**SUSTAV ZA EVIDENCIJU PRAĆENJA STUDENATA**

Nataša Vulević, Stipe Jurković, Marko Šimić, Domagoj Čović

**Sadržaj**

[1. Uvod 2](#_Toc191237805)

[2. Opis postojećeg stanja 3](#_Toc191237806)

[3. Postojeća rješenja za kontinuirano praćenje 3](#_Toc191237807)

[4. Funkcionalni zahtjevi 3](#_Toc191237808)

[5. Nefunkcionalni zahtjevi 4](#_Toc191237809)

[6. Rješenje ovog rada 4](#_Toc191237810)

[7. Software 6](#_Toc191237811)

[8. Hardware 7](#_Toc191237812)

[9. Prototip – implementacija 9](#_Toc191237813)

[10. ZAKLJUČAK 12](#_Toc191237814)

# Uvod

Pri izvedbi nastave na FESB-u postoji potreba za praćenjem prisutnosti studenata. Dosad se to činilo na način da bi student na početku sata (ili tijekom) poništio X-icu i to je bio dokaz da je prisutan na nastavi.

Problem s tim pristupom je što student može poništiti X-icu i izaći vani tj. ne postoji kontinuirano praćenje studentove prisutnosti tijekom cijeloga sata. Drugi problem je što jedan student može poništiti više X-ica (za sebe i kolege) te samim time ovaj sustav nije dovoljno pouzdan.

U ovome radu pokušat će se riješiti taj problem tako što će se uvesti bluetooth tehnologija koja će moći kontinuirano pratiti studentovu prisutnost tijekom cijeloga sata te osigurati da se svaki student samo jednom može evidentirati na nastavi.

Također evidencija se neće izvršavati pomoću X-ica već pomoću pametnih mobitela što će smanjiti sam trošak izrade X-ica.

# Opis postojećeg stanja

U svakoj učionici postoje RFID čitači na koje student prisloni svoju karticu (X-icu) koja radi na temelju RFID tehnologije, te ona očita njegov identifikacijski broj koji se poslije spremi u bazu podataka. Dovoljno je jednom tijekom sata očitati karticu. Profesor otključava učionicu svojom RFID karticom. Na taj način evidentira svoju prisutnost i označava početak sata. Tek nakon što se profesor prijavio studentske prijave se registriraju. Na kraju sata profesor mora provući karticu što označava završetak sata. Profesor u web aplikaciji (modulu službene stranice FESB-a) može vidjeti koji su studenti prisutni te može ručno dodati studenta. Studenti također preko službene stranice FESB-a mogu pratiti prisutnost na nastavi tijekom semestra.

# Postojeća rješenja za kontinuirano praćenje

Postoji mogućnost korištenja QR-koda s kojim bi se studenti prijavljivali pri ulasku u učionicu. Nedostatak toga je što studenti trebaju doći blizu uređaja te očitati taj QR-kod što može pridodati stvaranju gužvi i gubitku vremena. Sustav koji je predložen u ovom radu ne zahtjeva stajanje blizu uređaja već čim je bluetooth upaljen na mobitelu unutar učionice čitač ga očita.

# Funkcionalni zahtjevi

* Studenti se očitavaju preko mobitela
* Profesor otključava predavaonicu i započinje evidenciju nastave
* Evidencija prisustva svakih 15 min
* Web stranica
  + Treba se koristiti postojeća stranica FESB-a koja sadrži modul o prisutnosti na nastavi
* Mobilna aplikacija
  + Treba služiti za komunikaciju sa bluetooth uređajem i za pregled podataka
* Student može vidjeti prikaz svih predavanja na kojima je prisustvovao
* Profesor može vidjeti popis svih studenata koji su prisustvovali predavanjima
* Profesor može ručno unijeti studenta
* Spajanje na postojeću bazu podataka sa informacijama o e-rasporedu

# Nefunkcionalni zahtjevi

* Sigurnost
  + Treba omogućiti sigurnost podataka koji prolaza kroz sustav
* Pouzdanost
  + Treba omogućiti da sustav učinkovito prikuplja, pohranjuje i prikazuje podatke
* Skalabilnost
  + Mora moći raditi za veći broj korisnika
* Održavanje
  + Mora biti jednostavan za održavanje

# Rješenje ovog rada

Svaki student će na mobitel instalirati aplikaciju pomoću koje će povezati svoj identitet s MAC adresom smartphone uređaja U aplikaciju će se prijaviti svojim AAIEdu lozinkom i na taj način će se znati studentov identitet. Zatim student preko mob aplikacije upiše MAC adresu mobitela te se ti podaci šalju na server. Nakon toga je MAC adresa povezana sa imenom studenta. Zbog činjenice da je ime studenta vezano za MAC adresu samog uređaja ne može se dogoditi da na istom mobitelu osoba se ulogira dvaput sa različitim računima jer sustav to neće prihvatit. Na aplikaciji postoji gumb koji automatski pali bluetooth na mobitelu.

