

# Programsko inženjerstvo

Ak. god. 2023./2024.

## DentAll

Dokumentacija, *Rev. 1*

Grupa: *Tintilinci*

Voditelj: *Neven Lukić*

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>5</b>
<b>3 Specifikacija programske potpore</b>	<b>9</b>
3.1 Funkcionalni zahtjevi . . . . .	9
3.1.1 Obrasci uporabe . . . . .	11
3.1.2 Sekvencijski dijagrami . . . . .	25
3.2 Ostali zahtjevi . . . . .	32
<b>4 Arhitektura i dizajn sustava</b>	<b>33</b>
4.1 Baza podataka . . . . .	34
4.1.1 Opis tablica . . . . .	34
4.1.2 Dijagram baze podataka . . . . .	40
4.2 Dijagram razreda . . . . .	41
4.3 Dijagram stanja . . . . .	44
4.4 Dijagram aktivnosti . . . . .	45
4.5 Dijagram komponenti . . . . .	46
<b>5 Implementacija i korisničko sučelje</b>	<b>47</b>
5.1 Korištene tehnologije i alati . . . . .	47
5.2 Ispitivanje programskog rješenja . . . . .	48
5.2.1 Ispitivanje komponenti . . . . .	48
5.2.2 Ispitivanje sustava . . . . .	48
5.3 Dijagram razmještaja . . . . .	49
5.4 Upute za puštanje u pogon . . . . .	50
<b>6 Zaključak i budući rad</b>	<b>51</b>
<b>Popis literature</b>	<b>52</b>
<b>Indeks slika i dijagrama</b>	<b>53</b>

**Dodatak: Prikaz aktivnosti grupe**

**54**

# 1. Dnevnik promjena dokumentacije

## *Kontinuirano osvježavanje*

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napravljen predložak.	Neven Lukić	26.10.2023.
0.2	Započela opis projektnog zadatka.	Karla Šmuk	01.11.2023.
0.3	Dodani članovi tima i prva dva sastanaka.	Neven Lukić	1.11.2023.
0.3.1	Proširen opis projektnog zadatka. Dodani funkcionalni zahtjevi.	Karla Šmuk	04.11.2023.
0.4	Dodani <i>Use Case</i> dijagrami	Karla Šmuk	16.11.2023.
0.5	Dodan opis baze podataka, svih tablica i ER dijagram baze podataka.	Neven Lukić	16.11.2023.
0.6	Dodani opisi obrazaca upotrebe.	Karla Pišonić	16.11.2023.
0.7	Dodani ostali zahtjevi.	Filip Buljan	16.11.2023.
0.8	Dodani dijagrami razreda.	Roko Gligora	16.11.2023.
0.9	Dodani sekvencijski dijagrami. Promijenjeni neki dijelovi opisa projektnog zadatka.	Karla Šmuk	17.11.2023.

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

Rev.	Opis promjene/dodatka	Autori	Datum
1.0	Verzija samo s bitnim dijelovima za 1. ciklus	Neven Lukić	17.11.2013.

## 2. Opis projektnog zadatka

Cilj ovog projekta je razviti programsku potporu za stvaranje web aplikacije “DentAll” koja će omogućiti učinkovito upravljanje smještajem i prijevozom korisnika zdravstvenog turizma.

Porastom zdravstvenog turizma, zdravstvene ustanove trude se privući korisnike nudeći im cjelovite usluge, uključujući smještaj i prijevoz. U mnogim slučajevima skuplje zdravstvene usluge u državama od kuda strani korisnici dolaze, potiču potrebu za pretragom usluga u drugim državama. Međutim, unatoč pristupačnijim troškovima medicinske usluge, korisnici se suočavaju s mnogim drugim izazovima koji obeshrabruju njihovu odluku za potragom medicinske usluge izvan svoje države. Spomenuti problemi su troškovi putovanja, udaljenost, osjećaj nesigurnosti i nedostatak poznavanja destinacije u kojoj se zdravstvena usluga nudi.

