

# Pametni plastenik

---

**Vuković, Mihael**

**Undergraduate thesis / Završni rad**

**2022**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **Virovitica University of Applied Sciences / Veleučilište u Virovitici**

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:165:006857>

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

*Download date / Datum preuzimanja:* **2025-03-17**

*Repository / Repozitorij:*



[Virovitica University of Applied Sciences Repository](#) -  
[Virovitica University of Applied Sciences Academic Repository](#)



**VELEUČILIŠTE U VIROVITICI**

**MIHAEL VUKOVIĆ**

**PAMETNI PLASTENIK**

**ZAVRŠNI RAD**

VIROVITICA, 2022.

# **VELEUČILIŠTE U VIROVITICI**

**Preddiplomski stručni studij Elektrotehnike, smjer Telekomunikacije i informatika**

## **PAMETNI PLASTENIK**

### **ZAVRŠNI RAD**

Predmet : Digitalna elektronika

radi stjecanja zvanja stručnog  
prvostupnika elektrotehnike

Mentor:

Ivan Heđi, dipl. ing., v. pred.

Student:

Mihael Vuković

VIROVITICA, 2022.



OBRAZAC 1b

ZADATAK ZAVRŠNOG RADA

Student/ica: MIHAEL VUKOVIĆ JMBAG: 0307016551

Imenovani mentor: Ivan Heđi, dipl. ing., v. pred.

Imenovani komentor: -

Naslov rada:

*Pametni platenik*

Puni tekst zadatka završnog rada:

Koristeći današnje dostupne tehnologije moguće je automatizirati svakodnevne procese. Automatizacijom možemo stvoriti idealne uvjete za rast povrća. Kontrolom mikrokline mogu se povećati prinosi i kvaliteta proizvodnje. Uz to, uzgoj povrća u pametnom plateniku ne zahtijeva stalnu prisutnost osobe. Vaš zadatak je opisati tehnologije pomoću kojih možemo automatizirati platenik. Predložene tehnologije verificirati uz pomoć makete. Izraditi jedan ekosustav automatizacije platenika za proizvodnju povrća.

Datum uručenja zadatka studentu/ici: 28.07.2022..

Rok za predaju gotovog rada: 09.09.2022.

Mentor:

Ivan Heđi, dipl. ing., v. pred.

Dostaviti:

1. Studentu/ici
2. Povjerenstvu za završni rad - tajniku

## **PAMETNI PLASTENIK**

### **Sažetak**

U ovom radu navedeni su i opisani svi koraci i postupci izrade pametnog plastenika od ideje do gotove makete autonomnog sustava. Na samom početku upoznajemo se sa tehnologijama i komponentama koje su korištene za realizaciju ovoga autonomnog sustava. Nadalje biti će riječi o izradi android aplikacije za praćenje i upravljanje plastenikom uz pomoć svima dostupnog web alata za kojega nije potrebno predznanje iz područja programiranja. Na praktičnom dijelu zadatka upoznajemo se sa izradom električne sheme sustava, crtanja 3d objekata koje će izraditi 3d printer. Prolazimo svim koracima izrade makete plastenika počevši sa izradom stijenki od akrilnog stakla koje se mora precizno iskrojiti kako bi na kraju svaki komad mogao sjesti na svoje mjesto. Biti će obrađeni koraci sastavljanja i izrade tiskane pločice na kojoj je smješten ESP8266 NodeMCU mikro kontroler. Nakon uspješne izrade makete obrađuje se faza testiranja rada sustava.

**Ključne riječi:** pametni plastenik, MIT App Inventor, Firebase, ESP8266  
NodeMCU mikro kontroler, DHT11 senzor temperature i vlage

**Mentor:** Ivan Heđi, dipl. ing., v. pred.

## Basic documentation card

## SMART GREENHOUSE

**Abstract**

This paper lists and describes all the steps and procedures of making a smart greenhouse, from the idea to the finished model of the autonomous system. At the very beginning, we are introduced to the technologies and components that were used to realise this autonomous system. Furthermore, it will be discussed about the creation of an android application for monitoring and managing the greenhouse with the help of a web tool available to everyone, which does not require prior knowledge in the field of programming. In the practical part of the task, we get acquainted with the creation of the electrical scheme of the system, the drawing of 3D objects that will be created by a 3D printer. We go through all the steps of making a mock-up of a greenhouse, starting with the creation of acrylic glass walls, which must be precisely cut so that each piece can fit in its place at the end. The steps of assembling and manufacturing the printed circuit board on which the ESP8266 NodeMCU micro controller is located will be covered. After the successful creation of the model, the phase of testing the operation of the system is processed.

**Keywords:** Smart Greenhouse, MIT App Inventor, Firebase, ESP8266 NodeMCU microcontroller, DHT11 temperature and humidity sensor

**Supervisor:** Ivan Heđi, dipl. ing., v. pred.

## Sadržaj

1.	Uvod.....	1
2.	Primijenjene tehnologije .....	2
2.1.	ESP8266 NodeMCU .....	2
2.2.	Arduino integrirano razvojno okruženje .....	3
2.3.	Firestore .....	3
	.....	4
2.4.	3D ispis.....	4
2.5.	App Inventor .....	6
2.5.1.	Funkcionalnosti aplikacije za pametni plastenik.....	7
2.5.2.	Vizualni dio aplikacije .....	7
2.5.3.	Blokovsko dodjeljivanje funkcija.....	7
2.6.	DHT 11.....	8
2.7.	Senzor vlažnosti tla .....	8
2.8.	Senzor razine vode .....	9
2.9.	Servo motor.....	10
2.10.	Relejni modul.....	11
3.	Dijagram i shema .....	12
3.1.	Funkcijski dijagram sustava .....	12
3.2.	Električna shema sustava .....	13
4.	Realizacija funkcionalne makete pametnog plastenika.....	14
4.1.	Izrada i provjera rada elektronike.....	14
4.2.	Izrada programskog koda .....	14
4.3.	Testiranje rada elektronike i napisanoga koda .....	19
4.4.	Izrada tiskane pločice .....	19
4.5.	Izrada oklopa od akrilnog stakla .....	21
4.6.	Izrada sustava za navodnjavanje .....	22
4.7.	Izrada i pregled funkcija mobilne aplikacije .....	22
4.8.	Testiranje makete .....	25
5.	Zaključak.....	26
6.	Popis literature .....	27
7.	Popis slika .....	28
8.	Prilozi.....	29

# 1. Uvod

U današnje vrijeme klimatskih promjena i sve češćih ekstremnih vremenskih nepogoda napredne tehnologije uzgoja voća i povrća sve su traženije. Kako bi unatoč svim tim nepogodama imali dobar urod ljudi se sve više okreću plasteničkoj proizvodnji. Plastenik svojom najčešće metalnom konstrukcijom i najlonskom folijom izolira poljoprivredne kulture koje rastu u njemu od vanjskih utjecaja te je tako pogodan za stvaranje idealnih mikroklimatskih uvjeta u njemu. Tako ćemo u ovom radu uz pomoć nama danas dostupne tehnologije automatizirati plastenik tako da bi uz kontroliranu mikroklimu dobili što veći urod i kako bi produljili sezonu iskorištenja poljoprivrednih kultura unutar samog plastenika. Također automatizirani plastenik ne zahtijeva stalnu prisutnost osoba jer je proces automatiziran. Nadzor temperature zraka, vlage zraka i vlage tla se može nadzirati putem aplikacije za pametne telefone.

Proces automatizacije izvršavati će se uz pomoć ESP8266 NodeMCU mikro kontrolera koji će pomoću svojih senzora prikupljati podatke kao što su temperatura zraka, vlaga zraka i vlaga tla. Na temelju tih mjerenja izvršavati će se akcije provjetravanja, grijanja ili navodnjavanja. Sve izmjerene vrijednosti se šalju u Firebase bazu podataka kako bi bili dostupni korisnicima aplikacije za pametne telefone.



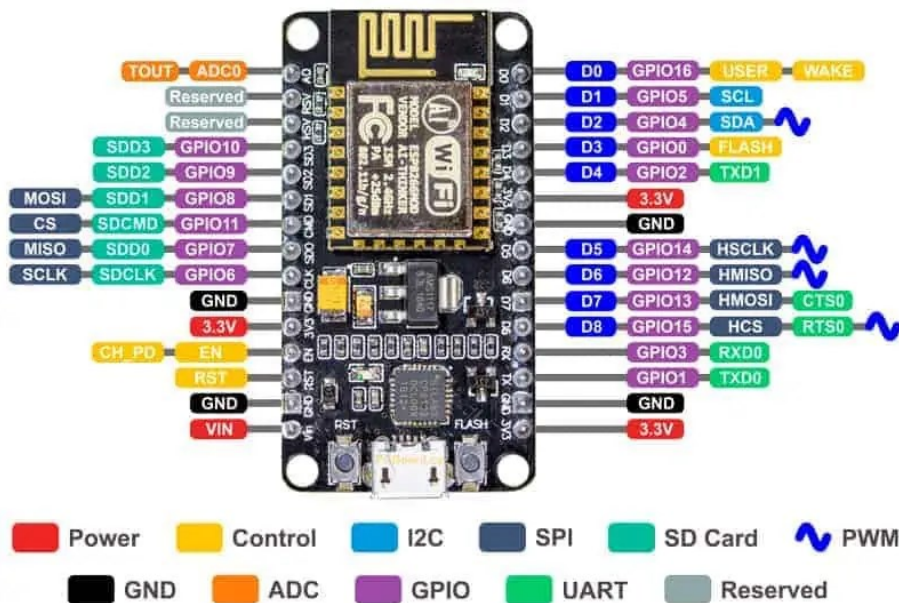
## 2. Primijenjene tehnologije

U daljnjem tekstu rada upoznati ćemo se sa tehnologijama i načinom rada komponenti koje su primijenjene u procesu automatizacije plastenika. Tako ćemo u nastavku spomenuti ESP8266 NodeMCU, DHT11, senzor vlage tla, Firebase, App Inventor, 3D ispis i još neke komponente koje će biti navedene u nastavku.

### 2.1. ESP8266 NodeMCU

ESP8266 NodeMCU je jeftini 32-bitni mikro kontroler koji je zbog svoje male cijene postao gotovo nezaobilazan u svim IoT projektima. NodeMCU se temelji na svom ESP8266 Wi-Fi mikročipu, koji ima ugrađen TCP/IP mrežni softver. Raspoloživo je sa 32KiB memorije za upute i 80KiB za korisničke podatke. Radi na frekvenciji 2.4GHz te također podržava WPA i WPA2 standard. Ulazni napon je od 4.5V do 10V no na sebi ima regulator napona koji stabilizira napon na njegov radni napon od 3.3V.

