

Elektronski sistem saobraćajne policije

Petar Komordžić

Fakultet tehničkih nauka

Univerzitet u Novom Sadu

komordzic.sr37.2020@uns.ac.rs

Sažetak: Elektronski sistem saobraćajne policije predstavlja sistem koja rešava brojne izazove u kontroli saobraćaja i upravljanju kaznama. Ovaj projekat ima za cilj poboljšanje efikasnosti procesa provere dokumenata, izdavanja kazni i upravljanja njihovim statusom. Sistem omogućava bržu i jednostavniju proveru dokumenata, automatizovano izdavanje kazni, transparentno praćenje statusa u sudu, prijavu krađe vozila i olakšan pristup informacijama o kaznama za građane.

Ključne reči: web aplikacija; saobraćajna policija; saobraćajna kazna; krađa vozila;

1. Uvod

Urbani saobraćajni sistemi suočavaju se sa brojnim izazovima kao što su povećanje broja vozača, sve češće nepridržavanje saobraćajnih propisa i nedostatak efikasnih mehanizama za otkrivanje i kažnjavanje prekršaja. Ovi problemi dovode do komplikovanog i dugotrajnog procesa kažnjavanja, gužvi u policijskim stanicama, nedostatak jednostavnog i transparentnog načina za proveru ličnih prekršaja od strane građana i sl. Upravo ovim elektronskim sistemom želimo da rešimo ove probleme uvođenjem brzog i intuitivnog procesa za proveru ličnih dokumenata i rukovanja prekršajima od strane službenika, kao i laku i brzu proveru svojih prekršaja i prijave krađe vozila od strane građanina.

U ovom radu opisano je i implementirano je pisanje prekršajnih naloga kao i direktno prosleđivanje tih naloga sudu sa ciljem da se proces pisanja kazni ubrza i olakša. Takođe je omogućena brza provera dokumenata čime omogućavamo da na licu mesta validiramo i proverimo lične dokumente. Građani mogu brzo i jednostavno proveriti svoje prekršajne naloge kao i prijaviti krađu vozila čime se smanjuju gužve i čekanja u samim policijskim stanicama. Na kraju, predloženi su načini za dalje unapređivanje sistema i potencijalni problemi sa trenutnom implementacijom.

2. Srodna istraživanja i pregled korišćenih tehnologija

U ovom poglavlju dat je primer aplikacije namenjene digitalizaciji i automatizaciji službe policije u Danskoj i navedene su tehnologije kojim je su korišćene za implementiranje sistema sličnih funkcionalnosti.

2.1. *Danska policija*

Danska policija [1] svojim građanima nudi web sajt pomoću koga građani mogu koristiti razne usluge koje policija obavlja online direktno sa svog uređaja. Sistem nudi mogućnost prijave zločina, pribavljanja dokumenta i podataka o prekršajima, dobavljanje podataka o kaznama, rukovanje dozvolama za korišćenje oružja, pregled policijskih stanica, mogućnost kontaktiranja hitne službe itd... Sistem je dosta obiman te se može reći da olakšava i ubrzava veliku većinu potreba građana u policiji. Mane ovog

sistema, prema mom mišljenju, bi bile te što se za većinu funkcionalnosti mora navigirati kroz veliki broj prozora pa to može biti neintuitivno za korisnike koji se ne snalaze najbolje sa tehnologijom.

2.2. Korišćene tehnologije

Ovaj podjeljak pojašnjava tehnologije koje su korišćene za rešavanje problema sistema elektronske policije. Za izradu serverskog dela projekta korišćen je programski jezik Go [2]. Za čuvanje podataka korišćena je MySQL [3] baza podataka. Klijentski deo projekta implementiran je korišćenjem je React [4] biblioteke. Komunikacija između različitih servisa i komponenti implementirana je korišćenjem REST [5] standarda za komunikaciju.

3. Specifikacija zahteva

U ovom poglavlju su objašnjeni funkcionalni i nefunkcionalni zahtevi sistema opisanog u ovom radu.

3.1. Funkcionalni zahtevi

U ovom odeljku su opisani funkcionalni zahtevi koji elektronski sistem policije mora podržavati i možemo ih videti na slici 1.



Slika 1. Use-case dijagram elektronskog sistema saobraćajne policije

Naziv	Prijava na sistem
Učesnici	Korisnik
Preduslovi	1. Korisnik postoji u bazi matičara
Rezultat	1. Korisnik je prijavljen na sistem
Koraci	1. Korisnik unosi kredencijale 2. Korisnik potvrđuje unos
Izuzeci	1. Pogrešni kredencijali (o čemu dobijamo poruku)

Tabela 1- Opis slučaja korišćenja "Prijava na sistem"

Naziv	Provera dokumenta
Učesnici	Policajac
Preduslovi	1. Korisnik je prijavljen na sistem kao policajac
Rezultat	1. Policajac vidi sve podatke o izabranom vozilu koje mu je MUP prosledio 2. Policajac vidi sve podatke o izabranoj osobi i njegovim vozilima koje mu je MUP prosledio
Koraci	1. Korisnik bira opciju provere dokumenata 3. Korisnik potvrđuje unos podataka
Izuzeci	1. Entitet ne postoji u bazi MUP-a

Tabela 2 – Opis slučaja korišćenja "Provera dokumenta"

Naziv	Provera osobe u sudu
Učesnici	Policajac
Preduslovi	1.Korisnik je prijavljen kao policajac
Rezultat	1.Korisnik vidi da li postoje aktivni slučajevi protiv izabranog lica koji su u toku.
Koraci	1. Korisnik bira opciju provere osobe u sudu 2. Korisnik unosi JMBG osobe 3. Korisnik potvrđuje svoj unos
Izuzeci	

Tabela 3 – Opis slučaja korišćenja "Provera osobe u sudu"

Naziv	Izdavanje prekršajnih naloga
Učesnici	Policajac
Preduslovi	1. Korisnik je prijavljen kao policajac
Rezultat	1. Prekršajni nalog se čuva u sistemu
Koraci	1. Korisnik bira opciju pisanja prekršajnog naloga 2. Korisnik unosi potrebne informacije o nalogu 3. Korisnik potvrđuje svoj unos
Izuzeci	1. Neki od polja su nevalidno unešeni (o čemu dobijamo poruku)

Tabela 4 – Opis slučaja korišćenja "Izdavanje prekršajnih naloga"

