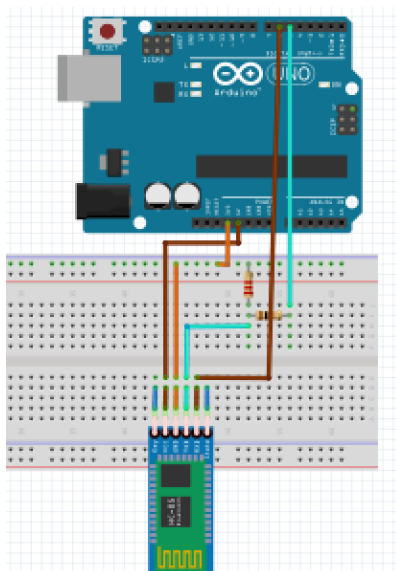
```
if ( temp < -4){  
    digitalWrite(6, HIGH);  
}  
  
else  
{digitalWrite(6, LOW);  
}  
  
if ( temp > 10){  
    digitalWrite(7,HIGH);  
}  
else {  
    digitalWrite(7,LOW);  
}
```

Slika 38: Ali funkcija za temperaturo

(Vir: Osebni vir)

Drugi del ali funkcij je namenjen temperaturi. Če je temperatura nižja od  $-4\text{ }^{\circ}\text{C}$ , je vklopljen priključek 6. Če pa je višja od  $10\text{ }^{\circ}\text{C}$ , je vklopljen priključek 7.

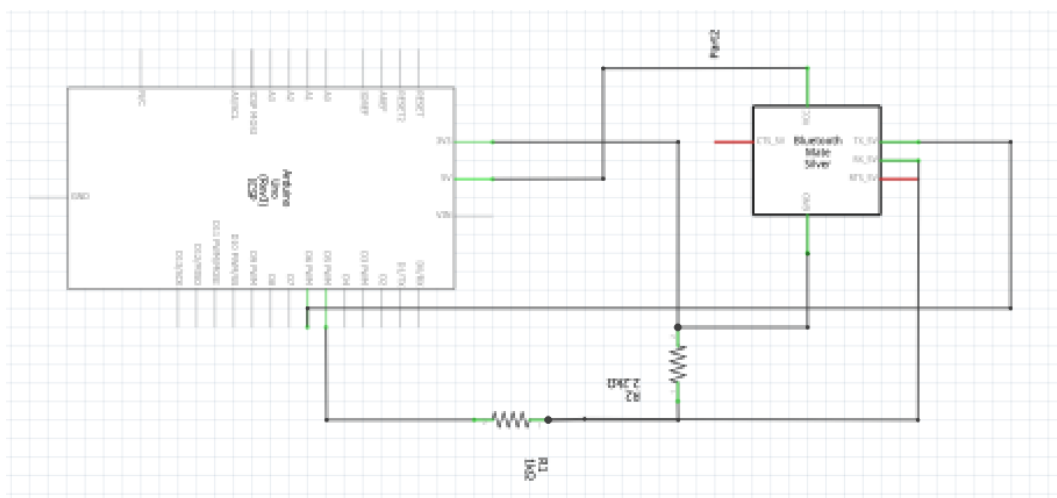
## 8.2 VEZJE ZA BLUETOOTH MODUL



Slika 39: Vezje HC-05 modula

(Vir: Osebni vir)

Bluetooth modul HC-05 deluje na osnovi štirih priključkov. Vezje je vidno na sliki 39. Prvi priključek je napetost 5 V. Drugi priključek je ozemljitev. Tretji priključek je RX, ki je uporaben za prejemanje signalov. RX priključek potrebuje delilnik napetosti, brez katerega bi lahko hitro prišlo do tega, da bi modul pregorel. Za delilnik smo uporabili dva upora, in sicer vrednosti 2.2 k $\Omega$  in 1 k $\Omega$ . Večjega smo povezali na ozemljitev, manjšega pa na prej določen priključek na Arduinou. Modul pa ima še en priključek, in sicer TX priključek za pošiljanje signala.