Slika 1. ESP8266 NodeMCU

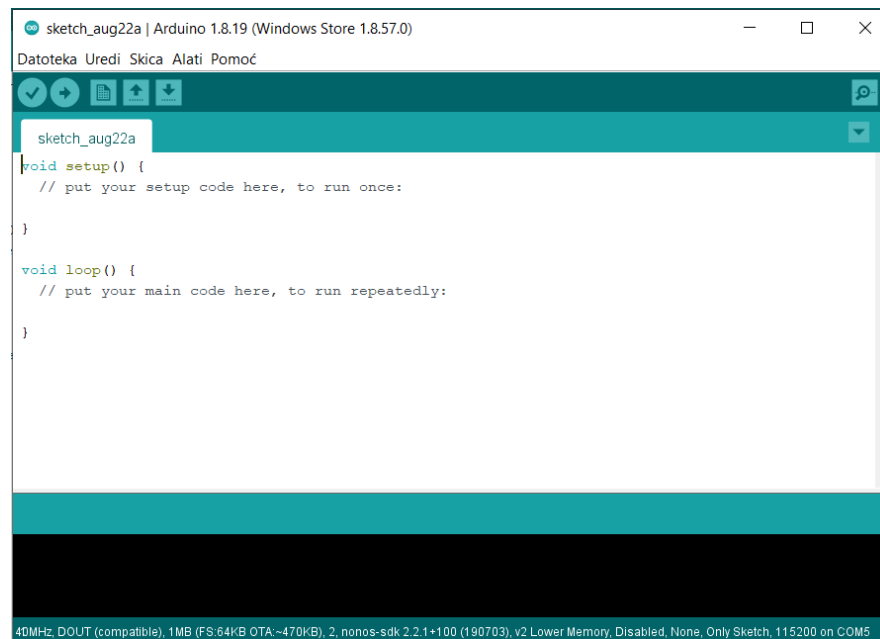


Izvor: <https://www.make-it.ca/wp-content/uploads/2021/09/nodemcu-pinout-functions.jpg> (20.08.2022.)

## 2.2. Arduino integrirano razvojno okruženje

Za programiranje ESP8266 NodeMCU-a koristimo besplatno arduino integrirano razvojno okruženje koje se sastoji od uređivača teksta u kojem se piše kod. Prije samog pisanja koda potrebno je u izborniku "Alati" izabrati vrstu pločice koja se koristi.

Za programiranje se koristi jezik koji je baziran na C i C++ jeziku.



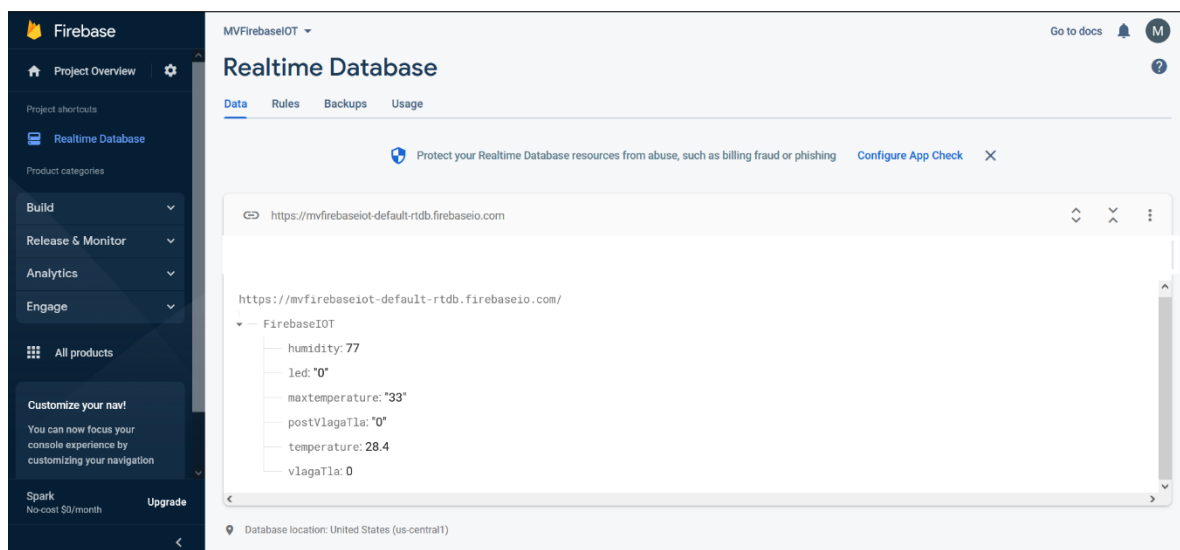
**Slika 2. Arduino integrirano razvojno okruženje**

*Izvor: autor*

Nakon što se napiše kod za NodeMCU potrebno ga je provjeriti tako što se stisne gumb sa kvačicom koji se nalazi na traci odmah iznad uređivača teksta. Ako je provjera bila uspješna prelazi se na kompiliranje koda i prijenos na pločicu mikro kontrolera.

## 2.3. Firebase

Firebase je platforma za izradu mobilnih i web aplikacija u vlasništvu Google-a. Firebase pruža alate za praćenje analitike, izvještavanje i popravljavanje padova aplikacija i stvaranje eksperimentalnih i marketinških proizvoda. Jedan od prvih proizvoda koje je Firebase izbacio je bila usluga Realtime Database (baza podataka koja se mijenja u stvarnom vremenu), a upravo ona se koristi za spremanje očitavanja senzora DHT11 i senzora vlažnosti tla, također se u toj bazi nalaze i varijable kao što su željena temperatura zraka, željena vlaga tla i vrijednost koja pokazuje radi li svijetlo za rast ili je isključeno.

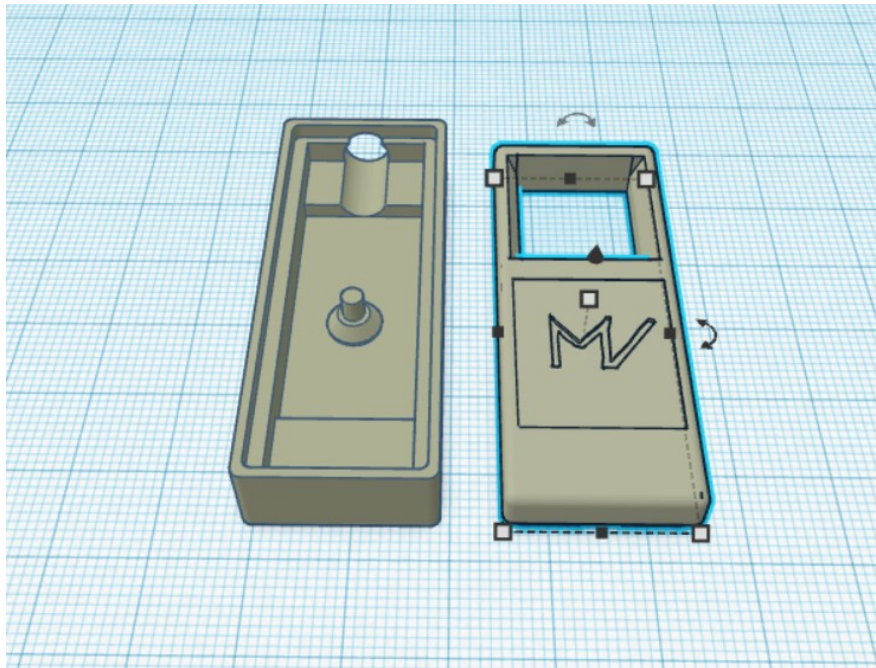


Slika 3. Firebase Realtime Database korisničko sučelje

Izvor: autor

## 2.4. 3D ispis

3D ispis je proces stvaranja trodimenzionalnih objekata od različitih materijala na način da se materijal nanosi sloj po sloj. To je proces koji omogućuje brzu izradu prototipa nekog objekta te je korišten prilikom izrade autonomnog plastenika. Za potrebe plastenika tehnikom 3d ispisa izrađeno je kućište za DHT11 senzor, linearni otvarač otvora plastenika sa nosačem servo motora, kućište senzora vlage tla i nosač pumpe za navodnjavanje od materijala skraćeno zvanog PLA( polilaktična kiselina) to je BIO plastični materijal. Za 3d ispis potrebno je napraviti 3d model objekta koji se misli izraditi te ga je potrebno spremi kao STL datoteku. STL datoteka je jednostavni prenosivi format u kojem se 3d model objekta interpretira pomoću sitnih trokuta, bitno je da rezolucija tih trokuta ne bude prevelika jer može doći do preklapanja trokuta, a ako je rezolucija preniska onda se pojavljuju rupe u objektima.



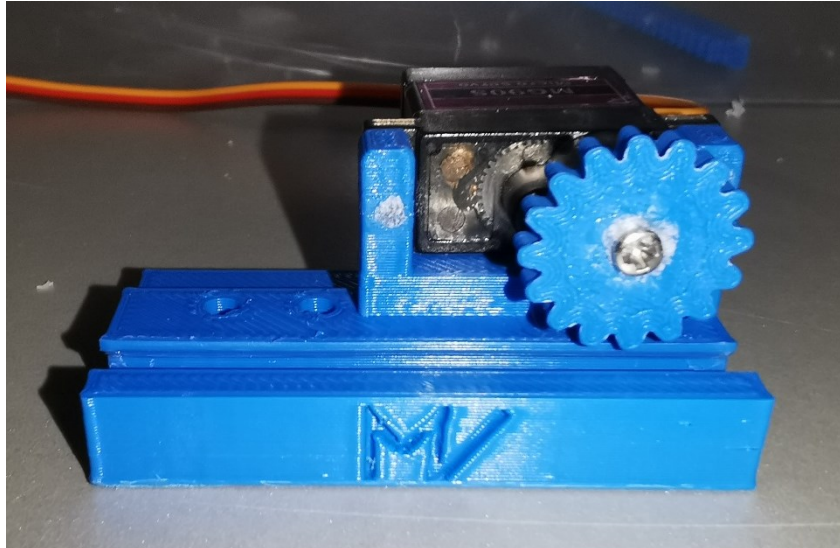
**Slika 4. 3D model DHT11 kućišta**

*Izvor: autor*



**Slika 5. DHT11 kućište izrađeno tehnikom 3d ispisa**

*Izvor: autor*



Slika 6. Linearni otvarač krovnog otvora izraden tehnikom 3d ispisa

*Izvor: autor*

## 2.5. App Inventor

Za izradu aplikacije za pametni telefon sa operacijskim sustavom Android koristi se MIT App Inventor. Pokrenuo ga je Google, a kasnije ga je preuzeo Massachusetts Institute of Technology (MIT). App Inventor je alat za izradu mobilnih aplikacija namijenjen svima jer za njegovu upotrebu ne treba imati iskustva u izradi aplikacija.

Proces izrade aplikacije se odvija u dva koraka designer i blocks. U designer dijelu osoba koja radi aplikaciju uzima elemente kao što su gumbi, padajući izbornici, prozorčići za slike i još puno toga, zatim ih slaže po svojoj volji tako da dobije svoj željeni vizualni dio aplikacije. Nakon uspješnog dizajniranja vizualnog dijela aplikacije prelazi se na block dio gdje se odabranim elementima dodjeljuju funkcije uz pomoć blokova koji se slažu jedan na drugi ili jedan u drugi u zavisnosti kakvu konačnu funkcionalnost želi osoba koja izrađuje aplikaciju. App Inventor ima mogućnost i testiranja aplikacije za vrijeme radan na njoj, a to se postigne odlaskom na izbornik Conect i odabirom AI companion. Nakon što skeniramo dobiveni QR kod možemo na našem pametnom telefonu isprobati aplikaciju. Za korištenje App Inventora nije potrebno ništa preuzimati nego sve se obavlja kroz internetski preglednik u web aplikaciji. Nakon završetka faze razvoja aplikacije korisnik može svoju aplikaciju podijeliti i sa drugima ili ju može preuzeti na svoj pametni telefon.

