

# Pametna knjižnica: analiza iskoristivosti potencijala interneta stvari za efikasno vođenje knjižnice

---

**Begić, Magdalena**

**Undergraduate thesis / Završni rad**

**2024**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **VERN University / Sveučilište VERN**

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:146:487053>

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

*Download date / Datum preuzimanja:* **2024-10-02**



*Repository / Repozitorij:*

[VERN' University Repository](#)



**SVEUČILIŠTE VERN'**

Zagreb

Internet stvari

**ZAVRŠNI RAD**

**Pametna knjižnica: analiza iskoristivosti potencijala  
interneta stvari za efikasno vođenje knjižnice**

Magdalena Begić

Zagreb, 2024.

**SVEUČILIŠTE VERN'**

Sveučilišni prijediplomski studij

Internet stvari

**ZAVRŠNI RAD**

**Pametna knjižnica: analiza iskoristivosti potencijala  
interneta stvari za efikasno vođenje knjižnice**

Mentor: doc. dr. sc. Petar Kolar

Studentica: Magdalena Begić

Zagreb, srpanj 2024.

# SADRŽAJ

PREDGOVOR.....	I
SAŽETAK.....	II
ABSTRACT .....	III
1. UVOD.....	1
2. TEORIJSKI PRIKAZ POTENCIJALNIH RJEŠENJA .....	2
2.1. Opis radiofrekvencijske identifikacije na primjeru knjižnice.....	3
2.2. Prednosti primjene IoT tehnologija u knjižnicama.....	5
3. PRIJEDLOG RJEŠENJA TEMELJEN NA RADIOFREKVENCIJSKOJ IDENTIFIKACIJI.....	8
3.1. Specifikacije i primjena Raspberry Pi mikroracunala .....	8
3.2. SPI protokol i njegova primjena u RC522 RFID modulu .....	10
4. EKSPERIMENTALNA DEMONSTRACIJA FUNKCIONALNOSTI PREDLOŽENOG RJEŠENJA .....	14
4.1 Metode i metodologija istraživanja.....	14
4.2 Razvoj konceptualnog modela.....	14
4.3 Komunikacija između komponenti .....	18
4.4 Izrada baze podataka .....	22
4.5 Izrada <i>web</i> -aplikacije .....	23
ZAKLJUČAK .....	34
LITERATURA.....	35
PRILOG: Kazalo kartica .....	37
POPIS TABLICA I SLIKA .....	38

## **PREDGOVOR**

Željela bih zahvaliti svojim roditeljima i sestri Miji na ogromnoj podršci i razumijevanju tijekom cijeloga mog studija.

Zahvaljujem i svom mentoru doc. dr. sc. Petru Kolaru na njegovom neiscrpnom trudu, angažmanu i pomoći tijekom izrade ovog praktičnog rada.

Posebnu zahvalnost dugujem svojem prijatelju Josilu (Josip Igrec), njegova stručnost u programiranju i spremnost da pomogne bili su neprocjenjivi.

Hvala svim profesorima Sveučilišta VERN' koji su mi pokazali da mogu postići sve što poželim, pogotovo profesorici doc. dr. sc. Danijeli Unić čiji su mi savjeti uvelike pomogli tijekom studiranja.

## SAŽETAK

Ovim radom analizira se iskoristivost potencijala Interneta stvari (IoT) za efikasno vođenje knjižnice. Istraživanje će obuhvatiti primjenu RFID tehnologije za praćenje i upravljanje knjižničnim fondom te izradu konceptualnog modela pametne knjižnice korištenjem Raspberry Pi mikroračunala i RFID modula. Također će biti razvijena *web*-aplikacija koja knjižničarima omogućava pregled i analizu podataka putem modernih *dashboardova* i *grafova*.

Ciljevi rada uključuju: objašnjenje tehnologije radiofrekvencijske identifikacije i njezinog potencijala korištenja u knjižnicama, izradu konceptualnog dokaza rada pametne knjižnice, razvoj primjera *web*-aplikacije za knjižničare, analizu strukture interneta stvari na konstruiranom primjeru, s naglaskom na načinu povezivanja i komunikaciji korištenih komponenti.

Primjena IoT tehnologija trebala bi značajno poboljšati operativnu efikasnost knjižnice, omogućiti precizno praćenje knjiga, smanjiti operativne troškove te unaprijediti korisničko iskustvo. *Web*-aplikacija će omogućiti knjižničarima detaljnu analizu podataka, što će im pomoći u donošenju informiranih odluka o nabavi knjiga, optimizaciji resursa i praćenju korisničkih aktivnosti. Očekuje se da će rezultati ovog istraživanja pokazati kako pametne tehnologije mogu unaprijediti rad knjižnica i pružiti temelj za daljnja istraživanja i razvoj sličnih sustava.

Ključne riječi: Internet stvari, RFID, pametna knjižnica, *web*-aplikacija, Raspberry Pi

## **ABSTRACT**

### **Smart Library: Analysis of the Potential Utilization of the Internet of Things for Efficient Library Management**

This paper analyzes the potential of Internet of Things (IoT) for efficient library management. The research will encompass the application of RFID technology for tracking and managing the library collection, as well as the development of a conceptual model of a smart library using Raspberry Pi microcomputers and RFID modules. A web application will also be developed to enable librarians to review and analyze data through modern dashboards and graphs.

The objectives of the paper include: explaining the technology of radio-frequency identification and its potential use in libraries, creating a conceptual proof of smart library operations, developing a sample web application for librarians, and analyzing the structure of the Internet of Things in the constructed example, with an emphasis on the connection and communication of the used components.

The application of IoT technology is expected to significantly improve the operational efficiency of the library, enable precise tracking of books, reduce operational costs, and enhance the user experience. The web application will allow librarians to perform detailed data analysis, helping them make informed decisions regarding book acquisitions, resource optimization, and monitoring user activities. It is anticipated that the results of this research will demonstrate how smart technologies can improve library operations and provide a foundation for further research and development of similar systems.

Keywords: Internet of Things, RFID, smart library, web application, Raspberry Pi

## 1. UVOD

U suvremenom dobu digitalne transformacije, knjižnice se suočavaju s izazovom kako unaprijediti svoje operacije i pružanje usluga koristeći napredne tehnologije. Internet stvari (IoT) predstavlja potencijalno rješenje za optimizaciju knjižničnih procesa i poboljšanje korisničkog iskustva. Među različitim IoT tehnologijama, radiofrekvencijska identifikacija (RFID) izdvaja se kao izuzetno korisna za primjenu u knjižnicama zbog njezinog omogućavanja efikasnog praćenja, upravljanja i organiziranja knjižničnih materijala.

Problem koji se istražuje u ovom radu odnosi se na potrebu za efikasnijim upravljanjem knjižničnim resursima i pružanjem naprednih usluga prvotno knjižničarima, a zatim korisnicima knjižnice. Predmet istraživanja je primjena RFID tehnologije u knjižnicama te analiza kako ta tehnologija zajedno s podatkovnim izvještajima može unaprijediti knjižnično poslovanje.

Motivacija za odabir ove teme proizlazi iz potrebe za modernizacijom knjižnica te unaprjeđenjem rada knjižničara. Korištenjem RFID tehnologije, zajedno s analizom izvještaja podataka dobivenih od nje, knjižnice mogu smanjiti operativne troškove, ubrzati procese zaduživanja i vraćanja knjiga, osigurati precizno praćenje inventara, poboljšati korisničko iskustvo te omogućiti knjižničarima više vremena da se fokusiraju na najbitnije zadatke.

Rad je strukturiran u nekoliko poglavlja. Nakon uvodnog dijela, drugo poglavlje pruža teorijski okvir s detaljnim opisom RFID tehnologije i njezine primjene. Treće poglavlje predlaže konkretno rješenje za primjenu RFID tehnologije u knjižnicama. Četvrto poglavlje prikazuje konceptualni dokaz rada pametne knjižnice, uključujući opis hardverskih komponenti (kartonski modeli police, kutije i stola, Raspberry Pi i RFID moduli) i razvijene *web*-aplikacije. Rad završava zaključkom koji sažima ključne nalaze.

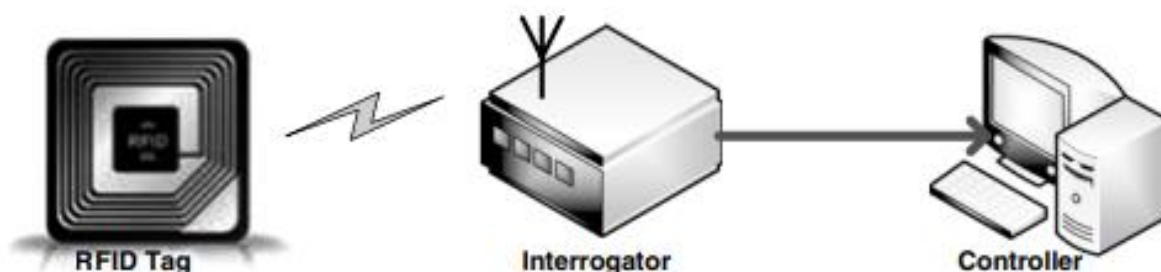


## 2. TEORIJSKI PRIKAZ POTENCIJALNIH RJEŠENJA

Prema CARNET-ovom Nacionalnom CERT-u<sup>1</sup> u suradnji s LS&S-om<sup>2</sup> (2007. g.) radiofrekvencijska identifikacija (eng. *Radio-Frequency Identification*, RFID) omogućava bežičnu komunikaciju između uređaja, u ovom slučaju između RFID čitača i RFID oznaka na knjigama. U RFID sustavu, komunikacija se ostvaruje putem elektromagnetskih polja. RFID čitač emitira radiovalove putem svoje antene, stvarajući elektromagnetsko polje. Kada RFID oznaka uđe u ovo polje, apsorbira energiju iz polja i koristi je za slanje svojeg jedinstvenoga identifikacijskog broja natrag čitaču. RFID čitač može očitavati više RFID oznaka u isto vrijeme što će kasnije biti i dokazano u praktičnom dijelu radu.