Profesor otključava učionicu tako što očita karticu na RFID čitaču. Tako se započinje evidencija sata, te bluetooth uređaj počinje sa očitavanjem signala. Profesorova kartica ima određeni identifikacijski broj. Identifikacijski brojevi profesora kojima je dozvoljen pristup su spremljeni u memoriji mikrokontrolera. Ako je identifikacijski broj prepoznat šalje se upit u bazu podataka da se provjeri ima li taj profesor tada ima sat. Ako je rezultat potvrdan vrata se otvaraju. U slučaju da nestane mrežne veze nema potvrde iz baze podataka već ako u memoriji mikrokontrolera postoji spremljen profesorov identifikacijski broj vrata se otvaraju.

Kad započne sat studenti upale bluetooth na njihovom uređaju i čitač očita MAC adresu prisutnih osoba. Svakih 15 minuta čitač šalje signal kojeg aplikacija očita. Dva uzastopna izostanka od 15 min se smatra da student ne prisustvuje nastavi. Na taj način se prati kontinuirana prisutnost studenata.

Sustav se spaja unutar zadanog radijusa. To se postiže tako što se mjeri jačina RRSI signala. Na taj se način izbjegava da netko izvan prostorije se spoji na čitač. U prostoriji postoje dva čitača radi bolje pokrivenosti, centralni čitač i periferni. Periferni samo očitava i šalje centralnom podatke koji ih sprema u privremenu memoriju zajedno sa podacima koje je sam očitao i šalje na server preko Ethernet kabela. Na serveru se podaci spremaju u bazu podataka. U bazu se šalju odmah, a spremljeni su određeno vrijeme i na mikrokontroler u slučaju da nestane mrežne veze nakon čega se brišu. U bazi podataka već postoji raspored svih studenata, profesora i učionica i ako osoba koja je očitana čitačem ima nastavu u to vrijeme i u toj učionici podaci se spreme.

Zatim se ti podaci mogu dohvatiti putem aplikacije ili web stranice gdje profesori mogu vidjeti spisak prisutnih studenata, a studenti svoje prisustvo na raznim kolegijima. Profesori u aplikaciji mogu ručno dodati nekog studenta.



# Software

Mobilna aplikacija

* Rađena je u Android Studio-u i Apple iOS- u za Android i iPhone pametne telefone
* Na početnom sučelju se nalazi login
* Preko aplikacije student inicijalno šalje svoju MAC adresu
* Login za studenta
  + Ako to pritisne gumb za paljenje bluetootha automatski mu se počne očitavati mobitel.
  + Nakon toga mu se otvori popis kolegija – ako je u tijeku predavanje to predavanje je u vrhu, ako ne kolegiji su poredani po abecednom redu.
  + Kada klikne na aktualni kolegij ima popis svoga prisustva i checkpointova
  + Ako klikne na ostale predmete ima popis prijašnih prisustava
* Login za profesora:
  + Vidi prisutnost svih učenika
  + Može ručno dodavati studente

Sigurnost:

* Svi podaci su enkriptirani

Web aplikacija

* Studenti i profesori mogu pregledavati prisutnost studenata.
* Profesori mogu ručno nadodati studenta u vremenu trajanja kolegija
* Popis kolegija izgleda slično kao i u mobilnoj aplikaciji

# Hardware

* Pametni telefoni koje posjeduje svaki student
* Bluetooth tehnologija koja se koristi je BLE (Bluetooth Low Energy) tehnologija koja troši manje energije te je pogodnija za low-power aplikacije kao što je ova.
* **BLE čitači** – sustav se sastoji od dva čitača :centralna jedinica (ESP32- NodeMCU-32S) i periferna jedinica (ESP32- NodeMCU-32S).