### 2.5.1. Funkcionalnosti aplikacije za pametni platenik

Za potrebe pametnog platenika izrađena je aplikacija u App Inventoru. Zahtjevi koje je aplikacija morala ispuniti su mogućnost očitavanja temperature zraka, vlage zraka, vlage tla i status radi li svijetlo za rast ili ne. Također potrebno je bilo napraviti i padajući izbornik kojim se namještaju željena temperatura u plateniku i željena vlaga tla. Za paljenje i gašenje svijetla za rast izrađena su dva gumba za paljenje i gašenje svijetla za rast.

### 2.5.2. Vizualni dio aplikacije

Za izradu vizualnog dijela aplikacije koristi se button, label, spinner, text te horizontalna i vertikalna poravnanja. Nakon završetka slaganja vizualnog dijela aplikacije još je potrebno staviti sličicu za ikonu aplikacije.

### 2.5.3. Blokovsko dodjeljivanje funkcija

Završenim postupkom slaganja vizualnog izgleda aplikacije prelazi se u karticu " blocks " gdje se sada za svaki pojedini element dodaju funkcionalnosti koje se slažu kao kockice.



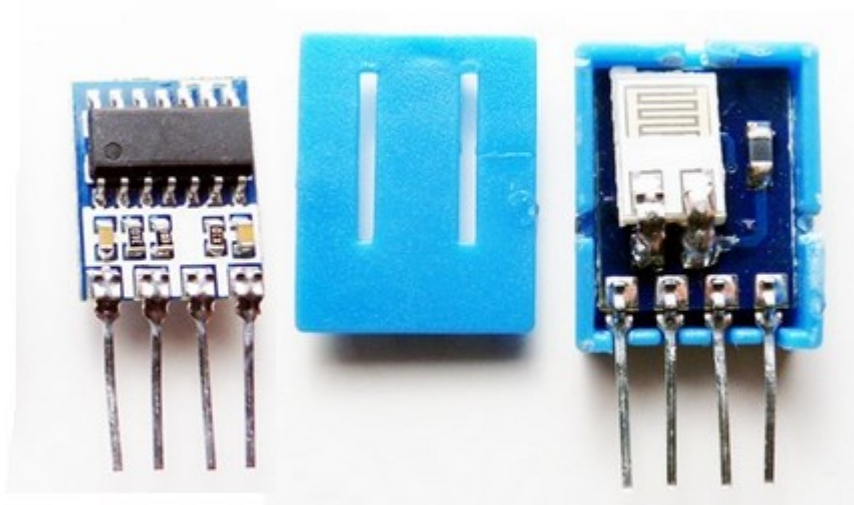
Slika 7. App Inventor blokovi *Izvor: autor*

Na slici 7 se može vidjeti kako izgleda primjer blokovskog programiranja događaja, pritiskom na gumb ON sprema se vrijednost 1 u bazu podataka u varijablu "led" koja u Firebase predstavlja podatak da li je svijetlo za rast upaljeno ili je svijetlo za rast ugašeno.

## 2.6. DHT 11

DHT 11 je relativno jednostavan i cjenovno pristupačan senzor koji služi za mjerenje temperature i vlažnosti zraka. Unutar kućišta se nalaze dvije vrste senzora. Za mjerenje temperature zraka senzor koristi termistor s negativnim temperaturnim koeficijentom, koji smanjuje vrijednost otpora s porastom temperature.

Za mjerenje vlage zraka koristi maleni kapacitivni senzor koji mijenja svoj kapacitet u zavisnosti od količine vlage između elektroda koje su postavljene jedna nasuprot druge. Temperaturni raspon DHT 11 senzora je od 0 do 50 stupnjeva Celzija. Raspon vrijednosti vlažnosti zraka koju može ovaj senzor mjeriti se kreće od 20 do 80% s 5% točnosti. U praksi se pokazalo da on ide iznad i ispod tih vrijednosti ali vjerojatnost točnog očitavanja je tada manja. Brzina uzorkovanja ovog senzora je 1Hz što nam daje podatak da se vrši jedno očitavanje svake sekunde. Radni napon senzora je od 3 do 5 volta, dok je maksimalna struja koju senzor prilikom mjerenja povlači 2,5 mA.



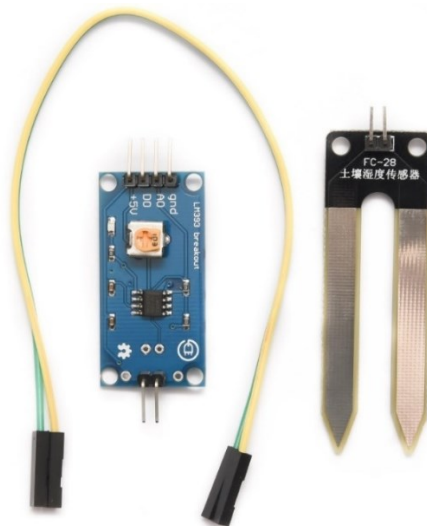
Slika 8. DHT 11 senzor

Izvor: <https://osoyoo.com/wp-content/uploads/2017/07/arduino126-3.jpg> (21.08.2022.)

## 2.7. Senzor vlažnosti tla

Senzor vlažnosti tla je jeftin i dostupan senzor čiji se rad zasniva na očitavanju promjene električnog otpora između elektroda senzora. Senzor se sastoji od dva dijela od mjerne sonde i elektronike sa komparatorom koji pretvara analogni signal očitavanja u digitalni. Mjerenje ovim senzorom se vrši tako da se zabode u tlo i tada se mjeri električni otpor tla,

što je tlo vlažnije električni otpor je manji, a što je tlo više suho to je električni otpor veći. Senzor dobiva analogna očitavanja u vrijednostima od 0 do 1023, ako želimo da nam vrijednost bude u postotcima te se vrijednosti moraju skalirati u vrijednosti od 0% do 100%. Ovaj senzor također omogućuje da ima analogni i digitalni izlaz pa u zavisnosti koji izlaz se koristi mora se paziti prilikom spajanja da se na pravu nožicu priključi senzor. Velika mana ovog senzora je što vrlo brzo dolazi do korodiranja i propadanja elektroda i ispuštaju se metali iz elektroda u zemlju i mogu na taj način trovati biljke. Jedno od rješenja je da senzor ne radi cijelo vrijeme već da mu se pušta napajanje samo onda kada vrši mjerenje, takvo postupanje će produžiti radni vijek senzora. Također nije ovo jedina vrsta senzora za mjerenje vlage tla postoji i kapacitivni senzori za mjerenje vlage tla kojemu su elektrode zaštićene od vanjskog okoliša i taj senzor mjeri kapacitet između dviju elektroda i na temelju toga određuje vlagu tla.



Slika 9. Senzor vlažnosti tla

Izvor: <https://e-radionica.com/hr/blog/2019/02/17/kkm-senzor-vlaznosti-tla/> (21.08.2022.)

## 2.8. Senzor razine vode

Senzor razine vode omogućuje detekciju visine vode u spremniku za navodnjavanje na način da se u plovku senzora nalazi magnet, a na vrhu senzora se nalazi magnetni prekidač čiji se kontakti otvore kada magnet dođe blizu njega.





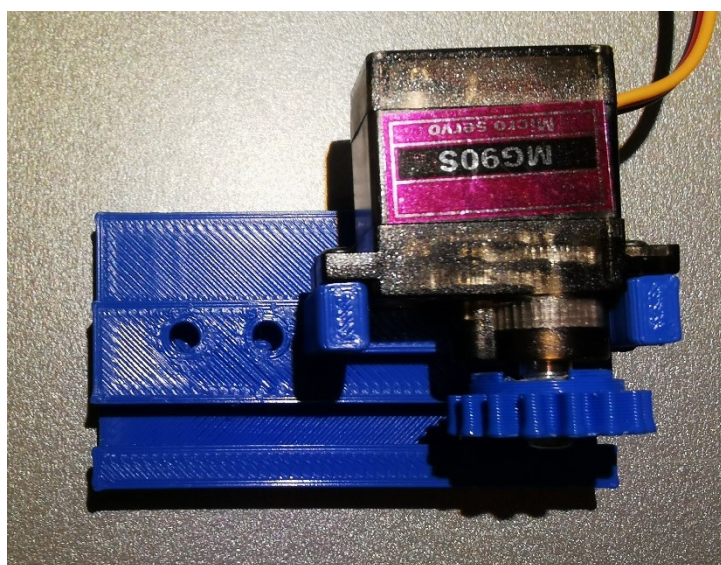
**Slika 10. Senzor razine vode**

*Izvor: <https://e->*

*[eadionica.com/media/catalog/product/cache/5/image/9df78eab33525d08d6e5fb8d27136e95/\\_1/\\_1088997-edit.jpg](https://e-eadionica.com/media/catalog/product/cache/5/image/9df78eab33525d08d6e5fb8d27136e95/_1/_1088997-edit.jpg) (21.08.2022.)*

## **2.9. Servo motor**

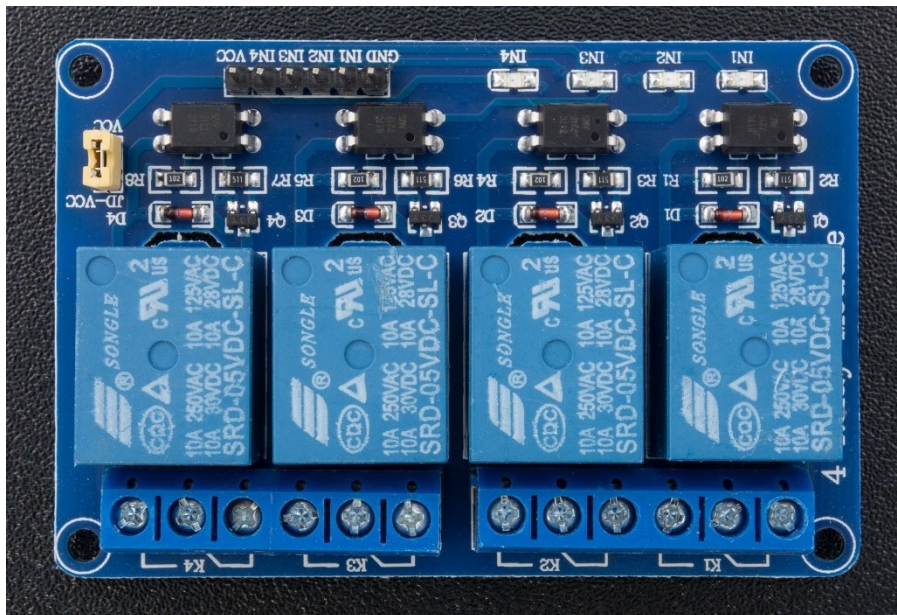
Servo motor vrsta je elektromotora kojem se može vrlo precizno odrediti kut zakretanja. Sastoji se od tri glavna dijela, a to su elektromotor, reduktor koji reducira veliki broj okretaja elektromotora na mali broj ali mu za isti omjer za koji mu je smanjena brzina poveća snagu. U kućištu servo motora se još nalazi i promjenjivi otpornik (potencijometar) koji se zakreće skupa sa izlaznom osovinom. Kut zakretanja se određuje mjereći otpor tog promjenjivog otpornika (potencijometra). Servo motor ima tri izlazne žice jedna je +5V, druga GND, a treća ulazni PWM signal za kontrolu brzine motora.



