Bežični NFC čitač

Zrnić, Matko

Master's thesis / Diplomski rad

2020

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:200:044953

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2025-03-21**

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU

FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA

Sveučilišni diplomski studij računarstva

BEŽIČNI NFC ČITAČ

Diplomski rad

Matko Zrnić

Osijek, 2020.

SADRŽAJ

1. UVOD 1

2. RFID TEHNOLOGIJA 2

2.1. Pasivna oznaka 2

2.2. Aktivna oznaka 3

2.3. Primjeri uporabe RFID tehnologije 3

3. BEŽIČNI RFID ČITAČ 5

3.1. Sklopovlje 6

3.1.1. Wemos D1 mini 7

3.1.2. PN532 9

3.1.3. OLED 128x64 10

3.2. Programska podrška 11

3.3. Web sustav za unos podataka 15

4. TESTIRANJE SUSTAVA 19

5. ZAKLJUČAK 22

SAŽETAK 24

ABSTRACT 25

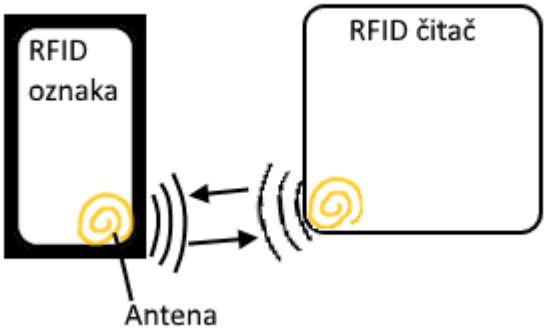
PRILOZI 27

1. UVOD

U ovome radu se konstruira bežični RFID *(eng. radio-frquency identification)* čitač. Potrebno je napraviti sklop koji ima mogućnost očitavanja RFID oznaka *(eng. RFID tag or smart label)* i bežičnu komunikaciju sa bazom podataka. Bežična komunikacija korisiti WiFi tehnologiju za lokalno povezivinja uređaja. Također je potrebno napraviti aplikaciju koja može komunicirati sa sklopovljem i sadrži funkcije koje upravljaju sa lokalnom MSSQL bazom podataka. Cilj rada je simulacija prijave korisnika koji imaju ovlašteni pristup. Za raspoznavanje korisnika se koriste RFID oznake, a za očitavanje oznaka se koristi RFID modul. Kontrola nad sklopovljem je omogućena pomoću mikroupravljača koji ima mogućnost povezivanja na internet. Programska podrška za mikroupravljač je izrađena u programu ArduinoIDE. ArduinoIDE omogućuje izradu programa za razne mikroupravljače, a poseban je zbog jednostavnosti korištenja i viskog stupnja interakcije između korisnika. Preko interneta se komunicira sa aplikacijom dizajniranom u programu Visual Studio. Visual Studio omogućuje korištenje Node.js i Express tehnologija za izradu aplikacije koja ima mogućnost komunikacije preko interneta. Node.js koristi JavaScript okruženje koje omogućuje izradu servera, odnosno internetske stranice preko koje se može vršiti interakcija sa aplikacijom. Interakcija sa aplikacijom je omogućena korištenjem Express tehnologije koja omogućuje izradu sučelja preko kojeg se vrši interakcija.

2. RFID TEHNOLOGIJA

RFID tehnologija se odnosi na tehnologiju gdje se bežično očitavaju podaci sa RFID oznaka (*eng. RFID tag*) ili naljepnica pomoću čitača sa antenom korištenjem elektromagnetskog polja. Na taj način se može automatski identificirati objekt i mogu se preuzeti podaci s tog objekta bez ili sa jako malo ljudskog sudjelovanja u procesu. Sastoji se od RFID oznake ili pametne naljepnice, RFID čitača i antene. RFID komunikacija je prikazana na slici 2.1. gdje se antena nalazi na čitaču. Oznake ili naljepnice se sastoje od integriranog kruga i antene pomoću čega prenose podatke čitaču pomoću radio valova. Čitač pretvara radio valove u korisniji oblik podataka koje računalo može koristiti [1].



Slika 2.1 RFID tehnologija.

Oznake ili naljepnice imaju jedinstveni ID ili UID *(eng. Unique Identifier)* pomoću kojeg se razlikuju. Svaka ima određenu količinu memorije. Razvrstavaju se na pasivne i aktivne.

2.1. Pasivna oznaka

Pasivne oznake sadrže integrirani sklop na kojemu se nalaze podatci i antenu koja služi za prijenos podataka. Čitač koji pokušava očitati oznaku emitira elektromagnetsko polje i pasivna oznaka se aktivira koristeći energiju tog elektromagnetskog polja. Zbog takvog načina napajanja, pasivne oznake su jako praktične i isplative za uporabu iz razloga što ne moraju imati svoje napajanje i sadrže manji broj dijelova. Najčešće se koriste u obliku kartica i naljepnica. Kartice su praktične iz razloga što su dobro zaštićene i zbog toga izdržive. Naljepnice se koriste iz razloga što su najmanje oznake i stoga imaju jako široku primjenu. Za prijenos informacija mogu koristiti tri različite frekvencije:

- niska frekvencije od 125 do 134 KHz
- visoka frekvencija ili NFC *(eng. Near-Field Communication)* 13.56 MHz je najraširenije korištena i koristi se u ovome radu
- ultra visoka frekvencija od 865 do 960 MHz.

Odabir frekvencije ovisi o dometu i brzini koja je potrebna za prijenos podataka iako je visoka frekvencija od 13.56 MHz najčešče korištena.

2.2. Aktivna oznaka

Aktivne oznake se razlikuju od pasivnih u tome što imaju svoj izvor napajanja. Prednost aktivnih oznaka je domet na kojemu mogu komunicirati sa čitačem. Zbog vlastitog izvora napajanja ne koristi se energija elektromagnetskog polja čitača što omogućuje mnogo veći domet komunikacije. Mana aktivnih oznaka je u veličini i ekonomičnosti, odnosno mnogo su skuplje za proizvesti od pasivnih. Iako su puno skuplje i veće od pasivnih, imaju široku primjenu iz razloga što je u nekim slučajevima potreban mnogo veći domet komunikacije od onoga što pasivne mogu ostvariti. Za komunikaciju se koristi frekvencija od 433 MHz ili 915 MHz. Aktivne oznake mogu komunicirati na dva različita načina. Prvi način je sličan pasivinim oznakama iz razloga što čekaju čitač da ih aktivira. Razlog takvog načina je ušteda baterije, a za razliku od pasivnih i dalje imaju mnogo veći domet komunikacije. Drugi način je slanje upita od aktivne oznake za svaki period vremena. Problem kod tog načina komunikacije je potrošnja baterije, ali čitač nije potreban za aktivaciju komunikacije.