Karakteristike ESP32- NodeMCU-32S:

* Mikrokontroler: ESP32 (dual-core, 32-bit, Xtensa LX6 processor).
* Brzina takta: Do 240 MHz.
* RAM: 520 KB SRAM + vanjski flash storage.
* Wi-Fi & Bluetooth:
* Wi-Fi: 802.11 b/g/n (podržava AP, STA, i AP+STA modove).
* Bluetooth: BLE 4.2 + klasični Bluetooth.
* GPIO pinovi: ~30 GPIOs (podržava for ADC, DAC, PWM, SPI, I2C, UART).
* Analogni & Digitalni I/O:
* ADC (Analog-to-Digital Converter): 12-bit (do 18 kanala).
* DAC (Digital-to-Analog Converter): 2 kanala.
* Interfejsovi: I2C, SPI, UART, CAN, PWM, itd.
* Napajanje: 3.3V (5V via USB).
* Deep Sleep Mode: Troši samo ~10 µA .
* Programiranje: Arduino IDE, MicroPython, ESP-IDF, PlatformIO.

Centralna jedinica

* Spojena je kabelom na izvor struje. Na mrežu je spojena Ethernet kabelom.
* Obrađuje podatke koje je prikupio svojim skeniranjem i skeniranjem perifernog ESP32 te ih šalje na server za pohranu i generiranje izvještaja o prisutnosti.

Periferni ESP32 uređaja

* Dodatni ESP32 mikrokontroler povezan je sa centralnim uređajem i koristi BLE za skeniranje signala prisutnih uređaja studenta.
* Periferni ESP32 uređaj ima svoje zadatke skeniranja, prikupljanje podataka o RSSI vrijednostima, i spremaju podatke u RX queove smještene u RAM-u za daljnju obradu.
* Skeneri filtriraju duplikate paketa pomoću identifikatora paketa ili vremena prijema, čime se osigurava da se samo relevantni podaci šalju centralnom ESP32 uređaju.
* ZA slanje podataka na centralni ESP32 se koristi ESP-NOW protokol koji omogućuje direktnu komunikaciju između više ESP32 uređaja bez potrebe za Wi-Fi mrežom ili internetom.
* **RFID-sustav** – služi za identifikaciju profesora, otključavanje i zaključavanje predavaonice te aktivaciju BLE čitača. Korištan je SunFounder PN532 NFC RFID

Osobine SunFounder PN532 NFC RFID

* Frekvencija: 13.56 MHz.
* Protokoli: I2C (defaultni, za jednostavnu komunikaciju), SPI (za high-speed komunikaciju), UART (Serial) (za direktnu konekciju s računalima ili mikrokontrolerima).
* NFC Standardi: ISO/IEC 14443 Type A & B (koristi se najviše pri NFC karticama), MIFARE (classic 1K, 4K, Ultralight, DESFire), FeliCa (koristi se u Japanu), NFC Peer-to-Peer (P2P) mode (služi za razmjenu podataka između dva NFC uređaja).
* Udaljenost očitanja: 3 to 5 cm
* Napajanje: 3.3V – 5V, kompatibilno sa Arduino, ESP32, Raspberry Pi.

RFID kartice- sadrže identifikacijski broj korisnika (profesora u ovom slučaju). Korištena je MIFARE Classic 1K kartica sa memorijom od 1 KB.

* **Ethernet modul za komunikaciju**

Osigurava mrežnu komunikaciju sa serverom.

* + WIZnet W5500 Ethernet modul
    - Protokoli: TCP/IP hardverski stog za stabilnu komunikaciju sa serverom.
    - Maksimalna brzina prijenosa podataka: 80 Mbps.
  + Sučelje: SPI za povezivanje s ESP32.
* **Napajanje -** DC napajanje ESP32 mikroprocesora
  + Mean Well LRS-35-5:
    - Izlazni napon: 5 V DC.
    - Snaga: 35 W.
  + PoE modul (Power over Ethernet) napajanje centralnog mikrokontrolera i njegove periferije
  + WIZnet PoE HAT: Kompatibilnost: Za W5500 Ethernet modul.
* **Server** - za server se koriste postojeći serveri na fesbu.

