

# Programsko inženjerstvo

Ak. god. 2023./2024.

## DentAll

Dokumentacija, Rev. 1.0

Grupa: Potplaćeni

Voditelj: Luka Kokić

Datum predaje: 17.11.2023.

Nastavnik: *Goran Rajić*

# Sadržaj

<b>1 Dnevnik promjena dokumentacije</b>	<b>3</b>
<b>2 Opis projektnog zadatka</b>	<b>4</b>
<b>3 Specifikacija programske potpore</b>	<b>8</b>
3.1 Funkcionalni zahtjevi . . . . .	8
3.1.1 Obrasci uporabe . . . . .	10
3.1.2 Sekvencijski dijagrami . . . . .	27
3.2 Ostali zahtjevi . . . . .	31
<b>4 Arhitektura i dizajn sustava</b>	<b>32</b>
4.1 Baza podataka . . . . .	33
4.1.1 Opis tablica . . . . .	34
4.1.2 Dijagram baze podataka . . . . .	43
4.2 Dijagram razreda . . . . .	44
4.3 Dijagram stanja . . . . .	46
4.4 Dijagram aktivnosti . . . . .	47
4.5 Dijagram komponenti . . . . .	48
<b>5 Implementacija i korisničko sučelje</b>	<b>49</b>
5.1 Korištene tehnologije i alati . . . . .	49
5.2 Ispitivanje programskog rješenja . . . . .	50
5.2.1 Ispitivanje komponenti . . . . .	50
5.2.2 Ispitivanje sustava . . . . .	50
5.3 Dijagram razmještaja . . . . .	51
5.4 Upute za puštanje u pogon . . . . .	52
<b>6 Zaključak i budući rad</b>	<b>53</b>
<b>Popis literature</b>	<b>54</b>
<b>Indeks slika i dijagrama</b>	<b>55</b>

**Dodatak: Prikaz aktivnosti grupe**

**56**

# 1. Dnevnik promjena dokumentacije

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napravljen predložak	Luka Kokić	25.10.2023.
0.2	Opis projektnog zadatka + (partial) specifikacija programske potpore	Ian Marković	31.10.2023.
0.3	Dodani opis arhitekture baze podataka kao i relacijska shema	Karlo Baljak	31.10.2023.
0.4	Početna verzija obrazaca uporabe	Ian Marković, Teo Musa	9.11.2023.
0.5	Konačna verzija obrazaca uporabe i ostali zahtjevi	Ian Marković, Teo Musa	10.11.2023.
0.6	Početak dijagrama obrazaca uporabe	Mislav Matić	11.11.2023.
0.7	Dijagrami razreda	Karlo Baljak	11.11.2023.
0.8	Dovršeni dijagrami obrazaca uporabe	Mislav Matić, Luka Kokić	15.11.2023.
0.9	Početak sekvencijskih dijagrama	Bruno Milaković	15.11.2023.
0.10	Dovršeni sekvencijski dijagrami	Bruno Milaković, Luka Kokić	16.11.2023.
1.0	Konačna verzija za prvu predaju	Luka Kokić	17.11.2023.

## 2. Opis projektnog zadatka

Cilj ovog projekta je razviti web aplikaciju „DentAll“ kojom će pružatelji usluga zdravstvenog turizma moći evidentirati i koordinirati lokalni smještaj i prijevoz korisnika usluga. Time bi se zdravstveni turizam učinio privlačnijim rješavanjem brige i cijene smještaja i prijevoza korisnika pri njihovom obitavanju u mjestu gdje koriste spomenute usluge. Uz to bi klijenti pružatelja usluga bili u mogućnosti unaprijed vidjeti geografsku sliku smještaja i okolice te moguće turističke opcije i kretanja tijekom privremenog obitavanja .

Aplikacija bi olakšala brige vlasnika usluga zdravstvenog turizma evidentiranjem, organizacijom i praćenjem smještaja i prijevoza klijenata uz programsku podršku interneta. Time se nastoji nadomjestiti nedostatak ovisnosti pružatelja zdravstvenih usluga o raznim i višestrukim pružateljima smještajnih usluga. Takvim pristupom kroz javnu i lagano dostupnu uslugu smanjio bi se broj posrednika u organizaciji i administraciji pružanja usluga zdravstvenog turizma. Također bi klijenti pronalazili sve potrebne informacije oko odabranog zdravstvenog turizma, tj. uz same zdravstvene usluge i o smještaju te prijevozu za svaki termin, čineći sam odabir prisustvovanja u inozemnim zdravstvenim uslugama jednostavniji i privlačniji.

Trenutna postojeća rješenja uključuju korištenje hotela i motela, što čini cijeli proces zdravstvenog turizma skupljim i kompliciranijim. Dugoročno bi pružateljima bilo isplativije iznajmljivati vlastite nekretnine u sklopu pružanja cjelokupnih usluga. Također je u trenutnim rješenjima prijevoz ostavljen na odgovornost klijenata, što otežava korištenje cijele usluge i njezinu privlačnost. Spajanjem oba problema u jedan sustav bi učinilo uslugu zdravstvenog turizma puno efektivnijom i jeftinijom, a time i više privlačnom mogućim budućim klijentima.

Postojeće vrste konkurentnih rješenja:

- javno dostupne informacije o načinima sudjelovanja u zdravstvenom turizmu i pripadnim članovima
- udruge sa svrhom spajanja svih dionika pružanja usluga zdravstvenog turizma

Primjer za usluge pružanja informacija je web stranica „MedicalTourism.com“ koja pruža korisnicima podatke o pružateljima zdravstvenih usluga, smještaja i mogućih tretmana koje spomenute usluge pružaju. Također sadrži i mnoge druge informacije kao vodiče za destinacije, usporedbe cijena i slično. No ipak je spomenuta web aplikacija napravljena za pružanje laganog pristupa svim potrebnim informacijama za moguće klijente, pružatelje zdravstvenih usluga, pružatelje smještaja te osiguravajuća društva na jednom mjestu te se time ne sukobljava do konkurentske razine sa svrhom kreiranja projektne aplikacije.

Dok je primjer druge vrste rješenja, čak konkurentnije ideji projekta, udruga Medical Tourism Association, koja spaja sve potrebne članove zdravstvenog turizma kroz program članstva. Udruga operira po cijelom svijetu i služi svrhu omogućavanja pružanja usluga zdravstvenog turizma kroz povezivanje potrebnih članova te svrhu informiranja javnosti o mogućnostima korištenja tih usluga. Također podržava edukaciju budućih pružatelja sa sveukupnim ciljem povećanja učestalosti zdravstvenog turizma u svijetu. Ovdje opisan projekt ipak sadrži određene funkcionalnosti koje nedostaju navedenoj konkurenciji, kao automatiziran proces organizacije i administracije, te lagano dostupni pregled informacija o obitavanju za klijente zdravstvenih usluga.



Slika 2.1: Mogućnosti povezivanja klijenata i pružatelja koje nudi web stranica "MedicalTourism.com"

Ciljani klijenti opisanih usluga su pružatelji usluga zdravstvenog turizma po cijelom svijetu, uključujući Hrvatsku i slične države sa jeftinom cijenom boravka. Optimalno bi bilo započeti pružanje usluga aplikacije najprije Europskim državama i okolici, a zatim, uz dovoljnu uspješnost i isplativost projekta, raširiti dostupnost usluge ostatku svijeta. Ciljani klijenti bi također bili i pružatelji lokalnih prijevoznih usluga, od privatnih firmi do javnih taksija, koje bi aplikacija spajala sa pružateljima zdravstvenih usluga za dogovor o prijevozu njihovih klijenata. Također bi, u manjoj perspektivi, evidencija većine korisnika zdravstvenog turizma na jednom mjestu olakšala statističke analize razvoja zdravstvenog turizma po cijelom svijetu.

U aplikaciji postoje tri uloge:

- smještajni administrator
- administrator prijevoznih usluga
- korisnički administrator

Jedan korisnik može imati više administratorskih uloga, dok uloga smještajnog administratora sadrži najveće ovlasti kojima mogu definirati druge korisnike i dodjeljivati uloge. Svaka uloga administratora sadrži određene posebne mogućnosti dodavanja i promjene informacija.

Smještajni administratori upravljaju podacima o smještaju; unose podatke o raspoloživom smještaju te definiraju smještajne kapacitete i unose ili brišu osnovne podatke o smještaju. Podatci smještaja se sastoje od tipa stana, kategorije opremljenosti, adrese i vremenskog perioda dostupnosti za korištenje. Uz svaki smještaj je dostupan grafički prikaz geografskog položaja uz Google Maps usluge.

Administratori prijevoznih usluga upravljaju podacima o prijevoznicima. To uključuje osnovne osobne i kontaktne podatke prijevoznika, vrstu i kapacitet prijevoznog sredstva te radno vrijeme raspoloživosti. Osnovni podatci prijevoznika se ne mogu mijenjati.

Korisnički administratori definiraju korisnike medicinskih usluga uz unos njihovih osnovnih podataka. Podatci korisnika se sastoje od imena, prezimena, kontaktnih podataka, vremena dolaska i odlaska u/iz zemlje te preferencije o veličini i kvaliteti smještaja. Detalji njihovih tretmana se preuzimaju iz vanjske aplikacije o evidenciji medicinskih usluga.

Aplikacija periodički provjerava status unesenih podataka i pridjeljuje adekvatni smještaj upisanim korisnicima te po zaključenju plana medicinskih usluga određenog korisnika im pridjeljuje prijevoznike od raspoloživih za svaki od termina. Sa završetkom spomenutog zaključenja sustav šalje poruku elektroničke pošte korisniku medicinskih usluga sa svim potrebnim informacijama te također šalje poruke elektroničke pošte prijevoznicima pridijeljenima terminima korisnika sa svim njima potrebnim informacijama.

Moguće dodatne funkcionalnosti za aplikaciju koje se mogu nadodati nakon izvršavanja jezgrenih funkcionalnosti su:

- prijavljivanje samog korisnika medicinskih usluga u aplikaciju čime mogu provjeriti osobne podatke, smještaj, prijevoznike i termine
- proširenje aplikacije u dodatnu svrhu prikaza općenitih podataka o medicinskom turizmu za privlačenje dodatnih klijenata
- mogućnost pridjeljivanja istog smještaja većem broju pacijenata u slučaju da je smještaj dovoljno velik.



## 3. Specifikacija programske potpore

### 3.1 Funkcionalni zahtjevi

#### Dionici:

1. Pružatelji zdravstvenih usluga (klinike)
2. Pružatelji prijevoznih usluga (prijevoznici)
3. Klijenti zdravstvenog turizma (pacijenti)
4. Korisnici (administratori)
  - a) administratori smještaja
  - b) administratori prijevoza
  - c) korisnički administratori
5. Razvojni tim

#### Aktori i njihovi funkcionalni zahtjevi:

1. Neprijavljeni korisnik (inicijator) može:
  - (a) se prijaviti na postojeći korisnički račun upisivanjem korisničkog imena i lozinke
2. Administrator smještaja (inicijator) može:
  - (a) unijeti, modificirati i brisati podatke o smještaju
  - (b) vidjeti postojeće smještaje, njihove podatke i raspoloživost (uz grafički prikaz)
  - (c) unijeti i brisati podatke o prijavljenim klinikama
  - (d) vidjeti postojeće prijavljene klinike
  - (e) vidjeti postojeće korisnike
  - (f) registrirati nove korisnike te dodjeljivati uloge i brisati postojeće
3. Administrator prijevoznih usluga (inicijator) može:
  - (a) unijeti i brisati podatke o prijevoznicima
  - (b) modificirati podatke raspoloživosti prijevoznika
  - (c) vidjeti postojeće podatke prijevoznika i njihove raspoloživosti

4. Korisnički administrator (inicijator) može:

- (a) unijeti i brisati podatke pacijenata
- (b) vidjeti postojeće pacijente i njihove podatke
- (c) preuzeti i vidjeti detalje tretmana klinika u kontekstu određenog pacijenta

5. Baza podataka (sudionik):

- (a) pohranjuje sve podatke o korisnicima i njihovim ovlastima
- (b) pohranjuje sve podatke o smještajima, raspoloživosti i smještajnim kapacitetima
- (c) pohranjuje sve podatke o prijevoznicima
- (d) pohranjuje sve podatke o pacijentima
- (e) pohranjuje sve podatke o dogovorenim terminima boravka pacijenta i prijevoza tijekom boravka