Naziv	Generisanje pdf fajla
Učesnici	Korisnik
Preduslovi	<ol style="list-style-type: none"> 1. Korisnik je prijavljen na sistem 2. Postoji bar jedan prekršajni nalog u sistemu
Rezultat	<ol style="list-style-type: none"> 1. Pdf fajl se čuva na našem sistemu i dobijamo url na kome ga možemo podeliti/preuzeti.
Koraci	<ol style="list-style-type: none"> 1. Korisnik bira opciju generisanja fajla
Izuzeci	<ol style="list-style-type: none"> 1. Greška prilikom generisanja PDF fajla 2. Greška priliko čuvanja fajla na sistem

Tabela 5 – Opis slučaja korišćenja "Generisanje pdf fajla"

Naziv	Prosleđivanje naloga sudu
Učesnici	Policajac
Preduslovi	<ol style="list-style-type: none"> 1. Korisnik je ulogovan kao policajac 2. Postoji bar jedan prekršajni nalog u sistemu
Rezultat	<ol style="list-style-type: none"> 1. Nalog sa attach-ovanim linkovima do priloženih fajlova(i obavezni pdf prekršajnog naloga) se šalje sudu
Koraci	<ol style="list-style-type: none"> 1. Korisnik bira nalog 2. Korisnik bira opciju da ga prosledi sudu 3. Korisnik unosi naslov i opis naloga 4. Korisnik prilaže željene fajlove 5. Korisnik potvrđuje unos
Izuzeci	<ol style="list-style-type: none"> 1. Neki od polja su nevalidno unešeni (o čemu dobijamo poruku)

Tabela 6 – Opis slučaja korišćenja "Prosleđivanje naloga sudu"

Naziv	Prijava krađe vozila
Učesnici	Korisnik
Preduslovi	1. Korisnik je prijavljen na sistem
Rezultat	1. Prijava krađe vozila se generiše i prosleđuje MUP-u
Koraci	1. Korisnik bira opciju prijave krađe vozila 2. Korisnik unosi potrebne podatke o krađi 3. Korisnik potvrđuje svoj unos
Izuzeci	1. Neka od polja su nevalidno popunjena (o čemu dobijamo poruku)

Tabela 7 – Opis slučaja korišćenja "Prijava krađe vozila"

Naziv	Pregled izdatih prekršajnih naloga
Učesnici	Policajac
Preduslovi	1. Policajac je prijavljen na sistem
Rezultat	1. Korisnik vidi listu prekršajnih 1naloga koje je on pisao
Koraci	1. Policajac bira opciju pregleda prekršajnih naloga 2. Policajac potvrđuje svoj izbor
Izuzeci	1. Ne postoje nalozi za prikaz(dobijamo poruku o tome)

Tabela 8 – Opis slučaja korišćenja "Pregled izdatih prekršajnih naloga"

Naziv	Pregled svojih prekršajnih naloga
Učesnici	Korisnik
Preduslovi	1. Korisnik je prijavljen na sistem
Rezultat	1. Korisnik vidi listu prekršajnih naloga koji su pisani za njega
Koraci	1. Korisnik bira opciju pregleda prekršajnih naloga 2. Korisnik potvrđuje svoj izbor
Izuzeci	1. Ne postoje nalozi za prikaz(dobijamo poruku o tome)

Tabela 9 – Opis slučaja korišćenja "Pregled svojih prekršajnih naloga"

Naziv	Provera statusa prosleđenih naloga
Učesnici	Policajac
Preduslovi	1. Korisnik je ulogovan kao policajac 2. Postoji bar jedan nalog prosleđen sudu
Rezultat	1. Policajac može da vidi sve naloga koje je poslao u sud kao i njihov trenutni status
Koraci	1. Korisnik bira opciju pregleda statusa poslatih naloga
Izuzeci	1. Ne postoje nalozi za prikaz(dobijamo poruku o tome)

Tabela 10 – Opis slučaja korišćenja "Provera statusa prosleđenih naloga"

Naziv	Pregled policijskih stanica
Učesnici	Korisnik
Preduslovi	1. Korisnik je ulogovan u sistem
Rezultat	1. Korisnik može da vidi sve postojeće policijske stanice sa njihovim osnovnim podacima
Koraci	1. Korisnik bira opciju za pregled policijskih stanica
Izuzeci	1. Ne postoji nijedna stanica (o čemu dobijamo poruku)

Tabela 11 – Opis slučaja korišćenja "Pregled policijskih stanica"

Naziv	Prilaganje slika
Učesnici	Policajac
Preduslovi	1. Policajac je ulogovan u sistem 2. Policajac je u procesu pisanja prekršajnog naloga
Rezultat	1. Slike su sačuvane u sistemu i njihovi linkovi su sačuvani u prekršajnom nalogu
Koraci	1. Korisnik bira opciju dodavanje slike 2. Korisnik bira slike koju želi da priloži 3. Korisnik potvrđuje unos
Izuzeci	1. Greška prilikom čuvanja slike u sistemu(o čemu dobijamo poruku)

Tabela 12 – Opis slučaja korišćenja "Prilaganje slika"