Naglasak se sve više stavlja na potrebu razvoja rješenja koje će omogućiti učinkovitu, brzu i jednostavnu koordinaciju smještaja i prijevoza. S obzirom na sve izazove koje bi korisnik trebao proći da se odluči za zdravstvenu uslugu u inozemstvu, zdravstvenim ustanovama nije dovoljno imati samo financijsku prednost već i mnoge druge. Ideja o izradi aplikacije za pomoć potencijalnim korisnicima usluga zdravstvenog turizma u pronalasku smještaja i prijevoza je ključna. Nije dovoljno privući korisnike samo povoljnim cijenama, nego je bitno pružiti im sigurnosti i udobnost tijekom boravka na novoj destinaciji. Organizacija smještaja i prijevoza uvelike bi povećala atraktivnost zdravstvenog turizma. Ovaj pristup omogućio bi korisnicima da se bolje informiraju i pripreme za njihovu medicinsku uslugu, bez potrebe za brigom o putovanju i smještaju. Organizacija smještaja i prijevoza do zdravstvenih ustanova korisnicima bi uzrokovala minimalan stres i smanjivala njihovu izgubljenost. To je ključni korak u motiviranju potencijalnih korisnika da se odluče za ovu uslugu.

Od ovakve aplikacije koristi bi imali korisnici, zdravstvene ustanove, prijevoznici te iznajmljivači smještaja. Aplikacija bi zdravstvenim ustanovama omogućila cjelovite usluge pacijentima povećavajući privlačnost njihove ponude. Prijevoznicima i iznajmljivačima smještaja pomogla bi u upravljanju svojim kapacitetima i vožnjama. Najbitniji korisnici imali bi najbolje moguće iskustvo jer bi im bilo

olakšano rezerviranje smještaja i prijevoza.

U aplikaciji postoje tri uloge korisnika:

- smještajni administrator
- administrator prijevoznih usluga
- korisnički administrator

Ulaskom u aplikaciju neprijavljeni korisnik dolazi na početnu stranicu. Može se odlučiti za prijavu u sustav te odabrati opciju “Prijava”. Korisnika se preusmjerava na stranicu za prijavu te tamo upisuje svoje podatke e-mail i lozinku. Dalje ga se preusmjerava ovisno o njegovoj ulozi ili ulogama.

### **Smještajni administrator**

Preusmjerava ga se na stranicu liste svih unesenih smještaja. Odabirom “Dodaj novi smještaj” može stvoriti novi smještaj. Također pritiskom na određeni smještaj preusmjerava ga se na stranicu detalja o tom smještaju. Tu se može vidjeti i prikaz smještaja na geografskoj karti. Odabirom “Uredi podatke” administrator može uređivati osnovne podatke te obrisati smještaj odabirom “Ukloni ovaj smještaj” nakon što dobije poruku potvrde kojom potvrđuje svoju odluku. Za kreiranje smještaja potrebni su podaci:

- adresa smještaja
- vrsta smještaja
- kategorija smještaja
- vremenski period dostupnosti ( dostupno od, dostupno do)
- lokacija ( u koordinatama ) za grafički prikaz geografskog položaja na karti

Ima najveće ovlasti te može definirati druge korisnike te im dodjeljivati različite uloge ( jedan korisnik može imati i više uloga ). Pritiskom na “Upravljanje administratorima” preusmjerava ga se na stranicu s listom svih administratora. Za kreiranje administratora potrebni su podaci:

- ime
- prezime
- e-mail
- uloga/e

Opcijom “Dodaj novog administratora” može unositi nove korisnike. Odabirom opcije “Promijeni ulogu” korisniku se može dodati nova uloga ili promijeniti postojeća. Također, odabirom “Izbriši administratora”, administrator ima mogućnost obrisati odabranog korisnika nakon što dobije poruku potvrde kojom potvrđuje svoju odluku.