### 3.1.1 Obrasci uporabe

#### UC1 - Prijava u sustav

- **Glavni sudionik:** Neprijavljeni korisnik
- **Cilj:** Dobiti pristup sustavu
- **Sudionici:** Baza podataka
- **Preduvjeti:**
  1. Postojanje korisničkog računa u bazi
- **Opis osnovnog tijeka:**
  1. Unos korisničkog imena i lozinke
  2. Sustav potvrđuje ispravnost unesenih podataka
  3. Sustav omogućava pristup funkcijama definirane ulogom korisničkog računa
- **Opis mogućih odstupanja:**
  - 2.a Neispravni podaci
    1. Aplikacija obavještava korisnika o neuspjeloj prijavi prikazivanjem poruke: "Incorrect username or password"

#### UC2 - Dodavanje novog korisnika

- **Glavni sudionik:** Smještajni korisnik
- **Cilj:** Kreiranje i dodavanje novog korisnika
- **Sudionici:** Baza podataka
- **Preduvjeti:**
  1. Korisnik je prijavljen na račun sa ulogom smještajnog administratora
- **Opis osnovnog tijeka:**
  1. Korisnik odabere opciju "Add new user"
  2. Aplikacija ponuđuje formu za popunjavanje informacija o novom korisniku
  3. Korisnik unosi sve tražene podatke: osobne podatke (*PIN, name, surname, phone i e-mail*) i korisničko-specifične podatke (*username, password i role*)
  4. Sustav u bazi stvara novog korisnika sa predanim podacima
- **Opis mogućih odstupanja:**
  - 3.a Unos postojećeg *username*-a
    1. Aplikacija obavještava korisnika o zauzetosti ponuđenog *username*-a porukom „Username already in use“

3.b Krivi format danog osobnog identifikacijskog broja (*PIN*), broja mobitela (*phone number*) ili adrese elektroničke pošte (*e-mail*)

1. Aplikacija obavještava korisnika o neispravnom formatu i sustav ne ažurira pripadajuće podatke u bazi podataka

### UC3 - Pregled korisnika

- **Glavni sudionik:** Smještajni korisnik
- **Cilj:** Pregled postojećih korisnika
- **Sudionici:** Baza podataka
- **Preduvjeti:**
  1. Korisnik je prijavljen na račun sa ulogom smještajnog administratora
- **Opis osnovnog tijeka:**
  1. Korisnik odabere opciju „View existing users“
  2. Aplikacija prikazuje podatke postojećih korisnika

### UC4 - Modificiranje podataka korisnika

- **Glavni sudionik:** Smještajni korisnik
- **Cilj:** Modificiranje podataka ciljanog korisnika
- **Sudionici:** Baza podataka
- **Preduvjeti:**
  1. Korisnik je prijavljen na račun sa ulogom smještajnog administratora
  2. Baza sadrži podatke o ciljanom korisniku
- **Opis osnovnog tijeka:**
  1. Korisnik odabere opciju „Modify user info“ pored podataka ciljanog korisnika
  2. Aplikacija ponuđuje formu za popunjavanje i izmjenjivanje informacija o korisniku
  3. Korisnik može izmijeniti osobne podatke (*phone number* i *e-mail*) i korisničko-specifične podatke (*password* i *role*)
  4. Sustav u bazi ažurira podatke ciljanog korisnika

### UC5 - Brisanje postojeće korisnika

- **Glavni sudionik:** Smještajni korisnik
- **Cilj:** Brisanje ciljanog korisnika
- **Sudionici:** Baza podataka
- **Preduvjeti:**

1. Korisnik je prijavljen na račun sa ulogom smještajnog administratora
2. Baza sadrži podatke o ciljanom korisniku
- **Opis osnovnog tijeka:**
  1. Korisnik odabere opciju „Delete“ pokraj podataka ciljanog korisnika
  2. Korisnik potvrđuje odabir nakon upita aplikacije
  3. Sustav briše podatke odabranog korisnika iz baze
- **Opis mogućih odstupanja:**
  - 2.a Korisnik odustane od brisanja tijekom procesa brisanja
    1. Aplikacija obavještava korisnika o prekidu brisanja

#### UC6 - Dodavanje novog smještaja

- **Glavni sudionik:** Smještajni korisnik
- **Cilj:** Kreiranje i dodavanje novog smještaja
- **Sudionici:** Baza podataka
- **Preduvjeti:**
  1. Korisnik je prijavljen na račun sa ulogom smještajnog administratora
- **Opis osnovnog tijeka:**
  1. Korisnik odabere opciju „Add new accomodation“
  2. Sustav ponuđuje formu za popunjavanje informacija o novom smještaju
  3. Korisnik unosi tražene podatke: osnovne podatke (*address, latitude, longitude, accomodation type* i *equipment category*) i da li je smještaj raspoloživ (*active*)
  4. Sustav u bazi stvara novi smještaj sa predanim podacima
- **Opis mogućih odstupanja:**
  - 3.a Unos postojeće adrese (*address*) ili geografske pozicije (*latitude* i *longitude*)
    1. Aplikacija obavještava korisnika o već postojećim unesenim podacima u bazi

#### UC7 - Pregled smještaja

- **Glavni sudionik:** Smještajni korisnik
- **Cilj:** Pregled unesenih smještaja
- **Sudionici:** Baza podataka
- **Preduvjeti:**
  1. Korisnik je prijavljen na račun sa ulogom smještajnog administratora
- **Opis osnovnog tijeka:**

1. Korisnik odabere opciju „View accomodations“
2. Aplikacija prikazuje podatke unesenih smještaja, uključujući prikaz geografske lokacije smještaja na karti

#### UC8 - Postavljanje raspoloživosti smještaja

- **Glavni sudionik:** Smještajni korisnik
- **Cilj:** Postavljanje željene raspoloživosti smještaja
- **Sudionici:** Baza podataka
- **Preduvjeti:**
  1. Korisnik je prijavljen na račun sa ulogom smještajnog administratora
- **Opis osnovnog tijeka:**
  1. Korisnik postavlja raspoloživost smještaja pomoću potvrdnog okvira
  2. Sustav u bazi ažurira raspoloživost smještaja

#### UC9 - Brisanje postojećeg smještaja

- **Glavni sudionik:** Smještajni korisnik
- **Cilj:** Brisanje ciljanog smještaja
- **Sudionici:** Baza podataka
- **Preduvjeti:**
  1. Korisnik je prijavljen na račun sa ulogom smještajnog administratora
  2. Baza sadrži podatke o ciljanom smještaju
- **Opis osnovnog tijeka:**
  1. Korisnik odabere opciju „Remove accomodation“ pokraj podataka ciljanog smještaja
  2. Korisnik potvrđuje odabir nakon upita aplikacije
  3. Sustav briše podatke odabranog smještaja iz baze
- **Opis mogućih odstupanja:**
  - 2.a Korisnik odustane od brisanja tijekom procesa brisanja
    1. Aplikacija obavještava korisnika o prekidu brisanja

#### UC10 - Dodavanje prijevoznika

- **Glavni sudionik:** Administrator prijevoznih usluga
- **Cilj:** Kreiranje i dodavanje novog prijevoznika
- **Sudionici:** Baza podataka
- **Preduvjeti:**

1. Korisnik je prijavljen na račun sa ulogom administratora prijevoznih usluga
- **Opis osnovnog tijeka:**
  1. Korisnik odabere opciju „Add new transporter“
  2. Aplikacija ponuđuje formu za popunjavanje informacija o novom prijevozniku
  3. Korisnik unosi tražene osobne i kontaktne podatke prijevoznika (*phone number, organization name, address* i *active*)
  4. Sustav u bazu sprema podatke o prijevozniku
- **Opis mogućih odstupanja:**
  - 3.a Krivi format danog broja mobitela (*phone number*)
    1. Aplikacija obavještava korisnika o neispravnom formatu i ne ažurira pripadajuće podatke u bazi podataka

#### UC11 - Pregled prijevoznika

- **Glavni sudionik:** Administrator prijevoznih usluga
- **Cilj:** Pregled unesenih prijevoznika
- **Sudionici:** Baza podataka
- **Preduvjeti:**
  1. Korisnik je prijavljen na račun sa ulogom administratora prijevoznih usluga
- **Opis osnovnog tijeka:**
  1. Korisnik odabere opciju „View transporters“
  2. Aplikacija prikazuje podatke unesenih prijevoznika

#### UC12 - Brisanje prijevoznika

- **Glavni sudionik:** Administrator prijevoznih usluga
- **Cilj:** Brisanje ciljanog prijevoznika
- **Sudionici:** Baza podataka
- **Preduvjeti:**
  1. Korisnik je prijavljen na račun sa ulogom administratora prijevoznih usluga
  2. Baza sadrži podatke ciljanog prijevoznika
- **Opis osnovnog tijeka:**
  1. Korisnik odabere opciju „Delete“ pokraj podataka ciljanog prijevoznika
  2. Korisnik potvrđuje odabir nakon upita aplikacije

3. Sustav briše podatke odabranog smještaja iz baze
- **Opis mogućih odstupanja:**
  - 2.a Korisnik odustane od brisanja tijekom procesa brisanja
    1. Aplikacija obavještava korisnika o prekidu brisanja

#### UC13 - Dodavanje vozila prijevoznika

- **Glavni sudionik:** Administrator prijevoznih usluga
- **Cilj:** Kreiranje i dodavanje novog vozila prijevoznika
- **Sudionici:** Baza podataka
- **Preduvjeti:**
  1. Korisnik je prijavljen na račun sa ulogom administratora prijevoznih usluga
- **Opis osnovnog tijeka:**
  1. Korisnik odabere opciju „Add new vehicle“ pod podacima ciljanog prijevoznika
  2. Aplikacija ponuđuje formu za popunjavanje informacija o novom vozilu
  3. Korisnik unosi tražene podatke o vozilu (*type of vehicle, capacity* i *active*)
  4. Sustav u bazi stvara novo vozilo i pridjeljuje ga ciljanom prijevozniku

#### UC14 - Pregled vozila prijevoznika

- **Glavni sudionik:** Administrator prijevoznih usluga
- **Cilj:** Pregled unesenih vozila određenog prijevoznika
- **Sudionici:** Baza podataka
- **Preduvjeti:**
  1. Korisnik je prijavljen na račun sa ulogom administratora prijevoznih usluga
- **Opis osnovnog tijeka:**
  1. Korisnik odabere opciju „View assigned vehicles“
  2. Aplikacija prikazuje podatke unesenih vozila ciljanog prijevoznika

#### UC15 - Postavljanje raspoloživosti vozila prijevoznika

- **Glavni sudionik:** Administrator prijevoznih usluga
- **Cilj:** Postavljanje raspoloživosti ciljanog vozila prijevoznik
- **Sudionici:** Baza podataka
- **Preduvjeti:**



1. Korisnik je prijavljen na račun sa ulogom administratora prijevoznih usluga
- **Opis osnovnog tijeka:**
  1. Korisnik postavlja raspoloživost ciljanog vozila pomoću potvrdnog okvira
  2. Sustav u bazi ažurira raspoloživost ciljanog vozila

#### UC16 - Brisanje vozila prijevoznika

- **Glavni sudionik:** Administrator prijevoznih usluga
- **Cilj:** Brisanje ciljanog vozila prijevoznika
- **Sudionici:** Baza podataka
- **Preduvjeti:**
  1. Korisnik je prijavljen na račun sa ulogom administratora prijevoznih usluga
  2. Baza sadrži podatke ciljanog vozila ciljanog prijevoznika
- **Opis osnovnog tijeka:**
  1. Korisnik odabere opciju „Remove vehicle“ pod podacima ciljanog prijevoznika
  2. Korisnik potvrđuje odabir nakon upita aplikacije
  3. Sustav briše podatke odabranog vozila prijevoznika iz baze
- **Opis mogućih odstupanja:**
  - 2.a Korisnik odustane od brisanja tijekom procesa brisanja
    1. Aplikacija obavještava korisnika o prekidu brisanja

#### UC17 - Dodavanje pacijenta

- **Glavni sudionik:** Korisnički administrator
- **Cilj:** Kreiranje i dodavanje novog pacijenta
- **Sudionici:** Baza podataka
- **Preduvjeti:**
  1. Korisnik je prijavljen na račun sa ulogom korisničkog administratora
- **Opis osnovnog tijeka:**
  1. Korisnik odabere opciju „Add new patient“
  2. Aplikacija ponuđuje formu za popunjavanje informacija o novom pacijentu
  3. Korisnik unosi tražene podatke: osobne i kontaktne podatke (*PIN, name, surname, phone, e-mail i residence address*) te podatke o tretmanu (*clinic i treatment*) i preferenciji smještaja (*accomodation preferences*)