3.2. Nefunkcionalni zahtevi

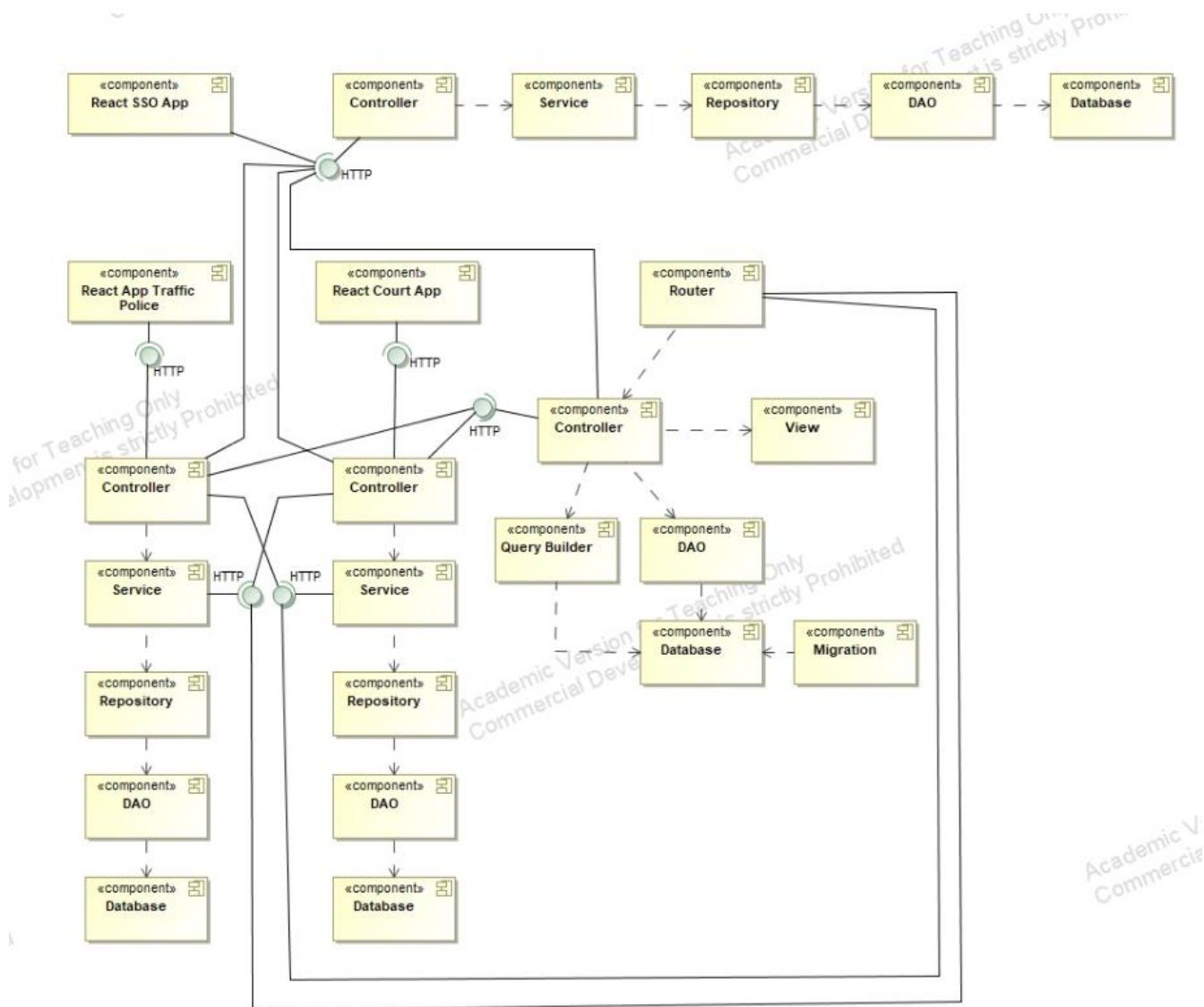
U ovom odeljku su opisani nefunkcionalni zahtevi koje elektronski sistem saobraćajne policije treba da ispuni.

Sistem mora da bude kompatibilan sa sistemima MUP-a i Suda i komunicirati sa njima na standardizovan način. Sistem mora biti implementirati kontrolu pristupa i biti kompatibilan sa SSO

sistemom e-uprave kako bi se na lak i jednostavan način bezbedno pristupalo sistemu. Sistem mora biti intuitivan i lak za korišćenje sa ciljem da se korisniku ovog sistema olakša i ubrza izvršavanje svih funkcionalnih zahteva.

4. Specifikacija dizajna

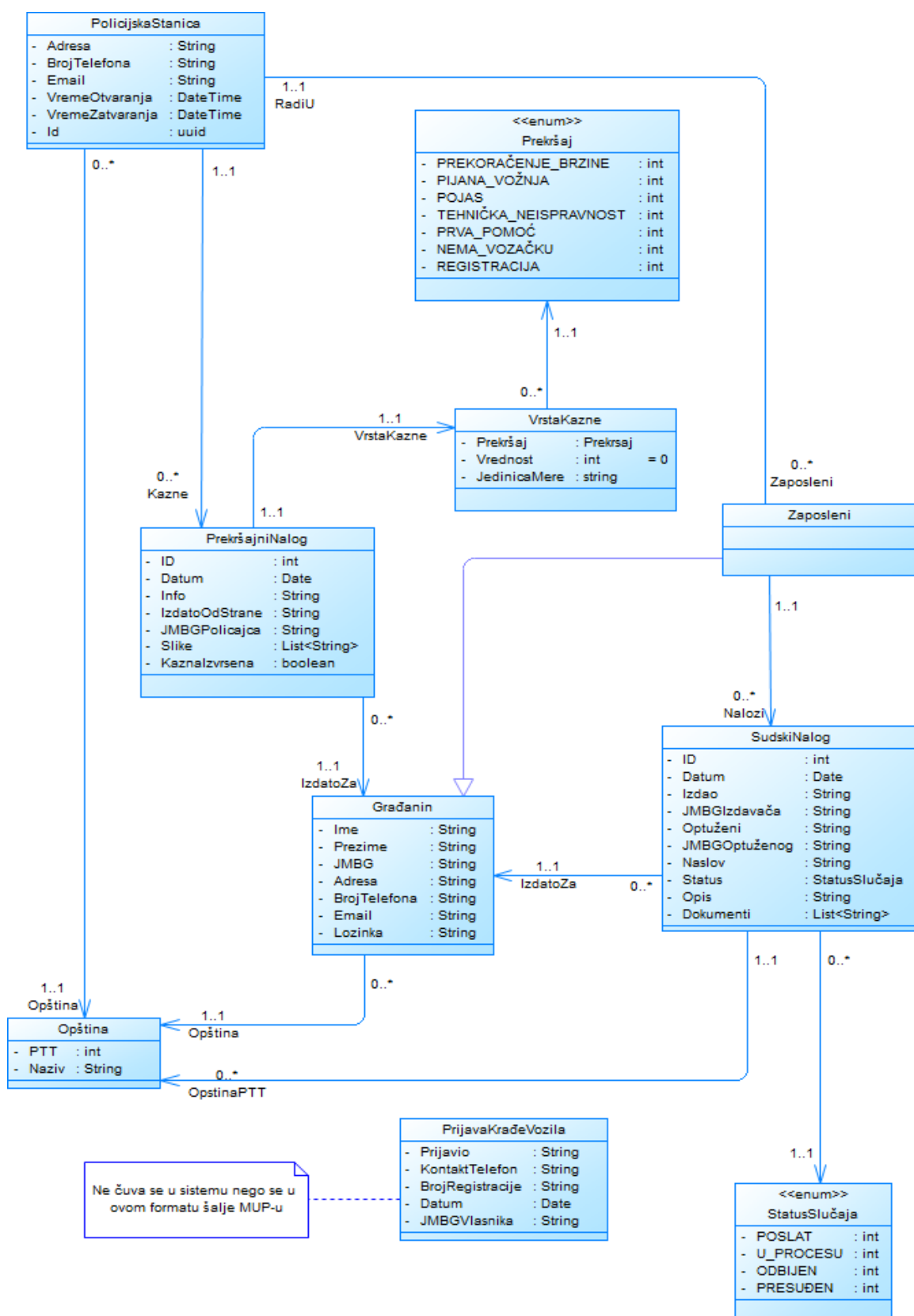
Ovo poglavlje objašnjava dizajn softverskog rešenja za elektronski sistem saobraćajne policije. Arhitektura sistema se može videti na slici 2.