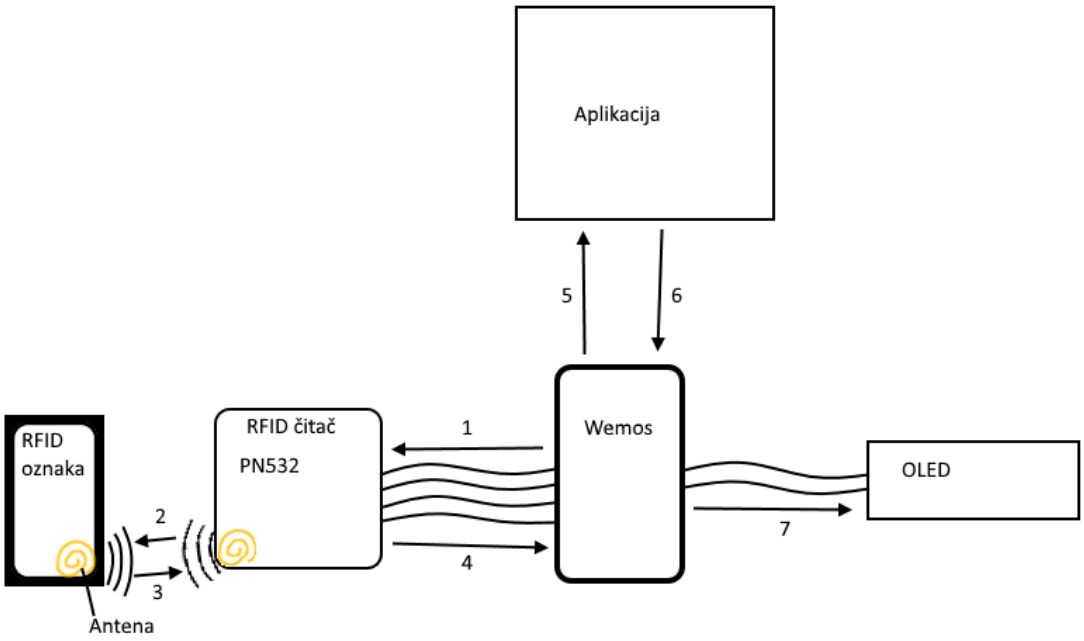
2.3. Primjeri uporabe RFID tehnologije

Iako je RFID tehnologija postojala već mnogo godina, primjena se počela tek nedavno pojavljivati. Razlog tome je što postaje isplativija i u mnogim slučajevima je praktičnija od tehnologija koje se trenutačno koriste. Najbolji primjer je bar kod tehnologija koja se trenutačno koristi u mnogo raznih slučajeve. Iako RFID tehnologija još nije isplativija od bar kod tehnologije, primjena je mnogo veća samo iz razloga što kod očitavanja RFID oznaka nije potreban vizualni kontakt dok to kod očitavanja bar koda nije slučaj. Iz tog razloga, najveća primjena RFID tehnologije je u označavanju stvari gdje sve može imati svoju oznaku koja se vrlo lako može očitati. Osim označavanja stvari, najveće primjene su u prijavama ljudi. Neke situacije toga su gradski prijevoz, zaposlenici itd. Prijava osobe funkcionira na način da svaka osoba ima svoju oznaku koju koristi za prijavu u određeni objekt. U ovome radu se pokušava

simulirati prijava osobe gdje se bilježi vrijeme te prijave i omogućuje se samo u slućajevima kada je pristup odobren [2].

3. BEŽIČNI RFID ČITAČ

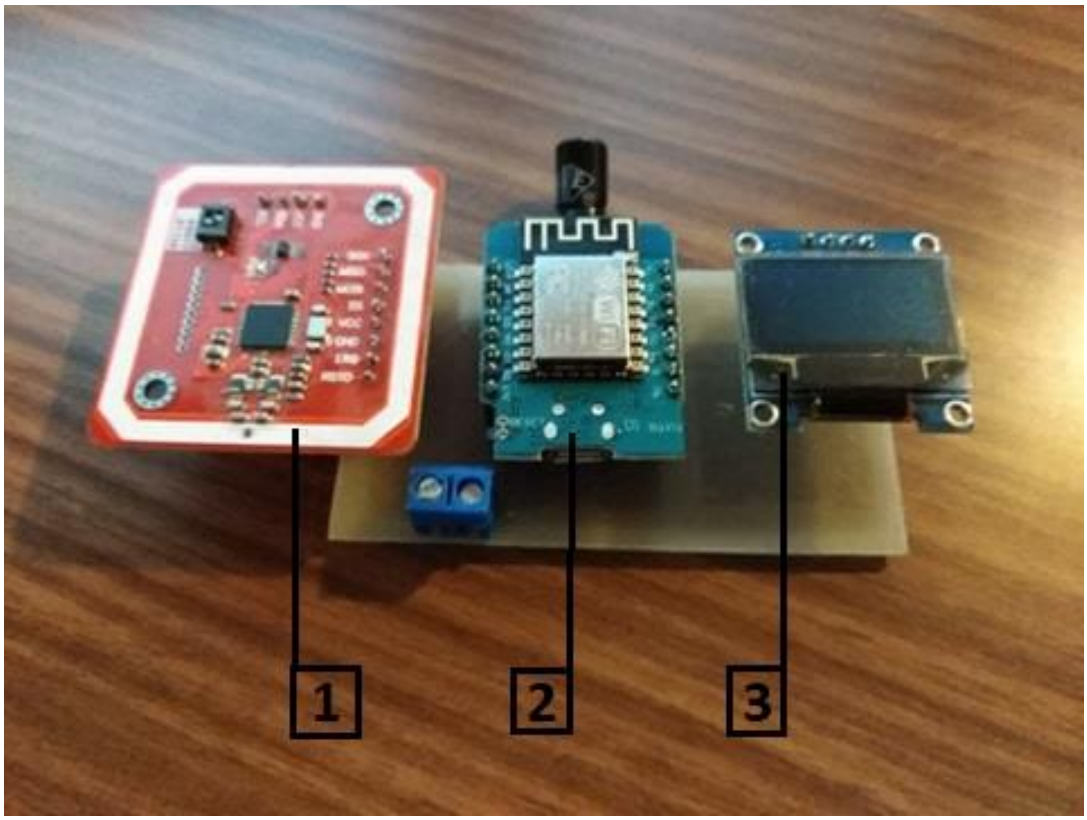
Bežični RFID čitač se razlikuje od običnog na način da nije povezan fizičkim medijem sa uređajem na kojem se nalazi baza podataka korisnika. Na slici 3.1 su prikazani koraci koje sklop prati kako bi obavio glavnu funkcionalnost. Prvi korak je komunikacija gdje Wemos šalje PN532 modulu naredbu da očita RFID oznake u blizini. U drugom koraku PN pokušava očitati oznake koje se nalaze u blizini. Koraci jedan i dva se ponavljaju sve dok se ne očita oznaka u blizini. Frekvencija ponavljanja koraka je dovoljno visoka da se dobije prividna perspektiva kako se oznake neprestano očitavaju. U slučaju da se očita oznaka, dolazi se do koraka tri u kojem oznaka dobije energiju elektromagnetskog polja pomoću koje šalje svoj UID čitaču. Korak četiri je slanje UID-a s čitača na Wemos. Wemos u koraku pet šalje UID na server gdje se provjerava nalazi li se u bazi. Šesti korak je slanje odgovora od servera Wemosu. Zadnji i finalni korak sedam je slanje dobivenog odgovora na OLED ekran gdje se odgovor prikazuje korisniku.