4. Sustav u bazi stvara novog pacijenta sa predanim podacima
- **Opis mogućih odstupanja:**
  - 3.a Krivi format danog osobnog identifikacijskog broja (*PIN*), broja mobitela (*phone number*) ili adrese elektroničke pošte (*e-mail*)
    1. Aplikacija obavještava korisnika o neispravnom formatu i ne ažurira pripadajuće podatke u bazi podataka

#### UC18 - Pregled pacijenata

- **Glavni sudionik:** Korisnički administrator
- **Cilj:** Pregled postojećih pacijenata
- **Sudionici:** Baza podataka
- **Preduvjeti:**
  1. Korisnik je prijavljen na račun sa ulogom korisničkog administratora
- **Opis osnovnog tijeka:**
  1. Korisnik odabere opciju „View patients“
  2. Aplikacija prikazuje podatke postojećih pacijenata, uključujući da li im je pridijeljen smještaj i prijevoz, te podacima o smještaju i prijevozu u slučaju da jesu pridijeljeni

#### UC19 - Dohvaćanje informacija o tretmanu

- **Glavni sudionik:** Sustav
- **Cilj:** Dohvaćanje informacija o tretmanu pacijenta pri registraciji/dodavanju pacijenta
- **Sudionici:** Baza podataka, baza podataka pacijentove klinike
- **Preduvjeti:**
  1. Baza sadrži sve potrebne podatke o pacijentu
  2. Baza podataka klinike sadrži potrebne podatke o tretmanu pacijenta
- **Opis osnovnog tijeka:**
  1. Sustav uspostavlja vezu sa bazom podataka klinike
  2. Sustav dohvaća podatke o tretmanu iz baze podataka klinike
  3. Sustav sprema dohvaćene podatke u bazu podataka sustava
- **Opis mogućih odstupanja:**
  - 1.a Sustav ne može uspostaviti vezu sa klinikom
    1. Aplikacija obavještava korisnika o neuspješnom povezivanju

#### UC20 - Brisanje pacijenta

- **Glavni sudionik:** Korisnički administrator
- **Cilj:** Brisanje ciljanog vozila prijevoznika
- **Sudionici:** Baza podataka
- **Preduvjeti:**
  1. Korisnik je prijavljen na račun sa ulogom korisničkog administratora
  2. Baza sadrži podatke ciljanog pacijenta
- **Opis osnovnog tijeka:**
  1. Korisnik odabere opciju „Delete“ pokraj podataka ciljanog pacijenta
  2. Korisnik potvrđuje odabir nakon upita aplikacije
  3. Sustav briše podatke odabranog pacijenta iz baze
- **Opis mogućih odstupa:**
  - 2.a Korisnik odustane od brisanja tijekom procesa
    1. Aplikacija obavještava korisnika o prekidu brisanja

#### UC21 - Periodičko pridieljivanje smještaja i prijevoza pacijentima

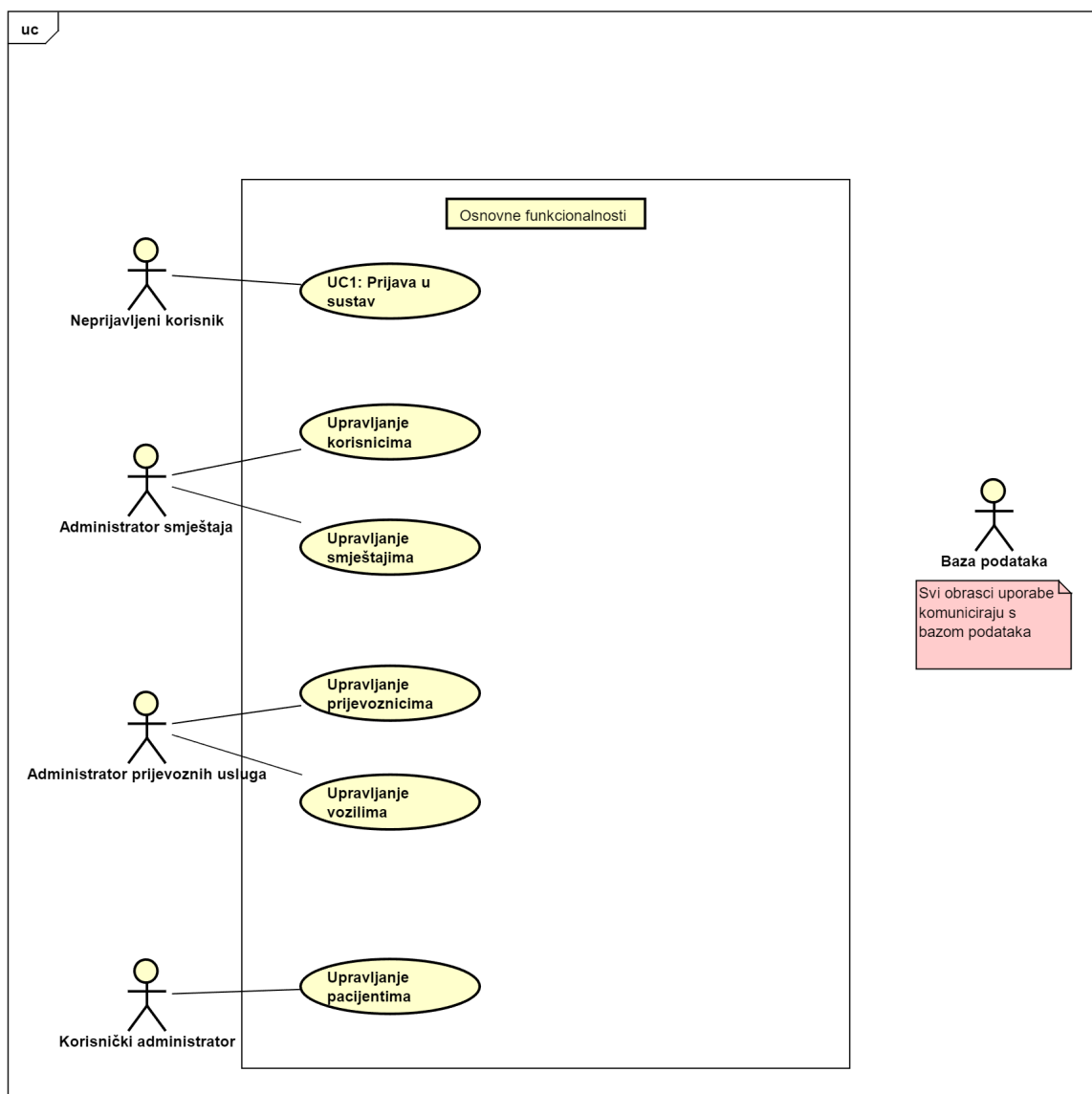
- **Glavni sudionik:** Korisnički administrator
- **Cilj:** Pridieljivanje adekvatnog smještaja i prijevoza upisanim pacijentima te obavještavanje pacijenta o uspješnom pridieljenju
- **Sudionici:** Baza podataka
- **Preduvjeti:**
  1. Baza sadrži potrebne podatke o pacijentima, smještajima i prijevoznima
  2. Baza sadrži barem jednu instancu svakog potrebnog podatka
- **Opis osnovnog tijeka:**
  1. Sustav provjerava unesene podatke za pacijente bez smještaja
  2. Sustav (ako je moguće) pridieljuje adekvatni smještaj pacijentu
  3. Sustav (ako je moguće) pridieljuje adekvatni prijevoz pacijentu pri zaključenju medicinskog plana
- **Opis mogućih odstupa:**
  - 2.a Sustav ne može naći adekvatni smještaj i/ili prijevoz pacijentu
    1. Sustav periodički pokušava ponovno pri unosu novih podataka

#### UC22 - Obavještavanje klijenata o uspješnom zaključenju plana

- **Glavni sudionik:** Sustav
- **Cilj:** Obavještavanje pacijenta, prijevoznika o uspješno zaključenom planu medicinskih usluga

- **Sudionici:** Baza podataka, pacijent, prijevoznik
- **Preduvjeti:**
  1. Baza sadrži potrebne podatke o pacijentima, smještajima i prijevoznima
  2. Sustav je uspješno zaključio medicinski plan pacijenta pridijeljivanjem smještaja i prijevoza pacijentu
- **Opis osnovnog tijeka:**
  1. Sustav provjerava zaključen plan te primjereno modificira podatke u bazi podataka (raspoloživost smještaja i vozila prijevoznika)
  2. Aplikacija šalje mail pacijentu sa podacima o datumu plana korištenja tretmana te smještaju i prijevozu tijekom plana tretmana
  3. Aplikacija šalje mail prijevozniku o zauzetosti vozila koje će se koristiti za pridijeljeni medicinski plan

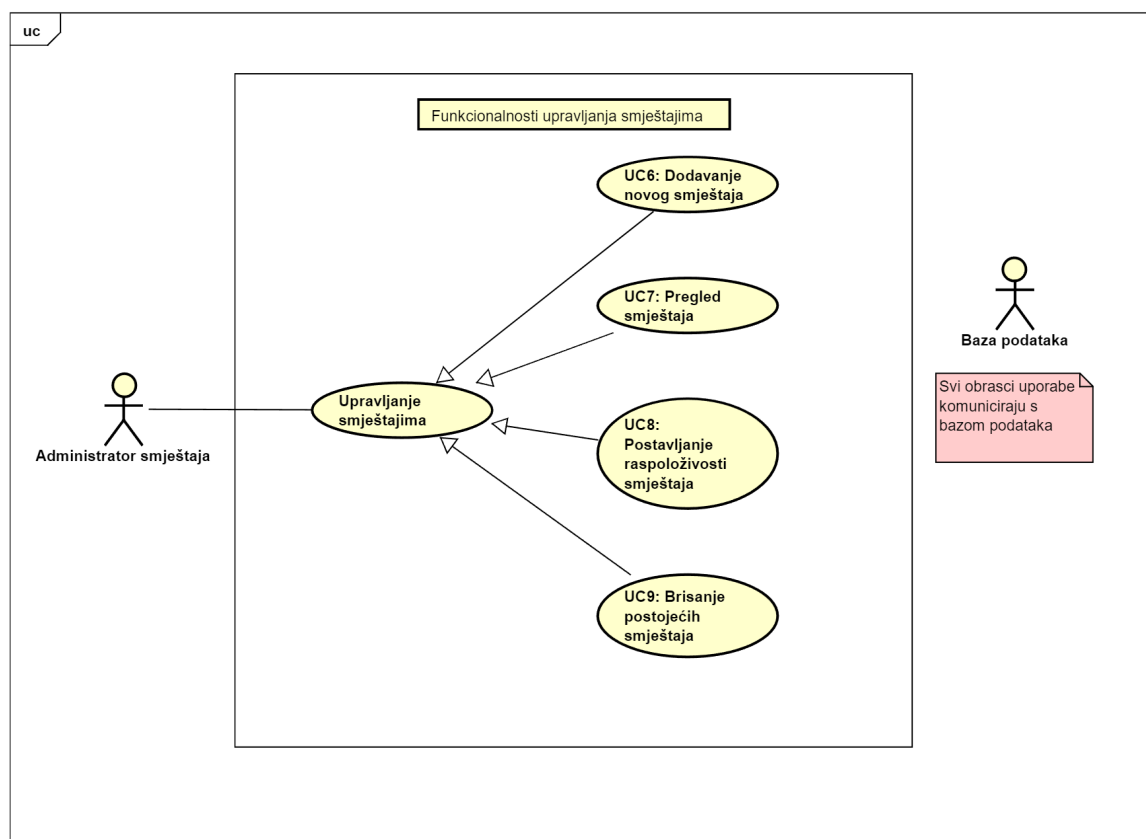
## Dijagrami obrazaca uporabe



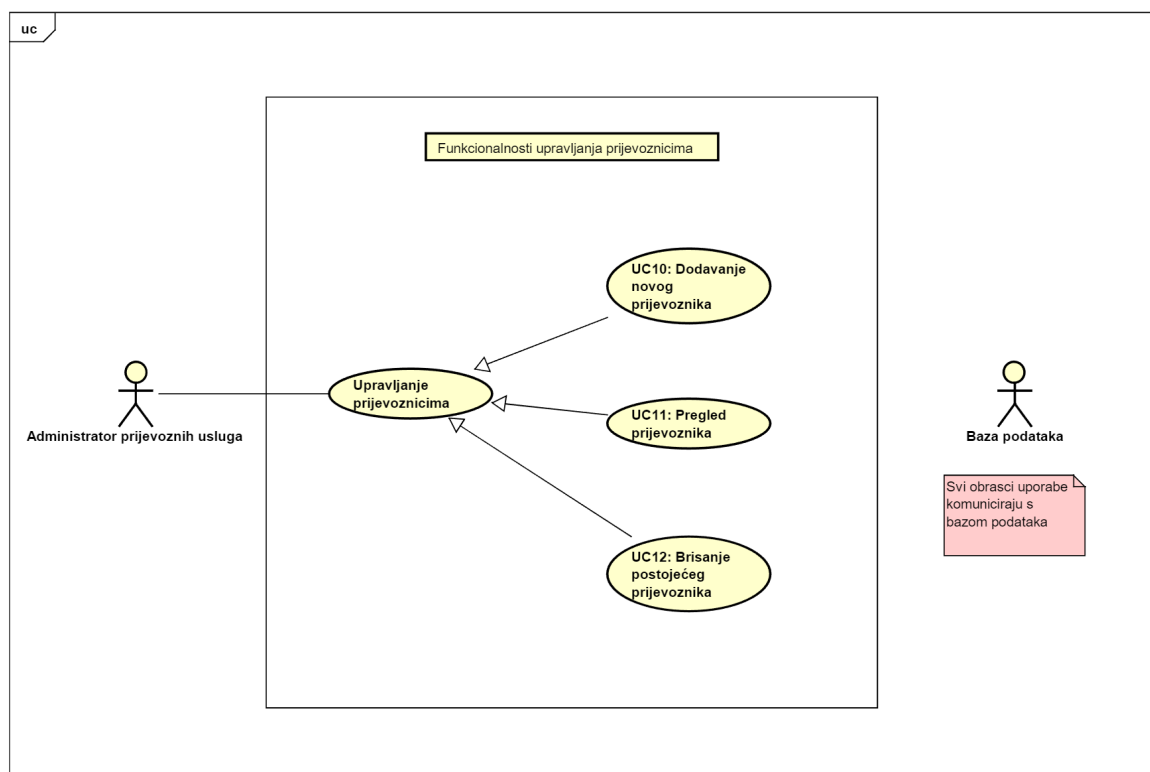
Slika 3.1: Dijagram obrasca uporabe, osnovne funkcionalnosti