Na slici 2.1 prikazan je osnovni koncept RFID sustava koji se sastoji od RFID oznake, RFID čitača (*interrogator*) i kontrolera. RFID oznaka komunicira s interogatorom putem radiovalova, a interogator prenosi informacije kontroleru koji ih obrađuje i pohranjuje u bazu podataka.

Slika 2.1. Prikaz RFID sustava



Izvor: Iqbal i sur., 2021.

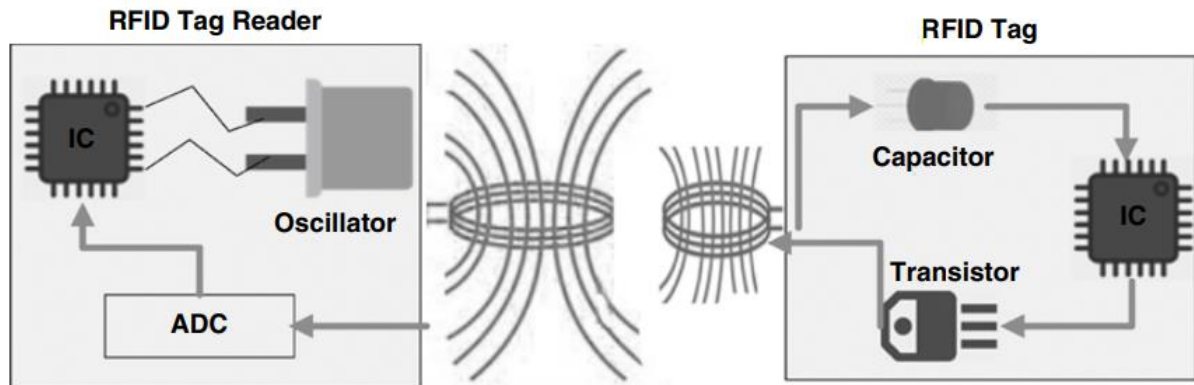
---

<sup>1</sup> CARNet CERT - nacionalno središte za sigurnost računalnih mreža i sustava.

<sup>2</sup> LS&S - laboratorij za sustave i signale pri Zavodu za elektroničke sustave i obradbu informacija Fakulteta elektrotehnike i računarstva Sveučilišta u Zagrebu.

Slika 2.2 prikazuje detaljan rad RFID tehnologije. RFID čitač koristi oscilator za generiranje elektromagnetskog polja koje aktivira RFID oznaku. Oznaka koristi dobivenu energiju za slanje svoga jedinstvenog identifikacijskog broja natrag čitaču.

Slika 2.2.: Prikaz rada RFID tehnologije



Izvor: Iqbal i sur., 2021.

Podskup RFID tehnologije koji radi na visokoj frekvenciji (13.56 MHz) i omogućava komunikaciju na vrlo kratkim udaljenostima je NFC (*Near Field Communication*). Međutim, za razliku od RFID-a, NFC čitači mogu očitavati samo jednu oznaku u jednom trenutku, što ograničava njegovu upotrebu u aplikacijama koje zahtijevaju brzu identifikaciju više objekata. RFID čitači mogu očitavati više oznaka istovremeno<sup>3</sup>, što ga čini idealnim za aplikacije koje zahtijevaju masovno praćenje objekata. Za bolje razumijevanje razlike između ove dvije tehnologije, korisno je napomenuti da se RFID često koristi kod ulaza u stambene zgrade ili za otvaranje kontejnera za smeće, dok se NFC koristi kod beskontaktnih bankovnih transakcija.

## 2.1. Opis radiofrekvencijske identifikacije na primjeru knjižnice

Kako RFID sustav funkcionira u kontekstu knjižnice (Iqbal i sur. 2021):

<sup>3</sup> WirelessLinks. (2021). RFID vs NFC: What's The Difference? <https://wlius.com/blog/rfid-vs-nfc-whats-the-difference/>

1. RFID oznake na knjigama: svaka knjiga u knjižnici ima malu RFID oznaku koja sadrži jedinstveni identifikator (UID). Ove oznake su obično nalijepljene unutar korica knjige.
2. RFID čitač: kada knjiga s RFID oznakom dođe u blizinu čitača, oznaka prima energiju od čitača i šalje svoj jedinstveni identifikator natrag čitaču.
3. Čitanje **jedinstvenog** ID-a: čitač hvata ovu transmisiju, uključujući jedinstvenu ID oznaku.
4. Usporedba s bazom podataka: čitač šalje uhvaćeni jedinstveni ID na računalo koje je povezano s njim.
5. Dohvaćanje informacija iz baze podataka: računalo provjerava jedinstveni ID u bazi podataka kako bi pronašlo informacije o odgovarajućoj knjizi, kao što su naslov, autor, lokacija u knjižnici, itd.
6. Obrada i akcija: na temelju operacije, sustav ažurira status knjige u bazi podataka (Iqbal i sur. 2021.)

Ovi koraci omogućuju efikasnije upravljanje knjižničnim fondom, smanjenje grešaka i bolju kontrolu nad inventarom, što rezultira poboljšanjem cjelokupnoga korisničkog iskustva.

Specifični problemi s kojima se susreće prilikom rada u knjižnici uključuju:

- Traženje literature na policama: iako svaka knjiga u knjižničarskim sustavima ima svoju određenu signaturu, tj. oznaku smještaja knjige s pomoću koje se pronalazi građa, najčešće klasificirana po UDK<sup>4</sup> broju, lokacija knjige može se promijeniti jer korisnici knjižnice ponekad premještaju knjige na pogrešne police ili ih ostavljaju na stolovima. Knjižničar tada mora izgubiti vrijeme na traženje knjige, a korisnik knjižnice mora čekati (ukoliko je od knjižničara zatražio da mu pronađe određenu literaturu).
- Velike gužve tijekom određenih perioda rada knjižnice: u sveučilišnim knjižnicama ponekad tri različita studija posuđuju istu literaturu za predmet i često imaju pauzu između nastave u isto vrijeme. Tu pauzu koriste za dolazak i posuđivanje literature, što stvara gužve. Knjižničar mora biti svjestan

---

<sup>4</sup> UDK – Univerzalna decimalna klasifikacija - opća knjižnična klasifikacija, jedna od najčešće upotrebljivanih u Europi, u širokoj primjeni i u Hrvatskoj, Hrvatska endiklopedija)

rasporeda svih studija kako bi na vrijeme upozorio druge zaposlenike, pripremio literaturu i osigurao da se proces posudbe odvija glatko.

- Planiranje naručivanja broja primjeraka određene literature: ponekad se naruči prevelik broj primjeraka određene knjige jer predavač odluči naručiti dodatne primjerke literature koju koristi već pet godina, ali statistika pokazuje da tu literaturu nitko nije koristio u posljednjih pet godina. Studenti su koristili isključivo predavačku skriptu. Takva se statistika može dobiti isključivo ako knjižničar osobno prati te podatke i ima te informacije na umu.

Ovi problemi zahtijevaju dodatno vrijeme i resurse knjižničara te otežavaju efikasno upravljanje knjižničnim fondom i korisničkim uslugama. S praktičnim dijelom ovoga rada pokušat će se pokazati koliko podatkovni izvještaji ukomponirani u *web*-aplikaciju zajedno s RFID tehnologijom mogu pomoći kod gore navedenih problema i situacija.

## **2.2. Prednosti primjene IoT tehnologija u knjižnicama**

Povezivanje različitih IoT komponenti (kao što su RFID čitači, senzori i Raspberry Pi mikroracunala) sa centralnom bazom podataka omogućuje:

- automatizirano praćenje i upravljanje - knjige opremljene RFID oznakama mogu se automatski pratiti, a informacije o njihovom statusu i lokaciji mogu se automatski ažurirati u bazi podataka
- povećanu efikasnost - automatizacija smanjuje potrebu za ručnim unosom podataka, čime se smanjuje mogućnost pogrešaka i povećava efikasnost rada knjižničara
- poboljšano korisničko iskustvo - korisnici mogu lako pretraživati dostupnost knjiga i pratiti njihove posudbe putem *web*-aplikacije, čime se poboljšava ukupno korisničko iskustvo.

Rješavanje problema pomoću interneta stvari podrazumijeva prikupljanje i analiziranje velikih količina podataka, što se može koristiti za izradu detaljnih izvještaja i grafova. Prednosti ovoga uključuju:

- bolje donošenje odluka - analiza podataka omogućava knjižničarima donošenje informiranih odluka na temelju stvarnih podataka, kao što su popularnost određenih knjiga ili žanrova
- identifikaciju trendova - grafovi i izvještaji omogućavaju praćenje trendova posudbe i vraćanja knjiga, što može pomoći u boljem planiranju nabave i organizacije fonda
- optimizaciju resursa - detaljni izvještaji omogućuju optimizaciju korištenja resursa knjižnice, uključujući prostor za skladištenje i radno vrijeme osoblja.

IoT tehnologije transformiraju tradicionalne knjižnice u pametne knjižnice, omogućujući im da postanu dinamičniji i interaktivniji prostori. Kroz povezivanje fizičkih objekata s internetom i korištenje podataka za donošenje odluka, knjižnice mogu pružiti bolje usluge svojim korisnicima i efikasnije upravljati svojim resursima.