**Slika 11. Servo motor ugrađen na linearni pokretač** *Izvor: autor*

## 2.10. Relejni modul

Relejni modul je tiskana pločica na kojoj se nalaze releji koji su galvanski odvojeni od upravljačkog sklopa koji okida releje kada mikro kontroler pošalje signal. Za realizaciju ovog platenika koristi se 4-kanalni relej modul. Releji su elektromehanička komponenta koja radi kao prekidač samo što taj prekidač spaja kontakte tako da elektromagnet unutar njega povuče kotvu koja sa sobom povlači kontakte releja u zatvoren položaj. Upravljački signal je odvojen od kontakata releja i napajanje zavojnice elektromagneta je također izvedena sa posebnim napajanjem tako da mikro kontroler samo šalje signal na opto-izolator koji okida tranzistor zavojnice releja.



Slika 12. 4-kanalni modul s relejima

Izvor : [https://e-](https://e-radionica.com/media/catalog/product/cache/5/image/9df78eab33525d08d6e5fb8d27136e95/d/s/dsc_5122.jpg)

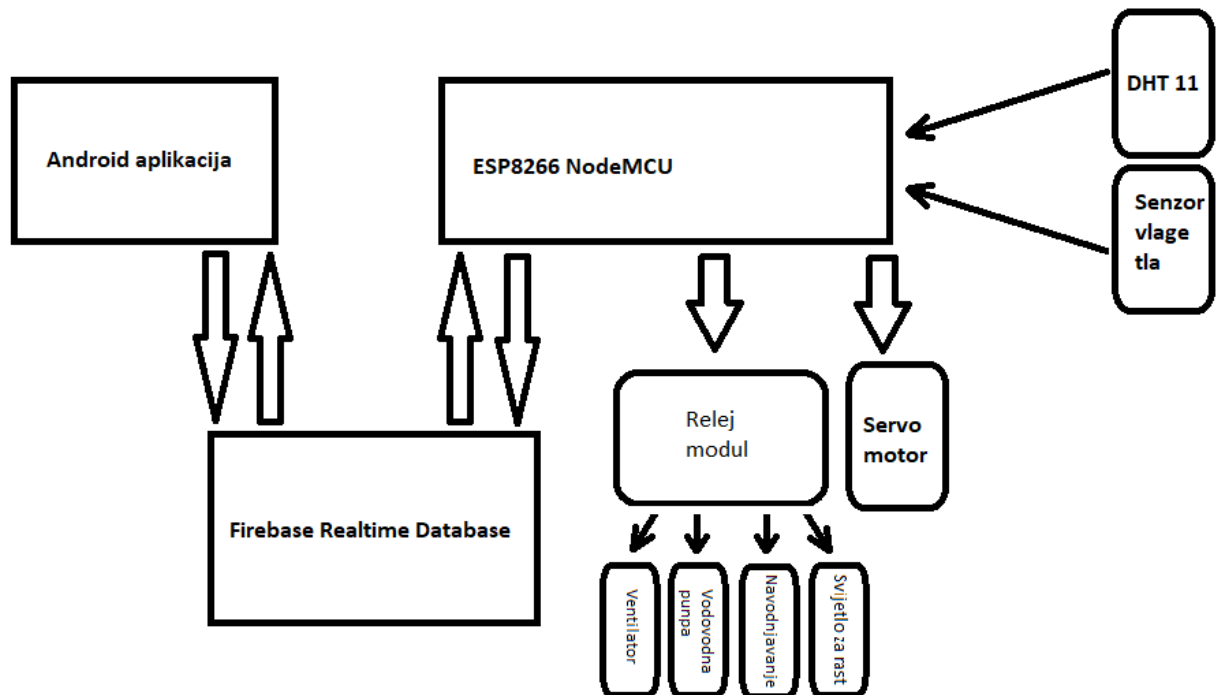
[radionica.com/media/catalog/product/cache/5/image/9df78eab33525d08d6e5fb8d27136e95/d/s/dsc\\_5122.jpg](https://e-radionica.com/media/catalog/product/cache/5/image/9df78eab33525d08d6e5fb8d27136e95/d/s/dsc_5122.jpg)  
(21.08.2022.)

### 3. Dijagram i shema

U ovom poglavlju objasnit će se kako sustav radi. Biti će prikazan funkcijski dijagram sustava i električna shema cijele instalacije sustava autonomnog plastenika.

#### 3.1. Funkcijski dijagram sustava

Sustav pametnog plastenika zasniva se na jednostavnom principu. NodeMCU očitava vrijednosti senzora i na temelju očitavanja izvršava radnje za održavanje mikroklimе unutar plastenika, također te vrijednosti šalje u Firebase Realtime Database. Da bi NodeMCU bio u mogućnosti slati podatke na Firebase server mora uključiti "FirebaseESP8266.h" biblioteku u kojoj su definirane funkcije za primanje i slanje podataka. Funkcija GET je zahtjev kojim NodeMCU traži vrijednosti iz baze podataka, a funkcija SET šalje podatke u bazu. Osim navedene biblioteke potrebno je da ESP8266 NodeMCU pristupi internetu putem Wi-Fi mreže. Sličan postupak je i za komunikaciju aplikacije pametnog telefona i Firebase baze podataka. Prvo je potrebno spojiti se na Internet mrežu zatim podatci za nadzor se u aplikaciju uvoze funkcijom GET, a postavljeni parametri temperature zraka, vlage tla i paljenja i gašenja svjetla se vrše funkcijom SET.

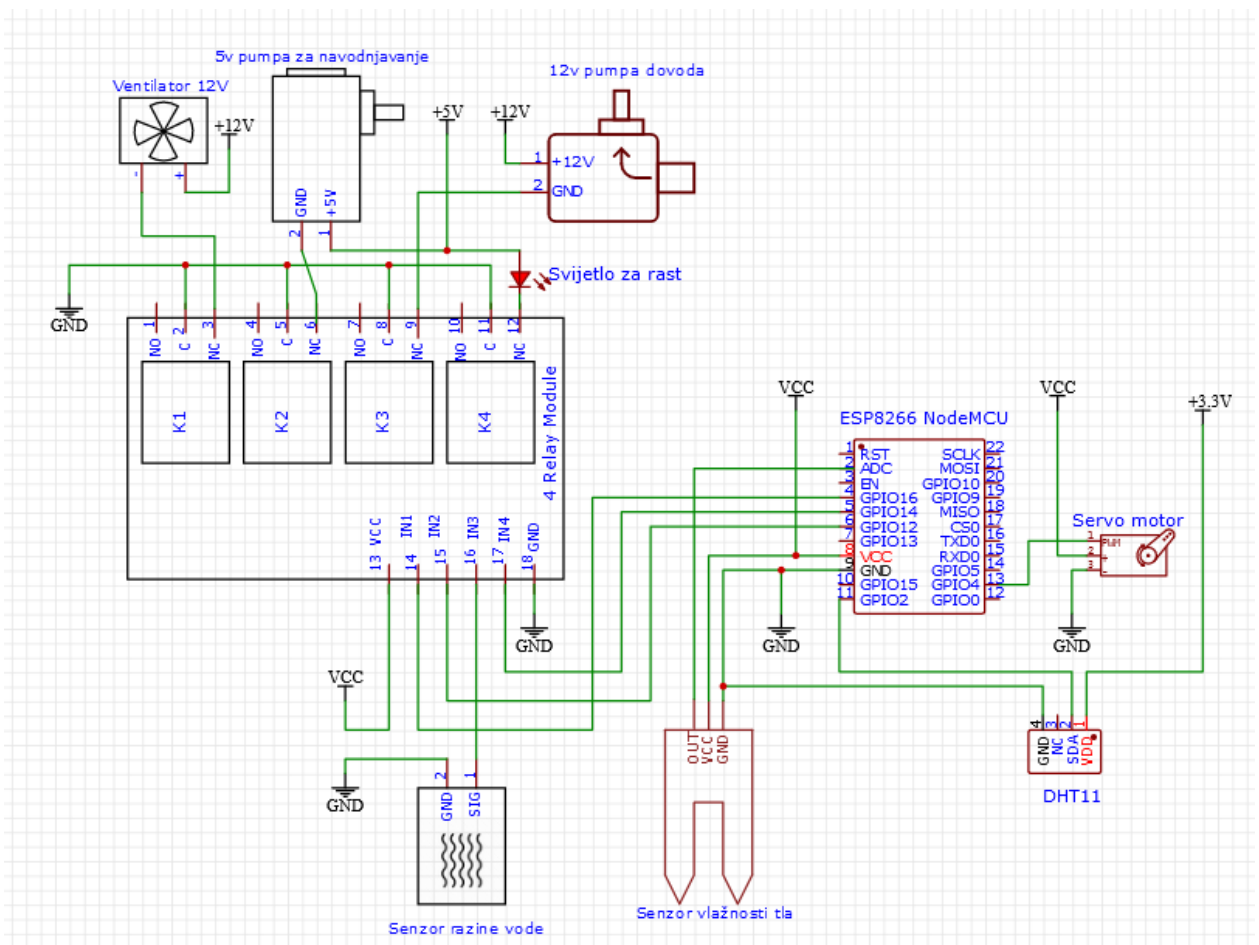


Slika 13. Funkcijski dijagram sustava *Izvor: autor*

### 3.2. Električna shema sustava

Električna shema izrađena je pomoću EasyEDA web aplikacije. Za rad je potrebno samo stvoriti korisnički račun i može se koristiti, baza raspoloživih komponenti je velika i pokriva mnoga područja.

Za izradu električne sheme bilo je prvo potrebno proučiti načine spajanja pojedinih senzora i modula, te se mora obratiti pozornost na napon kojim se napajaju senzori jer neki senzori rade na 3.3V, a neki na 5V.



Slika 14. Električna shema sustava *Izvor: autor*

## 4. Realizacija funkcionalne makete pametnog plastenika

Maketa pametnog plastenika izrađuje se od akrilnog stakla i podijeljena je u više zasebnih sustava kao što su upravljački sustav kojega čini ESP8266 NodeMCU, sustava za nadzor temperature i vlage zraka, sustava za nadzor vlažnosti tla, sustava za navodnjavanje i sustava za provjetravanje.

### 4.1. Izrada i provjera rada elektronike

Prvi korak kod izrade pametnog plastenika je uz pomoć nacrtane električne sheme sustava napraviti na eksperimentalnoj pločici prototip cijele elektronike sustava. Za izradu prototipa koristi se eksperimentalna pločica jer je naknadno lako promijeniti neke stvari na sklopu.

### 4.2. Izrada programskog koda

Nakon uspješno provedenog spajanja elektronike prelazi se na razvoj programskog koda koji će se učitati na ESP8266 NodeMCU.

Prvi korak prilikom rada s ESP pločicama potrebno je instalirati pogonski program kako bi računalo prepoznalo mikro kontroler. Nakon uspješno provedenog koraka instalacije pogonskih programa potrebno je otvoriti izbornik "Datoteka" i izabrati "preporučene vrijednosti". Na praznu crtu u retku "Additional Board Manager URLs: ubaciti slijedeću URL adresu " [http://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable/package_esp8266com_index.json)".