Slika 3.2: Dijagram obrasca uporabe, funkcionalnosti upravljanja korisnicima

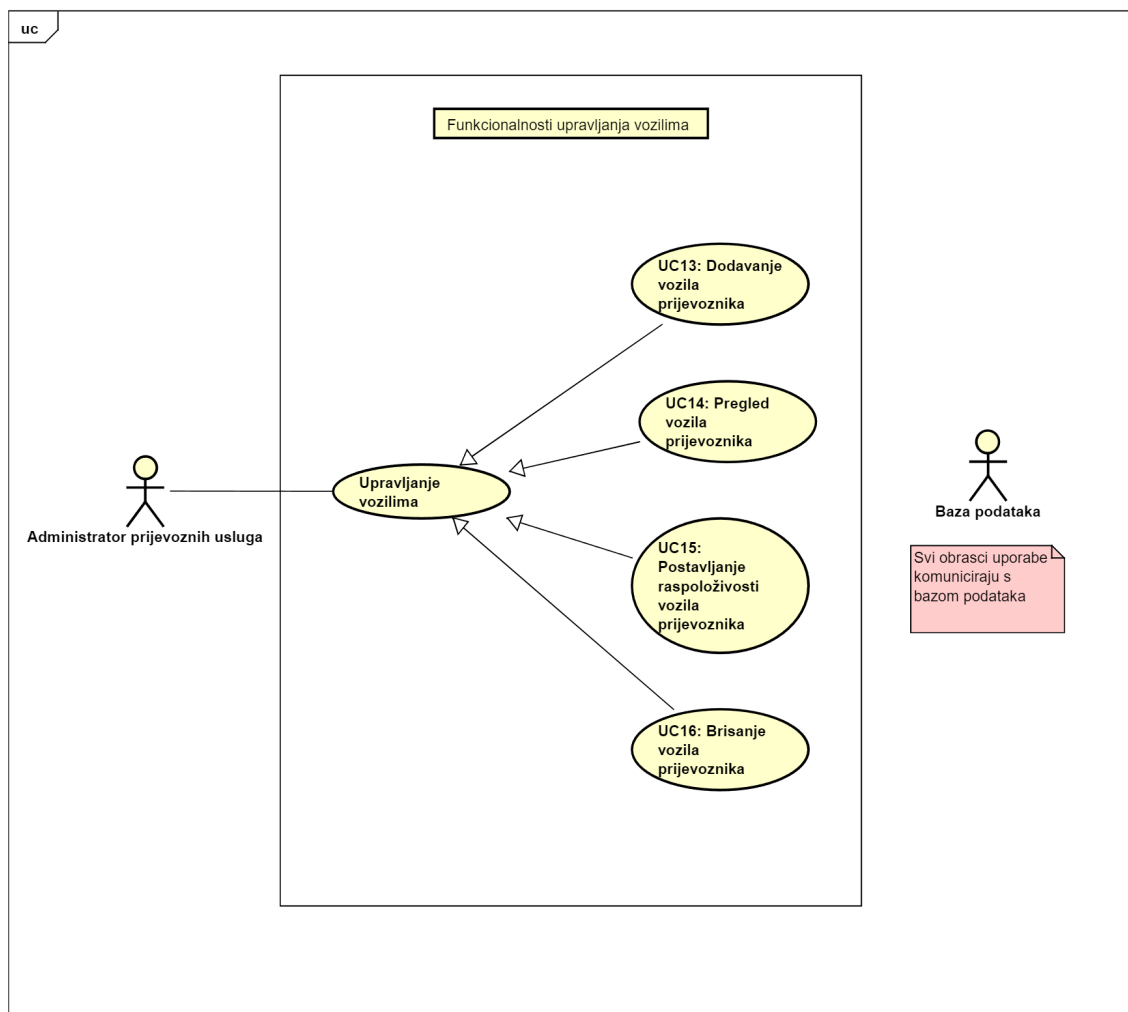


Slika 3.3: Dijagram obrasca uporabe, funkcionalnosti upravljanja smještajima

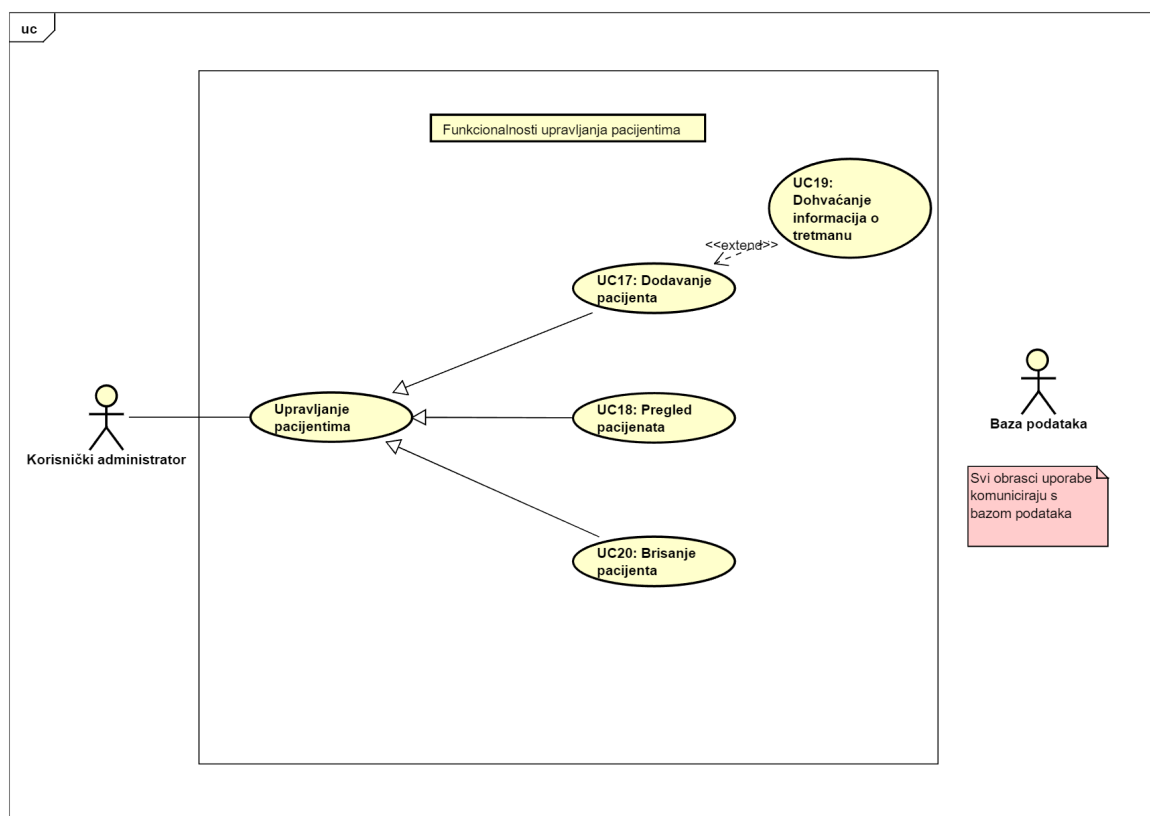


Slika 3.4: Dijagram obrasca uporabe, funkcionalnosti upravljanja prijevoznicima

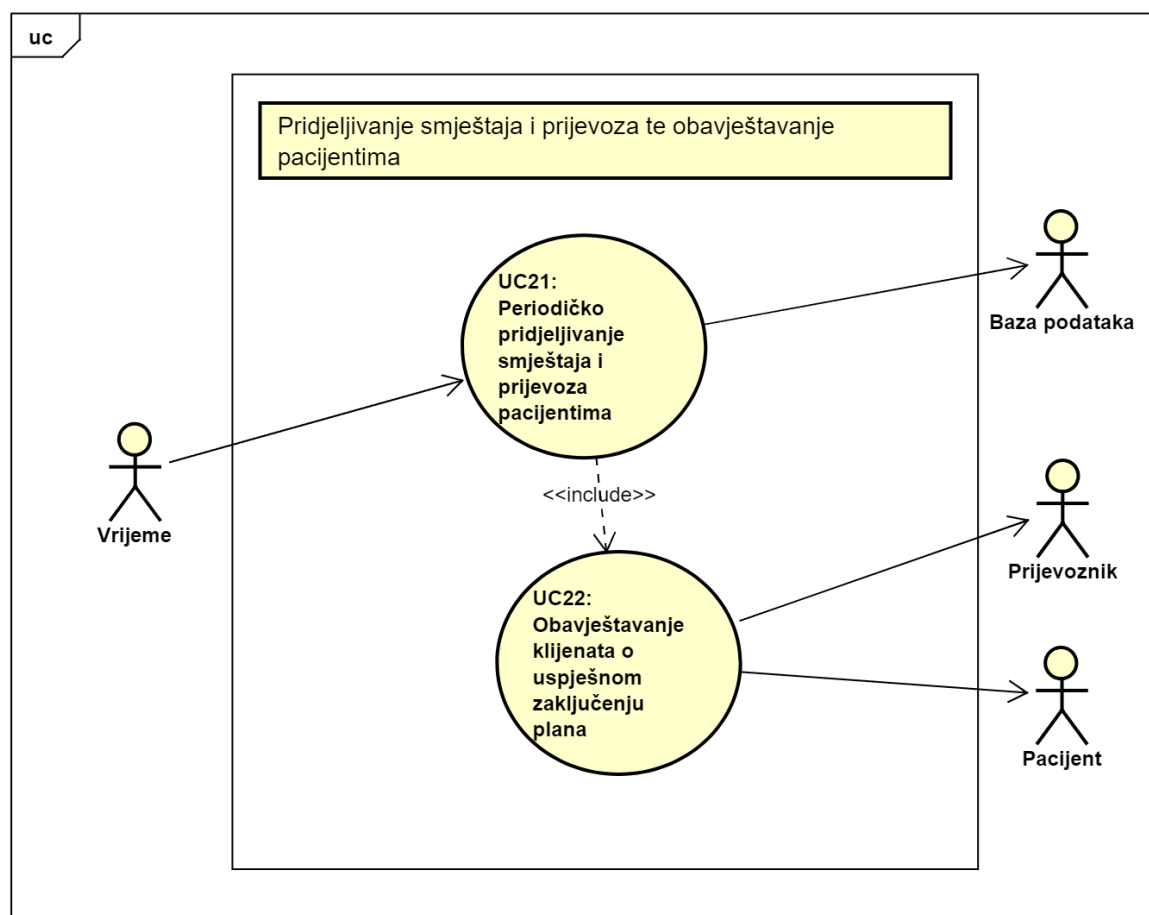




Slika 3.5: Dijagram obrasca uporabe, funkcionalnosti upravljanja vozilima



Slika 3.6: Dijagram obrasca uporabe, funkcionalnosti upravljanja pacijentima

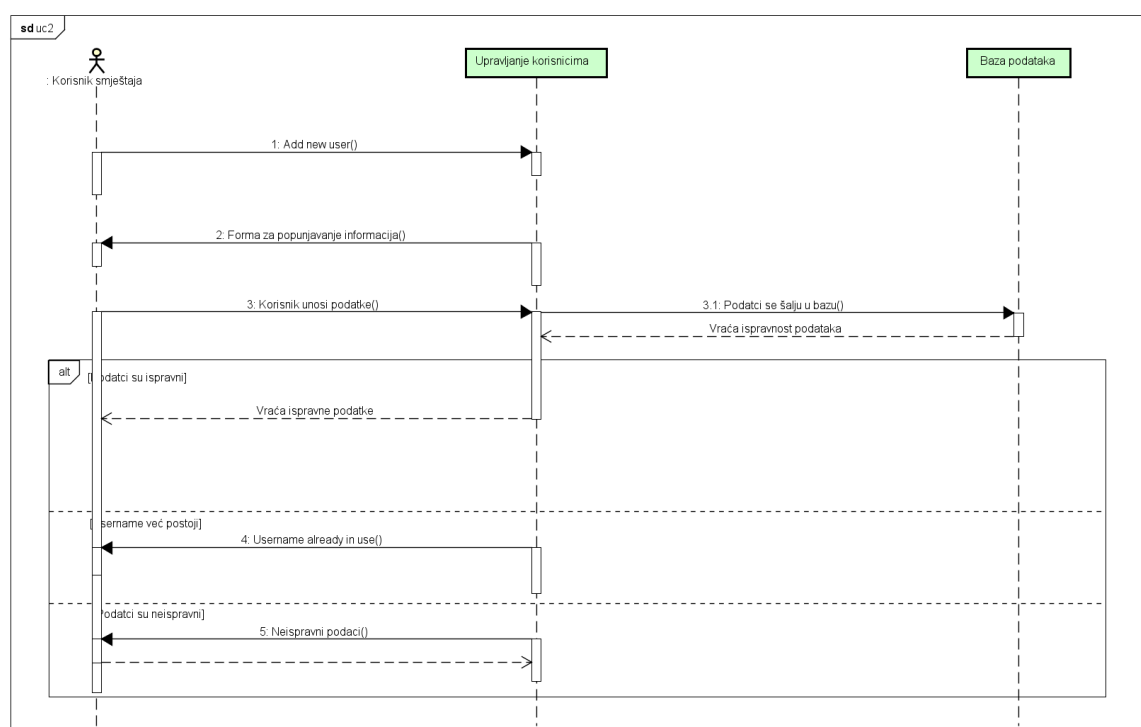


Slika 3.7: Dijagram obrasca uporabe, pridjeljivanje i obavještavanje

### 3.1.2 Sekvencijski dijagrami

#### Obrazac uporabe 2 - Dodavanje novog korisnika

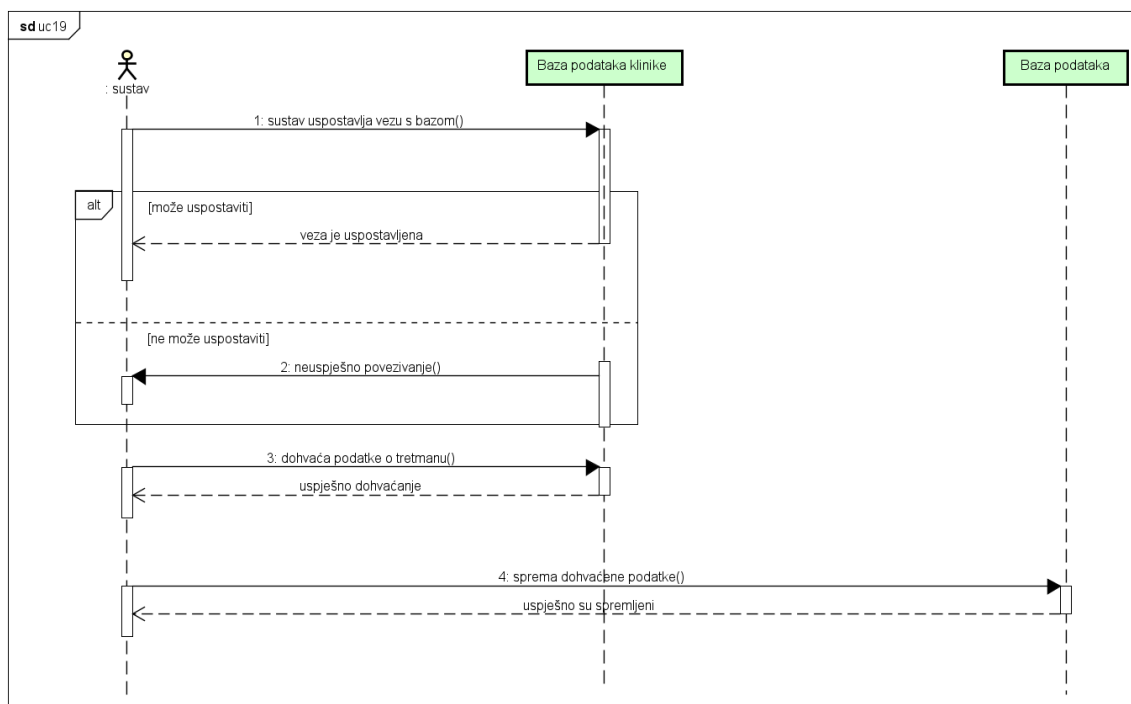
Kada korisnik koji ima ovlasti smještajnog administratora klikne na gumb "Add new user", pojavi mu se forma za ispunjavanje informacija o novom korisniku kojeg želi napraviti. Nakon što su podaci uneseni, oni se pošalju u bazu podataka koja potom javlja rezultat dodavanja novog korisnika u sustav, tj. ako nema nikakvih problema, novi korisnik je uspješno dodan u bazu podataka, no može doći do greške poput unošenja imena već postojećeg korisnika ili do unesenih neispravnih podataka kao što je krivi format e-mail-a, broja mobitela ili PIN-a.