Jedan od takvih primjera je gradska knjižnica u Požegi koja je iskoristila RFID tehnologiju za samposlužne uređaje i sigurnosna vrata knjižnice, predstavljajući javnosti poruku da knjižnice više nisu "samo mjesta gdje se čuvaju i daju na korištenje knjge", već da je moderna knjižnica 21. stoljeća kulturno središte u kojem se "čita, boravi, druži i uči"<sup>5</sup>.

S druge strane, jako je malo primjera knjižnica koji se mogu navesti da koriste moderna *web*-sučelja za knjižničare, a ako i koriste izvještaje u obliku grafikona u svojim aplikacijama oni se većinom odnose samo na analiziranje broja posjetitelja. "Knjižnice moraju usmjeriti svoje napore na praktične primjene vizualizacije podataka koje će im pomoći da steknu vrijedne uvide za donošenje važnih odluka, kao što je odabir resursa za obnovu." (Hsuanwei, 2017, str. 19). Hsuanwei također kaže da "Pokretačka snaga koja čini vizualizaciju podataka nužnom u knjižnicama najbolje se može objasniti riječima Jannette Finch i Angele Flenner. U raspravi o primjeru navedenom u njihovoj studiji, one navode da "na današnju knjižničku zbirku utječe mnogo faktora, kao što su akvizicije vođene potražnjom, pristup, *streaming* mediji,

---

<sup>5</sup> Slavonski.hr (2021). Gradska knjižnica Požega: Želimo da ljudi osvijeste koncept nove knjižnice u novoj zgradi. <https://slavonski.hr/gradska-knjiznica-pozega-ravnateljica-aleksandra-sutalo-zelimo-da-ljudi-osvijeste-koncept-nove-knjiznice-u-novoj-zgradi/> (1.7.2024.)

interdisciplinarni predmeti, entuzijazam za naručivanje, nova područja studija, politički pritisci, promjene dobavljača i pojedinačni članovi fakulteta koji slijede određenu liniju istraživanja. Ako knjižnice ne alociraju resurse na temelju podataka, tada bi moglo doći do subjektivne raspodjele sredstava, što bi utjecalo na percepciju pravednosti i štetilo reputaciji knjižnice na sveučilištu." Ovo je važan čimbenik koji se mora adresirati u današnjem svijetu." (Hsuanwei, 2017, str. 18).

Ovim radom pokušat će se pokazati kako se podatkovni izvještaji mogu koristiti za dobrobit rada knjižničara, koristeći podatke od samih knjiga i aktivnosti knjižnice.

### 3. PRIJEDLOG RJEŠENJA TEMELJEN NA RADIOFREKVENCIJSKOJ IDENTIFIKACIJI

Pojam "pametno" u kontekstu interneta stvari označava efikasnija, fleksibilnija i organiziranija rješenja s integracijom digitalnih procesa. Umjesto oslanjanja na tradicionalnu organizaciju polica za knjige, RFID omogućava precizno praćenje lokacije svake knjige na policama. Knjižničari mogu u realnom vremenu vidjeti trenutne knjige na svakoj lokaciji putem softverskog sučelja. Dodatno, sustav generira sveobuhvatne izvještaje i analitiku, uključujući grafikone, kako bi pružio uvide u obrasce korištenja knjiga.

Projekt je započet izradom vlastitog RFID čitača pomoću Raspberry Pi 4 mikroračunala i RFID modula RC-522<sup>6</sup>. Iako postoje napredniji uređaji s većim dometima i brzinom očitavanja oznaka, ovaj pristup omogućava osnovno razumijevanje i testiranje sustava.

#### 3.1. Specifikacije i primjena Raspberry Pi mikroračunala

Raspberry Pi je serija malih, pristupačnih mikroračunala koju je razvio Raspberry Pi Foundation, organizacije sa sjedištem u Ujedinjenom Kraljevstvu čiji je cilj promoviranje računalne edukacije i olakšavanje pristupa računalnoj tehnologiji. Raspberry Pi funkcionira unutar *open source* ekosustava: pokreće Linux (razne distribucije), a glavni podržani operativni sustav, Pi OS, je *open source*<sup>7</sup> i pokreće niz *open source* softvera. Shematski prikazi Raspberry Pi-ja redovito se objavljuju kao dokumentacija, ali ploča nije *open hardware*<sup>8</sup>. Tijekom izrade praktičnog dijela ovog rada, korišten je Raspberry Pi model 4.

---

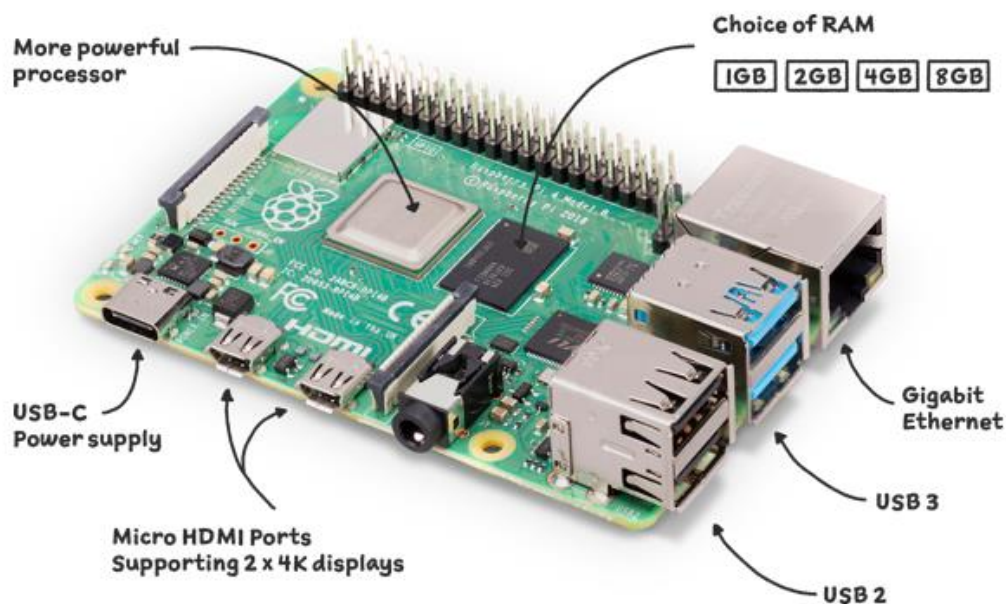
<sup>6</sup> Poveznica na dokumentaciju RFID modula RC-522: <https://www.nxp.com/docs/en/data-sheet/MFRC522.pdf>

<sup>7</sup> Termin Open source (otvoreni ili slobodni softver) se koristi kako bi se opisali softveri koje korisnici/e mogu koristiti potpuno besplatno i bez ograničavanja na koji način se koriste. opensource.com. (n. d.). Raspberry Pi. <https://opensource.com/resources/raspberry-pi> (20.5.2024.)

<sup>8</sup> Open hardware se odnosi na fizičke artefakte tehnologije koji su dizajnirani i ponuđeni na način koji omogućava svakome da ih pregleda, modificira, distribuira i koristi.

Slika 3.1 prikazuje izgled Raspberry Pi mikroračunala korištenog u ovom projektu. U ovom radu, Raspberry Pi je korišten za upravljanje RFID čitačima i komunikaciju s centralnim sustavom putem mreže.

Slika 3.1. Prikaz Raspberry Pi 4 mikroračunala



Izvor: Raspberry Pi, <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/>

U kontekstu ovog projekta, SSH je korišten za daljinsko upravljanje Raspberry Pi mikroračunalima. *Secure Shell* (SSH) je kriptografski mrežni protokol koji omogućava sigurno upravljanje mrežnim uslugama putem nesigurne mreže. SSH pruža snažnu autentifikaciju i sigurnu komunikaciju putem šifriranog kanala (Turopoli, 2015.)

U kasnijim poglavljima bit će prikazano kako su četiri Raspberry Pi mikroračunala spojena na istu mrežu i povezana s prijenosnim računalom preko SSH. Ovaj način povezivanja omogućava pokretanje Python skripti s prijenosnog računala te skeniranje RFID oznaka. Osim toga, na prijenosnom računalu je također postavljena *web*-aplikacija koja knjižničarima omogućava pregled i upravljanje podacima o knjigama.



Korištenjem SSH-a, cjelokupni sustav je siguran i jednostavan za upravljanje, što dodatno doprinosi praktičnosti i funkcionalnosti konceptualnog modela pametne knjižnice. Dokazano je da Raspberry Pi mikroračunala mogu učinkovito upravljati RFID čitačima i omogućiti komunikaciju s centralnim sustavom putem mreže. Primjena ovih mikroračunala omogućava fleksibilno i pristupačno rješenje za praćenje i upravljanje knjižničnim fondom

### 3.2. SPI protokol i njegova primjena u RC522 RFID modulu

SPI (*Serial Peripheral Interface*) je serijski komunikacijski protokol koji omogućava brzu komunikaciju između mikrokontrolera i perifernih uređaja. SPI je *full-duplex* protokol, što znači da se podaci mogu prenositi u oba smjera istovremeno.<sup>9</sup> SPI koristi četiri osnovne linije za komunikaciju<sup>10</sup>:

1. MOSI (*Master Out Slave In*) - linija za prijenos podataka od glavnog uređaja (*master*) prema podređenom uređaju (*slave*).
2. MISO (*Master In Slave Out*) - linija za prijenos podataka od podređenog uređaja (*slave*) prema glavnom uređaju (*master*).
3. SCLK (*Serial Clock*) - linija kojom master šalje serijski satni signal za sinkronizaciju.
4. SS/CS (*Slave Select/Chip Select*) - linija za odabir podređenog uređaja (*slave*) koji će komunicirati s glavnim uređajem (*master*).