Slika 40: Risba vezja

(Vir: Osebni vir)

Za povezavo preko Bluetooth modula med aplikacijo in sistemom je potrebno napisati kodo v programu Arduino IDE.

```
#include "DHT.h"
char Incoming_value = 0;

DHT dht (11, DHT22);

#include <SoftwareSerial.h>
SoftwareSerial bt(8,9);

int temp;
int hum;
```

*Slika 41: Prvi del programa*

*(Osebni vir)*

Koda za delovanje HC-05 modula je na slikah od 41 do 43. V tem delu deklariramo vse spremenljivke, ki jih bomo kasneje potrebovali v programu. S funkcijo »SoftwareSerial bt(8,9)« določimo, na katerih priključkih prideta priključka RX in TX iz Bluetooth modula. S funkcijo »int« pa shranjujemo številke oziroma vrednosti temperature in vlage.

```
void setup() {

  Serial.begin(9600);
  dht.begin();
  bt.begin(9600);

  float hum = dht.readHumidity();
  float temp = dht.readTemperature();

}
```

*Slika 42: Drugi del programa*

*(Vir: Osebni vir)*

Funkcijo »void setup()« uporabimo za inicializacijo spremenljivk in načinov priključkov. Funkcija ».begin()« nam omogoča zagnati sistem, DHT senzor in Bluetooth modul. S funkcijo »float« pa naredimo, da vrednosti vlage in temperature izpisuje z decimalko.

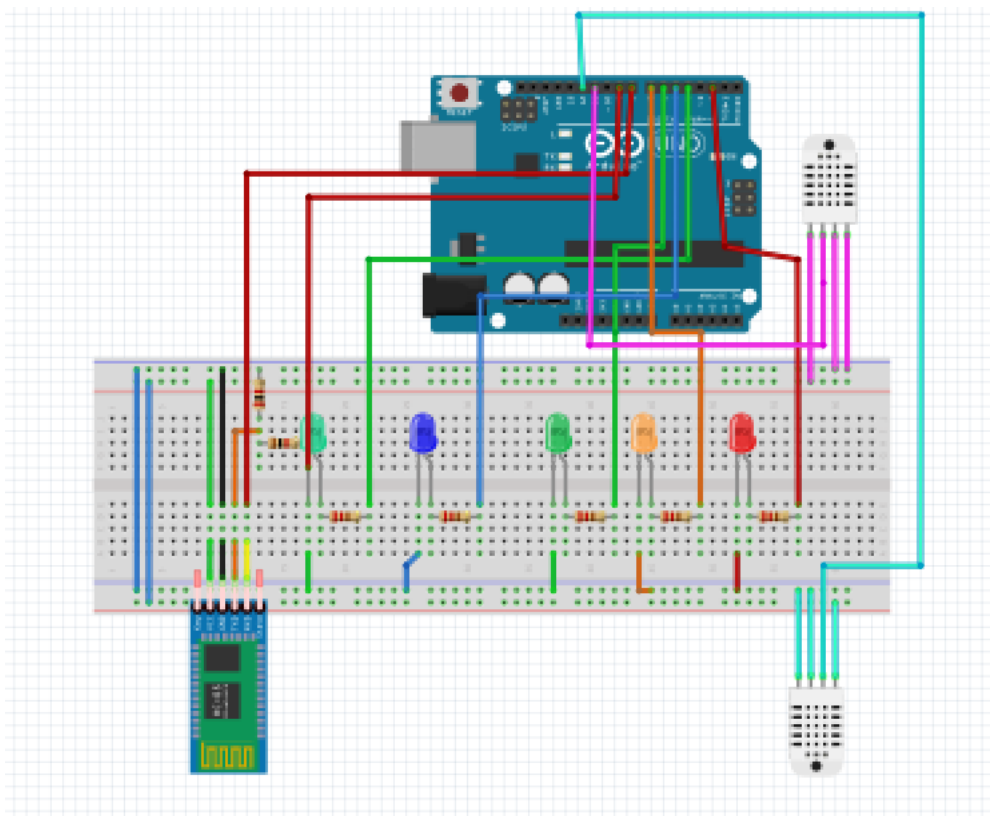
```
void loop(){  
    temp = dht.readTemperature();  
    hum = dht.readHumidity();  
  
    bt.print(temp);  
    bt.print(";");  
    bt.print(hum);  
    bt.println(";");  
    delay(200);  
}
```