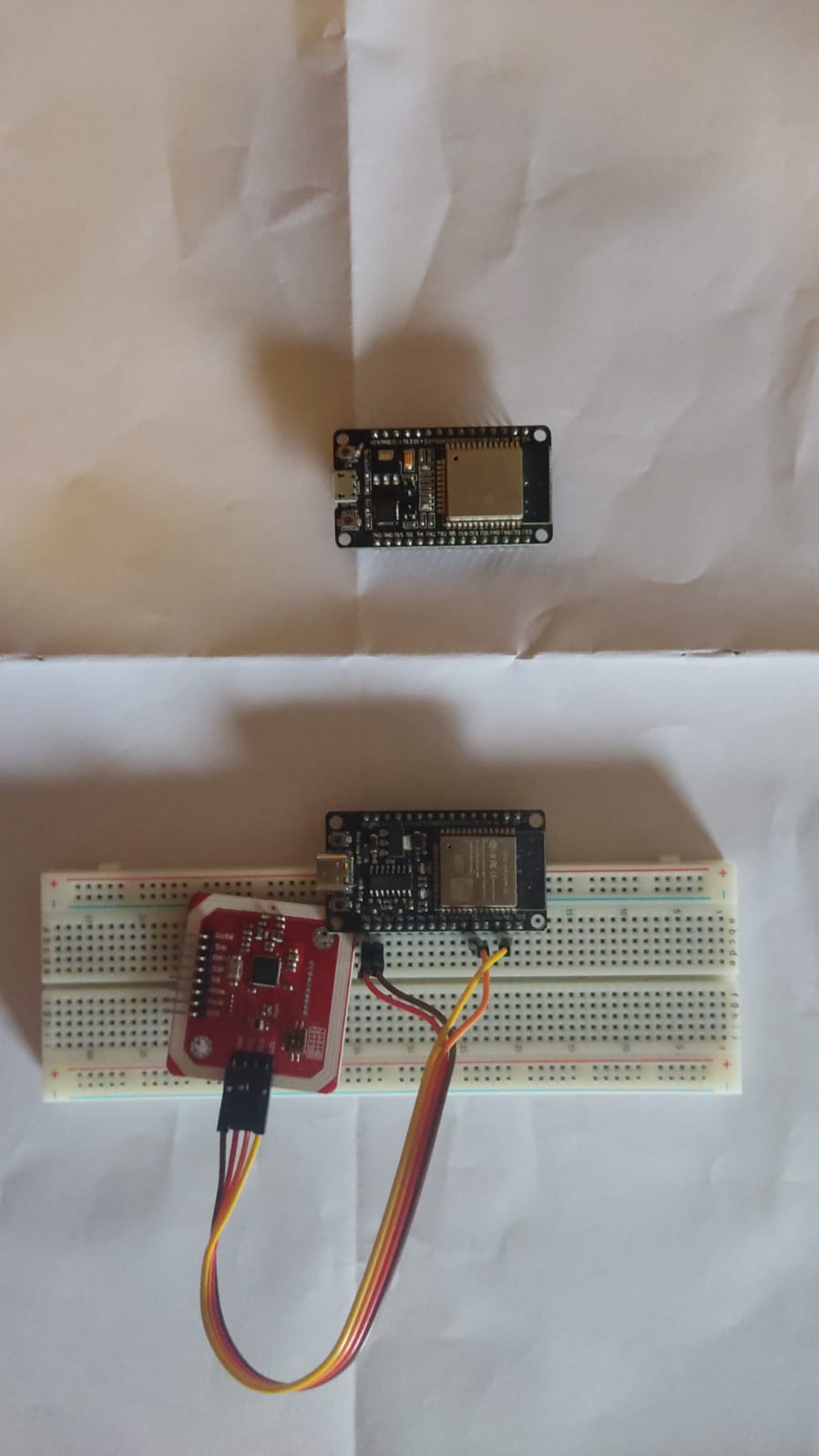
U serverima se nalazi baza podataka sa rasporedom predavanja za studente, profesore i učionice.

Nova baza podataka koja se treba napraviti je ona sa evidencijom o prisustvu.