Slika 3.1 Prikaz rada bežičnog RFID čitača

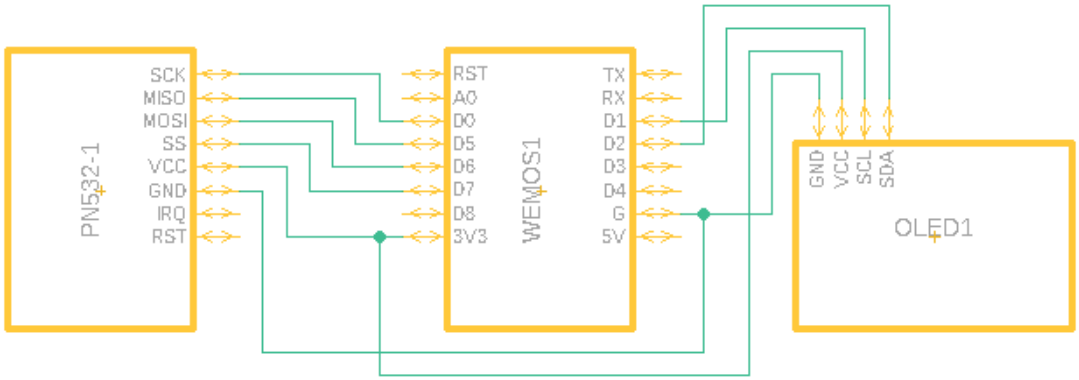
3.1. Sklopovlje

Sklopovlje je fizički dio rada koji uz programsku podršku omogućuje potrebnu funkcionalnost. Sklopovlje je prikazano na slici 3.2. Sastoji se od: Wemos D1 mini modula, PN532 modula, OLED 128x64 modula i naposkog pretvarača s a 12V na 3.3V. Napajanje sklopovlja je moguće preko USB micro kabela preko kojeg se mogu vršiti izmjene nad programskom podrškom uređaja, ali nije praktično kao izvor napajanja. Da se izbjegne problem korištenja USB kabela, koristi se vanjski izvor napajanja od 12V koji se preko naponskog pretvarača smanjuje na potrebnih 3.3V što je napon na kojem ostali moduli sklopovlja funkcioniraju.



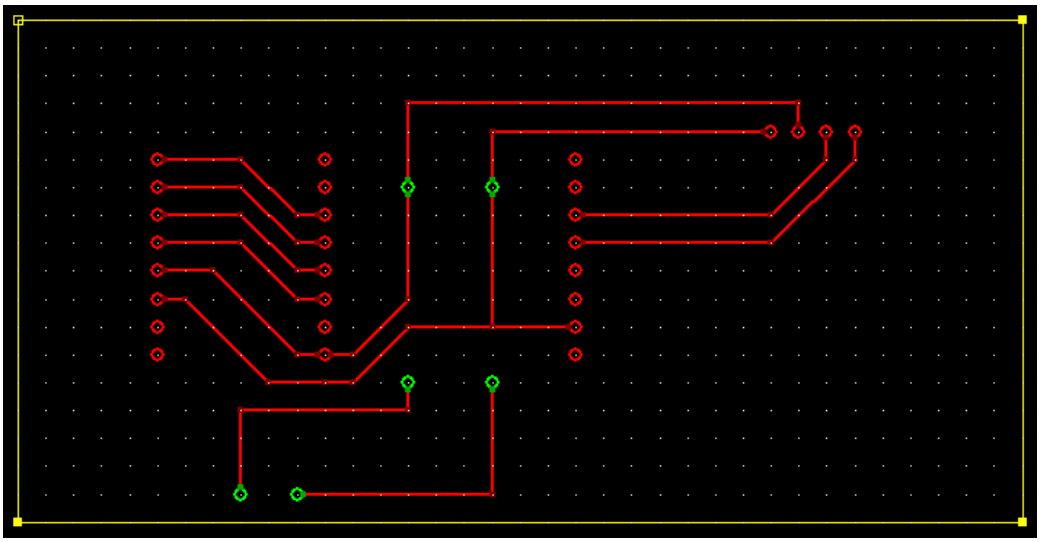
Slika 3.2 Sklopovlje bežičnog RFID čitača.

Za konekciju između elemenata sklopovlja je izrađena shema u programu Eagle prikazana na slici 3.3. Kao što se vidi na slici, elementi imaju povezane pinove za 3.3V i povezane pinove za uzemljenje. Ostali pinovi koji su povezani se koriste za komunikaciju. Između PN532 i Wemosa su povezani četiri pina dok je OLED povezan sa Wemosom pomoću dva pina.



Slika 3.3 Shema.

Prema shemi je izrađen projekt u programu ExpressPCB prikazan na slici 3.4. Taj projekt omogućuje izradu pločice *(eng. PCB)* na koju se postavljaju moduli. Moduli se leme, a pločica sadrži fizičke konekcije koje su potrebne između modula. Na taj način se dobije jednostavn, uredan i organiziran sklop.



Slika 3.4 ExpressPCB projekt.

3.1.1. Wemos D1 mini

Wemos D1 mini pro modul je programabilni SoC (*eng. System on Chip*) sa mogućnosti spajanja na Wi-Fi mrežu. Bazira se na esp8266 sustavu na čipu. Može raditi na 3.3V i 5V. Frekvencija signala takta na kojoj funkcionira može biti ili 80MHz ili 160MHz. Prikazan je na slici 3.2. pod brojem 2.

Wemos sadrži:

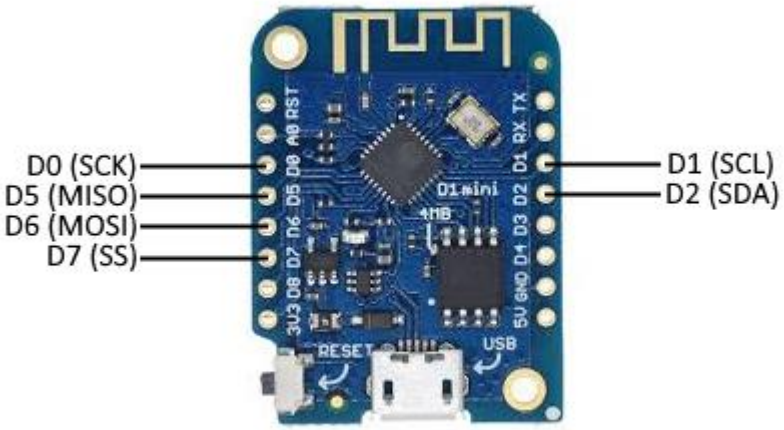
- 4MB flash memorije
- osam digitalnih pinova
- jedan analogni
- unutrašnju PCB antenu
- konektor za vanjsku antenu.

Iako se koristi kao mikroupravljač, prednost je u mogućnosti spajanja na Wi-Fi mrežu frekvencije 2.4 GHz što znači da ima mogućnost bežičnog komuniciranja sa drugim uređajima.

Wemos ima mogućnost komunikacije:

- SPI
- I2C
- UART.

Za komunikaciju sa PN532 modulom se koristi SPI, a za OLED modul se koristi I2C [3]. Na slici 3.5 je prikazan Wemos sa označenim pinovima koje koristi za komunikaciju [4].