Slika 43: Tretji del programa

(Vir: Osebni vir)

Ko ustvarimo funkcijo setup(), ki inicializira in nastavi začetne vrednosti, funkcija loop() naredi zaporedno zanko, ki omogoča, da se program spreminja in odziva. S funkcijo »temp = dht.readTemperature()« določimo metodo, ki jo uporabimo na objektu, da nam vrne temperaturo. S funkcijo »hum = dht.readHumidity()« določimo metodo, ki jo uporabimo na objektu, da nam vrne vlago. S funkcijo »bt.print()« bluetooth modul pošlje podatke oziroma vrednosti temperature in vlage aplikaciji, ki jih nato izpiše na zaslonu.

### 8.3 SKUPNO VEZJE



Slika 44: Skupno vezje v programu Fritzing

(Vir: Osebni vir)

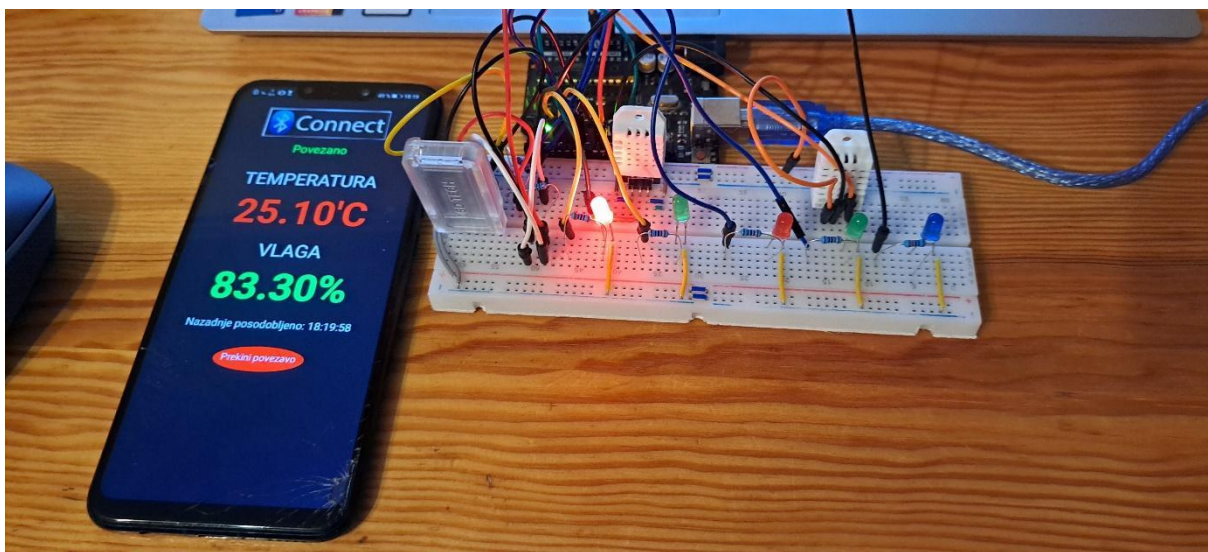
Zgornja slika prikazuje skupno vezje. LED diode v vezju predstavljajo naprave, ki jih bo sistem prižgal in ugašal. Na primer, ko bo vlaga prišla nad določeno vrednostjo, bo DHT senzor to zaznal in poslal signal mikrokrmilniku, v tem primeru Arduino UNO. Mikrokrmilnik bo ta signal predelal in vklopil priključek, ki je določen za napravo, ki bo vlago zmanjševala. Preko tega priključka bo stekla napetost, ki bo prekopila rele. Rele je stalno povezan na napetost 230 V. Ko bo prejel napetost 5 V preko Arduino, bo prekopil napravo iz stanja mirovanja v stanje delovanja. Na sliki vezja so releje zamenjale LED diode. Te so tudi uporabljene za boljše spremljanje, katere naprave so vklopljene in katere ne.