Ovaj protokol odabran je za izradu praktičnog rješenja ovog rada jer je poznat po svojoj brzini i efikasnosti te se često koristi za povezivanje mikrokontrolera s različitim perifernim uređajima kao što su senzori, memorijski čipovi, ekrani, i u ovom slučaju, RFID moduli.

Kako bi se ostvarila komunikacija između Raspberry Pi i RC522 modula, potrebno je spojiti odgovarajuće SPI pinove na oba uređaja.

Korišteni pinovi (pinout.xyz. (n. d.). SPI Pinout.):

---

<sup>9</sup> Biomedicinska instrumentacija. SPI (Serial Peripheral Interface) komunikacioni protokol. <http://kelm.ftn.uns.ac.rs/literatura/bmi/2022-23/SPI.pdf>

<sup>10</sup> NXP Semiconductors. (2016). MFRC522: Standard performance MIFARE and NTAG frontend. <https://www.nxp.com/docs/en/data-sheet/MFRC522.pdf> (20.6.2024.)

- SDA/SS (*Slave Select*) - povezuje se na GPIO pin 8 (SPI0 CE0) na Raspberry Pi.
- SCK (*Serial Clock*) - povezuje se na GPIO pin 11 (SPI0 SCLK) na Raspberry Pi.
- MOSI (*Master Out Slave In*) - povezuje se na GPIO pin 10 (SPI0 MOSI) na Raspberry Pi.
- MISO (*Master In Slave Out*) - povezuje se na GPIO pin 9 (SPI0 MISO) na Raspberry Pi.
- IRQ (*Interrupt*) - nije nužno koristiti, može ostati nepovezan.
- GND (*Ground*) - povezuje se na GND pin na Raspberry Pi.
- RST (*Reset*) - povezuje se na GPIO pin 25 na Raspberry Pi.
- 3.3V - povezuje se na 3.3V pin na Raspberry Pi.

Slika 3.2 prikazuje *pinout* Raspberry Pi mikroračunala s označenim GPIO pinovima koji se koriste za povezivanje s različitim komponentama putem SPI protokola.

Slika 3.2. Prikaz Raspberry Pi pinova za povezivanje pomoću SPI protokola

3v3 Power	1	•	•	2	5v Power
GPIO 2 (I2C1 SDA)	3	•	•	4	5v Power
GPIO 3 (I2C1 SCL)	5	•	•	6	Ground
GPIO 4 (GPCLK0)	7	•	•	8	GPIO 14 (UART TX)
Ground	9	•	•	10	GPIO 15 (UART RX)
GPIO 17 (SPI1 CE1)	11	•	•	12	GPIO 18 (SPI1 CE0)
GPIO 27	13	•	•	14	Ground
GPIO 22	15	•	•	16	GPIO 23
3v3 Power	17	•	•	18	GPIO 24
GPIO 10 (SPI0 MOSI)	19	•	•	20	Ground
GPIO 9 (SPI0 MISO)	21	•	•	22	GPIO 25
GPIO 11 (SPI0 SCLK)	23	•	•	24	GPIO 8 (SPI0 CE0)
Ground	25	•	•	26	GPIO 7 (SPI0 CE1)
GPIO 0 (EEPROM SDA)	27	•	•	28	GPIO 1 (EEPROM SCL)
GPIO 5	29	•	•	30	Ground
GPIO 6	31	•	•	32	GPIO 12 (PWM0)
GPIO 13 (PWM1)	33	•	•	34	Ground
GPIO 19 (SPI1 MISO)	35	•	•	36	GPIO 16 (SPI1 CE2)
GPIO 26	37	•	•	38	GPIO 20 (SPI1 MOSI)
Ground	39	•	•	40	GPIO 21 (SPI1 SCLK)

Izvor: Raspberry Pi Pinout. SPI. <https://pinout.xyz/pinout/spi> (22.6.2024.)

Kako bi se omogućila komunikacija putem SPI protokola potrebno je omogućiti SPI sučelje, tj. napraviti konfiguraciju na Raspberry Pi-ju. To je moguće napraviti pomoću naredbe u terminalu 'raspi-config', a zatim se odabere 'Interface Options' i omogući SPI. Također je potrebno instalirati potrebne biblioteke za Python koje omogućuju rad s SPI i RFID modulima, za to su potrebne sljedeće naredbe (PiMyLifeUp (n. d.). How to set up a Raspebrry Pi RFID RC522 Chip):

```
a. apt-get install python3-pip
```

```
b. pip3 install spidev
```

```
c. pip3 install mfrc522
```

RC522 RFID modul je 13.56 MHz RFID čitač/pisač (eng. *reader/writer*) baziran na MFRC522 integriranom čipu koji je razvio NXP Semiconductors. Unutarnji predajnik<sup>11</sup> MFRC522 čipa omogućava bežičnu komunikaciju s RFID karticama i primopredajnicima. Predajnik generira elektromagnetsko polje koje napaja RFID oznaku kada uđe u područje djelovanja. Oznaka koristi ovu energiju za slanje svog jedinstvenog identifikacijskog broja natrag čitaču. Ovaj proces omogućava učinkovitu komunikaciju između RFID čitača i kartice bez potrebe za dodatnim napajanjem ili aktivnim komponentama na strani kartice. (NXP Semiconductors, n. d.).

Slika 3.3 prikazuje MFRC-522 RFID modul koji radi na frekvenciji od 13.56 MHz. Modul omogućuje očitavanje podataka s RFID oznaka, što je ključno za praćenje knjiga i drugih objekata u knjižnici.

Slika 3.3. Prikaz MFRC-522 RFID modula radne frekvencije 13.56MHz

---

<sup>11</sup> Unutarnji predajnik (eng. "internal transmitter") odnosi se na komponentu unutar uređaja, kao što je RFID čitač, koja generira elektromagnetsko polje potrebnu za bežičnu komunikaciju s RFID oznakama.



Izvor: <https://soldered.com/product/mfrc-522-rfid-13-56mhz-reader/>

Dokazano je da je SPI protokol brz i efikasan za komunikaciju između mikrokontrolera i RFID modula. Samim time, ovaj protokol omogućava pouzdanu komunikaciju koja je ključna za učinkovito praćenje knjiga u knjižnici.

## **4. EKSPERIMENTALNA DEMONSTRACIJA FUNKCIONALNOSTI PREDLOŽENOG RJEŠENJA**

Ovo poglavlje prikazuje praktičnu primjenu predloženog rješenja korištenjem različitih metoda i tehnologija. Cilj je demonstrirati kako RFID tehnologija i IoT mogu poboljšati upravljanje knjižničnim fondom. Detaljno će se opisati korištene metode, razvoj konceptualnog modela, te implementacija sustava.

### **4.1 Metode i metodologija istraživanja**

U ovom radu korištene su sljedeće metode:

- pregled literature i analiza postojećih istraživanja o RFID tehnologiji i njezinoj primjeni u knjižnicama.
- Razvoj konceptualnog modela pametne knjižnice koristeći hardverske komponente kao što su kartonski modeli, Raspberry Pi mikroračunala i RFID moduli.
- izrada *web*-aplikacije za knjižničare koja demonstrira praktičnu upotrebu RFID sustava s naglaskom na prikaz izvještaja pomoću grafova.
- izrada baze podataka za uspoređivanje identifikacijskih oznaka i dohvaćanje informacija
- analiza strukture interneta stvari na konstruiranom primjeru, uključujući detalje o povezivanju i komunikaciji između komponenti.

Metodologija rada uključuje kvalitativnu analizu dostupnih podataka, razvoj i testiranje prototipa te evaluaciju učinkovitosti predloženih rješenja. Ovim dijelom dokazano je da su korištene metode (pregled literature, razvoj konceptualnog modela, izrada *web*-aplikacije i analiza IoT strukture) prikladne za demonstraciju funkcionalnosti RFID sustava u pametnoj knjižnici. Pristupi su omogućili sveobuhvatno testiranje i evaluaciju predloženih rješenja.

### **4.2 Razvoj konceptualnog modela**

Razvoj konceptualnog modela pametne knjižnice uključuje izradu kartonske police, kartonskog stola i kartonske kutije, pri čemu su svi ovi elementi opremljeni Raspberry Pi mikroračunalima i RFID modulima.

Kartonska polica izrađena je kako bi se simulirao klasični sustav organizacije knjiga u knjižnici. Na svakoj polici nalazi se 2 Raspberry Pi mikroracunala povezana s RFID modulima RC522. RFID moduli omogućuju identifikaciju knjiga pomoću RFID oznaka pričvršćenih na knjige. Kada se knjiga stavi na policu ili ukloni s nje, RFID modul komunicira s Raspberry Pi mikroracunalom kako bi zabilježio promjenu i ažurirao status knjige u sustavu.

Prijenosno računalo predstavlja računalo knjižničara, povezan je s mikroracunalima preko SSH protokola i prikazuje sustav pametne knjižnice, omogućujući knjižničarima prikaz informacija o lokaciji knjiga, tablice za upravljanje statusom o dostupnosti knjiga i podatkovne izvještaje pomoću grafova.

Kartonski stol također sadrži RFID modul i Raspberry Pi mikroracunalo za demonstraciju dodatnih funkcionalnosti kao što su brza identifikacija i provjera statusa knjiga. Ukoliko se neka knjiga želi posuditi, može se staviti na stol i pojaviti će se tablica pod nazivom "Trenutni procesi" gdje se status knjige može promijeniti jednim klikom na gumb "posudi".