Slika 3.8: Sekvencijski dijagram za UC2

**Obrazac uporabe 19 - Dohvaćanje informacija o tretmanu**

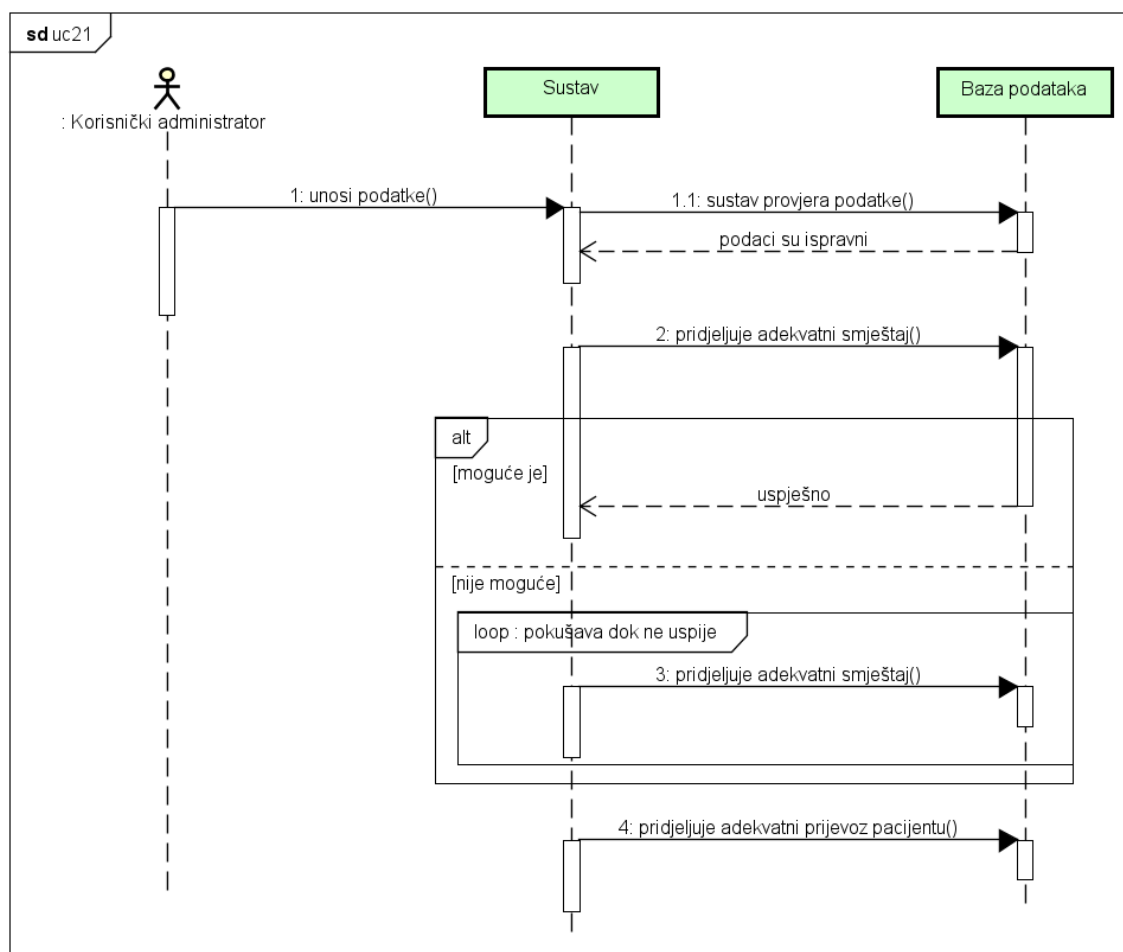
Nakon što je dodan novi pacijent, možemo iz vanjske baze podataka povući podatke o njegovom tretmanu. Prvotno sustav pokušava uspostaviti kontakt s vanjskom bazom podataka nakon čega može doći do dva slučaja: da se veza uspješno uspostavi, ili da se iz nekog razloga (npr. nedostatak interneta) ne uspije uspostaviti. Ako sustav uspije uspostaviti vezu, onda povuče tražene podatke te ih potom spremi u vlastitu bazu podataka.



Slika 3.9: Sekvencijski dijagram za UC19

**Obrazac uporabe 21 - Periodičko pridjeljivanje smještaja i prijevoza pacijentima**

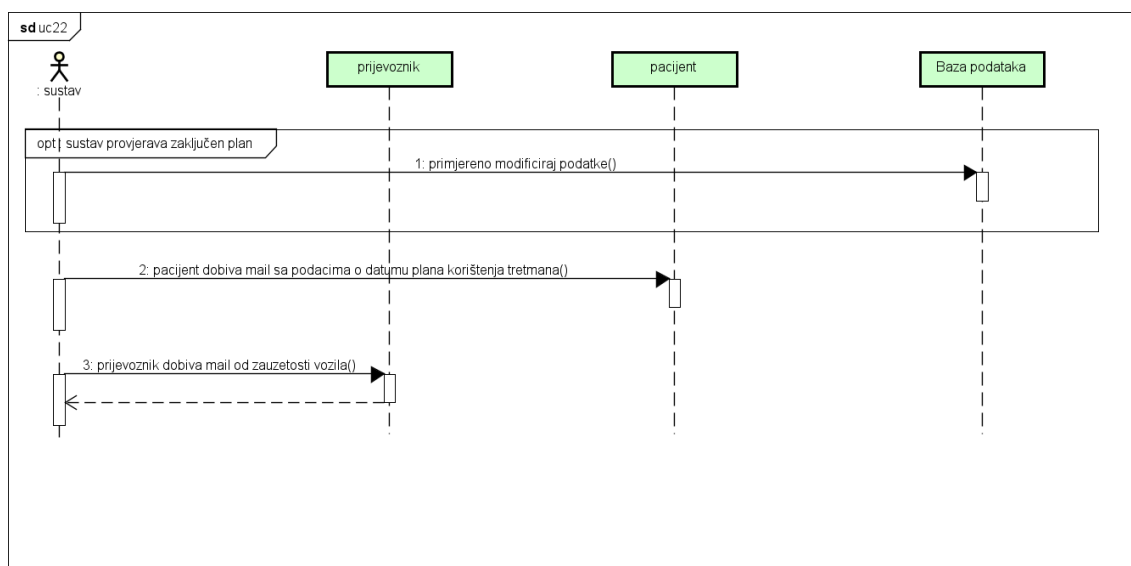
Svako malo sustav će pokušati dodijeliti postojećim pacijentima u bazi podataka koji još nemaju dodijeljen smještaj adekvatni smještaj. Ako sustav uspije pronaći adekvatan smještaj za pacijenta, onda se pacijentu dodijeli taj smještaj, a ako ne onda će se to ponovno pokušati nakon nekog vremena sve dok se ne uspije.