Slika 3.5 Wemos d1 mini sa pinovima koje koristi

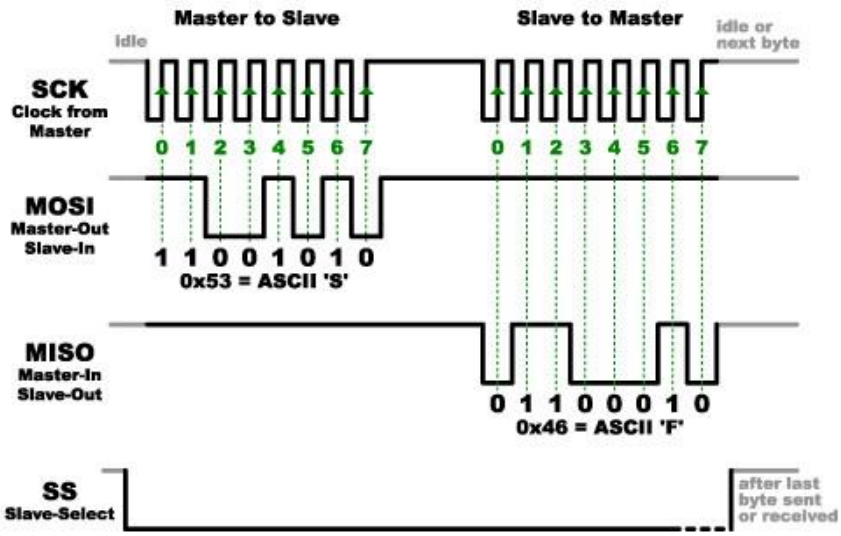
3.1.2. PN532

Svrha PN532 modula je bezkontaktna komunikacija koristeći frekvenciju od 13.56 MHz. Sadrži funkcionalnosti mikrokontrolera baziranog na 80C51 jezgri sa 40 KB ROM i 1 KB RAM memorije. Prikazan je na slici 3.2. pod brojem 1. Može koristiti SPI, HSU i I2C komunikaciu, a sa Wemos modulom komunicira koristeći SPI.

SPI komunikacija koriti četiri podatkovne linije:

- MOSI
- MISO
- SS
- CLK.

MOSI (eng. *master out, slave in*) služi kao podatkovna linija preko koje master, odnosno glavni uređaj što je u ovome slučaju Wemos šalje podatke slave-u što je u ovome slučaju PN532. MISO (*eng. master in, slavo out*) je obrnuto od MOSI-a, odnosno služi kako bi PN532 slao podatke Wemosu. SS *(eng. Slave select)* linija služi za odabir modula s kojime će master, odnosno Wemos komunicirati i potrebna je samo kada ima više modula. SS linija koristi pull-up mod, odnosno postavljena je na logičku jedinicu sve dok se ne želi ostvariti komunikacija i tada se postavlja na logičku nulu što označuje početak komunikacije. CLK *(eng. Clock)* podatkovna linija je oscilator za zadanu frekvenciju i služi za promjenu bitova, odnosno svakom promjenom oscilatora se definira vrijednost sljedećeg poslanog bita. Kao što se vidi na slici 3.6 podatci se šalju na način da se prvo SS linija postavi na nulu. To označava odabir uređaja za kumunikaciju. Nakon toga se na svaki rastući brid SCK linije gledaju stanja MOSI i MISO linije. To stanje označava jedan bit i tako se vrši komunikacija. Programiranje funkcionalnosti PN532 modula se vrši slanjem predodređeno definiranih okvira u kojima se nalazi željena naredba. Nakon slanja naredbi za inicijalizaciju modula u željeni način rada, periodično se šalju naredbe za pokretanje elektromagnetskog polja. U slučaju pisutnosti RFID oznake kada je aktivno elektromagnetsko polje, inicijalizira se oznaka i očitava se njen UID. Očitani UID se šalje Wemosu i PN zatim čeka sljedeću periodičnu naredbu [5].

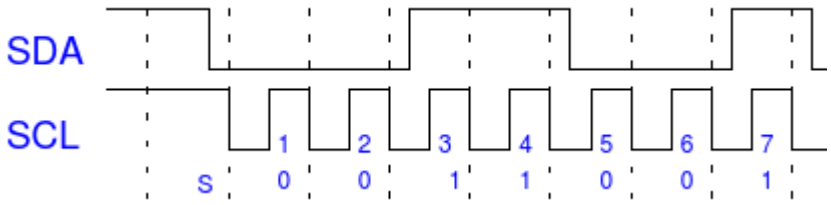


Slika 3.6 SPI komunikacija

3.1.3. OLED 128x64

OLED 128x64 sa 0.96“ dijagonalnom veličinom služi kao vizualni pokazatelj željenih informacija. Sastoji se od 128x64 individualnih piksela od kojih se svaki može uključiti i isključiti. Za kontrolu nad pikselima koristi SSD1306 integrirani sklop koji pomoću komunikacije s drugim uređajima određuje koji će pikseli biti uključeni u određenom trenutku. Prikazan je na slici 3.2. pod brojem 3.

Može komunicirati preko SPI i I2C komunikacije, a koristi se I2C komunikacija prikazana na slici 3.7.



Slika 3.7 I2C komunikacija

I2C komunikacija koristi dvije podatkovne linije preko kojih Wemos šalje podatke OLED-u. Nazivi linija su: SCL *(eng. Serial Clock Line)* i SDA *(eng. Serial Data Line)* gdje je SCL standardni oscilator željene frekvencije. SDA linija je podatkovna linija. SDA koristi pull-up mod u kojem je postavljena na logičku jedinicu kada nema komunikacije. Promjena iz logičke jedinice u nulu sa strane SDA linije kada je SCL linija na logičkoj jedinici označava početak komunikacije. Nakon toga, za vrijeme logičke jedinice SCL linije se gleda stanje SDA linije i

to stanje je bit koji se šalje. Prekid komunikacije se vrši na način da SDA linija promjeni stanje iz logičke nule u jedinicu za vrijeme logičke jedinice na SCL liniji [5].

3.2. Programska podrška

Programska podrška služi za upravljanje Wemos mikrokontrolerom koji je ujedno i „glavni“ modul među sklopovljem. Pod glavni se misli na to što je jedini koji se može programirati, a iz tog razloga upravlja sa ostalim uređajima. Programiranje Wemos-a se vrši pomoću Arduino IDE programa korištenjem C i C++ jezika. Glavne funkcije Arguino programa su setup i loop gdje je setup obična funkcija, a loop beskonačna petlja. Prvobitno kod programske podrške je uključivanje potrebnih biblioteka za mnogo lakši i pojednostavljen rad.

Korištenjem klasa iz biblioteka se deklariraju objekti:

- objekt za PN532 koji sadrži brojeve pinova pomoću kojih Wemos komunicira sa PN modulom
- objekt za komunikaciju sa OLED-om koji sadrži pinove koje Wemos koristi i dimenzije OLED-a koje su u ovom slučaju 128x64, vidi sliku 3.8.

Linija ***Kod***

```
1:      Adafruit_PN532 nfc(CLK, MISO, MOSI, SS);
        Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire,
2:      -1);
```

Slika 3.8 Deklaracija objekata

Za lakšu orijentaciju, kraći kôd i radi jednstavnosti se deklariraju konstantne vrijednosti. Pod to spadaju vrijednosti koje se koriste više puta i tako se postiže urednost i organiziranost kôda.

U funkciji setup se prvo inicijalizira brzina prijenosa podataka od 115200 bauda (115200 bitova po sekundi).

Nakon toga se vrši inicijalizacija:

- internetske konekcije pomoću jednostavne funkcije iz biblioteke koristeći naziv i loziknku WiFi mreže, vidi sliku 3.9.

Linija ***Kod***

```
1:      WiFi.begin(nam, pas);
2:      WiFi.mode(WIFI_STA);
3:      while (WiFi.status() != WL_CONNECTED){
4:        delay(500);
5:        Serial.println("Waiting for connection");}
6:      Serial.println("Connected");
```

Slika 3.9 Inicijalizacija internetske konekcije

- PN532 modula pomoću funkcije iz biblioteke za dohvaćanje verzije modula. Ako dobivena verzija modula nije točna, znači da nešto nije u redu sa uređajem, vidi sliku 3.10.