Sada je omogućen izbor odgovarajuće ESP pločice iz izbornika "Alati".

Kada se uspješno izabere pločica prvo što se uočava jest prazni prozor za uređivanje koda osim dvije "void loop" i "void setup" petlje. Unutar void setup petlje se piše dio koda koji će se izvršiti samo prilikom pokretanja koda ili nakon ponovnog pokretanja sustava, a unutar void setup petlje se piše dio programa koji se uvijek vrti u krug izvršava se od početka do kraja pa ponovno i tako dokle god mikro kontroler radi. Iznad void setup se uključuju biblioteke koje nam olakšavaju programiranje i upotrebu senzora i modula. Nakon uspješno napisanoga koda potrebno je još pod WIFI\_SSID unijeti ime Wi-Fi mreže na koje se planira spojiti NodeMCU mikro kontroler i pod WiFi\_PASSWORD lozinku lokalne Wi-Fi mreže na koje se vrši spajanje. Da bi NodeMCU znao u koju bazu će slati i povlačiti podatke unosi se firebase host adresa i firebase auth key.

## Algoritam 1: Programski kod pametnog plastenika

```
1.  /*Pametni platenik
2.  * Izradio : Mihael Vuković 3.ET
3.  * Veleučilište u Virovitici
4.  * Godina 2022.
5.  */
6.
7.
8.  #include <DHT_U.h>
9.  #include <DHTesp.h>
10. #include <Servo.h> // uključujemo library za servo motor
11.
12. //FirebaseESP8266.h mora biti uključen prije ESP8266WiFi.h
13.
14. #include <FirebaseESP8266.h> // uključujemo Firebase ESP8266 library
15. #include <esp8266wifi.h>
16. #include <DHT.h> // Uključujemo DHT11 Library i Adafruit Unified Sensor
    Library
17.
18.
19. #define FIREBASE_HOST "mvfirebaseiot-default-rtdb.firebaseio.com" //Firebase URL
20. #define FIREBASE_AUTH "3XwmwZm31VSsoid572GUfP8wpbjnLX0xx8p60v9q" //Firebase token
21. #define WIFI_SSID "Platenik" //naziv Wi-Fi mreže na koju se ESP8266 NodeMCU spaja
22. #define WIFI_PASSWORD "PlAsTeNiK" //lozinka Wi-Fi mreže
23.
24. #define DHTPIN 2 // Definiranje DHT11 pina
25. int led = 14; // definiranje pina svijetla za rast
26. int ventilator = 12; //definiranje pina za ventilator
27.
28. #define DHTTYPE DHT11
29. DHT dht(DHTPIN, DHTTYPE);
30.
31. //Definiranje FirebaseESP8266 podatkovnog objekta
32. FirebaseData firebaseData;
33. FirebaseData ledData;
34. FirebaseData maxtemperatureData;
35. FirebaseData postVlagaTlaData;
36. FirebaseJson json;
37.
38. // ----- servo dio -----
    --
39.
40. int SERVO_MOTOR = 4; //definiranje pina servo motora
41. int SERVO_CLOSED_VAL = 160;
42. int SERVO_OPEN_VAL = 10;
43. Servo servoMotor; // kreiramo servo objekt za upravljanje motorom
44. boolean ventsOpen; // za praćenje jesu li ventilacijski otvori otvoreni ili ne.
45.
46. // -----
    --
47.
48. // ----- senzor vlage dio -----
    --
49. int pumpa = 16; // definiranje pina pumpe za navodnjavanje
50. int sVlaga = A0; // definiranje pina za senzor vlažnosti tla
51. int ocitanje_vlage;
52. // -----
    --
53.
54.
55. void setup()
56. {
57.
58.   Serial.begin(9600);
59.
60.   dht.begin();
61.   pinMode(led, OUTPUT);
```

```

62. pinMode(ventilator, OUTPUT);
63.
64. WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
65. Serial.print("Connecting to Wi-Fi");
66. while (WiFi.status() != WL_CONNECTED)
67. {
68.     Serial.print(".");
69.     delay(300);
70. }
71. Serial.println();
72. Serial.print("Connected with IP: ");
73. Serial.println(WiFi.localIP());
74. Serial.println();
75.
76. Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
77. Firebase.reconnectWiFi(true);
78.
79. // ----- servo dio (setup) -----
80.
81. pinMode(SERVO_MOTOR, OUTPUT);
82. servoMotor.attach(SERVO_MOTOR);
83. servoMotor.write(SERVO_CLOSED_VAL);
84. ventsOpen = false;
85.
86. // -----
87.
88. // ----- senzor vlage dio (setup) -----
89.
90. pinMode(pumpa, OUTPUT);
91. pinMode(sVlaga, INPUT);
92.
93. // -----
94.
95. // ----- preventivno gašenje releja -----
96. digitalWrite(ventilator, HIGH);
97. digitalWrite(led, HIGH);
98. digitalWrite(pumpa, HIGH);
99. // -----
100.
101. }
102.
103. void sensorUpdate() {
104.     float h = dht.readHumidity(); //varijabli h je dodijeljena vrijednost vlage
    zraka
105.     float t = dht.readTemperature(); //varijabli t je dodijeljena vrijednost
    temperature zraka (u °C)
106.     float f = dht.readTemperature(true);
107.
108.
109.     float s = ocitanje_vlage;
110.     // Provjera jesu li svim varijablama dodijeljene vrijednosti.
111.     if (isnan(h) || isnan(t) || isnan(f)) {
112.         Serial.println(F("Neuspjelo čitanje DHT11 senzora!"));
113.         return;
114.     }
115.
116.     // provjerava dali je uspješno očitano senzor vlažnosti tla
117.     if (isnan(s)) {
118.         Serial.println(F("Očitavanje vlage tla nije uspjelo!"));
119.         return;
120.     }
121.
122.     Serial.print(F("Humidity: "));
123.     Serial.print(h);
124.     Serial.print(F("% Temperature: "));
125.     Serial.print(t);
126.     Serial.print(F("°C ,"));
127.     Serial.print(f);
128.     Serial.println(F("°F "));
129.

```

```

130.   if (Firebase.setFloat(firebaseData, "/FirebaseIOT/temperature", t))
131.   {
132.       Serial.println("PASSED");
133.       Serial.println("PATH: " + firebaseData.dataPath());
134.       Serial.println("TYPE: " + firebaseData.dataType());
135.       Serial.println("ETag: " + firebaseData.ETag());
136.       Serial.println("-----");
137.       Serial.println();
138.   }
139.   else
140.   {
141.       Serial.println("FAILED");
142.       Serial.println("REASON: " + firebaseData.errorReason());
143.       Serial.println("-----");
144.       Serial.println();
145.   }
146.
147.   if (Firebase.setFloat(firebaseData, "/FirebaseIOT/humidity", h))
148.   {
149.       Serial.println("PASSED");
150.       Serial.println("PATH: " + firebaseData.dataPath());
151.       Serial.println("TYPE: " + firebaseData.dataType());
152.       Serial.println("ETag: " + firebaseData.ETag());
153.       Serial.println("-----");
154.       Serial.println();
155.   }
156.   else
157.   {
158.       Serial.println("FAILED");
159.       Serial.println("REASON: " + firebaseData.errorReason());
160.       Serial.println("-----");
161.       Serial.println();
162.   }
163.
164.   //upload podatka vlage tla na firebase sa arduino
165.   if (Firebase.setFloat(firebaseData, "/FirebaseIOT/vlagaTla", s))
166.   {
167.       Serial.println("PASSED");
168.       Serial.println("PATH: " + firebaseData.dataPath());
169.       Serial.println("TYPE: " + firebaseData.dataType());
170.       Serial.println("ETag: " + firebaseData.ETag());
171.       Serial.println("-----");
172.       Serial.println();
173.   }
174.   else
175.   {
176.       Serial.println("FAILED");
177.       Serial.println("REASON: " + firebaseData.errorReason());
178.       Serial.println("-----");
179.       Serial.println();
180.   }
181. }
182. }
183.
184. void loop() {
185.     sensorUpdate();
186.
187.     if (Firebase.getString(ledData, "/FirebaseIOT/led")) {
188.         Serial.println(ledData.stringData());
189.         if (ledData.stringData() == "1") {
190.             digitalWrite(led, LOW); //LOW aktivira relej, modul je Low-level triggered
191.
192.         }
193.         else if (ledData.stringData() == "0") {
194.             digitalWrite(led, HIGH); //HIGH gašenje releja, modul je Low-level triggered
195.         }
196.     }
197.
198.     delay(100);
199.

```



```

200.
201. //-----kontrola temperature (otvaranje i zatvaranje otvora) -----
-----
202. float i = dht.readTemperature();
203. if (Firebase.getString(maxtemperatureData, "/FirebaseIOT/maxtemperature")) {
204.
205.     String i = maxtemperatureData.stringData();
206.     int y = i.toInt();
207.     int c = y;
208.     Serial.println(c);
209.     if (dht.readTemperature() >= c)
210.     {
211.         digitalWrite(ventilator, LOW); // LOW aktivira relej, modul je Low-level
triggered
212.
213.         //-----servo otvara otvor
214.         if (ventsOpen == false) {
215.             openServo();
216.             ventsOpen = true;
217.         }
218.         //-----
219.     }
220.     else
221.     {
222.         digitalWrite(ventilator, HIGH); //HIGH gašenje releja, modul je Low-level
triggered
223.
224.         // ----- servo zatvara otvor
225.         if (ventsOpen == true) {
226.             closeServo();
227.             ventsOpen = false;
228.         }
229.         // -----
230.     }
231.
232. }
233. delay(100);
234.
235.
236. //-----dio za vlagu tla -----
237.
238. ocitanje_vlage = analogRead(A0);
239. ocitanje_vlage = map(ocitanje_vlage, 400, 1002, 100, 0);
240. ocitanje_vlage = constrain(ocitanje_vlage, 0, 100); // izvršavanje skaliranja
analognog signala u postotke
241. Serial.print(„Vlaga tla iznosi : „);
242. Serial.print(ocitanje_vlage);
243. Serial.println(„%“);
244.
245. if (Firebase.getString(postVlagaTlaData, „/FirebaseIOT/postVlagaTla“) {
246.     String vt = postVlagaTlaData.stringData();
247.     int m = vt.toInt();
248.     int n = m;
249.     if (ocitanje_vlage < n) {
250.         digitalWrite(pumpa, LOW); // LOW aktivira relej, modul je Low-level
triggered
251.     }
252.     else
253.     {
254.         digitalWrite(pumpa, HIGH); //HIGH gašenje releja, modul je Low-level
triggered
255.     }
256. }
257. delay(500);
258.
259. }
260.
261.
262. /** Otvara krovni otvor kad se zadovolji uvjet ako već nisu otvorena */
263. void openServo() {

```

```
264.   if (servoMotor.read() != SERVO_OPEN_VAL) {
265.       servoMotor.write(SERVO_OPEN_VAL);
266.   }
267. }
268.
269. /** Zatvara krovni otvor kada je uvjet zadovoljen ako već nisu zatvorena */
270. void closeServo() {
271.     if (servoMotor.read() != SERVO_CLOSED_VAL) {
272.         servoMotor.write(SERVO_CLOSED_VAL);
273.     }
274. }
```