### **Administrator prijevoznih usluga**

Preusmjerava ga se na stranicu liste svih unesenih prijevoznika. Odabirom “Dodaj novog prijevoznika” može stvoriti novog prijevoznika. Za kreiranje prijevoznika potrebni su podaci:

- naziv prijevoznika
- e-mail
- broj telefona

Također pritiskom na određenog prijevoznika preusmjerava ga se na stranicu koja prikazuje listu unesenih transportnih vozila za tog prijevoznika. Za kreiranje vozila potrebni su podaci:

- tip vozila
- kapacitet

Odabirom “Uredi podatke” administrator može uređivati osnovne podatke. Osim toga, odabirom opcije “Ukloni vozilo”, administrator ima mogućnost izbrisati odabrano vozilo nakon što dobije poruku potvrde kojom potvrđuje svoju odluku.

### **Korisnički administrator**

Preusmjerava ga se na stranicu liste unesenih pacijenata. Odabirom opcije “Dodaj novog pacijenta” omogućuje se unos novih pacijenata. Za kreiranje korisnika potrebni su podaci:

- ime
- prezime
- PIN
- broj telefona
- e-mail
- datum i vrijeme dolaska i odlaska



- tip smještaja
- kategoriju smještaja

Na kraju završenog korisnik se može odjaviti pritiskom na gumb “Odjava” te se vraća na početnu stranicu.

Pacijent nema direktnu vezu s aplikacijom. Detalji o tretmanima ne unose se ručno u aplikaciju, već postoji umjetno ispitno sučelje koje komunicira s aplikacijom za evidenciju medicinskih usluga za koju se pretpostavlja da je već razvijena.

Nakon unosa novog pacijenta i potrebnih informacija aplikacija dodjeljuje raspoloživi smještaj pacijentu te je označuje kao zauzetu u periodu njegove medicinske usluge. Aplikacija periodički provjerava status medicinske usluge komunikacijom s aplikacijom medicinskih usluga. Ako aplikacija primi odgovor od aplikacije medicinskih usluga da je plan medicinskih usluga zaključan, to znači da su svi medicinski termini pacijenata potvrđeni te se tada može početi organizirati prijevoz za te termine. Potrebno je i označiti dodijeljene prijevoznike zauzetima u tim terminima.

Nakon završetka ukupnog plana puta, pacijentu će biti poslan e-mail o detaljima plana puta. Također, poruke će biti poslane i svakom prijevozniku s kontaktnim podacima korisnika te podacima o vremenima i adresama smještaja.

Jedna od mogućih nadogradnji ovoga zadatka bila bi dodavanje pristupa korisničkim iskustvima, točnije, omogućiti ocjenjivanje smještaja i prijevoza te pisanje recenzija kako bi korisnici mogli što bolje procijeniti kvalitetu usluga.

## 3. Specifikacija programske potpore

### 3.1 Funkcionalni zahtjevi

#### Dionici:

1. Vlasnik ( naručitelj )
2. Korisnici
  - (a) Smještajni administrator
  - (b) Administrator prijevoznih usluga
  - (c) Korisnički administrator
3. Razvojni tim

#### Aktori i njihovi funkcionalni zahtjevi:

1. Neprijavljeni korisnik (inicijator) može:
  - (a) se prijaviti u sustav za što su mu potrebni e-mail i lozinka
2. Smještajni administrator ( inicijator ) može:
  - (a) unositi nove smještaje
  - (b) mijenjati osnovne podatke smještaja
  - (c) brisati smještaje
  - (d) unositi nove korisnike
  - (e) dodjeljivati uloge korisnicima
3. Administrator prijevoznih usluga ( inicijator ) može:
  - (a) unositi nove prijevoznike
  - (b) mijenjati podatke prijevoznika
  - (c) brisati prijevoznike
  - (d) unositi vozila za prijevoznike
  - (e) brisati vozila
4. Korisnički administrator ( inicijator ) može:

- (a) unositi pacijente

5. Baza podataka ( sudionik ) može:

- (a) pohranjuje sve podatke o korisnicima i njihovim ovlastima
- (b) pohranjuje sve podatke o pacijentu
- (c) pohranjuje sve podatke o medicinskoj usluzi
- (d) pohranjuje sve podatke o smještaju
- (e) pohranjuje sve podatke o prijevozniku i vozilima
- (f) pohranjuje sve podatke o rezerviranju smještaja
- (g) pohranjuje sve podatke o rezerviranju prijevoznika

6. Aplikacija medicinske usluge ( sudionik ):

- (a) odgovorna za prikupljanje podatak o medicinskim tretmanima te naša aplikacija putem sučelja dohvaća podatke o tretmanima korisnika

### 3.1.1 Obrasci uporabe

#### UC1 - prijava

- **Glavni sudionik:** neprijavljeni korisnik
- **Cilj:** dobiti pristup korisničkom sučelju
- **Sudionici:** baza podataka
- **Opis osnovnog tijeka:**
  1. Na zahtjev se otvara stranica administratoru
  2. Administrator prenosi podatke korisnika u sustav
  3. Pohranjuju se promjene u bazi podataka

#### UC2 - odjava

- **Glavni sudionik:** administrator
- **Cilj:** odjaviti korisnika iz sustava
- **Sudionici:** baza podataka
- **Preduvjet:** korisnik mora postojati u sustavu, odnosno biti već u bazi podataka; administrator mora biti prijavljen u sustav
- **Opis osnovnog tijeka:**
  1. Administrator odabire opciju odjave korisnika iz sustava
  2. Korisnik je odjavljen iz sustava i pohranjene su promjene

#### UC3 - pregled svih administratora

- **Glavni sudionik:** smještajni administrator
- **Cilj:** dobiti uvid u podatke svih administratora
- **Sudionici:** baza podataka
- **Preduvjet:** administrator je prijavljen i dodijeljena mu je uloga smještajnog administratora
- **Opis osnovnog tijeka:**
  1. Smještajni administrator odabire opciju pregleda svih administratora kojima je on nadležan - korisničkih administratora i administratora prijevoznih usluga
  2. Otvara se popis svih administratora

#### UC4 - unos novog administratora

- **Glavni sudionik:** smještajni administrator
- **Cilj:** dodati administratora

- **Sudionici:** baza podataka
- **Preduvjet:** administrator je prijavljen i dodijeljena mu je uloga smještajnog administratora
- **Opis osnovnog tijeka:**
  1. Smještajni administrator odabere opciju dodavanja novog administratora
  2. Otvara se prozor za unos podataka
  3. Smještajni administrator upiše podatke o novom administratoru
  4. U bazu podataka se pohrani promjena
- **Opis mogućih odstupanja:**
  - 3.a Upisano je neispravno korisničko ime administratora
    1. Sustav obavještava smještajnog administratora o neuspjelom upisu

#### UC5 - brisanje administratora

- **Glavni sudionik:** smještajni administrator
- **Cilj:** obrisati administratora
- **Sudionici:** baza podataka
- **Preduvjet:** administrator je prijavljen i dodijeljena mu je uloga smještajnog administratora
- **Opis osnovnog tijeka:**
  1. Smještajni administrator odabere opciju brisanja administratora
  2. Otvara se prozor za unos podataka
  3. Smještajni administrator upiše podatke o administratoru
  4. Smještajni administrator odabere opciju 'Izbriši'
  5. Administrator se uklanja iz baze podataka

#### UC6 - dodjeljivanje uloga

- **Glavni sudionik:** smještajni administrator
- **Cilj:** dodijeliti ulogu administratoru
- **Sudionici:** baza podataka
- **Preduvjet:** administrator je prijavljen i dodijeljena mu je uloga smještajnog administratora
- **Opis osnovnog tijeka:**
  1. Smještajni administrator otvara popis svih administratora
  2. Smještajni administrator odabere administratora kojem želi promijeniti ulogu