Linija ***Kod***

```
1:      nfc.begin();
2:      uint32_t versiondata = nfc.getFirmwareVersion();
3:      if (! versiondata)
4:        Serial.println("PN532 not found");
```

Slika 3.10 Inicijalizacija PN532 modula

- OLED ekrana koristeći funkcije iz biblioteke za ispis na ekran. U ovome slućaju se ispisuje: „Starting...“ i to označva uspješnu inicijalizaciju ekrana vidi sliku 3.11.

Linija ***Kod***

```
1:      if(!display.begin(SSD1306_SWITCHCAPVCC, 0x3C))
2:        Serial.println("SSD1306 allocation failed");
3:      delay(2000);
4:      display.clearDisplay();
5:      display.setTextSize(1);
6:      display.setTextColor(WHITE);
7:      display.setCursor(0,0);
8:      display.println("Starting...");
9:      display.display();
```

Slika 3.11 Inicijalizacija OLED ekrana

Loop funkcija prvobitno postavlja prazan ekran te ispisuje: „Tag me“ što simbolizira da je uređaj spreman za učitavanje kartica. Nakon toga se izvodi očitavanje kartica u neposrednoj blizini. Iako se to očitavanje vrši samo trenutačno, zbog jednostavnosti beskonačne petlje u slučaju da se ne učita niti jedna kartica, prividno se dobije dojam da očitavanje kartica radi konstantno, vidi sliku 3.12.

Linija ***Kod***

```
1:      display.clearDisplay();
2:      display.setCursor(10,20);
3:      display.println("Tag me");
4:      display.display();
5:
        success = nfc.readPassiveTargetID(PN532_MIFARE_ISO14443A,
6:      uid, &uidLength);
```

Slika 3.12 Očitavanje oznaka

Ako se uspješno očita kartica, prvobitno se razmatra radi li se o mifare classic kartici ili mifare ultralight naljepnici. Razlika je u tome što classic ima 4 bajta UID dok ultralight ima 7, vidi sliku 3.13.

Linija ***Kod***

```
1:      if (success){
2:        if (uidLength == 4){
3:          Serial.println("\nFound mifare classic card"); x=4; }
4:        else{
5:          Serial.println("\nFound mifare ultralight tag"); x=7; }
6:
7:      get_me_UID(uid, SUID, x);
8:      SendPost(SUID);
```

Slika 3.13 Određivanje vrste oznake i pozivanje funkcija

Dobiveni UID je zapisan u heksadecimalnom obliku, a potreban je u char obliku. Da se dobije željeni oblik, napisana je funkcija koja pomoću ASCII tablice preračunava svaku brojku ili slovo u željeni oblik. Prvobitno uzima svaki broj zasebno od kojega izdvaja prvu i drugu brojku i određuje koja je to brojka ili slovo. Ako je brojka, dodaje se broj 48 iz razloga što je nula na 48. mjestu u Acsii tablici. Na slova se dodaje broj 55 iz razloga što je prvo slovo A na 55. mjestu. Dobivene brojke i slova se zatim stavljaju u sting po potrebnom poretku, vidi sliku 3.14.

Linija Kod

```
1:      void get_me_UID(uint8_t *uid, char *SUID, uint8_t x){
2:      uint8_t i=0, j=0, fst, scd;
3:      for(i=0; i<x; i++) {
4:        fst = uid[i]/16;
5:        scd = uid[i]-(fst*16);
6:        if(fst < 10)
7:          SUID[j] = fst + 48;
8:        else
9:          SUID[j] = fst + 55;
10:       if(scd < 10)
11:         SUID[j+1] = scd + 48;
12:       else
13:         SUID[j+1] = scd + 55;
14:       j=j+2; }}
```

Slika 3.14 Preoblikovanje UID-a u string

Finalni dio je provjera nalazi li se taj UID u bazi podataka. Prije slanja, na UID se dodaju dodatni znakovi iz razloga sto server prima podatke samo u određenom obliku. Zatim se provjerava uspostavljena li je internetska konekcija pomoću koje se komunicira sa serverom. U slučaju da je uspostavljena konekcija, šalju se podaci na adresu servera i čeka se odgovor. Odgovor daje informacije o UID-u ili serveru. UID se nalazi u bazi samo u slučaju da je odgovor 200 i tada se na ekran ispisuje „Authorised“ što označava uspještnost autentifikacije kartice. Dobivanjem odgovora 400 se na ekran ispisuje: „Wrong card“ što označava da je pokušana autentifikacija kartice koja se ne nalazi u bazi podataka. Osim 200 i 400 se mogu dobiti razni drugi odgovori od servera koji za svaki broj ima posebno značenje, ali skoro sigurno to znači da ima nekakvih problema sa serverom pa se na ekran ispisuje: „Error“, vidi sliku 3.15.

Linija ***Kod***

```
1:      void SendPost(char *SUID) {
2:        String postData = "{\"UID\":\"" + String(SUID) + "\"}";
3:        if(WiFi.status()== WL_CONNECTED){
4:          HTTPClient http;
5:          http.begin(HOST_ADDRESS);
6:          http.addHeader("Content-Type", "application/json");
7:          int httpCode = http.POST(postData);
8:          if(httpCode==200)
9:            display.println("Authorised");
10:         else if(httpCode==400){
11:           display.println("  Wrong");
12:           display.println("  card!");}
13:         else
14:           display.println("  Error");
15:         display.display();
16:         http.end(); }}
```

Slika 3.15 Slanje post metode

3.3. Web sustav za unos podataka

Sustav za unos podataka se sastoji od MSSQL lokalne baze podataka i metoda koje obrađuju potrebne zahtjeve. Glavni zadatak je da sadrži bazu podataka u kojoj se nalaze dvije tablice. Bežični čitač pomoću post metode salje UID očitane RFID kartice. Pomoću dobivenog UID-a se pretražuje tablica korisnika uspoređivanjem dobivenog UID-a sa UID-ima koji se nalaze u tablici. U slučaju da se ne nalazi u tablici, server vraća odgovor 400. U slučaju da se nalazi UID u tablici, za tog korisnika se pravi novi upis u drugu tablicu. Druga tablica sadrži prijave, odnosno svaki puta kada se dobije post metoda sa UID-om koji se nalazi u tablici, u drugu tablicu se upisuju podatci tog UID-a uključujući trenutačno vrijeme.

Prvobitno je potrebno napraviti korisničko sučelje što Visual Studio omogućuje pri izradi projekta. Jedino što je potrebno je modificirati na željeni način. Modificiranje se obavlja preimenovanjem automatski generiranih naziva, zatim brisanje nepotrebnih stvari i isto tako dupliciranje stvari koju su potrebne više puta. Kada se to sve obavi, dobije se samo izgled web stranice bez potrebnih funkcionalnosti. Na to se zatim dodaju funkcionalnosti koje omogućuju kretanje između napravljenih stranica.

Za izradu baza podataka je prvobitno potrebno napraviti dvije klase:

- klasa za korisnike osim primarnog ključa sadrži, ime i prezime korisnika te njegov pripadajući UID, vidi sliku 3.16 .