Kartonska kutija dizajnirana je za pohranu i transport knjiga unutar knjižnice. Kao i kod police i stola, i u ovoj kutiji je integrirano Raspberry Pi mikroracunalo s RFID modulom. RFID tehnologija omogućava praćenje knjiga dok su u kutiji, pružajući informacije o trenutnoj lokaciji i statusu knjiga.

Dokazano je da jednostavni modeli knjižničnih elemenata opremljeni RFID tehnologijom mogu efikasno simulirati stvarno okruženje pametne knjižnice. Ovaj pristup omogućava praćenje i upravljanje knjigama u realnom vremenu, što potvrđuje potencijal primjene RFID tehnologije u knjižnicama. Ova jednostavna, ali učinkovita konstrukcija omogućuje praktičnu demonstraciju primjene RFID tehnologije u knjižnicama, naglašavajući mogućnosti za poboljšanje upravljanja knjižničnim fondom i korisničkim iskustvom.

Slika 4.1 prikazuje konceptualni model pametne knjižnice izrađen od kartonskih komponenti, uključujući police, stol i kutiju. Svaka komponenta ima integrirani Raspberry Pi mikroračunalo i RFID modul, što omogućuje praćenje i upravljanje knjižničnim fondom u realnom vremenu.

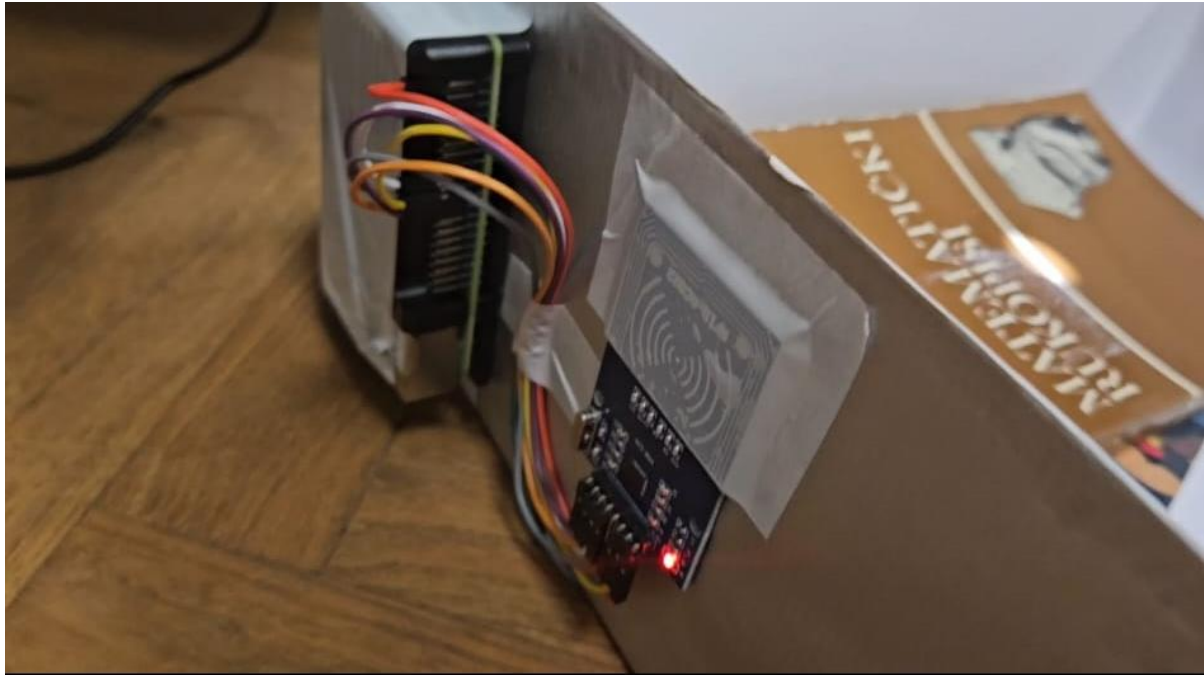
Slika 4.1. Prikaz konceptualnog modela "pametne" knjižnice



Izvor: autorski rad

Slika 4.2 prikazuje detaljan način spajanja Raspberry Pi mikroračunala i RFID modula. Prikazan je RFID modul RC522 povezan sa spojnim žicama na Raspebrry Pi mikroračunalo koji zajedno omogućuju pravilno funkcioniranje sustava.

Slika 4.2. Prikaz načina spajanja hardverskih komponenti



Izvor: autorski rad



Slika 4.3 prikazuje RFID ljepljivu oznaku koja se koristi za označavanje knjiga u knjižnici.

Slika 4.3 Prikaz RFID ljepljive oznake za knjige



Izvor: Autorski rad

### 4.3 Komunikacija između komponenti

U ovom projektu, četiri Raspberry Pi mikroračunala spojena su na istu lokalnu mrežu. Svako mikroračunalo ima instaliran SSH server, što omogućava daljinsko upravljanje sa središnjega prijenosnog računala. Povezivanje se ostvaruje pomoću SSH protokola, što omogućava sigurnu komunikaciju i upravljanje skriptama na svim mikroračunalima.

Na mikroračunalima se pokreću Python skripte<sup>12</sup> koje omogućuju skeniranje RFID oznaka, a na prijenosnom računalu se nalazi *web*-aplikacija koja pruža korisničko

---

<sup>12</sup> Skripta u Pythonu može se definirati kao datoteka koja se sastoji od Python koda ili programa. Završava nastavkom .py

sučelje za upravljanje knjižničnim resursima. Podatci o knjigama skeniranim na različitim lokacijama automatski se šalju u bazu podataka, gdje se mogu pratiti i analizirati preko *web*-aplikacije.

Kako bi se omogućila komunikacija između baze podataka i Python skripti, potrebno je kreirati `.env` datoteku koja sadrži URL na TURSO<sup>13</sup> bazu podataka i token za autentifikaciju. Svaka skripta se razlikuje po tome što unutar `.env` datoteke varijabla `LOCATION_ID` ima različitu vrijednost, koja označava specifičnu lokaciju u stvarnom vremenu. Na primjer, mikroracunalo broj 1 će imati `LOCATION_ID` postavljen na 3, što u bazi označava lokaciju "kutija". Za postavljanje vlastite baze podataka na ovoj platformi može se pratiti službena dokumentacija na sljedećoj poveznici: <https://docs.turso.tech/quickstart>

Primjer `.env` datoteke za mikroracunalo broj 1:

```
TURSO_URL=https://example.turso.com
TURSO_TOKEN=dobiveni_token
LOCATION_ID=3
```

Python skripta korištena u ovom projektu:

```
from typing import Union
import RPi.GPIO as GPIO
from mfrc522 import SimpleMFRC522
import libsql_experimental as libsql
import spidev
from dotenv import load_dotenv
import os

load_dotenv()

LOCATION_ID = os.getenv("LOCATION_ID")
```

---

<sup>13</sup> Turso je platforma za baze podataka koja se može koristiti za stvaranje stotina tisuća baza podataka po organizaciji. Podržava replikaciju na bilo koju lokaciju, uključujući vlastite servere, omogućujući pristup s mikrosekundnom latencijom.

```

LOCATION_ID = int(LOCATION_ID) if LOCATION_ID else None
DB_NAME = os.getenv("TURSO_DATABASE_URL", "library.db")
DB_AUTH_TOKEN = os.getenv("TURSO_AUTH_TOKEN", None)
print(f"DB_NAME={DB_NAME} | DB_AUTH_TOKEN={DB_AUTH_TOKEN} |
LOCATION_ID={LOCATION_ID}")

def main():
    spi = spidev.SpiDev()
    spi.open(1, 0) # Parametri (bus, device)

    reader = SimpleMFRC522()
    last_rfid_id = None
    try:
        reader.READER.Write_MFRC522(0x26, 127)
        while True:
            rfid_id, _ = reader.read()

            update_book(rfid_id, LOCATION_ID)

    except KeyboardInterrupt:
        os.exit(1)

    finally:
        GPIO.cleanup()

def db_exec(*args):
    global DB_NAME
    global DB_AUTH_TOKEN
    conn = libsql.connect(database=DB_NAME,
auth_token=DB_AUTH_TOKEN)
    res = conn.execute(*args)
    conn.commit()
    return res

```

```

def update_book(rfid_id: int, location_id: Union[int, None]):
    print(f"Updating rfid_id={rfid_id} to
location={location_id}")

    book_status = None
    book_id = None
    if location_id is None:
        book_status = "pending"
        obj = db_exec(
            "UPDATE books set location_id=NULL,
book_status='pending' where rfid_id=? RETURNING id",
            (rfid_id,),
        ).fetchone()
        book_id = obj[0]
    else:
        book_status = "available"
        obj = db_exec(
            "UPDATE books set location_id=?,
book_status='available' where rfid_id=? RETURNING id",
            (location_id, rfid_id),
        ).fetchone()
        book_id = obj[0]
    db_exec(
        "INSERT INTO book_log (book_id, book_status) values
(?, ?)",
        (book_id, book_status),
    )

if __name__ == "__main__":
    main()

```

## 4.4 Izrada baze podataka

Baza podataka sastoji se od nekoliko ključnih tablica koje pohranjuju podatke o knjigama, njihovim lokacijama i povijesti zaduživanja. Ovdje je opis strukture baze podataka i funkcionalnosti svake tablice:

1. Tablica „books“: pohranjuje osnovne informacije o svakoj knjizi u knjižnici. Polja u ovoj tablici uključuju:

- id: Jedinstveni identifikator knjige (INTEGER, PRIMARY KEY, AUTOINCREMENT).
- author: Autor knjige (TEXT).
- title: Naslov knjige (TEXT).
- year: Godina izdanja (INTEGER).
- udk: Univerzalna decimalna klasifikacija (INTEGER, može biti NULL).
- rfid\_id: Jedinstveni identifikator RFID oznake (INTEGER, može biti NULL).
- location\_id: Identifikator trenutne lokacije knjige (INTEGER, može biti NULL).
- book\_status: Status knjige (TEXT), moguće vrijednosti uključuju 'available', 'pending', 'lent'.
- created\_at: Vrijeme kada je knjiga dodana u sustav (TIMESTAMP).