Slika 3.10: Sekvencijski dijagram za UC21

**Obrazac uporabe 22 - Obavještavanje klijenata o uspješnom zaključenju plana**

Nakon što se plan za nekog pacijenta zaključi, on se provjeri te se mijenjaju podaci u bazi podataka u vezi raspoloživosti smještaja i vozila prijevoznika. Potom se pacijentu šalje mail u kojemu ga se obavještava o datumu tretmana te smještaju i prijevozu tijekom plana tretmana te se prijevozniku šalje mail o zauzetosti vozila koje će se koristiti za pridjeljeni medicinski plan.



Slika 3.11: Sekvencijski dijagram za UC22

## 3.2 Ostali zahtjevi

- Sustav treba omogućiti rad više korisnika u stvarnom vremenu
- Korisničko sučelje i sustav moraju podržavati hrvatsku abecedu (dijakritičke znakove) pri unosu i prikazu tekstualnog sadržaja
- Izvršavanje dijela programa u kojem se pristupa bazi podataka ne smije trajati duže od nekoliko sekundi
- Neispravno korištenje korisničkog sučelja ne smije narušiti funkcionalnost i rad sustava
- Aplikacija treba biti jednostavna za korištenje, korisnici se moraju znati koristiti sučeljem bez opširnih uputa
- Nadogradnja sustava ne smije narušavati postojeće funkcionalnosti sustava
- Aplikacija kao valutu koristi EUR
- Veza s bazom podataka mora biti kvalitetno zaštićena, brza i otporna na vanjske greške
- Pristup sustavu mora biti omogućen iz javne mreže pomoću HTTPS.



## 4. Arhitektura i dizajn sustava

Arhitektura sustava može se podijeliti na tri glavna podsustava: web preglednik, web poslužitelj i baza podataka.

- **Web preglednik** – program koji služi za pristupanje web stranicama. Putem web preglednika korisnik komunicira sa poslužiteljem koristeći princip zahtjev-odgovor ili šalje podatke (najčešće u obliku obrazaca). Daljnja zadaća web preglednika je osigurati da se traženi podaci ispravno prikazuju ili da se ispravno prosljeđuju i spremaju na poslužitelj.
- **Web poslužitelj** – glavni je dio web aplikacije. To je most koji povezuje korisnika i bazu podataka koji se temelji na protokolu HTTP. Na zahtjeve korisnika dohvaća tražene podatke (resurse) ili obrađuje i sprema poslane podatke od korisnika.
- **Baza podataka** – srce je sustava. U njoj su pospremljeni svi podaci. Skoro ne postoji sustav u kojem nema komunikacije između aplikacije i baze.

Aplikacije je izgrađena na modelu MVC (Model – View - Controller) softverske arhitekture uz male modifikacije. Controller dio strukture je ostvaren tako što je integriran unutar same baze, tj. funkcije koje manipuliraju podacima se nalaze unutar baze. Shodno tome aplikaciju onda dijelimo na tri komponente:

- **Model** – glavna komponenta sustava. Reprezentira strukturu podataka.
- **View** – komponenta zaslužna za reprezentaciju podataka.
- **Controller** – komponenta koja odrađuje svu logiku i komunikaciju između sučelja i baze.

Backend naše aplikacije je ostvaren direktno u bazi PostgreSQL (razvojno sučelje pgadmin) za što koristimo API napisan u Node.js frameworku Express, koji služi kao middleware. Za izradu frontend-a korišten je React uz pomoć Material UI. Razvojno okruženje koje smo koristili je bilo Visual Studio Code.

## 4.1 Baza podataka

U ovom projektu koristit ćemo relacijsku bazu podataka, čije su osnovne jedinice entiteti, definirani imenom i skupom atributa. Osnovna zadaća baze podataka je pohrana podataka te brza i efikasna obrada tih podataka u ovisnosti i korisničkim zahtjevima. U bazi podataka su pohranjeni podaci o korisnicima, njihovim ulogama, preferencijama, kao i o smještaju te dostupnosti smještaja. Dodatno uz navedeno, zbog zahtjeva organizacije prijevoza, baza također sprema informacije o vrstama vozila i dostupnosti tih vozila kao i o vremenu i mjestu boravka korisnika zdravstvenog turizma. Tako su i definirani sljedeći entiteti:

- Clinic
- Town
- AdminUser
- Credentials
- UserRole
- Patient
- Treatment
- PatientPreferences
- PatientArrival
- Accomodation
- AccomodationType
- Equipped
- AccomodationOccupied
- Transporter
- Vehicle
- VehicleType
- VehicleOccupied
- VehicleSchedule

### 4.1.1 Opis tablica

**Clinic** Ovaj entitet sadrži podatke o klinikama u odabranoj zemlji. Atributi su: ClinicID (primary key), clinicName, latitude, longitude, clinicAddress, TownID (foreign key). Ovaj je entitet u vezi Many-to-One sa entitetom Town preko atributa TownID, u Many-to-Many vazi sa entitetom Accomodation preko atributa AccomodationID i ClinicID, u Many-to-Many vezi sa entitetom Transporter preko atributa TransporterID i ClinicID, u One-to-Many vezi sa entitetom Patient preko atributa PatientID i ClinicID te u Many-to-Many vezi sa entitetom Treatment preko atributa TreatmentID.

Clinic		
ClinicID	BIGSERIAL	Jedinstveni brojčani identifikator klinike, automatski generiran
clinicName	VARCHAR	Ime klinike
Latitude	DECIMAL	Geografska širina
Longitude	DECIMAL	Geografska dužina
clinicAddress	VARCHAR	Adresa klinike
TownID	INT	Grad u kojem se klinika nalazi

**Town** Ovaj entitet sadrži podatke o gradovima u kojima se nalaze klinike u koje dolaze pacijenti na liječenje. Atributi su: TownID (primary key) i townName. Ovaj entitet je u vezi One-to-Many sa entitetom Clinic preko atributa TownID i u vezi One-to-Many sa entitetom Accomodation preko atributa TownID i u Many-to-Many vezi sa entitetom Transporter preko atributa TownID.

Town		
TownID	BIGSERIAL	Jedinstveni brojčani identifikator grada, automatski generiran
townName	VARCHAR	Ime grada

**AdminUser** Ovaj entitet sadrži podatke o korisnicima aplikacije; svi administratori i oni koji mogu dodavati ili ažurirati ili brisati podatke iz baze. Atributi su: UserID (primary key), PIN(personal identification number), firstname, lastname, phone, email. Ovaj je entitet u vezi Many-to-Many sa entitetom UserRole preko atributa RoleID te u vezi One-to-One sa entitetom Credentials preko atributa UserID.

AdminUser		
UserID	BIGSERIAL	Jedinstveni brojčani identifikator korisnika, automatski generiran
PIN	INT	Identifikacijski broj korisnika
firstname	VARCHAR	Ime korisnika
lastname	VARCHAR	Prezime korisnika
phone	VARCHAR	Broj mobitela korisnika
email	VARCHAR	Elektronička pošta korisnika

**Credentials** Ovaj entitet sadrži podatke o korisničkim računima administratora. Atributi su: UserID (foreign key), username i pass. Ovaj je entitet u One-to-One vezi sa entitetom AdminUser preko atributa UserID.

Credentials		
UserID	BIGSERIAL	ID korisnika kojem pripadaju korisničko ime i lozinka
username	VARCHAR	Jedinstveno korisničko ime
pass	BINARY	Lozinka korisnika za prijavu u aplikaciju

**UserRole** Ovaj entitet sadrži podatke o ulogama koje postoje u sustavu. Atributi su: RoleID i roleName. Ovaj je entitet u Many-to-One vezi sa entitetom UserAdmin preko atributa UserID.

UserRole		
RoleID	SMALLINT	Jedinstveni indentifikator uloge, automatski generiran
roleName	VARCHAR	ime uloge

**Patient** Ovaj entitet sadrži podatke o korisnicima zdravstvenog turizma. Atributi su: PatientID (primary key), PIN (personal identification number), firstname, lastname, phone, email, residenceAddress. Ovaj je entitet u vezi One-to-One sa entitetom Accomodation preko atributa AccomodationID, u vezi One-to-Many vezi sa entitetom Treatment preko atributa TreatmentID te u One-to-Many vezi sa entitetom Clinic preko atributa ClinicID i PatientID.

Patient		
PatientID	BIGSERIAL	Jedinstveni brojčani identifikator pacijenta, automatski generiran
PIN	INT	Identifikacijski broj pacijenta
firstname	VARCHAR	Ime pacijenta
lastname	VARCHAR	Prezime pacijenta
phone	VARCHAR	Broj mobitela pacijenta
email	VARCHAR	Elektronička pošta pacijenta
residenceAddress	VARCHAR	Mjesto prebivališta pacijenta

**Treatment** Ovaj entitet sadrži podatke o tretmanima. Atributi su: TreatmentID (primary key) i description. Ovaj je entite u Many-to-One vezi sa entitetom Patient preko atributa TreatmentID i PatientID te u One-to-Many vezi sa entitetom Clinic preko atributa TreatmentID i ClinicID.

Treatment		
TreatmentID	BIGSERIAL	Jedinstveni indentifikator tretmana, automatski generiran
description	text	opsi tretmana

**PatientPreferences** Ovaj entitet sadrži podatke o preferencijama pacijenta vezanih za smještaj. Atributi su: PatientID (foreign key), TypeID (foreign key), EquippedID (foreign key). Ovaj je entite u Many-to-One vezi sa entitetom Patient preko atributa PatientID, u One-to-One vezi sa entitetom AccomodationType preko atributa AccomodationID te u One-to-One vezi sa entitetom Equipped preko atributa EquippedID.

PatientPreferences		
PatientID	BIGSERIAL	Jedinstveni indentifikator pacijenta, automatski generiran
TypeID	SMALLINT	Jedinstveni indentifikator vrste smještaja, automatski generiran
EquippedID	SMALLINT	Jedinstveni indentifikator opremljenosti smještaja, automatski generiran

**PatientArrival** Ovaj entitet sadrži podatke o vremenu dolaska/odlaska pacijenta te također i gradu u kojem se liječi. Atributi su: ArrivalID(primary key), PatientID (foreign key), TownID (foreign key), dateOfArrival, dateOfDeparture. Ovaj je entite u One-to-One vezi sa entitetom Patient preko atributa PatientID, u One-to-One vezi sa entitetom Town preko atributa TownID.

PatientArrival		
ArrivalID	BIGSERIAL	Jedinstveni identifikator rekorda dolaska/odlaska pacijenta
PatientID	BIGSERIAL	Jedinstveni indentifikator pacijenta
TownID	SMALLINT	Grad u koji pacijent dolazi
dateOfArrival	DATETIME	Vrijeme i datum dolaska pacijenta
dateOfDeparture	DATETIME	Vrijeme i datum odlaska pacijenta

**Accommodation** Ovaj entitet sadrži podatke o smještaju, vrsti smještaja, njegovoj opremljenosti te adresi na kojoj se nalazi kao i koordinatama. Atributi su: AccomodationID (primary key),TypeID (foreign key), EquippedID (foreign key), latitude, longitude, address, TownID (foreign key), active i dateAdded. Ovaj je entitet u vezi Many-to-One sa entitetom Town preko atributa TownID, nadalje u vezi Many-to-One sa entitetom AccommodationType preko atributa TypeID, u vezi Many-to-One sa entitetom Equipped preko atributa EquippedID, u vezi One-to-One sa entitetom Patient preko atributa AccomodationID i PatientID.

Accommodation		
AccomodationID	BIGSERIAL	Jedinstveni brožani identifikator smještaja, automatski generiran
TypeID	INT	ID vrste smještaja
EquippedID	INT	ID opremljenosti smještaja
latitude	DECIMAL	Geografska širina
longitude	DECIMAL	Geografska dužina
address	VARCHAR	Adresa smještaja
TownID	INT	ID grada u kojem se smještaj nalazi
active	BIT	Je li smještaj upotrebljiv
dateAdded	DATE	Verijeme dodavanja smještaja u sustav

**AccommodationType** Ovaj entitet sadrži podatke o tipu smještaja (stan u zgradi, stan u kući, iznajmljeno, u vlasništvu klinike). Atributi su: TypeID (primary key), description. Ovaj je entitet u One-to-Many vezi s entitetom Accommodation preko

atributaTypeID.