# AVTOMATIZIRANA SUŠILNICA SUHOMESNATIH IZDELKOV

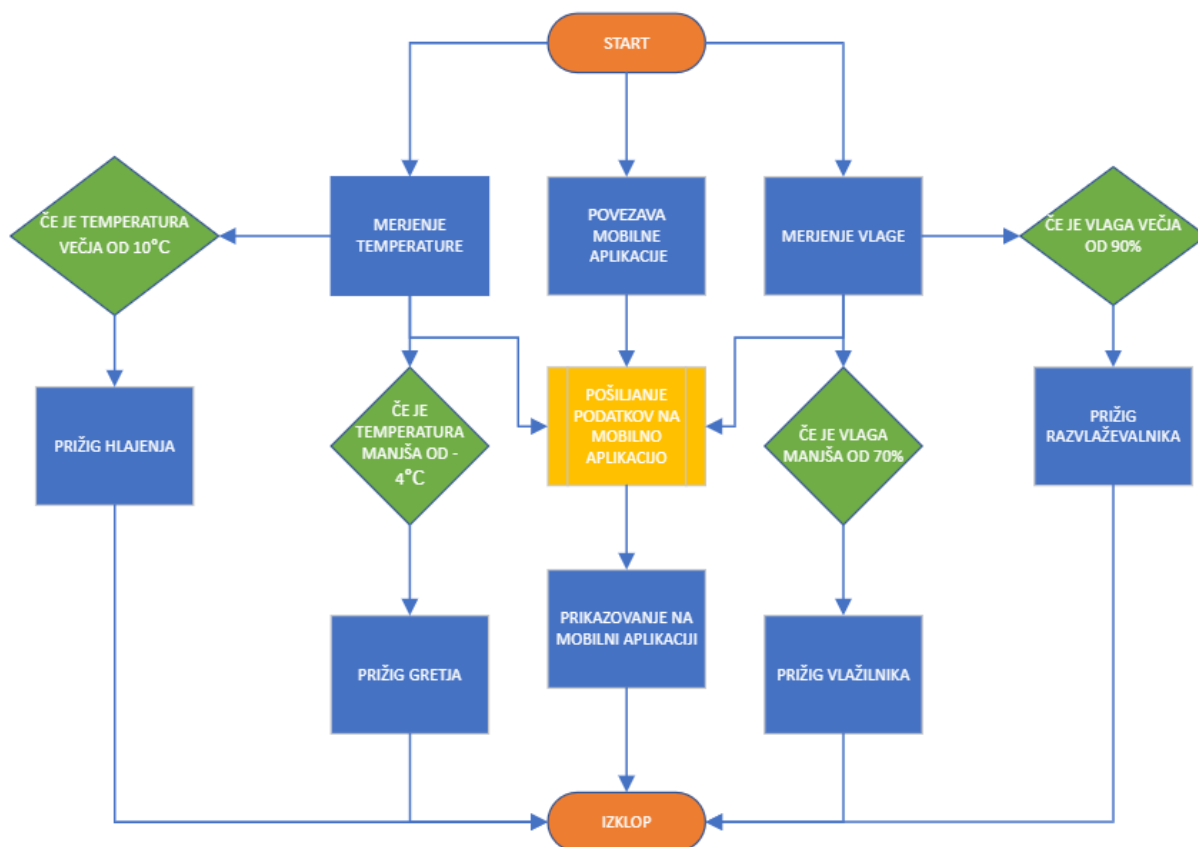
## Raziskovalna naloga



Slika 46: Vezje s priloženo aplikacijo

(Vir: Osebni vir)

Delovanje sistema je prikazano v spodnjem diagramu poteka.



Slika 47: Diagram poteka sistema

(Vir: Osebni vir)