2. Tablica „locations“: pohranjuje informacije o različitim lokacijama unutar knjižnice gdje se knjige mogu nalaziti. Polja u ovoj tablici uključuju:

- id: Jedinstveni identifikator lokacije (INTEGER, PRIMARY KEY, AUTOINCREMENT).
- name: Naziv lokacije (TEXT).
- created\_at: Vrijeme kada je lokacija dodana u sustav (TIMESTAMP).

3. Tablica „book\_log“: pohranjuje povijest zaduživanja i vraćanja knjiga. Polja u ovoj tablici uključuju:

- id: Jedinstveni identifikator zapisa (INTEGER, PRIMARY KEY, AUTOINCREMENT).
- book\_id: Identifikator knjige (INTEGER, FOREIGN KEY prema `books` tablici).
- book\_status: Status knjige u trenutku zapisa (TEXT), zadana vrijednost je 'available'.
- comment: Dodatni komentari (TEXT).
- created\_at: Vrijeme kada je zapis kreiran (TIMESTAMP, zadana vrijednost je trenutni datum i vrijeme).

Primjer koda za kreiranje tablice „book\_log“:

```
CREATE TABLE book_log (
    id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
    book_id INTEGER NOT NULL,
    book_status TEXT DEFAULT 'available',
    comment TEXT,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (book_id) REFERENCES books (id)
);
```

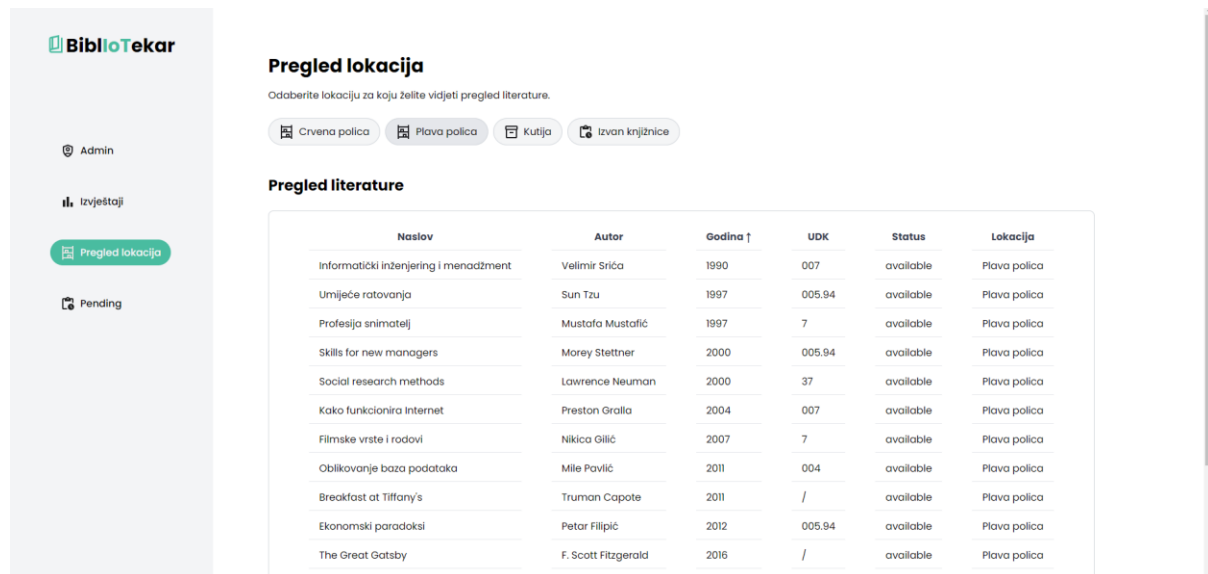
Dokazano je da primjena *cloud* baze umjesto lokalne baze podataka omogućuje knjižničarima pristup podacima u stvarnom vremenu s bilo koje lokacije, što pokazuje da knjižničari mogu imati pristup ažuriranim podacima i upravljati knjižničnim fondom s bilo koje lokacije, čime se poboljšava efikasnost i fleksibilnost rada. Time se zaključno dodatno naglašava potencijal IoT-a za modernizaciju knjižničnih usluga.

## 4.5 Izrada *web*-aplikacije

Primarni cilj kod izrade *web*-aplikacije bio je omogućiti knjižničarima odabir određene lokacije unutar knjižnice (npr. Crvena polica, Plava polica, Kutija) kako bi vidjeli pregled literature na toj lokaciji ili pregled literature koja trenutno nije u knjižnici. Osim

trenutne lokacije aplikacija pruža informacije o naslovu knjige, autoru, godini izdanja, UDK klasifikaciji, statusu knjige (dostupna ili posuđena). Tirkizna boja odabrana je kao primarna boja aplikacije zbog svoje asocijacije na pročišćenost i smirenost te izaziva pozitivne osjećaje.

Slika 4.4. Izgled web-aplikacije BiblioTekar



Izvor: Autorski rad

Slika 4.5. Prikaz tablice s knjigama sa statusom „posuđena“

## Pregled lokacija

Odaberite lokaciju za koju želite vidjeti pregled literature.

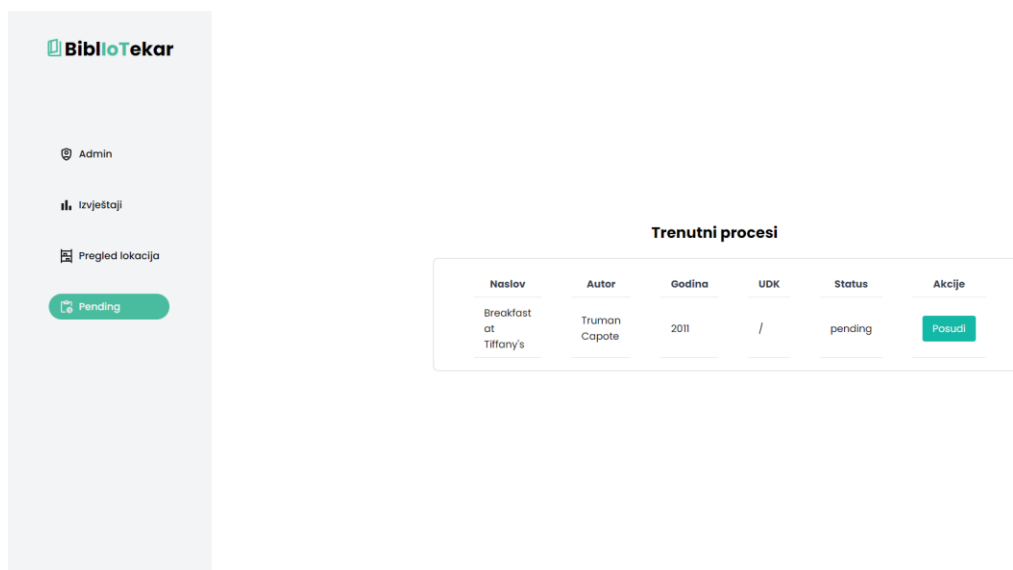


## Pregled literature

Naslov	Autor	Godina ↑	UDK	Status	Lokacija
Body language	Jane Lyle	1993	005.57	lent	Nepoznato
Corporate communication	Paul A. Argenti	1994	005.57	lent	Nepoznato
Vodstvo : teorija i praksa	Peter G. Northouse	2010	005	lent	Nepoznato
Organizacija	Pere Sikavica	2011	005.7	lent	Nepoznato
Poslovno odlučivanje	Pere Sikavica	2014	005	lent	Nepoznato
Journalism Next	Mark Briggs	2016	070	lent	Nepoznato

Izvor: Autorski rad

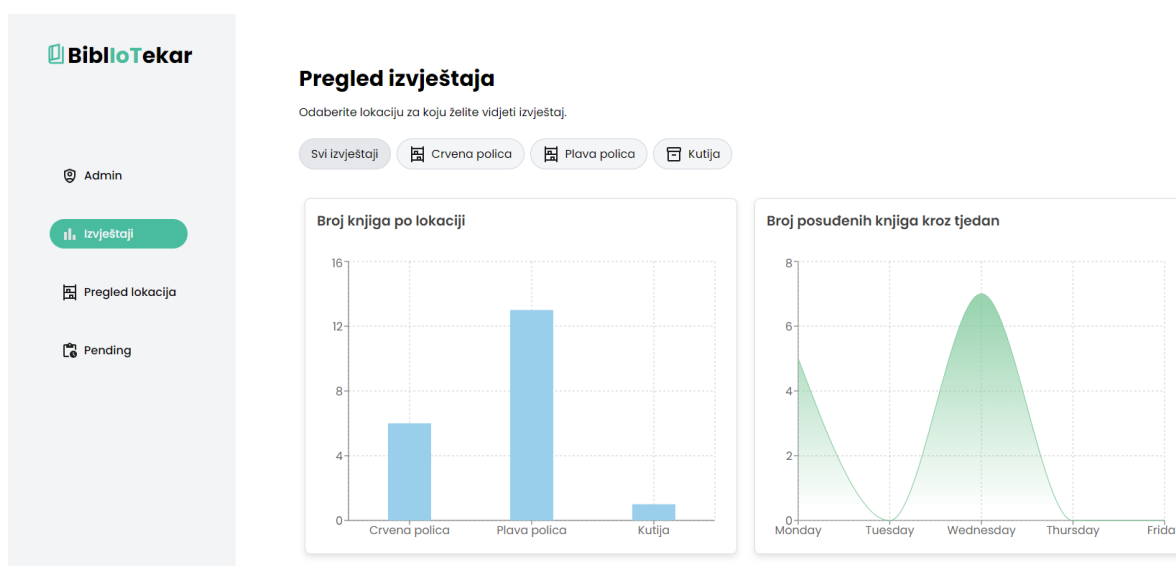
Slika 4.6. Prikaz tablice trenutnih procesa



Izvor: Autorski rad

Kako bi se još detaljnije i efikasnije pokušao riješiti problem unaprjeđenja rada knjižničara omogućilo se generiranje različitih izvještaja pomoću grafova.