AccommodationType		
TypeID	SMALLINT	Jedinstveni identifikator vrste smještaja, automatski generiran
description	TEXT	Opis vrste smještaja (stan u kući, stan u zgradi, soba u hotelu ili motelu)

**Equipped** Ovaj entitet sadrži podatke o vrsti opremljenosti smještaja (potpuno opremljen, djelomično opremljen). Atributi su: EquippedID (primary key), equipment. Ovaj je entitet u One-to-Many vezi sa entitetom Accommodation preko atributa EquippedID.

Equipped		
EquippedID	SMALLINT	Jedinstveni identifikator opremljenosti smještaja, automatski generiran
equipment	TEXT	Opis opremljenosti smještaja (potpuno opremljen, djelomično opremljen)

**AccommodationOccupied** Ovaj entitet sadrži podatke o zauzetosti smještaja kojima raspolaže pojedina klinika. Atributi su: PatientID (foreign key), AccommodationID (foreign key), dateTo, dateFrom. Ovaj je entitet u One-to-One vezi sa entitetom Accommodation preko atributa AccommodationID te u One-to-One vezi sa entiteom Patient preko atributa PatientID.

AccommodationOccupied		
ID	BIGSERIAL	Jedinstveni identifikator rekorda zauzeća smještaja
PatientID	BIGSERIAL	Jedinstveni identifikator pacijenta
AccommodationID	BIGSERIAL	Jedinstveni identifikator smještaja
dateFrom	DATE	Datum od kojeg je smještaj dostupan
dateTo	DATE	Datum do kojeg je smještaj dostupan



**Transporter** Ovaj entitet sadrži podatke o prijevoznicima s kojima klinika ima ugovore za prijevoz pacijenata. Atributi su: TransporterID (primary key), organisationName, phone, address, TownID (foreign key), active. Ovaj je entitet u Many-to-Many vezi sa entitetom Town preko atributa TownID, u Many-to-Many vezi sa entitetom Clinic preko atributa ClinicID te u One-to-Many vezi sa entitetom Vehicle preko atributa TransporterID.

Transporter		
TransporterID	BIGSERIAL	Jedinstveni identifikator prijevoznika, automatski generiran
organisationName	VARCHAR	Ime prijevoznika
phone	VARCHAR	Broj mobitela prijevoznika
address	VARCHAR	Adresa baze prijevoznika
TownID	INT	ID grada u kojem se smještaj nalazi
active	BIT	Je li prijevoznik radi

**Vehicle** Ovaj entitet sadrži podatke o vozilima kojima transporter raspolaže. Atributi su: VehicleID (primary key), capacity TypeID (foreign key), TransporterID (foreign key) te active. Ovaj je entitet u Many-to-One vezi sa entitetom Transporter preko atributa TransporterID, u Many-to-Many vezi sa entitetom VehicleOccupied preko atributa VehicleID, u One-to-One vezi sa entitetom VehicleType preko atributa VehicleID te u One-to-Many vezi sa entitetom VehicleAvailability preko atributa VehicleID.

Vehicle		
VehicleID	BIGSERIAL	Jedinstveni identifikator vozila, automatski generiran
capacity	SMALLINT	Kapacitet vozila (2 osobe, 4 osobe, 5 osoba...)
TypeID	SMALLINT	Vrsta vozila
TransporterID	INT	Jedinstveni identifikator prijevoznika kojem vozilo pripada

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

Vehicle		
active	BIT	Je li vozilo u funkciji

**VehicleType** Ovaj entitet sadrži podatke o vrsti vozila. Atributi su:TypeID(primary key), description. Ovaj je entitet u One-to-One vezi sa entitetom Vehicle preko atributa VehicleID.

VehicleType		
TypeID	SMALLINT	Jedinstveni identifikator vrste vozila, automatski generiran
description	TEXT	Opis vrste vozila (auto, kombi, min-bus...)

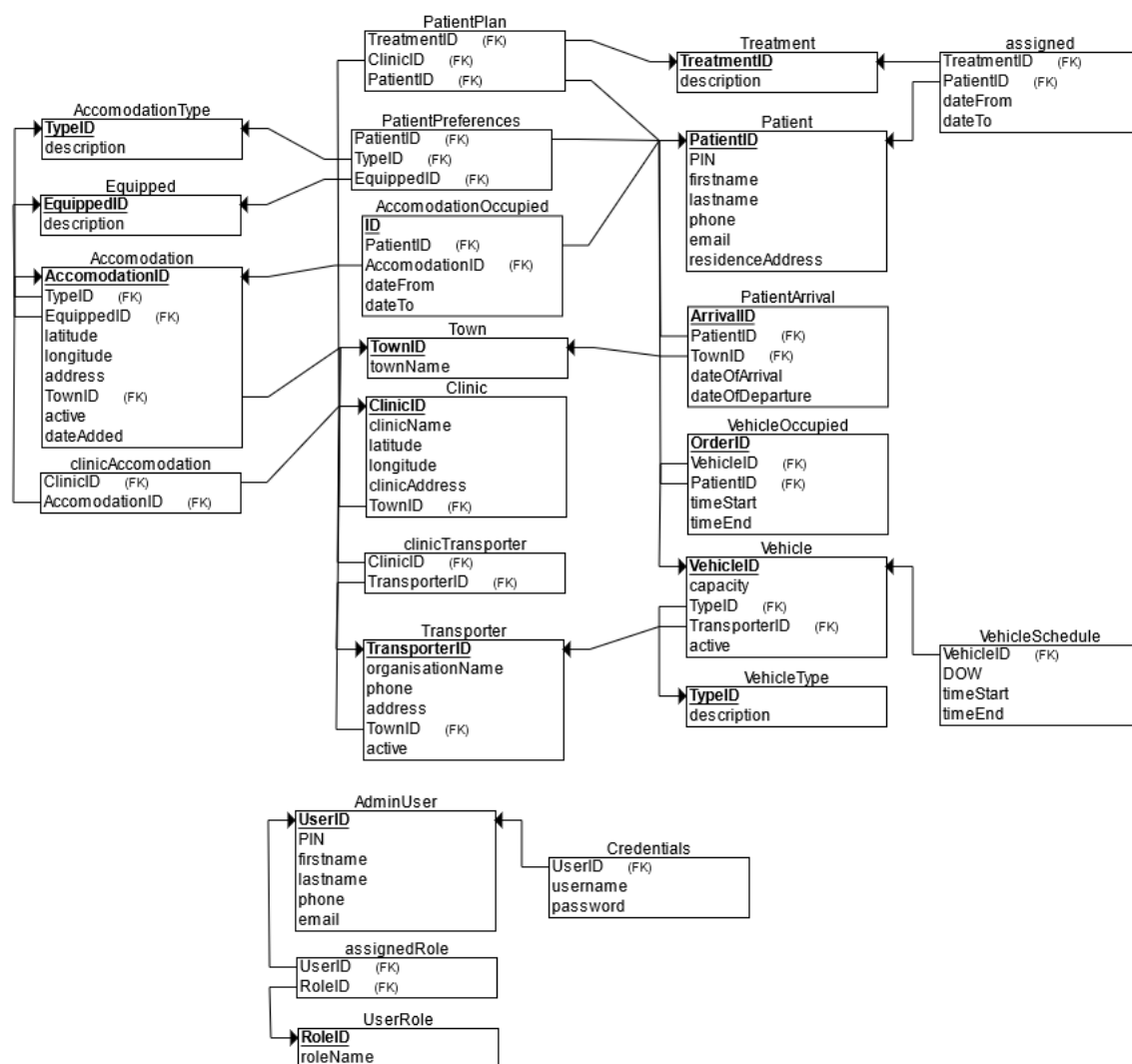
**VehicleOccupied** Ovaj entitet sadrži podatke o vremenima kada je koje vozilo zauzeto, tj. kada prevozi pacijente od smještaja do klinike te natrag. Atributi su: VehicleID (foreign key), PatientID (foreign key), timeStart, timeEnd. Ovaj je entitet u One-to-Many vezi sa entitetom Vehicle preko atributa VehicleID te u One-to-One vezi sa entitetom Patient preko atributa PatientID.

VehicleOccupied		
OrderID	BIGSERIAL	Jedinstveni identifikator rekorda zauzeća vozila
VehicleID	BIGSERIAL	Jedinstveni identifikator vozila koje je zauzeto
PatientID	BIGSERIAL	Jedinstveni identifikator pacijenta kojem je vozilo dodijeljeno
timeStart	TIME	Vrijem od kada je vozilo zauzeto
timeEnd	TIME	Vrijeme do kada je vozilo zauzeto

**VehicleSchedule** Ovaj entitet sadrži podatke o vremenima kada je koje vozilo dostupno, tj. radno vrijeme radnih dana. Atributi su: VehicleID (foreign key), DOW (day of the week), timeStart, timeEnd. Ovaj je entitet u Many-to-Many vezi sa entitetom Vehicle preko atributa VehicleID.

VehicleSchedule		
VehicleID	INT	Jedinstveni identifikator vozila
DOW	SMALLINT	Dan u tjednu u kojem je vozilo slobodno
timeStart	TIME	Vrijem od kada je vozilo slobodno
timeEnd	TIME	Vrijeme do kada je vozilo slobodno