### 8.3.1 Skupna koda

Skupna koda je sestavljena iz treh glavnih delov: za uravnavanje temperature in vlage, za pošiljanje podatkov na mobilno aplikacijo in za odpiranje okna. Z deklaracijo `#define DHTPINz 11` in `#define DHTPINn 12` smo določili priključka DHT22 senzorjev. Ker uporablja sistem zunanji in notranji senzor, smo to določili s črkama `z` in `n` na koncu besede `DHTPIN`. Da smo lahko Bluetooth modul priključili na priključka 8 in 9, smo morali v kodo vstaviti tudi knjižico `SoftwareSerial`. Ta knjižica omogoča, da modul prejema in pošilja podatke, na katerem koli priključku. Drugače je to mogoče samo na priključku 0 in 1. Skupna koda je na slikah od 48 do 53.

```
#include "DHT.h"
char Incoming_value = 0;

#define DHTPINz 11
#define DHTTYPEz DHT22
DHT dhtz(DHTPINz, DHTTYPEz);

#define DHTPINn 12
#define DHTYPEn DHT22
DHT dhtn(DHTPINn, DHTYPEn);

#include <SoftwareSerial.h>
SoftwareSerial bt(8,9); // RX, TX

int tempz;
int vlagaz;

int temp;
int hum;
```

---

Slika 48: Priprava skupne kode

(Vir: Osebni vir)

```
void setup() {  
  
    Serial.begin(9600);  
    dhtz.begin();  
    dhtn.begin();  
    bt.begin(9600);  
  
    pinMode(7, OUTPUT);  
    pinMode(6, OUTPUT);  
    pinMode(5, OUTPUT);  
    pinMode(4, OUTPUT);  
    pinMode(2, OUTPUT);  
  
}
```

*Slika 49: Void setup skupne kode*

*(Vir: Osebni vir)*

V »void setup« zaženemo vse komponente vezja, torej oba senzorja in modul. Zaženemo pa tudi serijski vmesnik za lažje spremljanje spreminjanja stanja. Določimo tudi vse priključke, ki bodo uporabljeni kot preklopniki relejev. Ko bo skozi stekla električna napetost, bo rele preklopil iz odprtega v zaprto stanje in naprava, ki je namenjena recimo večanju vlage v prostoru, se bo avtomatsko vklopila.

```
void loop() {  
  
    float vlagaz = dhtz.readHumidity();  
    float tempz = dhtz.readTemperature();  
  
    float hum = dhtn.readHumidity();  
    float temp = dhtn.readTemperature();  
  
    Serial.print(tempz);  
    Serial.print(" ; ");  
    Serial.print(vlagaz);  
    Serial.println("zunaj");  
  
    delay(500);  
  
    Serial.print(temp);  
    Serial.print(" ; ");  
    Serial.println(hum);  
}
```

Slika 50: Del kode za prikazovanje na serijskem vmesniku

(Vir: Osebni vir)

»Serial.print« napiše v serijski vmesnik veličine, ki jih izmerita oba senzorja. Funkcija »float« pa je podatkovni tip za številke s plavajočimi točkami oziroma število, ki ima decimalno točko.

```
bt.print(temp);  
bt.print(";");  
bt.print(hum);  
bt.println(";");  
delay(200);  
  
if ( hum < 70) {  
    digitalWrite(7, HIGH);  
}  
  
else  
{digitalWrite(7, LOW);  
}  
  
if ( hum > 90) {  
    digitalWrite(6, HIGH);  
}  
else {  
    digitalWrite(6, LOW);  
}
```

Slika 51: Del kode za prenos podatkov na aplikacijo in uravnavanje vlage v prostoru

(Vir: Osebni vir)

V vrsticah, kjer je uporabljena funkcija »bt.print«, pošilja sistem signale Bluetooth modulu, ta pa naprej aplikaciji. V tem delu je napisan tudi del kode za uravnavanje temperature. V spodnjem delu pa je napisan del kode za uravnavanje vlage.

```
if ( temp < -4){  
    digitalWrite(5, HIGH);  
}  
  
else  
{digitalWrite(5, LOW);  
}  
  
if ( temp > 10){  
    digitalWrite(4,HIGH);  
}  
else {  
    digitalWrite(4,LOW);  
}
```

Slika 52: Del kode za uravnavanje temperature

(Vir: Osebni vir)