# Prototip – implementacija

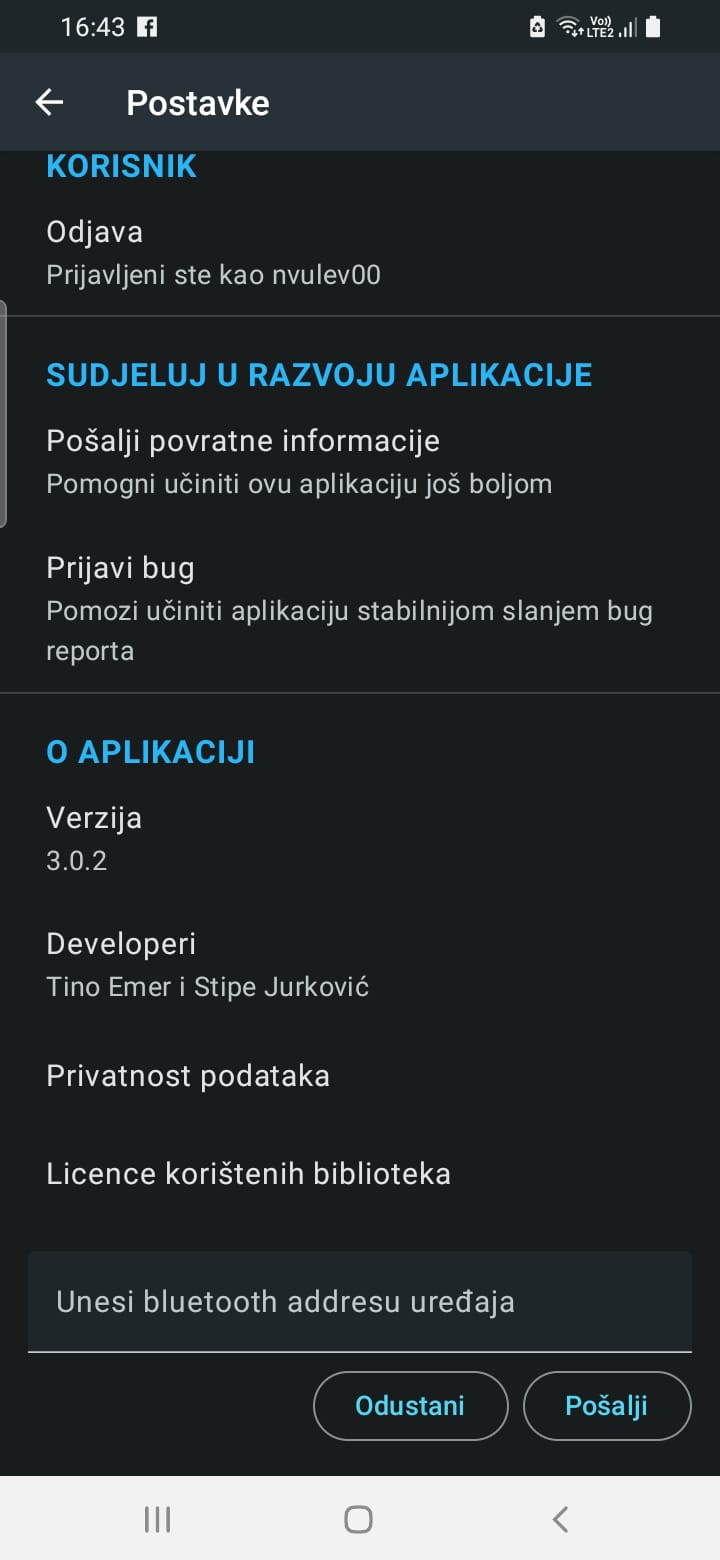
Implementacija sustava se sastoji od dva ESP32 čitača, centralnog i perifernog, RFID čitača, mobilne aplikacije, te servera koji se sastoji od web sučelja i baze podataka.

Korištena su dva ESP32- NodeMCU-32S, jedan kao centralni drugi kao periferni. RFID čitač je SunFounder PN532 NFC RFID. Karticom se aktivira RFID čitač koji pošalje signal ESP32 da počne očitavati. Očitavaju centralni i periferni čitač. Kada ESP32 detektira Bluetooth Classic advertising paket, dohvaća MAC adresu uređaja. Očitavanje koje napravi periferni ESP šalje centralnom ESP-u preko ESP-NOW protokola. Centralni ESP šalje podatke serveru. Server provjerava dolazne podatke i ako se MAC adresa nalazi u bazi podataka upisuje prisutnost studenta.