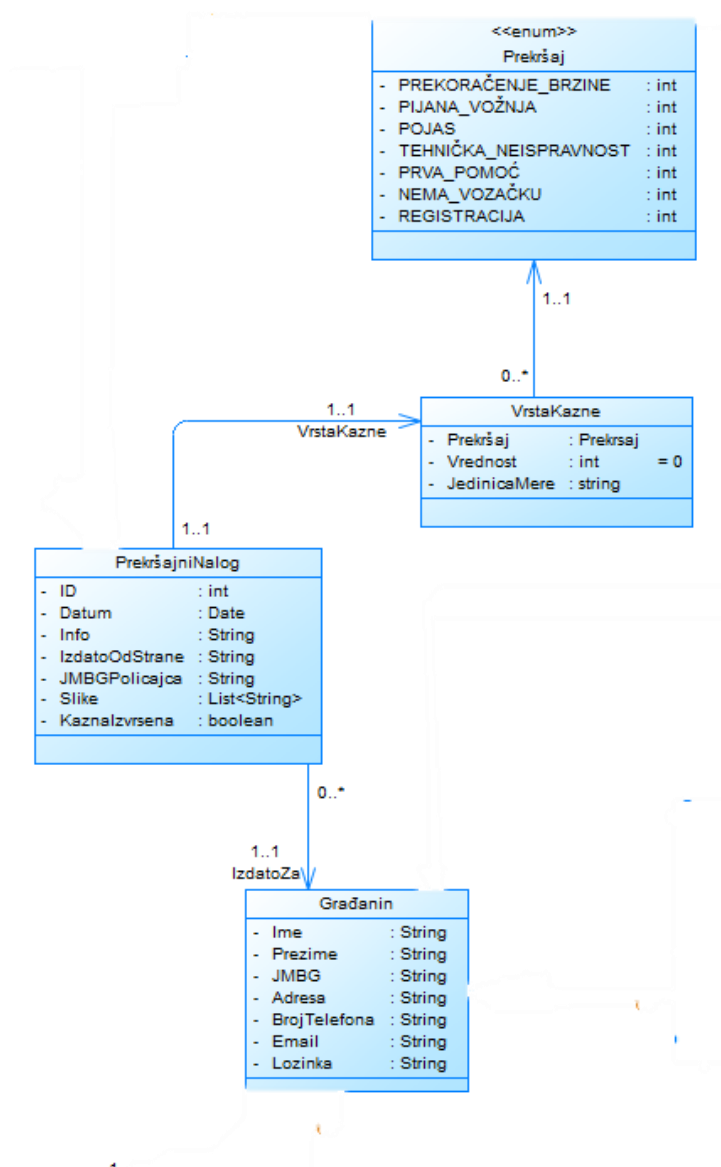
Slika 2 – Dijagram komponenti sistema e-uprave

Sistem komunicira sa servisima Suda i MUP-a radi razmene podataka i dobavljanja potrebnih informacija. Rukovanje sa korisničkom autorizacijom se vrši preko SSO servisa sa kojim takođe komunicira. Arhitektura sistema je standardna Web arhitektura koja se sastoji od kontrolera servisa i repozitorijuma koji komuniciraju sa odvojenim klijentskim delom koda (GUI).

Domenski model sistema je izmodelovan na klasnom dijagramu koji je predstavljen slikom 3.