Za odpiranje okna morajo biti zunanje razmere v določenih omejitvah. To nadziramo z delom kode, ki je prikazan na sliki spodaj. S stavkom if določimo, da se bo okno odprlo, če bodo izpolnjeni pogoji, ki so zadani v oklepaju. Pogoj je napisan s pomočjo funkcije in jo v C++ jeziku napišemo kot dva &&. Z oklepaji je koda urejena tako, da uspešno deluje. V prvem oklepaju je določeno, da mora biti izpolnjen pogoj temperature. Temperatura mora biti med 5 °C in 2 °C. Poleg te omejitve mora biti v mejah tudi vlaga, ki je določena v drugem oklepaju. Ta mora biti med 70 % in 90 %. Če sta izpolnjeni obe zahtevi, določeni v oklepajih, pošlje mikrokrmilnik signal na priključek 2, ta pa preko releja vklopi motorček za odpiranje okna.

```
if (((tempz < 5)&&(tempz > -2))&&((vlagaz < 90)&&(vlagaz > 70))){  
    digitalWrite(2,HIGH);  
    delay(5000)  
    digitalWrite(2,HIGH);  
}  
  
else{  
    digitalWrite(2,LOW);  
}  
}
```

*Slika 53: Del kode za odpiranje okna*

*(Vir: Osebni vir)*

## 9 PREDSTAVITEV REZULTATOV

### 9.1 PRIMERJAVA STROŠKOV IZDELAVE Z CENO TRGA

Naš namen je bil, da ustvarimo sistem, ki bo cenovno ugodnejši, kot sistem, ki bi ga vzpostavilo podjetje, ki se s takšnimi sistemi ukvarja.

Stroški izdelave:

- 1 Arduino UNO: 15.99 €
- 1 testna plošča: 2.99 €
- 2 DHT22 senzorja: 18.66 €
- 1 HC-05 Bluetooth modul: 15.78 €
- 4 releji: 10 €
- 2 m dolg kabel za priklop Arduina: 3.5 €
- Ostali kabli: 10 €
- 2 predupora: 0.10 €
- 3 vtičnice: 8.40 €

Skupna cena komponent je bila 85.42 €. Ker smo vse naprave za uravnavanje temperature in vlage že imeli predhodno v lasti, je ta sistem v primerjavi s tem, kar ponuja trg, veliko cenejši. Po raziskavi trga smo ugotovili, da bi v povprečju vzpostavitev sistema stala okoli 500 €, kar pomeni, da smo s tem prihranili približno 400 €.

### 9.2 PREDSTAVITEV HIPOTEZ

Sistem je v zadostni meri nadzoroval temperaturo in toploto v prostoru ter olajšal nadzorovanje razmer, saj smo jih lahko nadzorovali preko mobilne aplikacije. Rezultati hipotez so razvidni v naslednji tabeli. Prikaz temperature in vlage preko mobilne aplikacije je zelo uporaben, saj ni potrebno priti v prostor, da ju vidimo. Rezultati so vzpodbudni, ker sistem deluje po pričakovanjih, ki smo si jih zadali. Velikokrat pa se zgodi, da se določene naprave za krajši čas vklopijo in nato izklopijo, kar pa je seveda razumljivo, če upoštevamo dejstvo, da so izpolnjene zahteve, ki jih določa Arduino koda.

*(Vir: Osebni vir)*

*Tabela 6: Potrjevanje in zavrnitev hipotez*



AVTOMATIZIRANA SUŠILNICA SUHOMESNATIH IZDELKOV  
Raziskovalna naloga

---

H1 – S sistemom bomo prihranili čas, saj ne bo potrebno vsakodnevno ročno uravnavanje pogojev.	Potrjena.
H2 – Spremljanje toplote in vlage bo lažje, saj bo vse mogoče spremljati preko mobilnega telefona.	Zavrjeno.
H3 – Sistem je cenovno ugodnejši, kot jih ponuja trg.	Potrjena.

Hipotezo 1 smo potrdili, ker je delovanje sistema pripomoglo h temu, da so se suhomesnati izdelki sušili bolje kot prej. S tem pa so potrebovali manj čiščenja in nadziranja. Čeprav velikokrat pride aplikacija prav, smo hipotezo 2 zavrnil. Preden se vzpostavi povezava z Bluetooth modulom lahko preko drugih brezžičnih merilnih naprav, ki so nameščene v prostoru preverimo kakšne so razmere. Sistem pa je veliko cenejši, kot jih ponuja trg, zato smo hipotezo 3 potrdili.