Slika 4.7. Prikaz podatkovnih izvještaja na *web*-aplikaciji BibloTekar





## Pregled izvještaja

Odaberite lokaciju za koju želite vidjeti izvještaj.

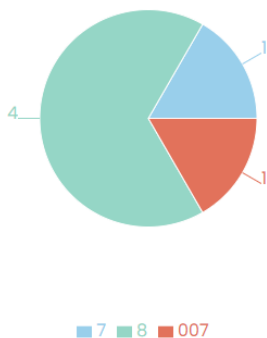
Svi izvještaji

Crvena polica

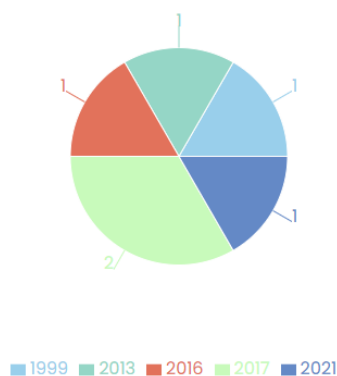
Plava polica

Kutija

Broj knjiga po UDK-u



Broj knjiga po godini



Izvor: autorski rad

Odabirom na tipku “Svi izvještaji” prikazuju se grafovi s obzirom na sve lokacije i sve knjige u knjižničnoj bazi, primjerice koliko je određenih knjiga na kojoj lokaciji. Kada se na *web*-aplikaciji odabere određena lokacija prikazuju se dva grafikona: prvi analizira broj knjiga s obzirom na UDK<sup>14</sup>, a drugi s obzirom na godinu izdanja. Knjižničar tada može vidjeti je li možda slučajno na tu lokaciju stavio knjigu koja tamo ne pripada, primjerice ako je cijela polica ispunjena knjigama sa klasifikacijskom oznakom 007 i samo jedna knjiga sa oznakom 8.

Podatkovni izvještaji mogu prikazivati različite informacije, neki od njih su:

- učestalost posuđivanja određenih knjiga: omogućuju knjižničarima uvid u koje su knjige najpopularnije među korisnicima što im može pomoći u donošenju odluka o nabavi dodatnih primjeraka popularnih naslova
- praćenje starih knjiga: grafovi mogu prikazivati knjige koje nisu posuđivane duže vrijeme, što može pomoći u odluci o dezinvestiranju ili zamjeni tih knjiga

<sup>14</sup> Univerzalna decimalna klasifikacija (UDK) kao indeksni jezik predstavlja alat za dobivanje informacija o sadržaju publikacija neovisno o prirodnom jeziku ili mediju na kojem su objavljene.

- analiza vremena posudbi: prikaz prosječnog trajanja posudbi i učestalosti vraćanja knjiga može pomoći u optimizaciji posudbenih pravila i procedura
- prediktivna analitika: korištenje povijesnih podataka za predviđanje budućih trendova posudbi može pomoći knjižničarima u dugoročnom planiranju i strategiji nabave.

Za lakši pristup i pregled koda *web*-aplikacije, u radu je uključen QR kod koji vodi na GitHub repozitorij projekta. Ovaj repozitorij sadrži sve datoteke i upute potrebne za postavljanje i pokretanje aplikacije i python skripte.

Slika 4.8. Prikaz QR koda za pristup repozitoriju Projekta

(<https://github.com/magdalenabegic/BibloTekar>)



Različite tehnologije korištene u izradi aplikacije:

- Next.js: popularni React (JavaScript biblioteka za izradu korisničkih sučelja) razvojni okvir<sup>15</sup> koji omogućava jednostavno kreiranje *server-side rendered*<sup>16</sup> i statičkih *web*-aplikacija. Omogućava automatsko generiranje ruta, optimizaciju performansi te integraciju s različitim alatima i servisima (nextjs.org (n. d.)).

---

<sup>15</sup> Razvojni okvir (eng. framework) je strukturirani skup alata i knjižnica koji pružaju temeljne funkcionalnosti i smjernice za razvoj softvera. Omogućava programerima bržu i efikasniju izradu aplikacija pružanjem unaprijed definiranih modula i funkcija za uobičajene zadatke, poput upravljanja podacima, rukovanja događajima i interakcije s korisnikom.

<sup>16</sup> Server-side rendering (SSR) je tehnika u kojoj se HTML sadržaj generira na serveru umjesto u pregledniku korisnika. To omogućava brže učitavanje stranica i bolje performanse, posebno za korisnike sa sporijim internetskim vezama ili uređajima.

- Drizzle: biblioteka koja olakšava rad s bazama podataka (Drizzle (n. d.)). Omogućava jednostavno definiranje shema, migracija i manipulaciju podacima kroz čisti i intuitivni API (Application Programming Interface). API je skup definiranih pravila i protokola koji omogućuju različitim softverskim aplikacijama da međusobno komuniciraju. API-ji omogućuju programerima pristup funkcionalnostima i podacima druge aplikacije ili usluge, bez potrebe za znanjem o internim detaljima implementacije. Na primjer, API može omogućiti aplikaciji pristup podacima iz baze podataka, interakciju s vanjskim servisima ili korištenje funkcionalnosti operativnog sustava (Goodwin, 2024).
- Tailwind CSS: CSS<sup>17</sup> razvojni okvir koji omogućava brzo i efikasno kreiranje prilagođenih dizajna. Umjesto pisanih prilagođenih stilova, koristi klasične stilove direktno u HTML-u kako bi omogućio fleksibilno i responzivno oblikovanje (tailwindcss.com (n. d.)).
- tRPC (*TypeScript Remote Procedure Call*): alat koji omogućava jednostavnu komunikaciju između frontend i backend dijelova aplikacije. Kroz koncept "routera", omogućava definiranje i upravljanje API pozivima na siguran i tipiziran način. Glavna prednost tRPC-a je mogućnost dijeljenja tipova između klijentske i serverske strane aplikacije, što omogućava statičko tipiziranje i automatsko dovršavanje koda, čime se smanjuje mogućnost grešaka (trpc.io (n. d.)).

#### Rutiranje (usmjeravanje):

- mapa "pages" je mjesto gdje su smještene sve rute aplikacije.
- datoteka nazvana primjerice "moja-stranica.tsx" bit će prikazana na ruti "/my-page"
- kad god je stavljen bilo koji dio naziva datoteke u uglate zagrade ([ i ]), bit će tretiran kao dinamički segment rute i može biti zamijenjen bilo kojom vrijednošću u URL-u.

---

<sup>17</sup> CSS (Cascading Style Sheets) je stilski jezik koji se koristi za opisivanje prezentacije dokumenta napisanog u jeziku za označavanje, poput HTML-a. CSS omogućava razdvajanje sadržaja od dizajna, pružajući programerima mogućnost da definiraju stilove za elemente na web stranici, uključujući boje, fontove, raspored i druge vizualne aspekte.

## Komunikacija s *backendom*:

- komunikacija sa usmjernikom je apstrahirana korištenjem tRPC-a. tRPC ima koncept "*routera*" koji se može koristiti za usmjeravanje zahtjeva prema *backendu*
- mapa "routers" je mjesto gdje je smještena sva *backend* logika.
- primjer jedne datoteke smještene u "routers" mapi:

```
import { z } from "zod";
import { createTRPCRouter, publicProcedure } from "../trpc";
import { db } from "../db";
import { bookLog, books } from "../db/schema";
import { eq, sql } from "drizzle-orm";
import { BookStatus } from "~/utils/constants/book";

export const booksRouter = createTRPCRouter({
  getAll: publicProcedure.query(() => {
    return db.query.books.findMany({
      with: {
        location: true, // Fetch the related location
      },
    });
  }),

  getByLocation: publicProcedure
    .input(z.object({ locationId: z.number().nullable() }))
    .query(({ input }) => {
      return db.query.books.findMany({
        where: (fields, op) =>
          input.locationId
            ? op.eq(fields.locationId, input.locationId)
            : op.and(op.isNull(fields.locationId),
              op.eq(fields.bookStatus, BookStatus.Lent)),
      });
    });
});
```

```

    with: {
      location: true, // Fetch the related location
    },
  });
}),

getPending: publicProcedure.query(() => {
  return db.query.books.findMany({
    where: (fields, op) => op.eq(fields.bookStatus,
BookStatus.Pending),
  });
}),

getByTimestamp: publicProcedure.query(() => {
  return db.query.books.findMany({
    // where: (fields, op) => op.gt(fields.createdAt, new
Date(Date.now() - 24 * 60 * 60 * 1000)),
  });
}),

getOne: publicProcedure
  .input(
    z.object({
      id: z.number(),
    }),
  )
  .query(({ input }) => {
    return db.query.books.findFirst({
      where: (fields, op) => op.eq(fields.id, input.id),
    });
  }),

updateStatus: publicProcedure