3. Otvori se prozor s podacima administratora
  4. Smještajni administrator odabere jednu od ponuđenih uloga (korisnički administrator ili administrator prijevoznih usluga)
  5. U bazu podataka se pohrani promjena
- **Opis mogućih odstupanja:**
    - 3.a Upisano je neispravno korisničko ime administratora
      1. Sustav obavještava smještajnog administratora o neuspjelom dodjeljivanju uloge administratoru

#### UC7 - pregled svih smještaja

- **Glavni sudionik:** smještajni administrator
- **Cilj:** pregledati sve smještaje
- **Sudionici:** baza podataka
- **Preduvjet:** administrator je prijavljen i dodijeljena mu je uloga smještajnog administratora, postoji barem jedan smještaj unesen u sustav
- **Opis osnovnog tijeka:**
  1. Administrator odabere opciju pregleda svih smještaja
  2. Prikaže se popis svih smještaja koji postoje u sustavu

#### UC8 - pregled smještaja

- **Glavni sudionik:** smještajni administrator
- **Cilj:** pregledati traženi smještaj
- **Sudionici:** baza podataka
- **Preduvjet:** administrator je prijavljen i dodijeljena mu je uloga smještajnog administratora
- **Opis osnovnog tijeka:**
  1. Administrator odabere opciju pregleda svih smještaja
  2. Prikaže se popis svih smještaja koji postoje u sustavu
  3. Administrator odabere iz popisa traženi smještaj

#### UC9 - unos novog smještaja

- **Glavni sudionik:** smještajni administrator
- **Cilj:** dodati novi smještaj
- **Sudionici:** baza podataka
- **Preduvjet:** administrator je prijavljen i dodijeljena mu je uloga smještajnog administratora

- **Opis osnovnog tijeka:**
  1. Administrator odabere opciju dodavanja smještaja
  2. Otvara se prozor za unos podataka o novom smještaju
  3. Administrator unese podatke o smještaju
  4. U bazu podataka se pohrani promjena
- **Opis mogućih odstupanja:**
  - 3.a Upisano je neispravno ime smještaja
    1. Sustav obavještava smještajnog administratora o neuspjelom upisu

#### UC10 - izmjena podataka o smještaju

- **Glavni sudionik:** smještajni administrator
- **Cilj:** izmijeniti podatke o smještaju
- **Sudionici:** baza podataka
- **Preduvjet:** administrator je prijavljen i dodijeljena mu je uloga smještajnog administratora
- **Opis osnovnog tijeka:**
  1. Administrator odabere opciju pregleda svih smještaja
  2. Prikaže se popis svih smještaja koji postoje u sustavu
  3. Administrator odabere iz popisa traženi smještaj
  4. Otvori se stranica s podacima o smještaju
  5. Administrator izmijeni podatke o smještaju
  6. Administrator odabere opciju 'Spremi promjene'
  7. U bazu podataka se pohrani promjena
- **Opis mogućih odstupanja:**
  - 3.a Traženi smještaj ne postoji u sustavu
    1. Sustav obavještava smještajnog administratora da smještaj ne postoji u sustavu

#### UC11 - brisanje smještaja

- **Glavni sudionik:** smještajni administrator
- **Cilj:** obrisati smještaj
- **Sudionici:** baza podataka
- **Preduvjet:** administrator je prijavljen i dodijeljena mu je uloga smještajnog administratora
- **Opis osnovnog tijeka:**
  1. Administrator odabere opciju brisanja smještaja

2. Otvara se prozor za unos podataka
3. Administrator unese ime smještaja i obriše ga
4. Smještaj se uklanja iz baze podataka

### UC12 - prikaz smještaja na karti

- **Glavni sudionik:** smještajni administrator
- **Cilj:** prikazati smještaj na karti
- **Sudionici:** OpenMaps
- **Preduvjet:** administrator je prijavljen i dodijeljena mu je uloga smještajnog administratora
- **Opis osnovnog tijeka:**
  1. Administrator odabire opciju pregleda svih smještaja
  2. Prikaže se popis svih smještaja koji postoje u sustavu
  3. Administrator odabere iz popisa traženi smještaj
  4. Administrator odabere opciju prikaza smještaja na karti
  5. Otvori se prozor gdje se vidi lokacija smještaja na karti
- **Opis mogućih odstupanja:**
  - 3.a Traženi smještaj ne postoji u sustavu
    1. Sustav obavještava smještajnog administratora da smještaj ne postoji u sustavu
  - 4.a Smještaj se ne može locirati na karti
    1. Sustav obavještava smještajnog administratora da smještaj nije moguće locirati