Linija Kod

```
1:     public class User {
2:       public int ID { get; set; }
3:       public string UID { get; set; }
4:       public string LName { get; set; }
5:       public string FName { get; set; }
         public virtual ICollection<Access_date> Access_dates {
6:     get; set; } }
```

Slika 3.16 Klasa za korisnike

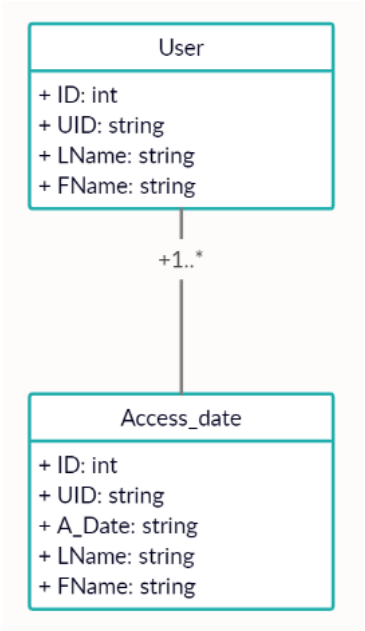
- klasa za datume prijave sadrži primarni ključ koji je povezan sa primarnim ključem klase za korisnike. Objekti klasa koji imaju isti UID imaju povezane primarne ključeva i tako je napravljeno međudjelovanje. Za razliku od korisničke klase, dodano je vrijeme i datum potrebno za označavanje vremena prijave, vidi sliku 3.17.

Linija Kod

```
1:     public class Access_date {
2:       public int ID { get; set; }
3:       public string UID { get; set; }
4:       public DateTime A_Date { get; set; }
5:       public string A_LName { get; set; }
6:       public string A_FName { get; set; }
7:       public virtual User User { get; set; } }
```

Slika 3.17 Klasa za prijave

Napravljene klase su predlošci za tablice. Tablice se popunjavaju tako da svaki element tablice je objekt jedne od klasa. Tablice su prikazane na slici 3.18.



Slika 3.18 Tablice baze i veza između njih

Kako bi se omogućilo jednostavno popunjavanje tablice, na tablicu za korisnike je dodan gumb Create. Gumb omogućuje da se ručno unesu podatci za korisnika, odnosno upiše se ime, prezime i UID korisnika i to se sprema u bazu podataka, vidi sliku 3.19.

Linija Kod

```
      public ActionResult Create([Bind(Include =
1:    "ID,UID,LName,FName")] User user){
2:      if (ModelState.IsValid) {
3:        db.User.Add(user);
4:        db.SaveChanges();
5:        return RedirectToAction("Index"); }
6:      return View(user); }
```

Slika 3.19 Funkcija za kreiranje objkta korisničke klase

Kao i za popunjavanje tablice, ekvivalentrno je napravljeno brisanje. Brisanje se obavlja na način da uzme primarni ključ korisnika kojeg se želi obrisati. Pomoću primarnog ključa se može taj korisnik naći u bazi podataka i obrisati pomoću jednostavne funkcije za uklanjanje objekta, vidi sliku 3.20.

Linija Kod

```
1:      public ActionResult DeleteConfirmed(int id) {
2:        db.User.Remove(user);
3:        db.SaveChanges();
4:        return RedirectToAction("Index"); }
```

Slika 3.20 Funkcije za brisanje korisničkih objekata

Za razliku od popunjavanja tablice za korisnike, popunjavanje tablice za prijave se obavlja kada se dobije post zahtjev. U post zahtjevu se nalazi samo UID korisnika koji se koristi kako bi se pretražila tablica za korisnike. Ako se UID ne pronađe u tablici, znači da je pokušaj nedozvoljenog pristupa i server zatim šalje kod 400. U slučaju da je pronađen UID, razmatra se koliko je prijava trenutno u bazi. Ako baza sadrži deset ili više prijava, briše se ona sa najdaljim vremenom od trenutnog. Nakon toga se popunjavaju podatci objekta datuma prijave. Prvobitno se uzima trenutačni datum i vrijeme koje označuje kada je obavljena prijava. Nakon toga se uzimaju ime i prezime koje pripadaju korisniku sa pripadajućim UID-om. Objekt se dodaje u bazu podataka, spremaju se promjene i kako bi se označio uspjeh, šalje se kod 200 koji označava uspješan zapis prijave u bazu, vidi sliku 3.21.

Linija Kod

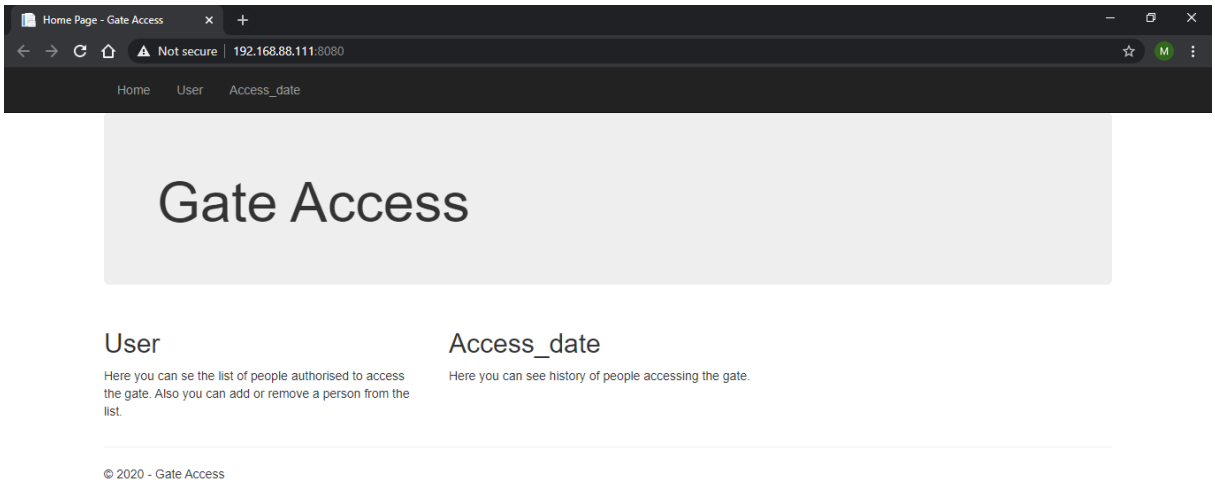
```
        public ActionResult Create1([Bind(Include = "UID")]
1:      Access_date access_date){
2:        List<Access_date> ads = db.Access_date.ToList();
3:        List<User>  users = db.User.ToList();
4:        foreach (var user in users) {
5:          if (user.UID == access_date.UID) {
6:            if (ads.Count >= 10)

7:      db.Access_date.Remove(db.Access_date.FirstOrDefault());
8:            access_date.A_Date = DateTime.Now;
9:            access_date.A_FName = user.FName;
10:           access_date.A_LName = user.LName;
11:           db.Access_date.Add(access_date);
12:           db.SaveChanges();
              return new HttpStatusCodeResult(HttpStatusCode.OK); }}
          return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }
```