## 10 MOŽNOSTI NADALJNEGA RAZISKOVANJA

Možnosti za nadaljnje raziskovanje je kar nekaj. Lahko bi dodali senzor za zaznavanje plesni, ki bi nam sporočil, ko bi bili izdelki preveč plesnivi in bi jih morali očistiti. Lahko bi tudi dodali napravo za avtomatsko čiščenje želodcev, ki bi jih bilo potrebno očistiti. Menimo, da bi bila uporabna tudi video kamera za slikovni nadzor nad prostorom. Ker je prostor zaprt, bi bilo potrebno dodati kamero s kakovostnim nočnim vidom ali pa napeljati osvetljavo, ki bi jo lahko prižigali na daljavo. Možnost bi bila tudi namestitev naprave za preverjanje suhosti izdelka, ta bi nam sporočila, koliko še izdelku manjka do popolnosti. Lahko pa bi ta sistem ustvarili v nekakšni majhni škatli. V njej pa bi sušili manjše količine mesa in drugih izdelkov, ki se sušijo na zraku. Ustvarili bi lahko tudi robotski sistem, ki bi avtomatsko obračal želodce. Kar bi nam prihranilo še dodaten čas.

## 11 VIRI IN LITERATURA

[1] Arduino opisi (Svetovni splet). (citirano 12. 2. 2023). Dostopno na naslovu:

[https://www.arduino.cc/reference/en/?\\_gl=1\\*10fza9n\\*\\_ga\\*MTk4MjkzNTc5Ni4xNjY0MDkzNTk5\\*\\_ga\\_NEXN8H46L5\\*MTY3NjIxMjE4Mi4yNC4xLjE2NzYyMTIzODQuMC4wLjA](https://www.arduino.cc/reference/en/?_gl=1*10fza9n*_ga*MTk4MjkzNTc5Ni4xNjY0MDkzNTk5*_ga_NEXN8H46L5*MTY3NjIxMjE4Mi4yNC4xLjE2NzYyMTIzODQuMC4wLjA).

[2] Arhiv vremenskih razmer meseca januarja (Svetovni splet). (citirano 9. 2. 2023). Dostopno na naslovu:

<https://meteo.arso.gov.si/met/sl/app/webmet/#webmet==8Sdwx2bhR2cv0WZ0V2bvEGcw9ydlJWblR3LwVnaz9SYtVmYh9iclFGbt9SaulGdugXbsx3cs9mdl5WahxXYyNGapZXZ8tHZv1WYp5mOnMHbvZXZulWYnwCchJXYtVGdlJnOn0UQQdSf>.

[3] DHT22 senzor (Svetovni splet). (citirano 9. 2. 2023). Dostopno na naslovu:

<https://www.instructables.com/How-to-use-DHT-22-sensor-Arduino-Tutorial/>.

[4] MIT app inventor (Svetovni splet). (citirano 16. 2. 2023). Dostopno na naslovu:

<http://appinventor.mit.edu/>.

[5] MIT app inventor Wikipedija (Svetovni splet). (citirano 16. 2. 2023). Dostopno na naslovu:

[https://en.wikipedia.org/wiki/MIT\\_App\\_Inventor](https://en.wikipedia.org/wiki/MIT_App_Inventor).

[6] Modul HC-05 (Svetovni splet). (citirano 10. 2. 2023). Dostopno na naslovu:

<https://microcontrollerslab.com/hc-05-bluetooth-module-priključekout-examples-arduino-applications/>.

[7] Programski jezik C++ (Svetovni splet). (citirano 10. 2. 2023). Dostopno na naslovu:

<https://sl.wikipedia.org/wiki/C%2B%2B>.

[8] Rele opis (Svetovni splet). (citirano 15. 2. 2023). Dostopno na naslovu:

<https://sl.wikipedia.org/wiki/Rele#:~:text=Delovanje%20releja.%20Pri%20priklopu%20na%20izvor%20enosmerne%20napetosti,popusti%20in%20kontakti%20se%20vrnejo%20v%20prvotni%20polo%C5%BEaj>.

[9] 3dsvet – Kaj je Arduino? (Svetovni splet). (citirano 9. 2. 2023). Dostopno na naslovu:

<https://www.3dsvet.eu/kaj-je-arduino/>.