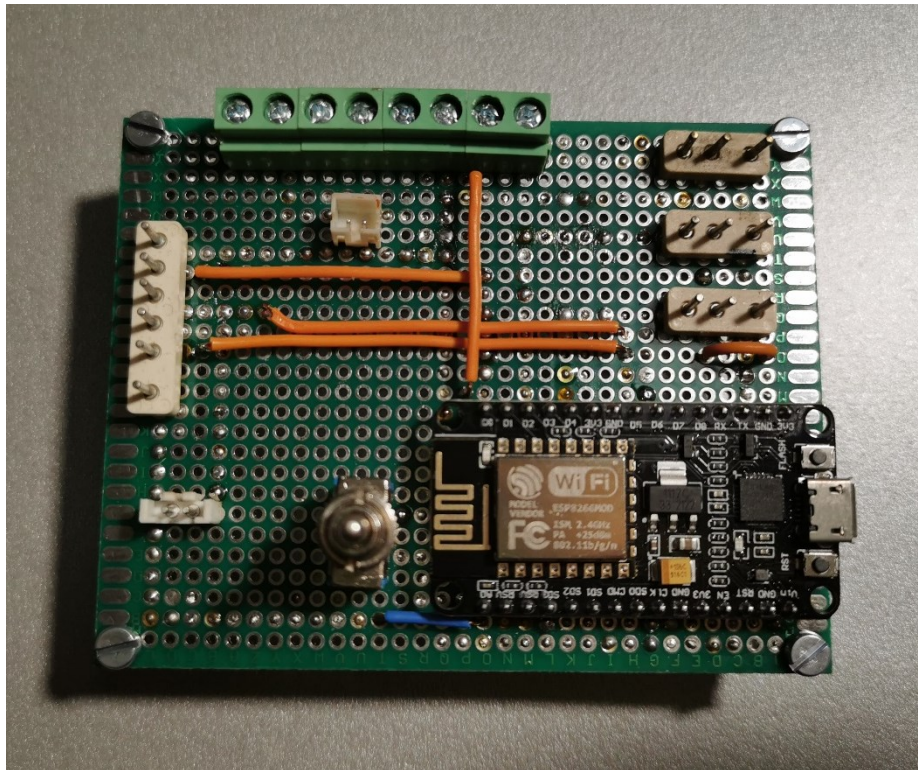
### 4.3. Testiranje rada elektronike i napisanoga koda

Testiranje se vrši kako bi se provjerilo je li električna shema bila točna i da se vidi ponaša li se razvijeni kod onako kako se predviđalo u fazi njegova razvoja.

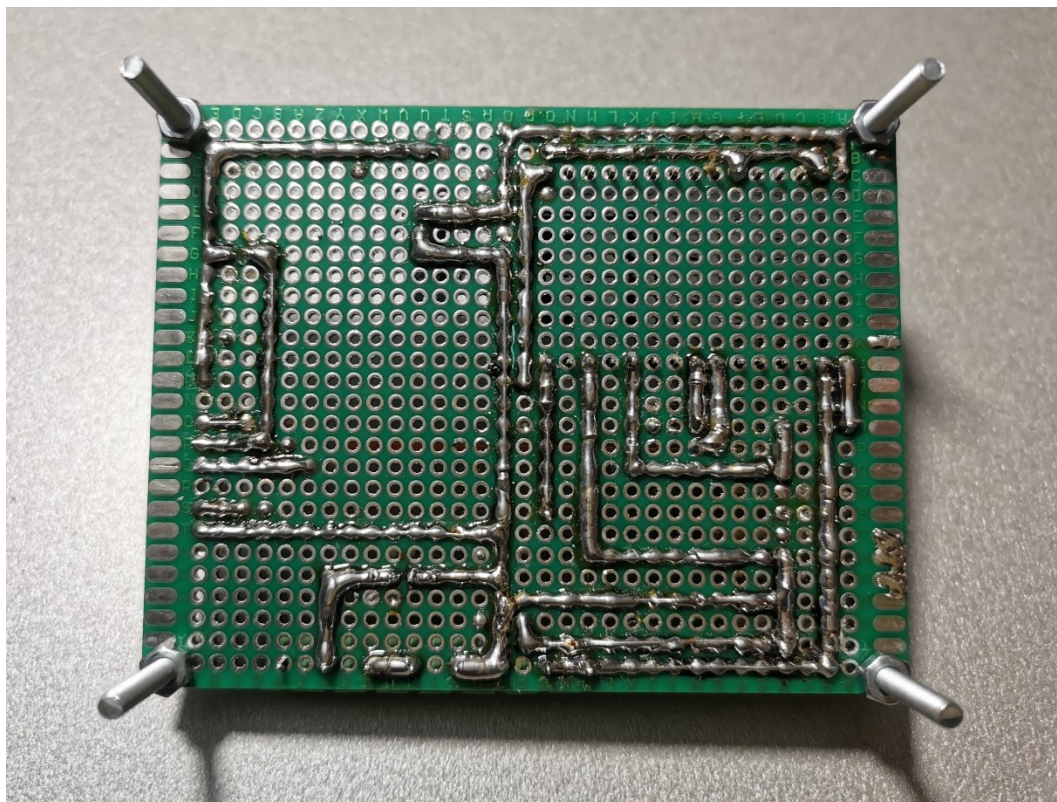
Za testiranje sustava potrebno je dovesti napajanje od 5V i 12V. 5V napajanje je za pojedine senzore i za mikro kontroler koji tih 5V pretvara u 3.3V koje koristi za svoj rad i za rad DHT11 senzora.

### 4.4. Izrada tiskane pločice

Nakon uspješno provedenih testiranja prelazi se na izradu tiskane pločice za elektroniku. Tiskana pločica je komad stakloplastike na koji je postavljen tanki sloj bakra koji se tijekom izrade tiskane pločice oblikuje u vodiče. Za potrebe ovoga rada neće se koristiti metode crtanja vodova i njihovo jetkanje u mješavini solne kiseline i vodikovog peroksida, već se koristi univerzalna tiskana pločica sa bakrenim kružićima presvučeni lemnom legurom. U takvu pločicu se postave elementi i tada se postupkom lemljenja uz pomoć lemne legure izrađuju vodovi po pločici.



Slika 15. Prednja strana izradene tiskane pločice *Izvor: autor*



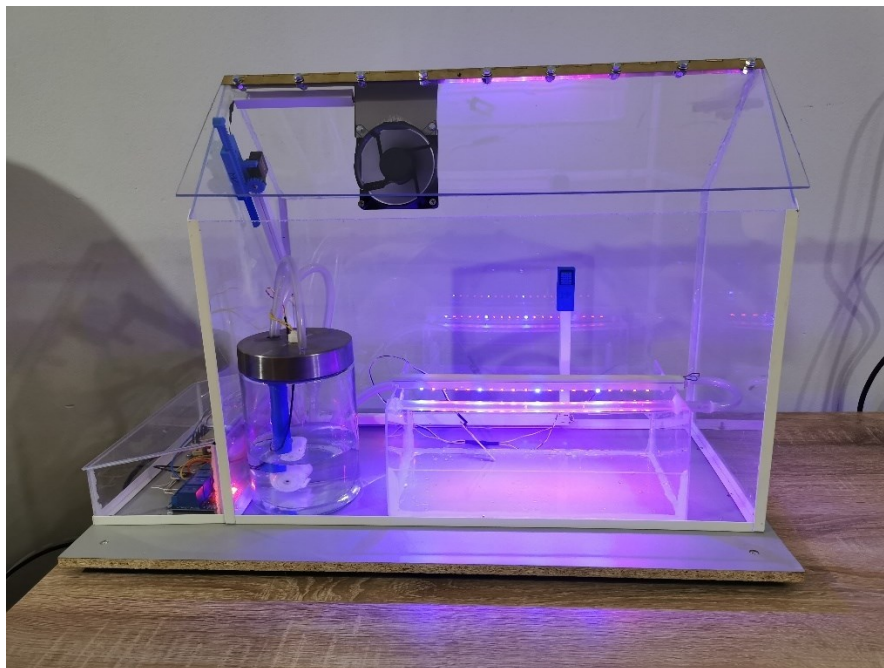
Slika 16. Stražnja strana tiskane pločice *Izvor: autor*

## 4.5. Izrada oklopa od akrilnog stakla

Maketa pametnog plastenika je izrađena od akrilnog stakla koji je spajan dvo-komponentnim epoksidnim ljepilom. Stijenke su debljine 2.5mm što osigurava dovoljno veliku čvrstoću za navedenu maketu.

Na maketi je izrađen veliki ventilacijski otvor koji se nalazi na krovu i on osigurava uz pomoć ugrađenog ventilatora efikasno prozračivanje plastenika i na taj način ukoliko temperatura poraste iznad željene plastenik se brzo uspije rashladiti. Mehanizam za otvaranje otvora na krovu realiziran je pomoću linearnog pokretača koji se sastoji od nosača servo motora koji ima na sebi zupčanik. Taj zupčanik pomiče zupčastu letvu naprijed za otvaranje otvora i nazad za zatvaranje.

Uz oklop od akrilnog stakla je još izrađena vanjska kutija sa poklopcem gdje je smještena tiskana pločica i relej modul. Posuda za sadnju je također izrađena od akrilnog stakla, ima integrirane nosače crijeva za navodnjavanje i svijetlo pogodno za rast biljaka koje je napravljeno uz pomoć kombinacije crvenih i plavih LED dioda. Različitom omjerom crvenih i plavih LED dioda ostvaruju se idealni uvjeti za pojedine poljoprivredne kulture. Za potrebu ove makete izrađeno je svijetlo u omjeru 4:1 (četiri crvene i jedna plava dioda) sve ukupno u svijetlu za rast ima 16 crvenih i 4 plave diode. Omjer 3:1, 4:1 i 5:1 je idealan za uzgoj salate i špinata. Omjer 7:1, 8:1, 9:1 je optimalan za uzgoj paradajza, patlidžana, krastavaca i paprike.