```

```

.input(
  z.object({
    id: z.number(),
    status: z.nativeEnum(BookStatus),
  }),
)
.mutation(async ({ input }) => {
  const { id, status } = input;

  return db.transaction(async (tx) => {
    const book = await db
      .select({
        id: books.id,
      })
      .from(books)
      .where(eq(books.id, id))
      .limit(1)
      .get();

    if (!book) {
      return {
        success: false as const,
        message: "Book not found",
      };
    }

    await tx.insert(bookLog).values({
      bookId: book.id,
      bookStatus: status,
    });

    const res = await tx

```

```

        .update(books)
        .set({
            bookStatus: status,
        })
        .where(eq(books.id, id))
        .returning();

const newBook = res.at(0);

if (!newBook) {
    return {
        success: false as const,
        message: "Book not found",
    };
}

return {
    success: true as const,
    data: book,
};
});
}),

```

```

countLentBooks: publicProcedure.query(async () => {
    const [result] = await db
        .select({
            count: sql`count(*)`.as('count'),
        })
        .from(books)
        .where(eq(books.bookStatus, BookStatus.Lent));

    return result ? result.count : 0;
}

```

```
}),  
});
```

Slika 4.9. Prikaz QR koda za pristup videu konceptualnog modela pametne knjižnice (<https://www.youtube.com/watch?app=desktop&v=7BhDWkwuYFo>)



Dokazano je da moderna *web*-aplikacija, ukoliko je izrađena na pravi način, može značajno unaprijediti rad knjižničara omogućujući im generiranje izvještaja i analizu podataka uz ostale svakodnevne zadatke. Zaključno, ovim radom naglašava se potencijal IoT tehnologija i *cloud* infrastrukture za transformaciju knjižnica u dinamične i interaktivne prostore.



## ZAKLJUČAK

Ovim radom dokazano je da primjena Interneta stvari (IoT) u knjižnicama donosi brojne prednosti koje značajno unaprjeđuju način upravljanja knjižničnim fondom, poboljšavaju korisničko iskustvo i olakšavaju rad knjižničara. Korištenjem radiofrekvencijske identifikacije, zajedno s analizom podataka, moguće je postići veću učinkovitost, detaljniji uvid u organizaciju knjiga i bolje upravljanje knjižničnim fondom.

Izrađena *web*-aplikacija i sustav prikazani u ovom radu pokušavaju riješiti neke probleme rada knjižnice iz perspektive studentice koja je radila u knjižnici. Primjena ovako razvijenog sustava omogućava istovremeno precizno praćenje dostupnosti literature i lokacije knjiga na policama, dok analiza podataka putem grafova pruža knjižničarima vrijedne uvide u obrasce korištenja knjiga. Ovaj sustav omogućava knjižničarima da bolje planiraju nabavu knjiga, optimiziraju raspodjelu resursa i unaprijede cjelokupno korisničko iskustvo.

Osim toga, pokazano je kako se uz pomoć tehnologija poput Raspberry Pi mikroračunala i SSH protokola može ostvariti učinkovita komunikacija između različitih komponenti sustava. Na taj način, knjižničari mogu daljinski upravljati sustavom i pratiti stanje knjiga u realnom vremenu, što dodatno doprinosi efikasnosti i fleksibilnosti rada knjižnice.

Ovaj rad također naglašava važnost modernizacije knjižnica u digitalnom dobu te potrebu za kontinuiranim unaprjeđenjem knjižničnih usluga. Preporučuje se daljnje istraživanje i razvoj novih tehnologija koje mogu doprinijeti još većoj efikasnosti i kvaliteti knjižničnih usluga, kao i širenje primjene IoT tehnologija u knjižnicama širom svijeta.

U konačnici, implementacija pametnih rješenja poput ovih može značajno unaprijediti rad knjižničara, omogućujući im da se fokusiraju na ostale važne segmente knjižničnog poslovanja. Ovaj rad može poslužiti kao temelj za buduća istraživanja i razvoj sličnih sustava koji će dodatno unaprijediti knjižničarske poslove i korisničko iskustvo.

## LITERATURA

### KNJIGE

1. Iqbal, M. A., Hussain, S., Xing, H., Imran M. A. (2021.) *Enabling the Internet of Things: Fundamentals, Design and Applications*. Wiley-IEEE Press.

### ZNANSTVENI I STRUČNI ČLANCI

1. Hsuanwei, M.C. (2017). Information Visualization Meets Libraries. *Library Technology Reports* 53(3), str. 17-20. Preuzeto s:  
<https://www.journals.ala.org/index.php/ltr/article/view/6290/8210>

### INTERNETSKI IZVORI I OSTALO:

1. Raspberry Pi. What Is A Raspberry Pi?  
<https://opensource.com/resources/raspberry-pi>
2. Cert.hr. (2007). CERT-PUBDOC-2007-01-179.  
<https://www.cis.hr/www.edicija/LinkedDocuments/CCERT-PUBDOC-2007-01-179.pdf>
3. Drizzle (n. d.) <https://orm.drizzle.team/>
4. Goodwin, M.. IBM, 2024. What is an API?. <https://www.ibm.com/topics/api>
5. Guru99. (n. d.). Kako pokrenuti Python skripte. <https://www.guru99.com/hr/run-python-scripts.html>
6. Hrvatska enciklopedija. (n. d.) Univerzalna decimalna klasifikacija. U: Hrvatska enciklopedija <https://www.enciklopedija.hr/clanak/univerzalna-decimalna-klasifikacija>
7. NSK. (n. d.). UDK katalogizacija. <https://katalogizacija.nsk.hr/udk/>
8. opensource.com. (n. d.). Raspberry Pi.  
<https://opensource.com/resources/raspberry-pi>
9. Raspberry Pi Pinout. SPI. <https://pinout.xyz/pinout/spi> (22.6.2024.)

10. Raspberry Pi. Raspberry Pi 4. <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/>
11. nextjs.org (n. d.) <https://nextjs.org/>
12. NXP Semiconductors. (2016). MFRC522: Standard performance MIFARE and NTAG frontend. <https://www.nxp.com/docs/en/data-sheet/MFRC522.pdf>
13. PiMyLifeUp (n. d.). How to set up a Raspebrry Pi RFID RC522 Chip. Preuzeto s: <https://pimylifeup.com/raspberry-pi-rfid-rc522/>
14. Slavonski.hr (2021). Gradska knjižnica Požega: Želimo da ljudi osvijeste koncept nove knjižnice u novoj zgradi. <https://slavonski.hr/gradska-knjiznica-pozega-ravnateljica-aleksandra-sutalo-zelimo-da-ljudi-osvijeste-koncept-nove-knjiznice-u-novoj-zgradi/>
15. Soldered Electronics. (n. d.). MFRC-522 RFID 13.56MHz Reader. <https://soldered.com/product/mfrc-522-rfid-13-56mhz-reader/>
16. Tailwindcss.com (n. d.) <https://tailwindcss.com/>
17. trpc.io (n. d.) <https://trpc.io/>
18. Turopoli, D. 2015. *SSH protokol* [završni rad]. Sveučilište J.J. Strossmayera u Osijeku
19. WirelessLinks, 2021. RFID vs NFC: What's The Difference? <https://wlius.com/blog/rfid-vs-nfc-whats-the-difference/>
20. Biomedicinska instrumentacija. SPI (Serial Peripheral Interface) komunikacioni protokol. <http://kelm.ftn.uns.ac.rs/literatura/bmi/2022-23/SPI.pdf>

## **PRILOG: Kazalo kartica**

API - *Application Programming Interface*

CSS - *Cascading Style Sheets*

GPIO - *General Purpose Input/Output*

IoT – Internet stvari (*Internet of Things*)

NFC - *Near Field Communication*

RFID – radiofrekvencijska identifikacija

SPI – Serijsko periferno sučelje (*Serial Peripheral Interface*)

SSH – *Secure Shell*

tRPC - *TypeScript Remote Procedure Call*

UDK – Univerzalna decimalna klasifikacija

## POPIS TABLICA I SLIKA

Slika 2.1. Prikaz RFID sustava.....	2
Slika 2.2.: Prikaz rada RFID tehnologije.....	3
Slika 3.1. Prikaz Raspberry Pi 4 mikroračunala.....	8
Slika 3.2. Prikaz Raspberry Pi pinova za povezivanje pomoću SPI protokola.....	10
Slika 3.3. Prikaz MFRC-522 RFID modula od 13.56MHz.....	11
Slika 4.1. Prikaz konceptualnog modela “pametne” knjižnice.....	14
Slika 4.2. Prikaz načina spajanja hardverskih komponenti.....	14
Slika 4.3 Prikaz RFID ljepljive oznake za knjige.....	15
Slika 4.4. Izgled <i>web</i> -aplikacije BiblloTekar.....	21
Slika 4.5. Prikaz tablice s knjigama sa statusom „posuđena“.....	21
Slika 4.6. Prikaz tablice trenutnih procesa.....	22
Slika 4.7. Prikaz podatkovnih izvještaja na <i>web</i> -aplikaciji BiblloTekar.....	22
Slika 4.8. Prikaz QR koda za pristup repozitoriju projekta.....	24
Slika 4.9. Prikaz QR koda za pristup videu konceptualnog modela pametne knjižnice.....	30