### UC13 - pregled prijevoznika

- **Glavni sudionik:** administrator prijevoznih usluga
- **Cilj:** pregledati sve prijevoznike
- **Sudionici:** baza podataka
- **Preduvjet:** administrator je prijavljen i dodijeljena mu je uloga administratora prijevoznih usluga; postoji barem jedan prijevoznik u sustavu
- **Opis osnovnog tijeka:**
  1. Administrator odabere opciju pregleda svih prijevoznika
  2. Prikaže se popis svih prijevoznika
  3. Administrator odabere traženog prijevoznika

### UC14 - izmjena podataka prijevoznika



- **Glavni sudionik:** administrator prijevoznih usluga
- **Cilj:** izmijeniti podatke traženog administratora
- **Sudionici:** baza podataka
- **Preduvjet:** administrator je prijavljen i dodijeljena mu je uloga administratora prijevoznih usluga; postoji barem jedan prijevoznik u sustavu
- **Opis osnovnog tijeka:**
  1. Administrator odabire opciju pregleda svih prijevoznika
  2. Prikaže se popis svih prijevoznika koji postoje u sustavu
  3. Administrator odabere iz popisa traženog prijevoznika
  4. Otvori se stranica s podacima prijevoznika
  5. Administrator izmijeni podatke o prijevozniku
  6. Administrator odabere opciju 'Spremi promjene'
  7. U bazu podataka se pohrani promjena
- **Opis mogućih odstupanja:**
  - 3.a Traženi prijevoznik ne postoji u sustavu
    1. Sustav obavještava administratora da odabrani prijevoznik ne postoji u sustavu

#### UC15 - unos novog vozila

- **Glavni sudionik:** administrator prijevoznih usluga
- **Cilj:** unijeti podatke o novom vozilu
- **Sudionici:** baza podataka
- **Preduvjet:** administrator je prijavljen i dodijeljena mu je uloga administratora prijevoznih usluga
- **Opis osnovnog tijeka:**
  1. Administrator odabere opciju dodavanja novog vozila
  2. Otvara se prozor za unos podataka o novom vozilu
  3. Administrator unese podatke o vozilu
  4. U bazu podataka se pohrani promjena

#### UC16 - brisanje vozila

- **Glavni sudionik:** administrator prijevoznih usluga
- **Cilj:** obrisati vozilo
- **Sudionici:** baza podataka
- **Preduvjet:** administrator je prijavljen i dodijeljena mu je uloga administratora prijevoznih usluga

- **Opis osnovnog tijeka:**

1. Administrator odabere opciju brisanja vozila
2. Otvara se prozor za unos podataka
3. Administrator unese ime vozila i obriše ga
4. Vozilo se uklanja iz baze podataka

#### UC17 - pregled svih prijevoznika

- **Glavni sudionik:** administrator prijevoznih usluga
- **Cilj:** dobiti uvid u sve prijevoznike
- **Sudionici:** baza podataka
- **Preduvjet:** administrator je prijavljen i dodijeljena mu je uloga administratora prijevoznih usluga
- **Opis osnovnog tijeka:**
  1. Administrator odabire opciju pregleda svih prijevoznika
  2. Prikaže se popis svih prijevoznika

#### UC18 - brisanje prijevoznika

- **Glavni sudionik:** administrator prijevoznih usluga
- **Cilj:** obrisati prijevoznika
- **Sudionici:** baza podataka
- **Preduvjet:** administrator je prijavljen i dodijeljena mu je uloga administratora prijevoznih usluga
- **Opis osnovnog tijeka:**
  1. Administrator odabire opciju brisanja prijevoznika
  2. Otvara se prozor za unos podataka
  3. Administrator unese ime prijevoznika i obriše ga
  4. Prijevoznik se uklanja iz baze podataka