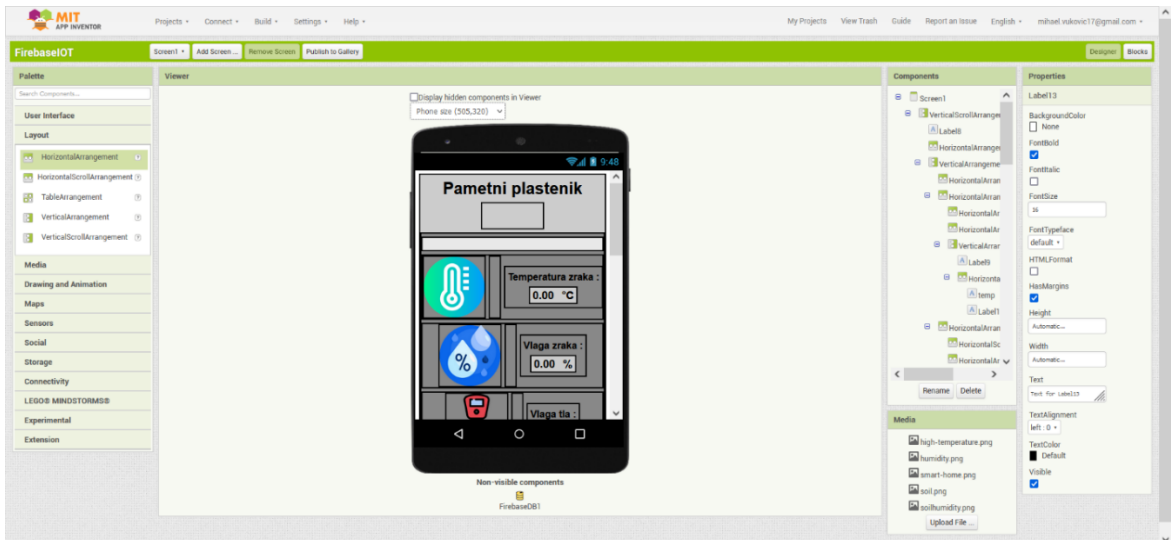
Slika 17. Prikaz završenog modela pametnog plastenika *Izvor: autor*

## **4.6. Izrada sustava za navodnjavanje**

Sustav navodnjavanja plastenika izrađen je od dva rezervoara. Glavni rezervoar zapremine 1.1L je opremljen motornom pumpom koja tjera vodu iz rezervoara kroz cijevi sve do biljaka. Na vrhu poklopca spremnika nalazi se senzor razine vode kojemu je zadaća upaliti pumpu u drugom pomoćnom rezervoaru koji predstavlja dubinski bunar kako bi se iz njega voda pumpala u glavni rezervoar. Na taj način se dodatno automatizirao proces proizvodnje jer nije potrebno dolaziti i puniti spremnik s vodom kada se isprazni već se spremnik sam puni.

## **4.7. Izrada i pregled funkcija mobilne aplikacije**

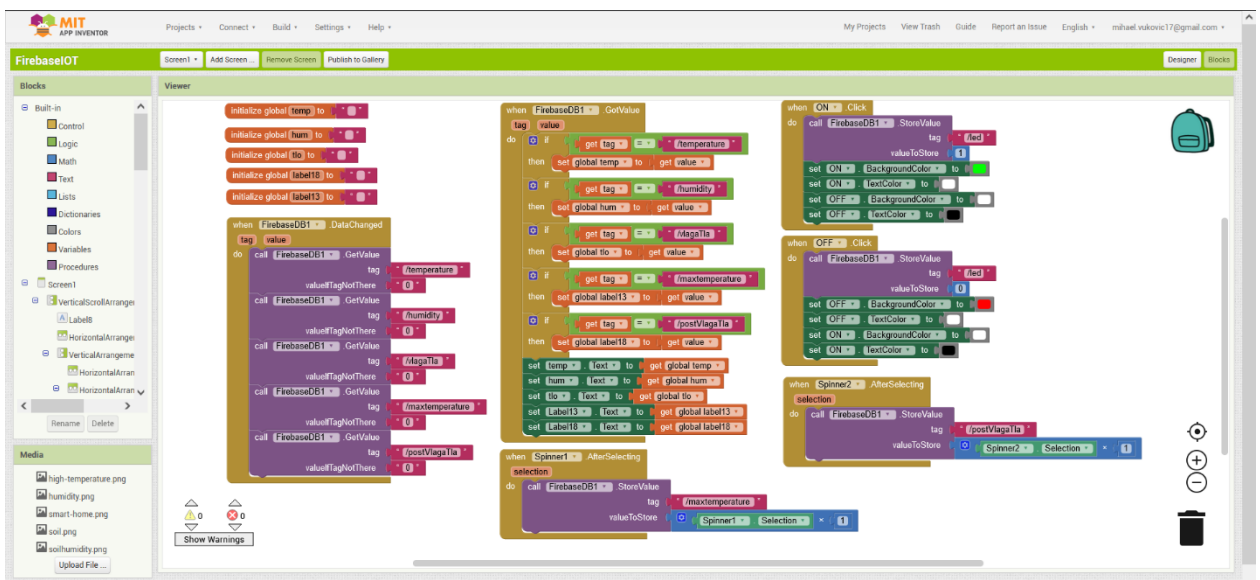
Kako bi plastenik imao mogućnost nadzora temperature zraka, vlažnosti tla i vlažnosti zraka za potrebu rada izrađena je aplikacija za android pametni telefon. Aplikacija također ima i mogućnost paljenja i gašenja svijetla za rast tako da vlasnik plastenika ne mora svaki dan dolaziti ručno paliti svijetlo. Postavljanje željene temperature zraka i vlažnosti tla se vrši putem aplikacije s bilo koje lokacije u svijetu na kojoj vlasnik plastenika ima pristup internetu. Za izradu aplikacije korišten je besplatni MIT App Inventor. To je besplatna web aplikacija koju je razvio Google, a kasnije njezino održavanje je preuzeo Massachusetts Institute of Technology (MIT). Ovaj alat za izradu aplikacija za mobilni telefon je vrlo jednostavno koristiti jer nije potrebno ništa programirati već se sve svodi na "Drag&Drop" metodu. Vizualni izgled aplikacije se slaže po želji tako da se stavljaju elementi unutar virtualnog pametnog telefona i tako osoba koja razvija tu aplikaciju odmah vidi kako će izgledati.



Slika 18. Prikaz kartice "designer" u kojoj se izrađuje vizualni izgled aplikacije

Izvor: autor

Za dodjeljivanje funkcija tim postavljenim elementima u kartici "blocks" pomoću blokova koji predstavljaju određene funkcije, radnje ili logičke operatore slažu se željene funkcionalnosti.



Slika 19. Prikaz kartice "blocks" gdje se dodjeljuju funkcionalnosti aplikaciji

Izvor: autor

Nakon završetka razvoja aplikacije prije preuzimanja moguće ju je testirati i provjeriti radi li sve baš onako kako je predviđeno. Ako je aplikacija na testiranju postigla zadovoljavajuće rezultate aplikacija se može preuzeti i instalirati na pametni telefon.

Slika 20. Konačni izgled aplikacije



Izvor: autor

## 4.8. Testiranje makete

Zadnji korak uoči trajnog puštanja u rad makete pametnog plastenika vrši se provjera svih modula i senzora. Sistem za navodnjavanje testira se na način da se provjeri uključuje li se pumpa za navodnjavanje kada se izvuče senzor vlažnosti tla iz zemlje. Ako je senzor ispravan otpor će mu porasti i pumpa se mora upaliti. Za provjeru sistema za održavanje temperature i ventilacije potrebno je grijaće tijelo koje će za ovo testiranje predstavljati žarulja sa žarnom niti koja će se postaviti u plastenik kako bi zagrijala unutarnji zrak. Ukoliko je DHT11 senzor ispravan otvara se krovni otvor i uključuje se ventilator koji tada upuhuje vanjski zrak. Svijetlo za rast se jednostavno provjerava vizualnim putem tako da se unutar aplikacije pritisne gumb za paljenje svijetla i ako se ono tada upali, a nakon gašenja na gumb za gašenje ugasi svijetlo je ispravno.



## 5. Zaključak

Živi se u vremenu kada je svježija hrana sve skuplja, a potražnja u stalnom porastu. U isto vrijeme proizvodnja iste nikada nije bila teža i izazovnija, zbog sve ekstremnijih vremenskih prilika koje pogađaju naš planet. Kada se pogleda sve ovo navedeno ne čudi sve veći broj firmi koje se bave procesima automatizacije poljoprivredne proizvodnje. Zadatak ovoga rada bio je napraviti automatizirani proces proizvodnje povrća u pametnom plasteniku uz idealnu mikroklimu za rast i razvoj biljaka. Na samom početku bilo je potrebno odlučiti koji su to najbitniji parametri koji se moraju mjeriti i kontrolirati kako bi postigli idealne uvjete za rast. Donesena je odluka da se mjere temperatura zraka, vlaga tla i vlaga zraka. Za kontrolu temperature zraka važna su mjerenja sa senzora DHT11 na temelju kojih platenik odlučuje treba li otvoriti ventilacijski otvor na krovu ili ga treba držati zatvorenim. Za brz razvoj biljke potrebno je i navodnjavanje, a ono je riješeno također automatiziranim procesom tako da senzor vlažnosti tla konstantno mjeri vlagu tla i kada vlaga padne ispod vrijednosti koja se postavila putem mobilne aplikacije za nadzor i upravljanje plastenika pali se elektromotorna pumpa koja pumpa vodu u sustav navodnjavanja. Da bi proces navodnjavanja bio u potpunosti lišen ljudske prisutnosti napravljeno je automatsko dopunjavanje spremnika za navodnjavanje koje se puni iz drugog spremnika koji predstavlja dubinski bunar. Kako je potreba za hranom i po zimi velika stavljeno je svjetlo za rast koje omogućuje povrću dovoljan broj osvjetljenih sati u danu onda kada sunce nije dostupno i time se rast biljke ne usporava i ne prekida dokle god vlada idealna mikroklima unutar plastenika. Za potrebe praćenja vrijednosti mjerenja i upravljanje svjetlom izrađena je posebna aplikacija s kojom je moguće pratiti očitavanja senzora te je moguće upravljanje svjetlom za rast s bilo kojeg kraja svijeta, samo je jedino bitno imati pristup internetu. Osim već navedenih sustava u plasteniku za potpunu autonomiju nedostaje vlastiti izvor električne energije, a to se može postići izgradnjom fotonaponskog sustava sa baterijama koje bi napajale platenik električnom energijom u vrijeme kada nema sunca.