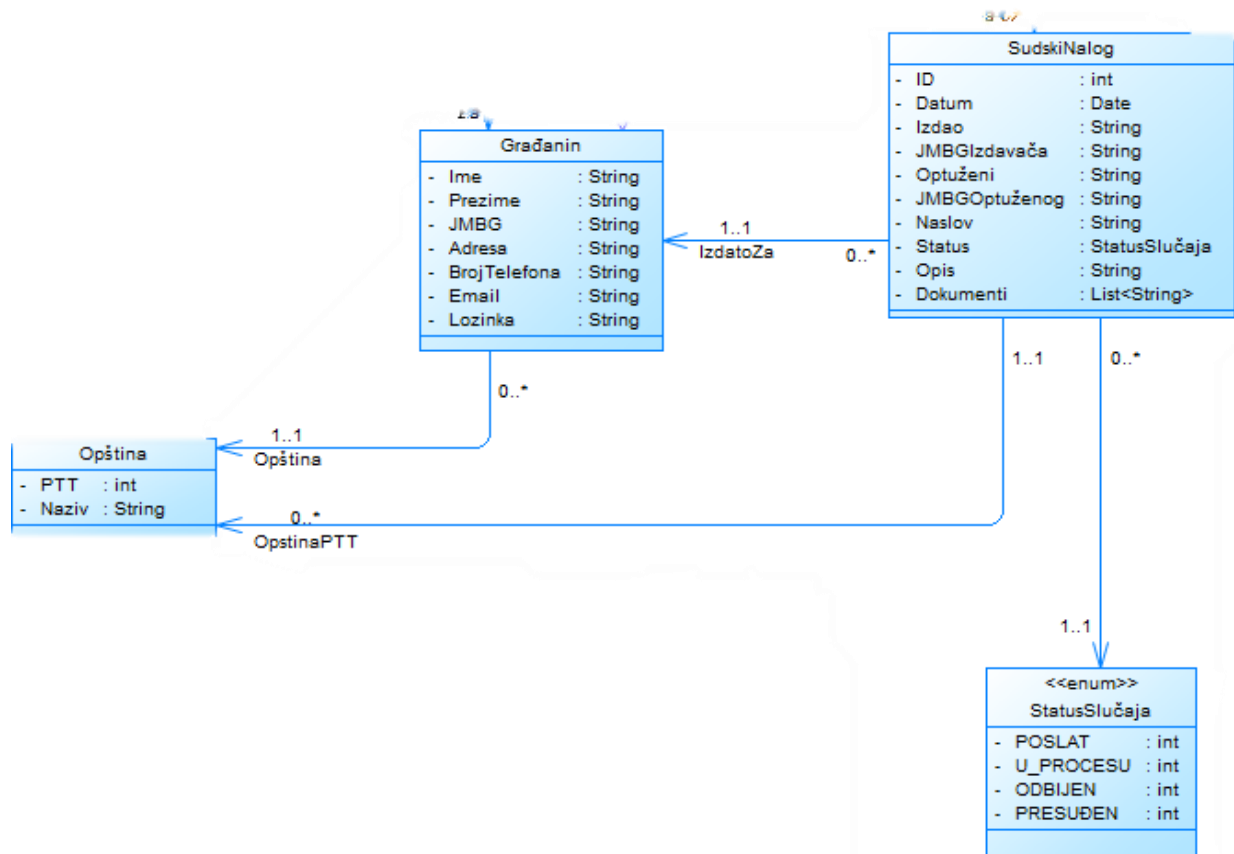


Slika 3 – Klasni dijagram domenskog modela



Slika 4 – Dijagram klasa

Klasa prekršajni nalog reprezentuje kaznu koju policijac može napisati I sadrži osnovne podatke o građaninu kojem se piše kazna kao I o službenom licu koje je piše. Klasa vrsta kazne nam omogućava da imamo više tipova prekršaja kao I vrednosti za one tipove za koje je to predviđeno(brzina, alkohol...) Klasa sadrži listu slika tj. listu URL stringova do putanje na kojoj su nam slike sačuvane što nam omogućava da priložimo neograničen broj slika kao dodatne dokaze pri pisanju naloga.



Slika 5 – Dijagram klasa

Klasa sudski nalog predstavlja nalog koji možemo poslati sudu. Generiše se na osnovu prekršajnog naloga iz kog se polja sa podacima građanina I sluđbenika automatski popune dok ostale detalje I dokaze možemo sami priložiti. Može sadržati listu URL linkova koja sadrži sve slike I pdf dokumente koje prilažemo kao dokaze. U sebi sadrži I enumeraciju koja predstavlja status slučaja u sudu I menja se kada se status promeni u sistemu suda.

5. Implementacija

Ovo poglavlje objašnjava kako su implementirane neke od funkcionalnosti elektronskog sistema saobraćajne policije

5.1 Čuvanje prekršajnih naloga

U ovom podeoku objašnjena je implementacija čuvanja prekršajnih naloga na serverskom delu aplikacije. Metodu koja čuva nalog prikazana je na listingu 1.

```
func (s saobracjanaServiceImpl) SaveNalog(noviNalog data.PrekršajniNalog) (*data.PrekršajniNalogDTO, error) {
    noviNalog.Datum = time.Now()
    if noviNalog.KaznaIzvršena {
        points := CalculatePointsForTicket(noviNalog)
        if points != 0 {
            err := s.mupService.SendPoints(noviNalog.JMBGZapisanog, points)
            if err != nil {
                log.Fatal(err)
                return nil, errors.New("There was problem while sending nalog points to MUP")
            }
        }
    }
    nalog, err := s.saobracjanaRepo.SaveNalog(noviNalog)
    //Ostatak koda...
```

Listing 1 – Čuvanje prekršajnog naloga.

Ovoj metodi se prosleđuje objekat koji predstavlja novi nalog sa podacima koje smo poslali sa klijentskog dela aplikacije. Nakon što smo postavili datum kada smo obradili nalog (trenutno vreme) proveravamo da li želimo izvršiti kaznu automatski. Ako je ta opcija izabrana to znači da će se kazna izvršiti a kazneni poeni oduzeti (ako za tim ima potrebe). Kazne koje su izvršene nemaju pravo da se naknadno pošalju sudu. Vidimo da se poeni računaju unutar funkcije CalculatePointsForTicket čiju implementaciju možemo videti na listingu 2. Ako su poeni veći od 0, što znači da treba da ih oduzmemo na vozačkoj dozvoli, šaljemo zahtev servisu MUP-a koji će obraditi taj zahtev i oduzeti prosleđen broj bodova. Nakon toga ako se nije desila nikakva greška čuvamo nalog u našem sistemu.

```

const dozvoljeniPromili = 0.3
const poeniZaRegularnePrekrasaje = 1
const poneiZaVecePrekrasaje = 1

func CalculatePointsForTicket(kazna data.PrekrasajniNalog) int {
    switch kazna.TipPrekrasaja {
    case "POJAS":
        return poeniZaRegularnePrekrasaje
    case "PREKORACENJE_BRZINE":
        points := (*kazna.Vrednost / 10) * 2
        if *kazna.Vrednost%10 > 0 {
            points += 1
        }
        return int(points)
    case "PIJANA_VOZNJA":
        if *kazna.Vrednost > dozvoljeniPromili {
            points := int(math.Floor((*kazna.Vrednost - dozvoljeniPromili) / 0.2))
            return points
        }
        return 0
    case "TEHNICKA_NEISPRAVNOST":
        return poneiZaVecePrekrasaje
    case "REGISTRACIJA":
        return poneiZaVecePrekrasaje
    }
    return 0
}

```

Listing 2 – Računanje kaznenih poena za prekršajni nalog

U ovom kodu vidimo switch blok sa različitim slučajevima koji se odnose na tip prekršaja. Ako su tipovi prekršaja neispravnost ili ne posedovanje registracije to se smatra većim prekršajem i vraćamo broj poena definisan konstantom za veće prekršaje. Nevezivanje pojasa se smatra manjom kaznom te tu vraćamo vrednost definisanu konstantom za te vrste prekršaja. Kod prekoračenja brzine prvo gledamo u koliko intervala od po 10km/h smo prekoračili brzinu i za svaki dodeljujemo 2 poena. Ako vrednost prekoračenja ima ostatak pri deljenju sa 10 dodajemo još jedan poen jer to znači da samo prekoračili i malo preko prethodno izračunatih desetina (na primer 24km/h). Kod pijane vožnje gledamo da li je vrednost promila veća od minimalno dozvoljene vrednosti definisane u konstanti iznad. Ako jeste onda ima smisla računati poene, a ako ne vraćamo 0 poena. Poeni za pijanu vožnju se računaju tako što računamo koji broj intervala od 0.2 promila (jer želimo da se na svaka 0.2 promila povećava broj poena) postoji i zaokružujemo taj broj da bi dobili broj poena koje želimo oduzeti.