Slika 3.21 Funkcija za obradu post metode i kreiranje objekta prijave

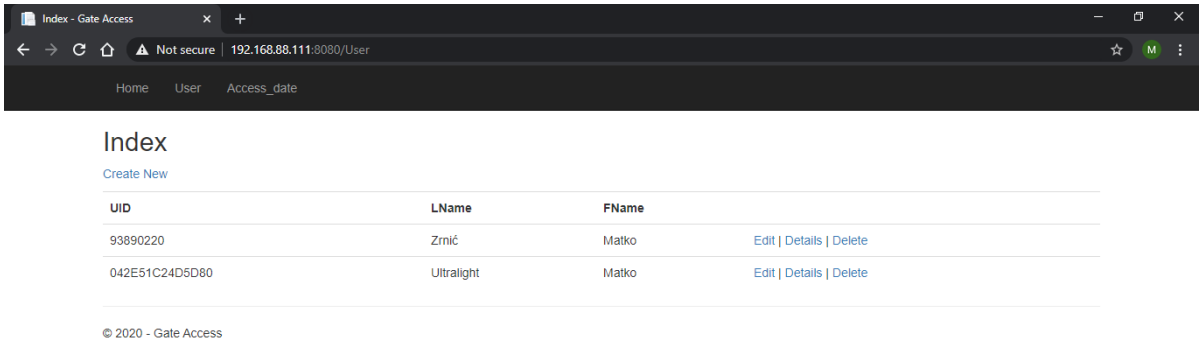
4. TESTIRANJE SUSTAVA

Kako bi se olakšala orijentacija, dodana je naslovna stranica prikazana na slici 4.1. Naslovna stranica služi kao referentna stranica preko koje se može pristupiti ostalima. Daje pristup stranici za korisnike koja prikazuje tablicu upisanih korisnika. Osim korisnika, daje pristup i stranici za datume prijave na kojoj je prikazana tablica prethodnih prijava. Kao pomagalo i radi estetskog izgleda, na naslovnoj stranici se nalaze upute za korištenje koje mogu pomoći pri organizaciji.



Slika 4.1 Naslovna stranica

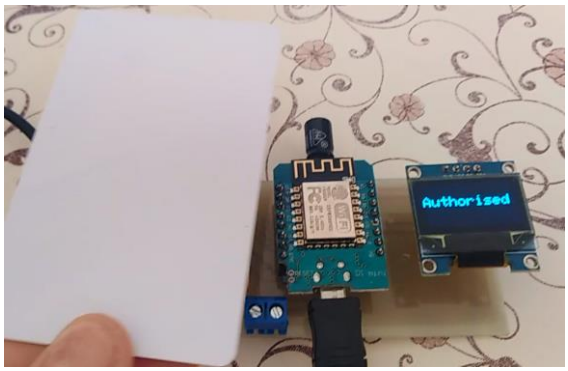
U svrhu testiranja, dodana su dva korisnika. Jedan korisnik sadrži UID mifare classic kartice. Drugi korisnik sadrži UID mifare ultralight naljepnice. Razlika je u tome što mifare classic kartice imaju UID od četiri bajta dok UID mifare ultralight naljepnice je veličine od sedam bajtova. Korisnički podatci su prikazani na slici 4.2.



Slika 4.2 Baza korisnika

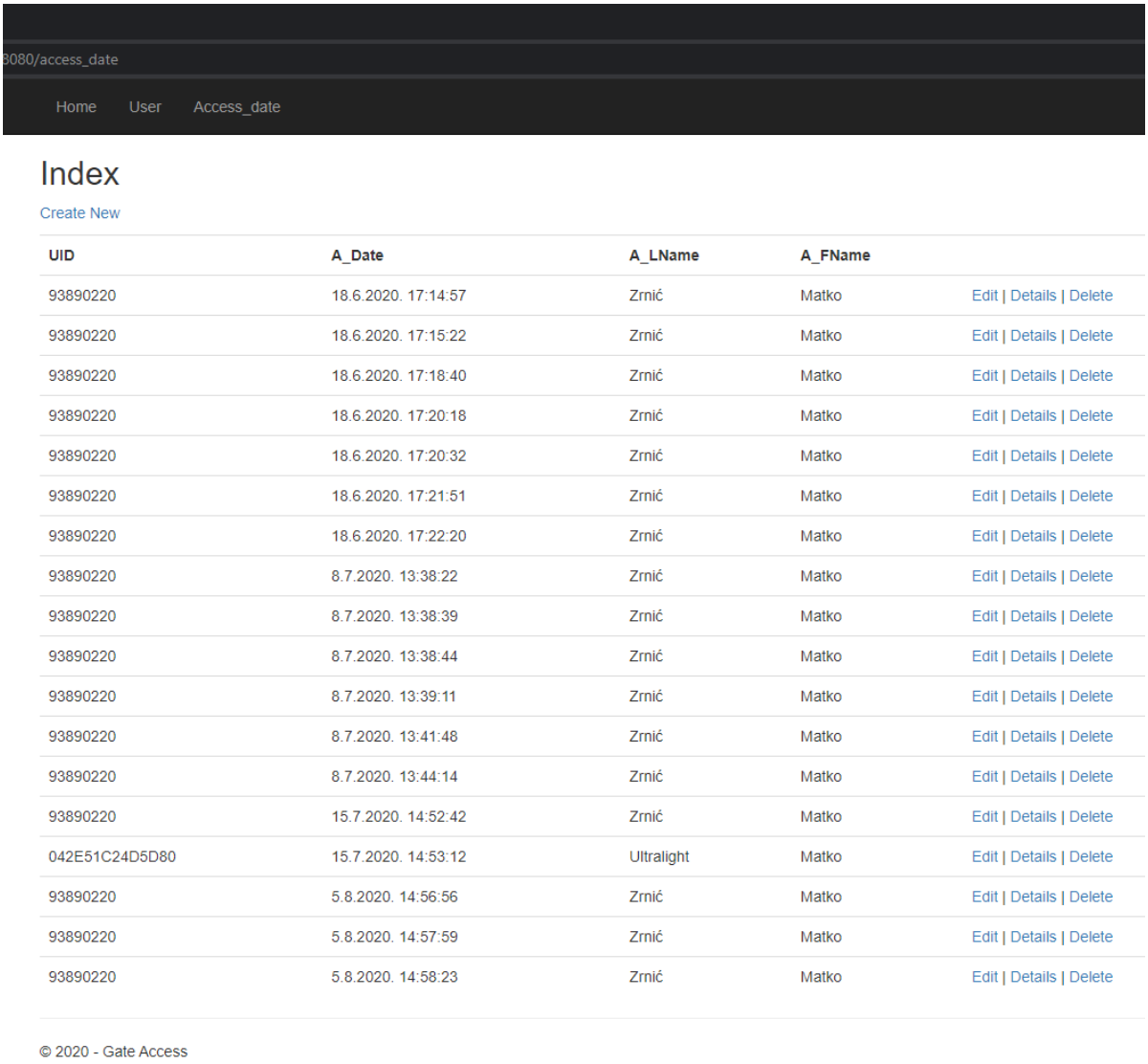
Sa upisanim korisnicima, odnosno dvije RFID oznake koje su upisane u bazi i nekoliko oznaka koje se ne nalaze u bazi, sustav je zatim testiran. Korištenjem oznaka koje nisu upisane u bazi, dobiva se odgovor sa servera i zatim se ispisuje „Wrong card!“ na ekranu. Korištenjem oznaka koje su upisane u bazi podataka, nakon komunikacije sa serverom i dobivanjem potvrde, na ekranu se ispisuje „Authorised“. Prikaz ne dozvoljenog pristupa se nalazi na slici 4.3. a pristup dozvoljenog pristupa na slici 4.4.