## 6. Popis literature

### Internetski izvori

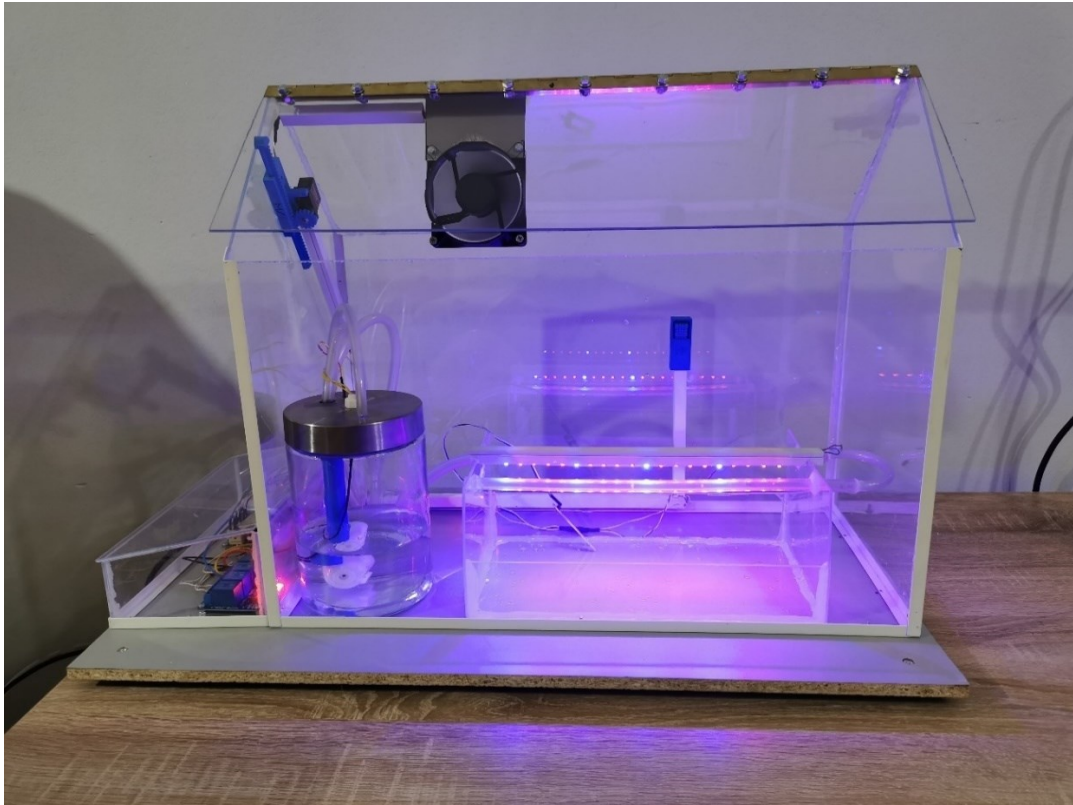
1. Insight Into ESP8266 NodeMCU Features & Using It With Arduino IDE <https://lastminuteengineers.com/esp8266-nodemcu-arduino-tutorial/> (21.08.2022.)
2. Firebase <https://en.wikipedia.org/wiki/Firebase> (18.08.2022.)
3. DHT11 <https://e-radionica.com/hr/dht11-senzor-temperature-i-vlage.html> (21.08.2022.)
4. Senzor vlažnosti tla <https://e-radionica.com/hr/senzor-vlaznosti-tla.html> (21.08.2022.)
5. Senzor razine vode <https://pijaeducation.com/arduino/sensor/float-switch/> (22.08.2022.)
6. Svijetlo za rast <https://www.led-lights.com/different-red-blue-ratio-a-140> (15.08.2022.)

## 7. Popis slika

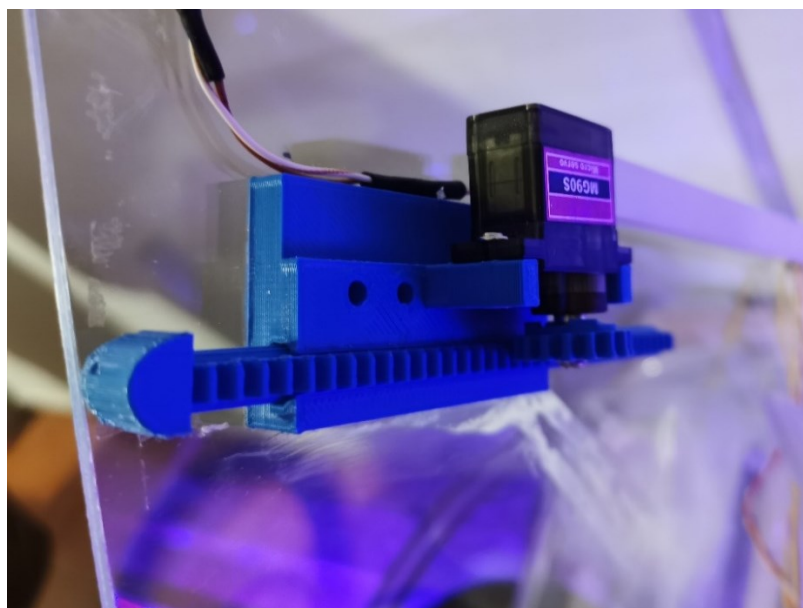
Slika 1. ESP8266 NodeMCU.....	2
Slika 2. Arduino integrirano razvojno okruženje.....	3
Slika 3. Firebase Realtime Database korisničko sučelje.....	4
Slika 4. 3D model DHT11 kućišta.....	5
Slika 5. DHT11 kućište izrađeno tehnikom 3D ispisa.....	5
Slika 6. Linearni otvarač krovnog otvora izrađen tehnikom 3D ispisa.....	6
Slika 7. App Inventor blokovi.....	7
Slika 8. DHT 11 senzor.....	8
Slika 9. Senzor vlažnosti tla.....	9
Slika 10. Senzor razine vode.....	9
Slika 11. Servo motor ugrađen na linearni pokretač.....	10
Slika 12. 4-kanalni modul s relejima.....	11
Slika 13. Funkcijski dijagram sustava.....	12
Slika 14. Električna shema sustava.....	13
Slika 15. Prednja strane izrađene tiskane pločice.....	20
Slika 16. Stražnja strana tiskane pločice.....	20
Slika 17. Prikaz završenog modela pametnog plastenika.....	21
Slika 18. Prikaz kartice "designer" u kojoj se izrađuje vizualni izgled aplikacije.....	23
Slika 19. Prikaz kartice "blocks" gdje se dodjeljuju funkcionalnosti aplikaciji.....	23

## 8. Prilozi

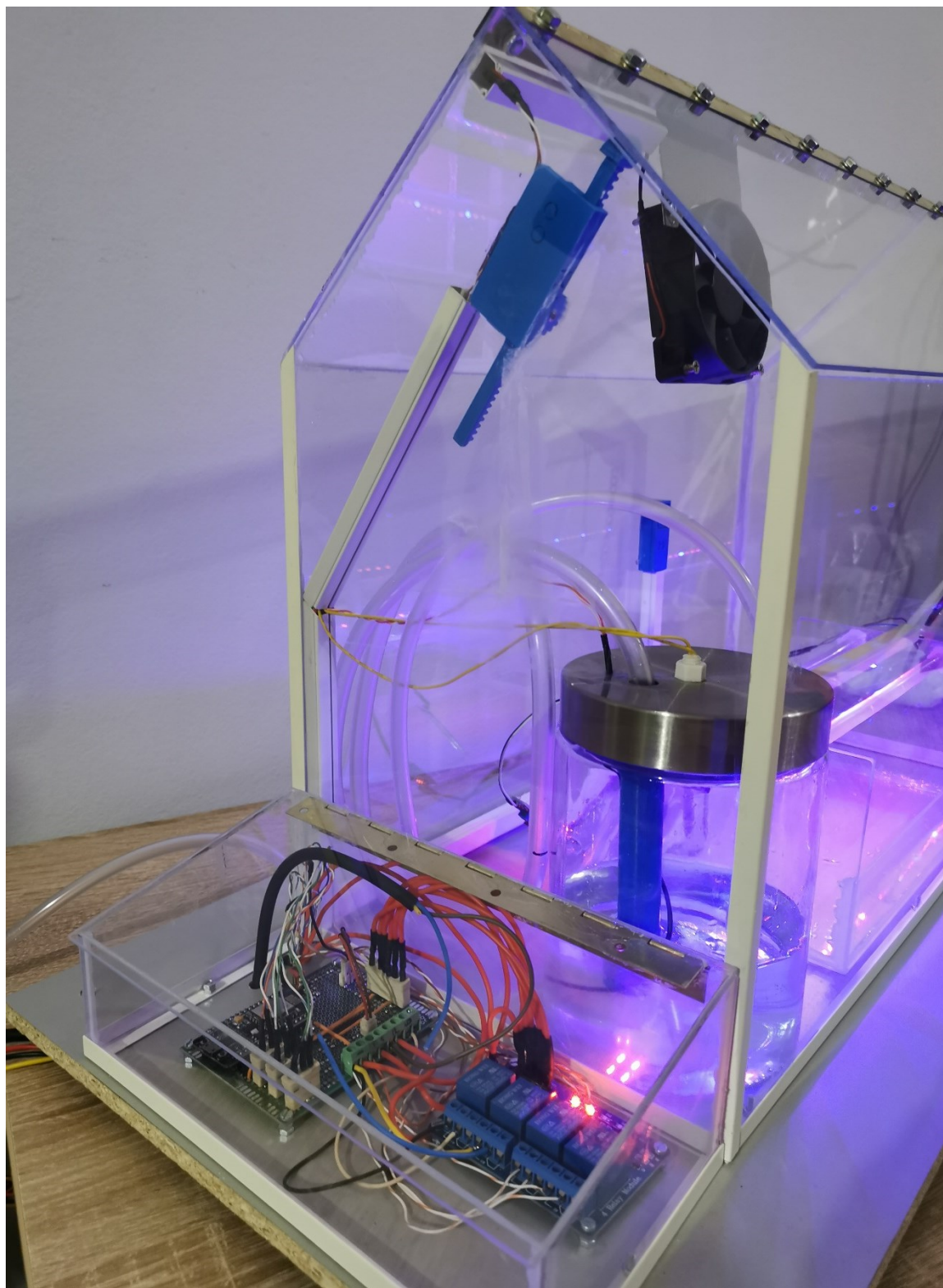
### Prilog 1: Slika modela pametnog plastenika



### Prilog 2: Linearni pokretač za otvaranje ventilacijskog otvora



### Prilog 3: Pogled na bočnu stranu modela plastenika





Veleučilište u Virovatici

**OBRAZAC 5**

### IZJAVA O AUTORSTVU

Ja, MIHAEL VUKOVIĆ

izjavljujem da sam autor/ica završnog/diplomskog rada pod nazivom

PANETIJI PLASTELIJA

Svojim vlastoručnim potpisom jamčim sljedeće:

- da je predani završni/diplomski rad isključivo rezultat mog vlastitog rada koji se temelji na mojim istraživanjima i oslanja se na objavljenu literaturu, a što pokazuju korištene bilješke i bibliografija,
- da su radovi i mišljenja drugih autora/ica, koje sam u svom radu koristio/la, jasno navedeni i označeni u tekstu te u popisu literature,
- da sam u radu poštivao/la pravila znanstvenog i akademskog rada.

**Potpis studenta/ice**

Michael Vuković



**OBRAZAC 6**

**ODOBRENJE ZA POHRANU I OBJAVU  
ZAVRŠNOG/DIPLOMSKOG RADA**

Ja MIHAEL VUKOVIĆ

dajem odobrenje za objavljivanje mog autorskog završnog/diplomskog rada u javno dostupnom digitalnom repozitoriju Veleučilišta u Virovitici te u javnoj internetskoj bazi završnih radova Nacionalne i sveučilišne knjižnice bez vremenskog ograničenja i novčane nadoknade, a u skladu s odredbama članka 83. stavka 11. Zakona o znanstvenoj djelatnosti i visokom obrazovanju (NN 123/03, 198/03, 105/04, 174/04, 02/07, 46/07, 45/09, 63/11, 94/13, 139/13, 101/14, 60/15, 131/17).

Potvrđujem da je za pohranu dostavljena završna verzija obranjenog i dovršenog završnog/diplomskog rada. Ovom izjavom, kao autor navedenog rada dajem odobrenje i da se moj rad, bez naknade, trajno javno objavi i besplatno učini dostupnim:

- a) široj javnosti
- b) studentima i djelatnicima ustanove
- c) široj javnosti, ali nakon proteka 6 / 12 / 24 mjeseci (zaokružite odgovarajući broj mjeseci).

Potpis studenta/ice

Michael Vuković

U Virovitici, 26.08.2022.

*\*U slučaju potrebe dodatnog ograničavanja pristupa Vašem završnom/diplomskom radu, podnosi se pisani obrazloženi zahtjev.*