5.2 Prosleđivanje naloga sudu

U ovom podeoku je objašnjen proces slanja sudskih naloga sistemu suda na serverskom delu aplikacije što se može videti na listingu 3.

```
func (s saobracjanaServiceImpl) SaveSudskiNalog(nalog data.SudskiNalog) (*data.SudskiNalog, error) {
    zaposleni, err := s.saobracjanaRepo.GetZaposleni(nalog.JMBGSluzbenika)
    if err != nil {
        log.Fatal(err.Error())
        return nil, errors.New("There is no policeman with that jmbg")
    }
    nalog.Datum = time.Now()
    nalog.OpstinaPTT = zaposleni.RadiU.Opstina.PTT

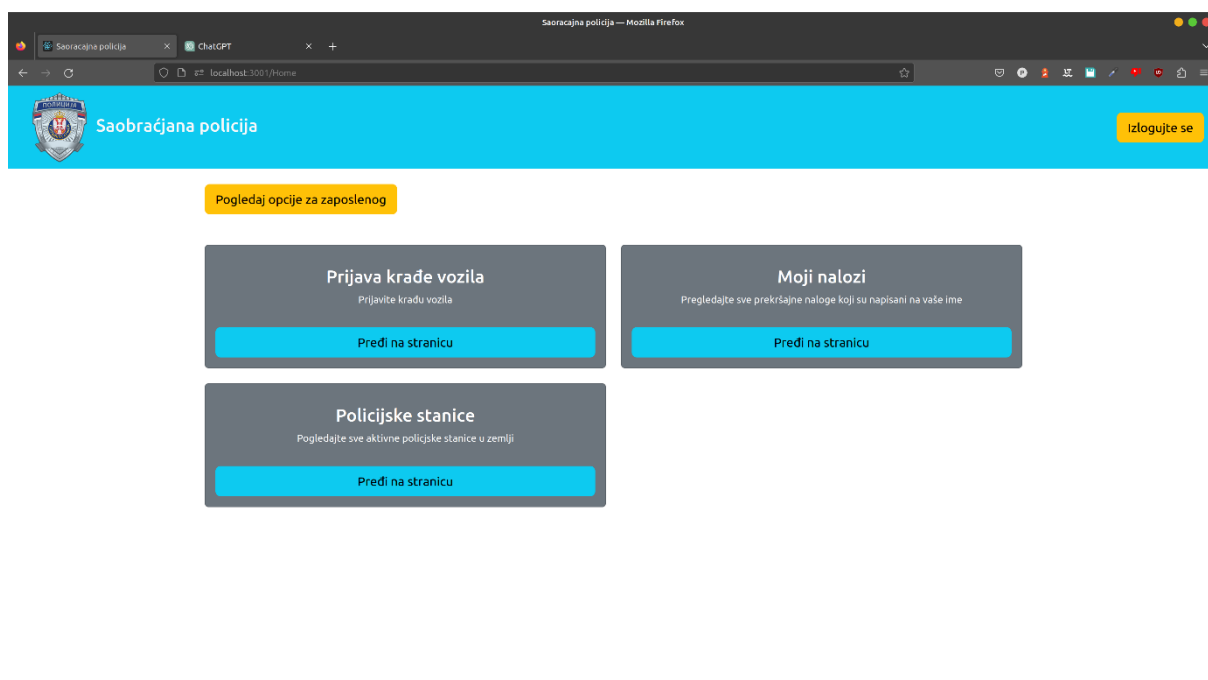
    nalog.StatusSlucaja = "U_PROCESU"
    savedNalog, err := s.saobracjanaRepo.SaveSudskiNalog(nalog)
    if err != nil {
        log.Fatal(err.Error())
        return nil, errors.New("There is problem with saving nalog to db")
    }
    nalog.Id = savedNalog.Id
    err = s.sudService.SendNalog(nalog)
    if err != nil {
        log.Fatal(err.Error())
        s.saobracjanaRepo.RemoveNalog(savedNalog.id)
        return nil, errors.New("There is problem with sending nalog to sudService")
    }
    return savedNalog, nil
}
```

Listing 3 – Slanje i čuvanje sudskog naloga

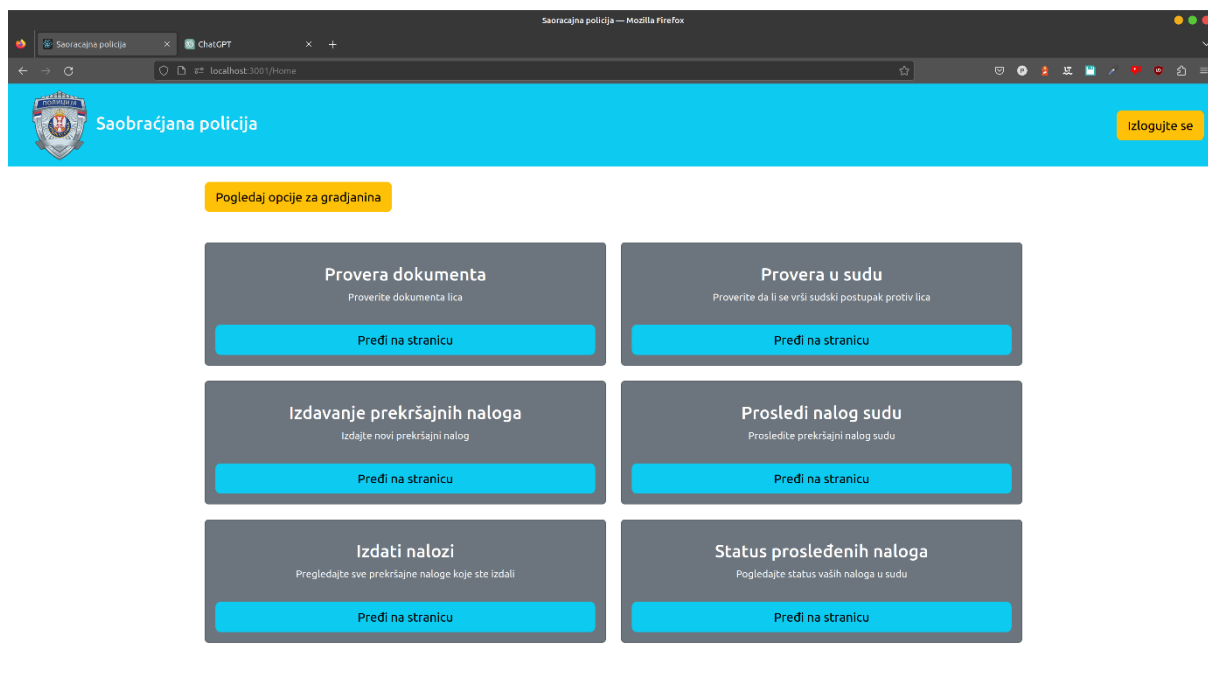
Ova metoda prima sudski nalog dobijen sa klijentskog dela aplikacije koji treba sačuvati i proslediti servisu suda. U prvom koraku dobivljamo zaposlenog na osnovu JMBG polja iz naloga i validiramo da on zaista postoji u sistemu. Nakon toga na osnovu službenika i opštine u kojoj radi postavljamo opštinu samog naloga da se poklapa sa njom. Nakon toga čuvamo taj nalog u našem sistemu kako bi dobili automatski generisan ID koji takođe treba proslediti sudu. U sledećem koraku šaljemo taj nalog sudskom servisu i čekamo odgovor o tome da li je zahtev uspešno obrađen. U slučaju greške nalog se briše iz sistema i korisniku se vraća greška.