Slika 4.3 Nedozvoljen pristup



Slika 4.4 Dozvoljen pristup

Na slici 4.5 je prikazano nekoliko pristupa. Kao što se vidi, svaki pristup ima svoje vrijeme kada je prislonjena kartica, ime i prezime korisnika koji je obavio pristup i UID kartice koja se koristila.



UID	A_Date	A_LName	A_FName	
93890220	18.6.2020. 17:14:57	Zrnić	Matko	Edit | Details | Delete
93890220	18.6.2020. 17:15:22	Zrnić	Matko	Edit | Details | Delete
93890220	18.6.2020. 17:18:40	Zrnić	Matko	Edit | Details | Delete
93890220	18.6.2020. 17:20:18	Zrnić	Matko	Edit | Details | Delete
93890220	18.6.2020. 17:20:32	Zrnić	Matko	Edit | Details | Delete
93890220	18.6.2020. 17:21:51	Zrnić	Matko	Edit | Details | Delete
93890220	18.6.2020. 17:22:20	Zrnić	Matko	Edit | Details | Delete
93890220	8.7.2020. 13:38:22	Zrnić	Matko	Edit | Details | Delete
93890220	8.7.2020. 13:38:39	Zrnić	Matko	Edit | Details | Delete
93890220	8.7.2020. 13:38:44	Zrnić	Matko	Edit | Details | Delete
93890220	8.7.2020. 13:39:11	Zrnić	Matko	Edit | Details | Delete
93890220	8.7.2020. 13:41:48	Zrnić	Matko	Edit | Details | Delete
93890220	8.7.2020. 13:44:14	Zrnić	Matko	Edit | Details | Delete
93890220	15.7.2020. 14:52:42	Zrnić	Matko	Edit | Details | Delete
042E51C24D5D80	15.7.2020. 14:53:12	Ultralight	Matko	Edit | Details | Delete
93890220	5.8.2020. 14:56:56	Zrnić	Matko	Edit | Details | Delete
93890220	5.8.2020. 14:57:59	Zrnić	Matko	Edit | Details | Delete
93890220	5.8.2020. 14:58:23	Zrnić	Matko	Edit | Details | Delete

Slika 4.5 Popis prijava

5. ZAKLJUČAK

Cilj diplomskog rada je izvršen na način da je napravljen sklop i programska podrška koji komunicira sa serverom koji sadrži baze podataka. Sklop se sastoji od tri modula. Glavni modul je Wemos d1 mini i on je zaslužan za komunikaciju između ostalih modula i servera. Sa drugim modulom PN532 komunicira koristeći spi metodu pomoću četiri linije. Wemos prvobitno šalje naredbe za PN532 modul tako da se očitavaju RFID kartice koje se nalaze u blizini. Nakon očitavanja kartica, PN532 šalje nazad UID te kartice. Wemos zatim prosljeđuje dobiveni UID serveru koristeći internetsku konekciju. Zbog korištenja internetske konekcije se može reći da je sklop bežični jer ne koristi fizički medij. Server sadrži bazu podataka u kojoj se nalaze dvije tablice. Prva tablica sadrži korisnike gdje svaki ima UID koji pripada kartici. Druga tablica sadrži vremena kada su obavljene prijave gdje je svakom vremenu pridružen korisnik koji je obavio prijavu. Primanjem UID-a, server provjerava nalazi li se u tablici za korisnike. U slučaju da dobiveni UID nije upisan u bazi, šalje se odgovor 400. Ako UID pripada nekom od korisnika iz baze, prvobitno se upisuje prijava u drugu tablicu i vraća se odgovor 200. Ovisno o odgovoru, Wemos pomoću I2C komunikacije šalje OLED modulu naredbe za prikaz. Nakon što se dobiju naredbe, OLED prikazuje odgovarajuću poruku na ekran. Iako sklop obavlja sve potrebne funkcije, može se znatno unaprijediti. Kako bi se unaprijedio sklop, moguće je smanjiti dimenzije korištenjem manjih modula ili mijenjanjem rasporeda modula na pločici. Samim mijenjanjem rasporeda je potrebno osmisliti novu i kvalitetniju pločicu koja bi znatno poboljšala pouzdanost i kvalitetu sklopa. Osim dimenzija, moguće je napraviti kučište koje bi osim boljeg izgleda osiguralo bolju zaštitu i time smanjilo rizik od kvarova i oštećenja.

Literatura

[1] K.F. RFID Handbook, Fundamentals and Applications in Contactless Smart Cards and Identification, Second Edition, John Wiley & Sons, Ltd. Munich, Germany, 2003.

[2] DACHCOM, smart-tec, https://www.smart-tec.com/en/auto-id-world/rfid-technology, pristupljeno 13. rujna 2020.

[3] Espressif Systems, ESP32-WROOM-32 Datasheet V2.9, 2019.

[4] Steve Kemp, Steve.fi, https://steve.fi/hardware/d1-pins/, pristupljeno 13. rujna 2020.

[5] B.V. UM0701-02 PN532 User Manual, NXP, 05. studeni 2007.

[6] WINSTAR, WEO012864C

SAŽETAK

U diplomskom radu se koristi bežična RFID tehnologija sa primjenom prijave korisnika. Bežični aspekt se ostvaruje spajanjem mikroupravljaća na WiFi lokalnu mrežu. RFID se ostvaruje korištenjem modula sa mogučnošću očitavanja Mifare Classic i Mifare Ultralight oznaka. Rezultat se prikazuje na modulu sa ekranom. Sklopovlje je spojeno na posebno izrađenoj pločici. Kontrola nad sklopovljem je ostvarena programiranjem mikroupravljača koji također vrši komunikaciju sa serverom. Server se nalazi na lokalnom računalu i preko WiFi mreže komunicira sa sklopovljem. Sadrži bazu podataka u kojoj se nalaze korisnici i vremena prijava korisnika. Cijeloukupna funkcionalnost se simulira prijavom korisnika pomoću osobne oznake i ako se nalazi u bazi podataka, zapiše se vrijeme i odobrena je prijava.

Ključne riječi: RFID, RFID oznaka, Wemos, PN532.

ABSTRACT

In this masters, used technology is RFID technology with aplications in checking-in a user. Wireless aspect is acomplished by connecting microcontroler on WiFi local network. RFID is achived by using a moudle that detects Mifare Classic and Mifare Ultralight tags. Result is then showed on a display module. Modules are conected on a specialy made PCB *(Printed circut board).* Hardware is controled by programing the microcontroler that also comunicates with server. Server is located on a local computer and comunicates with modules using WiFi. Server containes a database with users and times of check-ins. Funcionality is simulated by user cheking-in with personal tag and if he is in the database, time is saved and access is authorised.

Key words: RFID, RFID tag, Wemos, PN532.

ŽIVOTOPIS

Matko Zrnić je rođen u Virovitici 25.06.1996. U Križevcima je pohađao osnovnu školu Vladimira Nazora. Nakon osnovne škole upisuje Gimnaziju Petra Preradovića u Virovitici. Gimnaziju završava 2014. godine nakon čega upisuje Fakultet Elektrotehnike, Računarstva i Informacijskih Tehnologija u Osijeku.

PRILOZI

Prilozi se nalaze nalaze na CD-u *(eng. Compact disk).*

Prilog 1	Programski kod: INO datoteka (programska podrška Wemos modula)
Prilog 2	Programski kod: Visual Studio Solution datoteka (Programska podrška servera i baze podataka)
Prilog 3	Shema: EAGLE schematic datoteka (Shema sklopovlja)