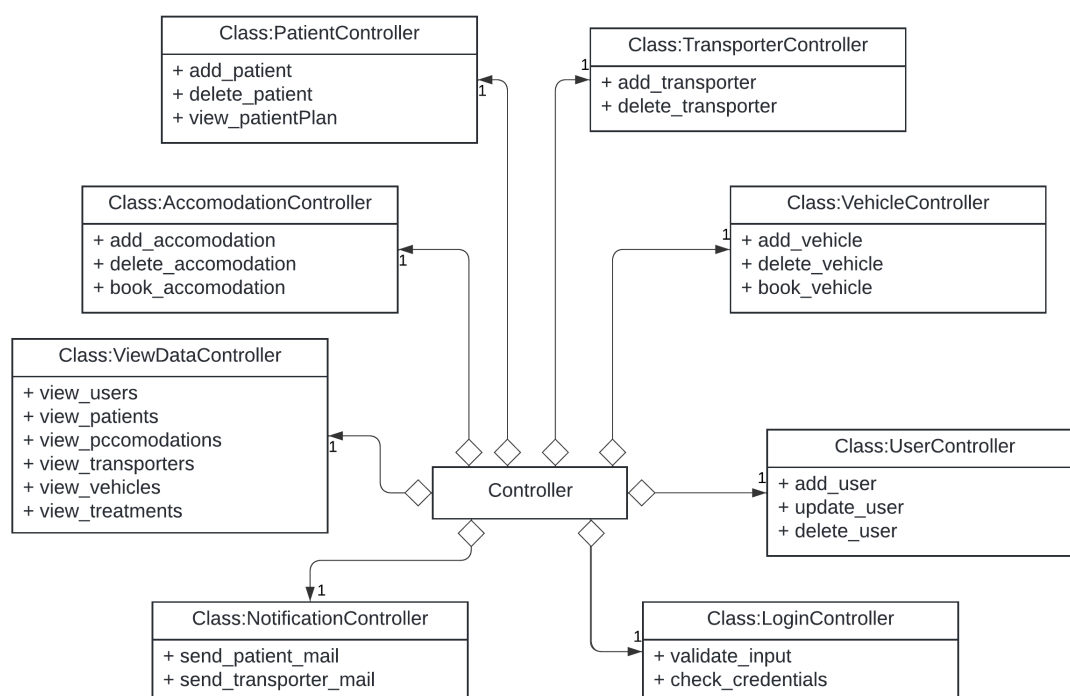
## 4.1.2 Dijagram baze podataka



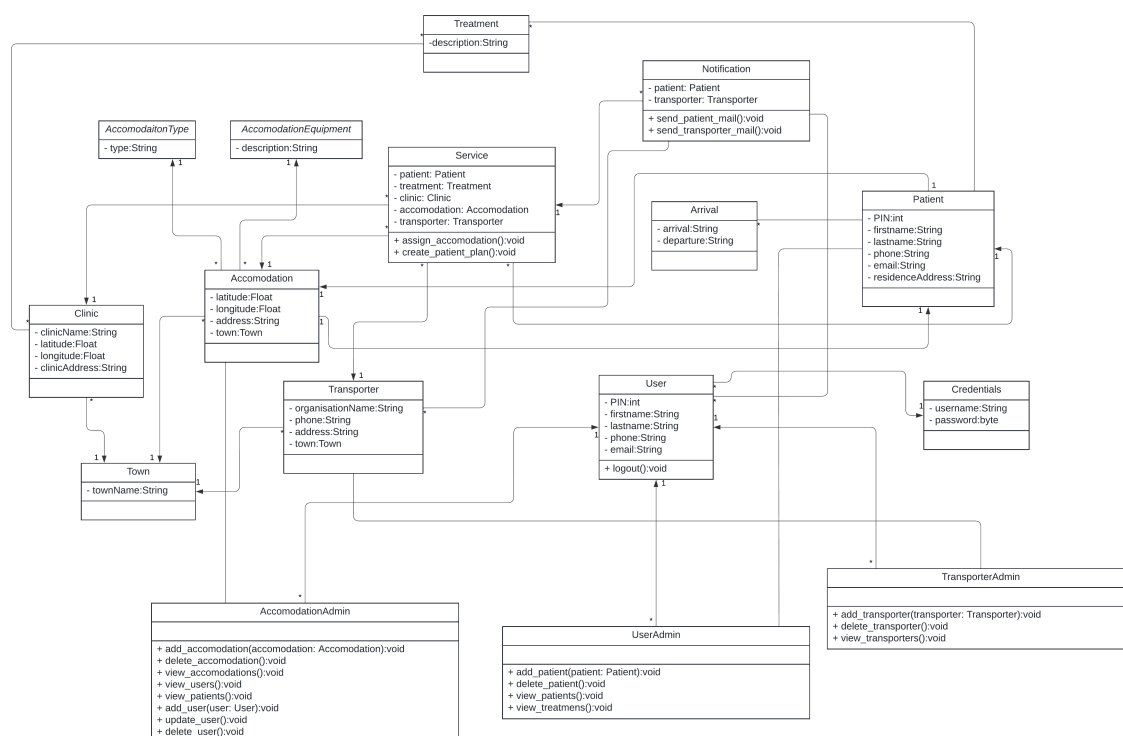
Slika 4.1: Sheme baze podataka

## 4.2 Dijagram razreda

Na slikama 4.2 i 4.3 su prikazani razredi *Model* i *Controller* iz MVC arhitekture. Razredi prikazani slikom 4.2 nasljeđuju razred *Controller*. Metode koje smo definirali unutar tih razreda pripremaju podatke i šalju ih bazi koja ih onda obrađuje. Baza manipulira modelima te na kraju vraća podatke kako bi ih *View* mogao prikazati. Model razredi, prikazani slikom 4.3, prikazuju strukturu baze podataka te integrirane funkcije koje služe za obradu, slanje ili primanje podataka.



Slika 4.2: Dijagram razreda Controller



Slika 4.3: Dijagram razreda Model

**dio 2. revizije**

*Prilikom druge predaje projekta dijagram razreda i opisi moraju odgovarati stvarnom stanju implementacije*

## 4.3 Dijagram stanja

### *dio 2. revizije*

*Potrebno je priložiti dijagram stanja i opisati ga. Dovoljan je jedan dijagram stanja koji prikazuje **značajan dio funkcionalnosti** sustava. Na primjer, stanja korisničkog sučelja i tijekom korištenja neke ključne funkcionalnosti jesu značajan dio sustava, a registracija i prijava nisu.*

## 4.4 Dijagram aktivnosti

### *dio 2. revizije*

*Potrebno je priložiti dijagram aktivnosti s pripadajućim opisom. Dijagram aktivnosti treba prikazivati značajan dio sustava.*



## 4.5 Dijagram komponenti

### *dio 2. revizije*

*Potrebno je priložiti dijagram komponenti s pripadajućim opisom. Dijagram komponenti treba prikazivati strukturu cijele aplikacije.*

## 5. Implementacija i korisničko sučelje

### 5.1 Korištene tehnologije i alati

#### *dio 2. revizije*

*Detaljno navesti sve tehnologije i alate koji su primijenjeni pri izradi dokumentacije i aplikacije. Ukratko ih opisati, te navesti njihovo značenje i mjesto primjene. Za svaki navedeni alat i tehnologiju je potrebno **navesti internet poveznicu** gdje se mogu preuzeti ili više saznati o njima.*

## 5.2 Ispitivanje programskog rješenja

### *dio 2. revizije*

*U ovom poglavlju je potrebno opisati provedbu ispitivanja implementiranih funkcionalnosti na razini komponenti i na razini cijelog sustava s prikazom odabranih ispitnih slučajeva. Studenti trebaju ispitati temeljnu funkcionalnost i rubne uvjete.*

### 5.2.1 Ispitivanje komponenti

*Potrebno je provesti ispitivanje jedinica (engl. unit testing) nad razredima koji implementiraju temeljne funkcionalnosti. Razraditi **minimalno 6 ispitnih slučajeva** u kojima će se ispitati redovni slučajevi, rubni uvjeti te izazivanje pogreške (engl. exception throwing). Poželjno je stvoriti i ispitni slučaj koji koristi funkcionalnosti koje nisu implementirane. Potrebno je priložiti izvorni kôd svih ispitnih slučajeva te prikaz rezultata izvođenja ispita u razvojnom okruženju (prolaz/pad ispita).*

### 5.2.2 Ispitivanje sustava

*Potrebno je provesti i opisati ispitivanje sustava koristeći radni okvir Selenium<sup>1</sup>. Razraditi **minimalno 4 ispitna slučaja** u kojima će se ispitati redovni slučajevi, rubni uvjeti te poziv funkcionalnosti koja nije implementirana/izaziva pogrešku kako bi se vidjelo na koji način sustav reagira kada nešto nije u potpunosti ostvareno. Ispitni slučaj se treba sastojati od ulaza (npr. korisničko ime i lozinka), očekivanog izlaza ili rezultata, koraka ispitivanja i dobivenog izlaza ili rezultata.*

*Izradu ispitnih slučajeva pomoću radnog okvira Selenium moguće je provesti pomoću jednog od sljedeća dva alata:*

- *dodatak za preglednik **Selenium IDE** - snimanje korisnikovih akcija radi automatskog ponavljanja ispita*
- ***Selenium WebDriver** - podrška za pisanje ispita u jezicima Java, C#, PHP koristeći posebno programsko sučelje.*

*Detalji o korištenju alata Selenium bit će prikazani na posebnom predavanju tijekom semestra.*

---

<sup>1</sup><https://www.seleniumhq.org/>

## 5.3 Dijagram razmještaja

### *dio 2. revizije*

Potrebno je umetnuti **specifikacijski** dijagram razmještaja i opisati ga. Moguće je umjesto specifikacijskog dijagrama razmještaja umetnuti dijagram razmještaja instanci, pod uvjetom da taj dijagram bolje opisuje neki važniji dio sustava.

## 5.4 Upute za puštanje u pogon

### *dio 2. revizije*

*U ovom poglavlju potrebno je dati upute za puštanje u pogon (engl. deployment) ostvarene aplikacije. Na primjer, za web aplikacije, opisati postupak kojim se od izvornog kôda dolazi do potpuno postavljene baze podataka i poslužitelja koji odgovara na upite korisnika. Za mobilnu aplikaciju, postupak kojim se aplikacija izgradi, te postavi na neku od trgovina. Za stolnu (engl. desktop) aplikaciju, postupak kojim se aplikacija instalira na računalo. Ukoliko mobilne i stolne aplikacije komuniciraju s poslužiteljem i/ili bazom podataka, opisati i postupak njihovog postavljanja. Pri izradi uputa preporučuje se **naglasiti korake instalacije uporabom natuknica** te koristiti što je više moguće **slike ekrana** (engl. screenshots) kako bi upute bile jasne i jednostavne za slijediti.*

*Dovršenu aplikaciju potrebno je pokrenuti na javno dostupnom poslužitelju. Studentima se preporuča korištenje neke od sljedećih besplatnih usluga: Amazon AWS, Microsoft Azure ili Heroku. Mobilne aplikacije trebaju biti objavljene na F-Droid, Google Play ili Amazon App trgovini.*

## 6. Zaključak i budući rad

### *dio 2. revizije*

*U ovom poglavlju potrebno je napisati osvrt na vrijeme izrade projektnog zadatka, koji su tehnički izazovi prepoznati, jesu li riješeni ili kako bi mogli biti riješeni, koja su znanja stečena pri izradi projekta, koja bi znanja bila posebno potrebna za brže i kvalitetnije ostvarenje projekta i koje bi bile perspektive za nastavak rada u projektnoj grupi.*

*Potrebno je točno popisati funkcionalnosti koje nisu implementirane u ostvarenoj aplikaciji.*

# Popis literature

1. Programsko inženjerstvo, FER ZEMRIS, <http://www.fer.hr/predmet/proinz>
2. Astah Community, <http://astah.net/editions/uml-new>

# Indeks slika i dijagrama

2.1	Mogućnosti povezivanja klijenata i pružatelja koje nudi web stranica "MedicalTourism.com" . . . . .	5
3.1	Dijagram obrasca uporabe, osnovne funkcionalnosti . . . . .	20
3.2	Dijagram obrasca uporabe, funkcionalnosti upravljanja korisnicima . . . . .	21
3.3	Dijagram obrasca uporabe, funkcionalnosti upravljanja smještajima . . . . .	22
3.4	Dijagram obrasca uporabe, funkcionalnosti upravljanja prijevoznima . . . . .	23
3.5	Dijagram obrasca uporabe, funkcionalnosti upravljanja vozilima . . . . .	24
3.6	Dijagram obrasca uporabe, funkcionalnosti upravljanja pacijentima . . . . .	25
3.7	Dijagram obrasca uporabe, pridjeljivanje i obavješćavanje . . . . .	26
3.8	Sekvencijski dijagram za UC2 . . . . .	27
3.9	Sekvencijski dijagram za UC19 . . . . .	28
3.10	Sekvencijski dijagram za UC21 . . . . .	29
3.11	Sekvencijski dijagram za UC22 . . . . .	30
4.1	Sheme baze podataka . . . . .	43
4.2	Dijagram razreda Controller . . . . .	44
4.3	Dijagram razreda Model . . . . .	45



# Dodatak: Prikaz aktivnosti grupe

## Dnevnik sastajanja

### 1. sastanak

- Datum:18.10.2023.
- Prisustvovali: Karlo Baljak, Luka Kokić, Ian Marković, Mateo Martić, Mislav Matić, Bruno Milaković, Teo Musa
- Teme sastanka:
  - Međusobno upoznavanje
  - Rasprava o projektnom zadatku

### 2. sastanak

- Datum:25.10.2023.
- Prisustvovali: Karlo Baljak, Luka Kokić, Ian Marković, Mateo Martić, Mislav Matić, Bruno Milaković, Teo Musa
- Teme sastanka:
  - Podjela zadataka na projektu
  - Rasprava o alatima koje ćemo koristiti

### 3. sastanak

- Datum:10.11.2023.
- Prisustvovali: Karlo Baljak, Luka Kokić, Ian Marković, Mateo Martić, Mislav Matić, Bruno Milaković, Teo Musa
- Teme sastanka:
  - Rasprava o finalizaciji finijih detalja vezano uz obrasce uporabe i dijagrame te njihova finalizacija
  - Raspodjela zadataka koji trebaju biti obavljeni do prve predaje

## Tablica aktivnosti

	Luka Kokić	Karlo Baljak	Ian Marković	Mateo Martić	Mislav Matić	Bruno Milaković	Teo Musa
Upravljanje projektom	7						
Opis projektnog zadatka			4				
Funkcionalni zahtjevi			3				
Opis pojedinih obrazaca			8				8
Dijagram obrazaca					7		
Sekvencijski dijagrami						11	
Opis ostalih zahtjeva			1				
Arhitektura i dizajn sustava		3					
Baza podataka		15					
Dijagram razreda		5		5			
Dijagram stanja							
Dijagram aktivnosti							
Dijagram komponenti							
Korištene tehnologije i alati							
Ispitivanje programskog rješenja							
Dijagram razmještaja							
Upute za puštanje u pogon							
Dnevnik sastajanja	1						
Zaključak i budući rad							
Popis literature					1		

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

	<b>Luka Kokić</b>	<b>Karlo Baljak</b>	<b>Ian Marković</b>	<b>Mateo Martić</b>	<b>Mislav Matić</b>	<b>Bruno Milaković</b>	<b>Teo Musa</b>
Ukupno dokumentacija	8	23	16	5	8	11	8
Sastanci	14	14	14	12	12	12	12
Istraživanje informacija i tehnologija	10	13	5	4	4	7	4
Deployment		10					
Izrada baze podataka		7					
Spajanje s bazom podataka		5					
Backend	2	10					
Frontend	3	5					
Ukupno razvoj projekta	37	87	35	21	24	30	24

## Dijagrami pregleda promjena

### *dio 2. revizije*

*Prenijeti dijagram pregleda promjena nad datotekama projekta. Potrebno je na kraju projekta generirane grafove s gitlaba prenijeti u ovo poglavlje dokumentacije. Dijagrami za vlastiti projekt se mogu preuzeti s [gitlab.com](https://gitlab.com) stranice, u izborniku Repository, pritiskom na stavku Contributors.*