6. Demonstracija

Ovo poglavlje prikazuje način korišćenja elektronskog sistema saobraćajne policije. Prilikom prijavljivanja na sistem korisniku se prikazuje početni ekran kao što je prikazano na slikama 6 i 7.



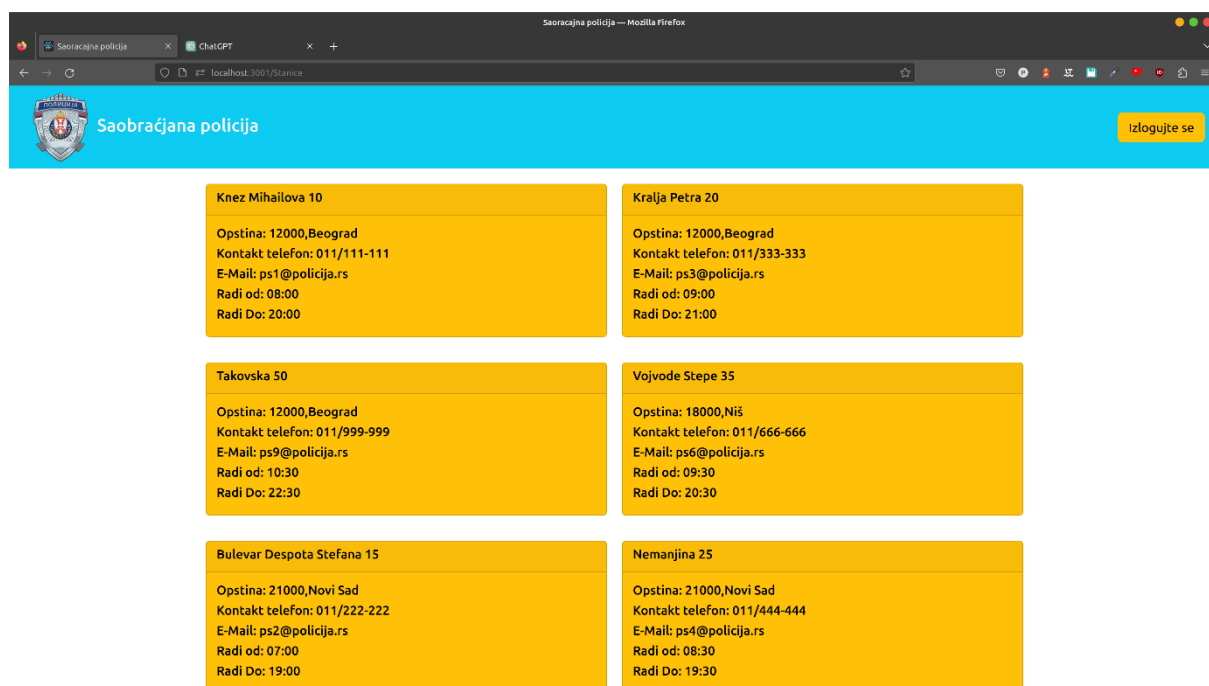
Slika 6 – Izgled početnog ekrana za građane



Slika 7 – Izgled početnog ekrana za zaposlene

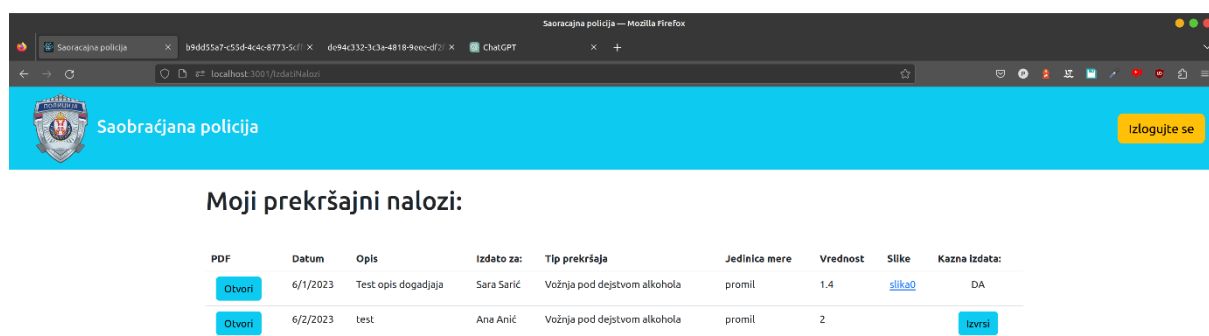
U gornjem delu ekrana se nalazi navigaciona traka na kojoj imamo opciju odjave sa sistema. U centralnom delu prozora se nalaze kartice sa mogućim funkcionalnostima sistema. Klikom na neku od kartice korisnik se prikazuje stranica na kojoj se izvršava ta radnja.

Biranjem opcije pregleda policijskih stanica korisniku se prikazuju sve policijske stanice u našem sistemu kao što je prikazano na slici 8.