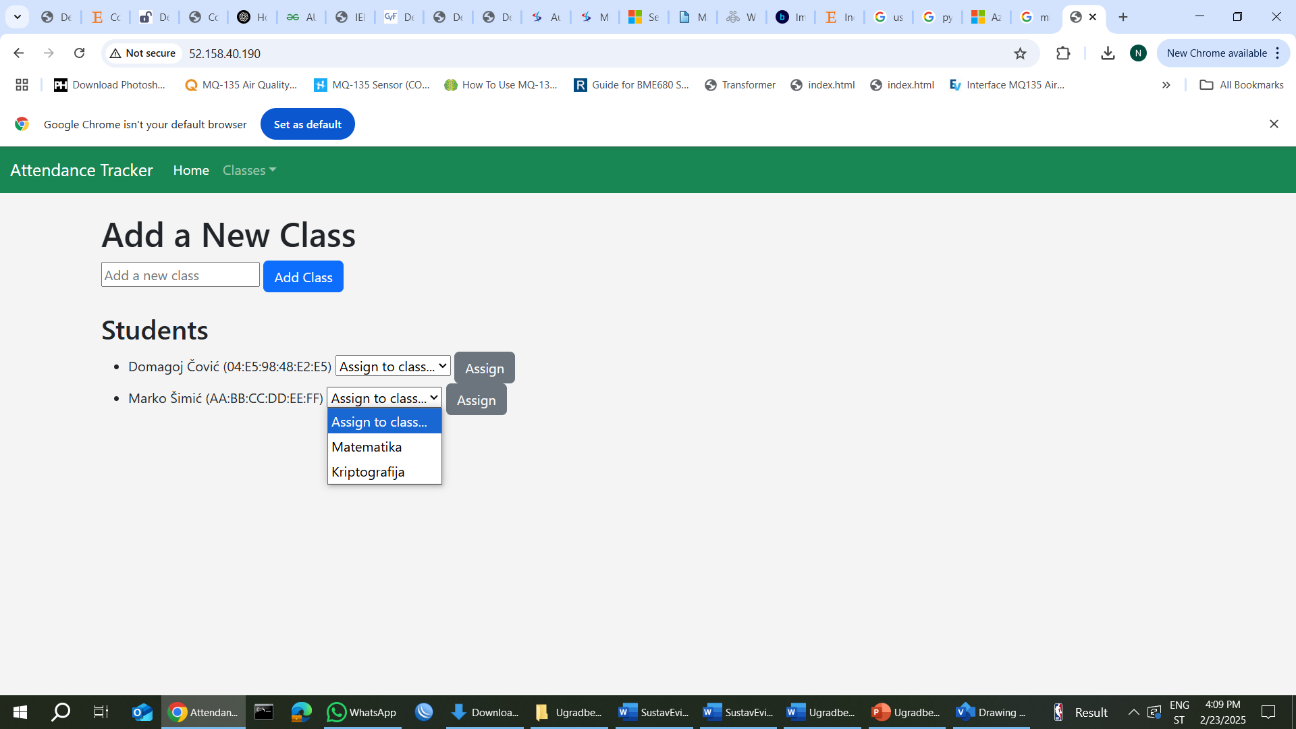
Kod za ESP32 i RFID je programiran u PlatformIO ekstenziji VSCoda.

Aplikacija rađena u Android Studio-u i ostvarena je kao modul postojeće Fesb Companion aplikacije koja sadrži podatke s FESB-ovog servera. U njoj se student prijavljuje sa svojim AAI@EduHr računom kako bi osigurao jedinstvenu identifikaciju, te unosi MAC adresu uređaja u aplikaciju koja se šalje na server. MAC adresa se sprema u bazu podataka i trajno je povezana sa korisničkim računom studenta.



Server je podignut na Microsoft Azure servisu korištenjem Flask frameworka. Server se sastoji od baze podataka i web sučelja za upravljanje prisutnošću. Korištena je SQLite baza podataka. Centralni ESP32 šalje HTTP POST zahtjev serveru s podacima o prisutnim studentima. Flask server provjerava dolazne podatke i evidentira prisutnost u bazu podataka. Kontinuirano se povećava broj instanci skeniranja MAC adrese studenta, u zadanom intervalu od 15 min što predstavlja kontinuiranu prisutnost studenta u predavaonici..

Web sučelje je razvijano u Bootstrap frameworku. Profesori mogu koristiti sučelje za pregled prisutnosti studenata u realnom vremenu te mogu dodavati nove studente i predmete. Studenti mogu pregledavati svoju evidenciju prisutnosti.



# ZAKLJUČAK

Ovaj bluetooth sustav praćenja prisutnosti omogućava modernizirano i poboljšano praćenje prisutnosti od prošlog sustava. Može se pratiti kontinuirano prisustvo studenta na nastavi, te omogućiti da se jedan student očita samo jednom, te je smanjena mogućnosti da se lažno evidentira student koji nije na nastavi. Također je sustav dugoročno jeftiniji je nema potreba za izdavanjem X-ica s obzirom da se koriste pameti telefoni. Ne postoji potreba za timbravanjem te su gužve pri početku nastave manje.

Ovo rješenje doprinosi digitalizaciji obrazovanja, unapređujući učinkovitost i iskustvo svih sudionika nastavnog procesa.