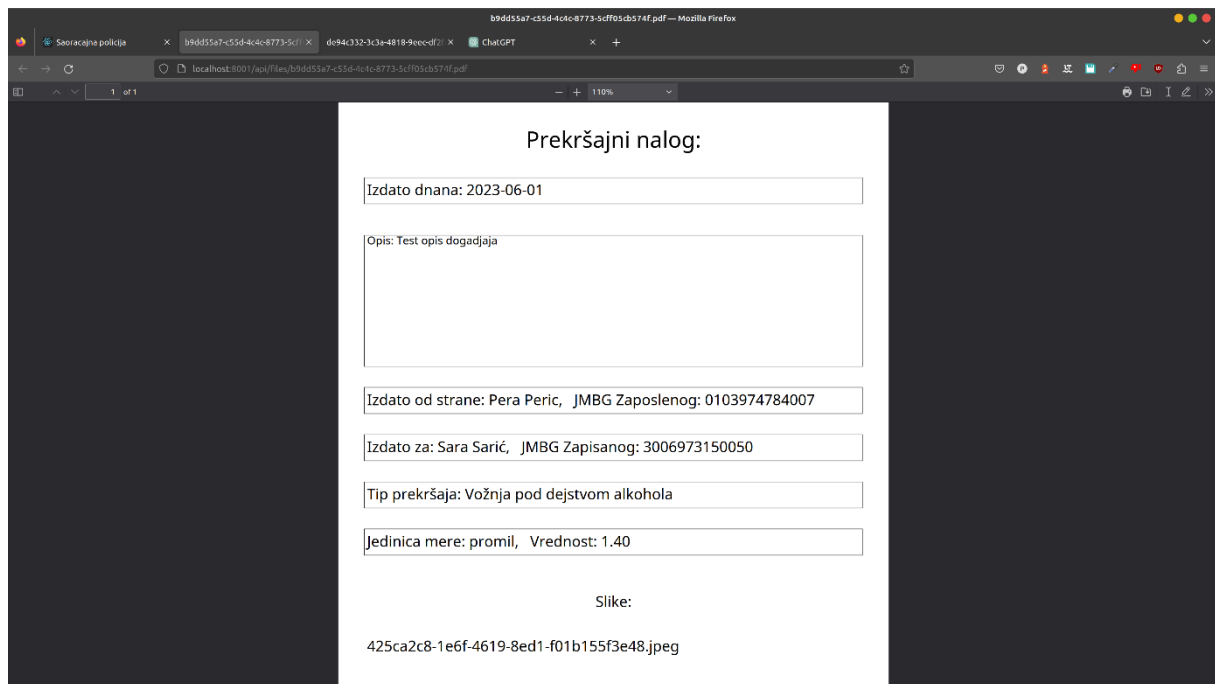
Slika 8 – Izgled stranice za pregled policijskih stanica

Biranjem opcije za pregled prekršaja (izdatih ili sopstvenih) korisniku se prikazuju svi prekršajni nalozi koji zadovoljavaju te kriterijume kao što je prikazano na slici 9.



Slika 9 – Izgled stranice za pregled prekršajnih naloga

Ako izaberemo opciju da otvorimo PDF korisniku će se prikazati PDF dokument prekršajnog naloga kao na slici 10.



Slika 10 – Izgled PDF fajla prekršajnog naloga

Biranjem opcije za proveru osobe u MUP-u policajac može videti sve informacije o ličnim dokumentima građanina kao što je prikazano na slikama 11 i 12.

Saobraćajna policija

izlogujte se

Provera osobe/vozila:

Unesite JMBG: 2004993175065 **Pretraži**

Vozačka Dozvola

Broj vozačke dozvole: F9FF5519

Kategorije vozila: B1

Datum izdavanja: 1/1/1

Datum isteka: 6/30/2033

Broj kaznenih poena: 0

Status vozačke dozvole: Aktivna

Unesite broj tablice: NS123NS **Pretraži**

Saobraćajna Dozvola

Marka: test

Model: test

Godina proizvodnje: 0

Slika 11 – Provera vozačke dozvole na stranici za proveru dokumenta

Unesite JMBG:

Unesite broj tablice:

Saobraćajna Dozvola

Marka: test

Model: test

Godina proizvodnje: 0

Boja: test

Registar: NS123NS

Snaga motora: 123 kW

Maksimalna brzina: 200 km/h

Broj sedišta: 2

Težina: 1 kg

Tip vozila: Putničko vozilo

Status registracije:

Prijavljena krađa: N/A

Kreiraj prekršajni nalog:

Unesite opis događaja:

Slika 12 – Provera saobraćajne dozvole na stranici za proveru dokumenta

Na ovoj stanici je moguće direktno kreirati prekršajni nalog, ali ta opcija je moguća i odlaskom na stranicu kreiranja novog naloga kao što je prikazano na slici 13.

Kreiraj prekršajni nalog:

Unesite opis događaja:

Unesite vaše ime:

Unesite JMBG lica kome pišete nalog:

Unesite ime lica kome pišete nalog:

Krivično delo:

Jedinica mere:

Vrednost:

Prinesite dostupne slike

No files selected.

☐ Izvrši kaznu automatski:

Slika 13 – Izgled stranice za kreiranje prekršajnog naloga

Na ovoj stranici policajac može uneti sve potrebne podatka kao i priložiti slike kao dokaze. Izborom opcije prijava nalog se čuva u našem sistemu. Kreirani nalozi za koje nismo izvršili kaznu se mogu proslediti sudu na stranici za prosleđivanje naloga sudu koja je prikazana na slici 14.

7. Zaključak

U ovom radu predstavljeno je softversko rešenje sistema saobraćajne policije. Predstavljani softver ima za cilj da olakša i ubrza repetitivne procese izdavanja i rukovanja sa kaznama, proveru dokumenata na licu mesta, ali i da olakša proveru osnovnih podataka od strane građana. Sistem takođe rasterećuje fizičke lokacije policije, MUP-a ili suda jer sada nema potrebe da ode uživo da bi se izvadilo ili predao neki dokument, saznala informacija i slično. U odnosu na srodne sisteme koji se koriste u svetu ovaj sistem je dosta manjeg obima s obzirom da je razvijan od strane jednog studenta u mnogo kraćem vremenskom roku. Imajući to u vidu ovaj softver i dalje sadrži dovoljno funkcionalnosti da bi se mogao koristiti za šta je namenjen i time zaista olakšao procese pokrivene njegovom implementacijom. Dalje opcije za unapređivanje ovog sistema bi mogle uključivati mogućnost plaćanja svojih kazni online korišćenjem nekih od platformi za elektronsko plaćanje kao što je Stripe [6]. Još jedna od opcija koje bi mogle biti od značaja se mogu preuzeti iz srodnih rešenja a to bi bila mogućnost kontaktiranja hitnih službi online u vidu nekakvog real time chat-a ili sistema notifikacija u samoj policijskoj stanici na klik dugmeta pozovi.

8. Reference

- [1] Danish police Politi, <https://politi.dk>
- [2] Go, programming language, <https://go.dev/>
- [3] MySQL, sql database, <https://www.mysql.com/>
- [4] React, javascript library, <https://react.dev/>
- [5] Lekosh G. 2022, What is Rest, <https://restfulapi.net/>
- [6] Stripe 2023, Payments infrastructure for the internet, <https://stripe.com/>