#### UC19 - unos novog prijevoznika

- **Glavni sudionik:** administrator prijevoznih usluga
- **Cilj:** unijeti novog prijevoznika
- **Sudionici:** baza podataka
- **Preduvjet:** administrator je prijavljen i dodijeljena mu je uloga administratora prijevoznih usluga
- **Opis osnovnog tijeka:**
  1. Administrator odabire opciju dodavanja prijevoznika

2. Otvara se prozor za unos podataka o novom prijevozniku
  3. Administrator unese podatke o prijevozniku
  4. Administrator odabere opciju 'Dodaj prijevoznika'
  5. U bazu podataka se pohrani promjena
- **Opis mogućih odstupanja:**
    - 3.a Upisano je neispravno ime prijevoznika
      1. Sustav obavještava administratora o neuspjelom unosu

### UC20 - unos novog pacijenta

- **Glavni sudionik:** korisnički administrator
- **Cilj:** unijeti podatke o novom pacijentu
- **Sudionici:** baza podataka, aplikacija medicinskih usluga
- **Preduvjet:** administrator je prijavljen u sustav i ima ulogu korisničkog administratora
- **Opis osnovnog tijeka:**
  1. Administrator odabire opciju unosa novog pacijenta
  2. Otvara se prozor za upis podataka novog pacijenta
  3. Administrator upiše podatke o novom pacijentu
  4. Pohranjuju se promjene u bazu podataka

### UC21 - dohvaćanje termina pacijenta

- **Glavni sudionik:** korisnički administrator
- **Cilj:** dohvatiti podatke o terminu medicinskih usluga
- **Sudionici:** baza podataka, aplikacij medicinskih usluga
- **Preduvjet:** administrator je prijavljen u sustav i ima ulogu korisničkog administratora
- **Opis osnovnog tijeka:**
  1. Administrator otvara aplikaciju medicinskih usluga
  2. Administrator iz aplikacije medicinskih usluga preuzima podatke preferiranog dolaska i odlaska pacijenta
  3. Podatci o terminu se upisuju u sustav
  4. Pohranjuju se promjene u bazu podataka
- **Opis mogućih odstupanja:**
  - 2.a Za navedeno razdoblje dolaska i odlaska nema dostupnih termina
    1. Sustav javlja administratoru da u navedenom razdoblju nema dostupnih termina

**UC22 - dodjeljivanje smještaja pacijentu**

- **Glavni sudionik:** korisnički administrator
- **Cilj:** +dodijeliti smještaj pacijentu
- **Sudionici:** baza podataka, aplikacija medicinskih usluga
- **Preduvjet:** administrator je prijavljen u sustav i ima ulogu korisničkog administratora
- **Opis osnovnog tijeka:**
  1. Smještajni administrator unosi u aplikaciju podatke o raspoloživom smještaju
  2. Administrator unosi korisnika u sustav
  3. Administrator pridjeljuje raspoloživu smještajnu jedinicu
  4. Smještajna jedinica se označava kao zauzeta u danom periodu

**UC23 - dodjeljivanje prijevoznika terminima**

- **Glavni sudionik:** korisnički administrator
- **Cilj:** dodijeliti prijevoznika
- **Sudionici:** baza podataka, aplikacija medicinskih usluga
- **Preduvjet:** administrator je prijavljen u sustav i ima ulogu korisničkog administratora
- **Opis osnovnog tijeka:**
  1. Administrator prijevoznih usluga u aplikaciju unosi podatke o prijevoznicima
  2. Administrator unosi korisnika u sustav
  3. Zaključavanje plana medicinskih usluga s listom termina
  4. Administrator pridjeljuje raspoložive prijevoznike za svaki od termina
  5. Administrator označava prijevoznike zauzete u tim terminima

**UC24 - slanje e-maila prijevozniku**

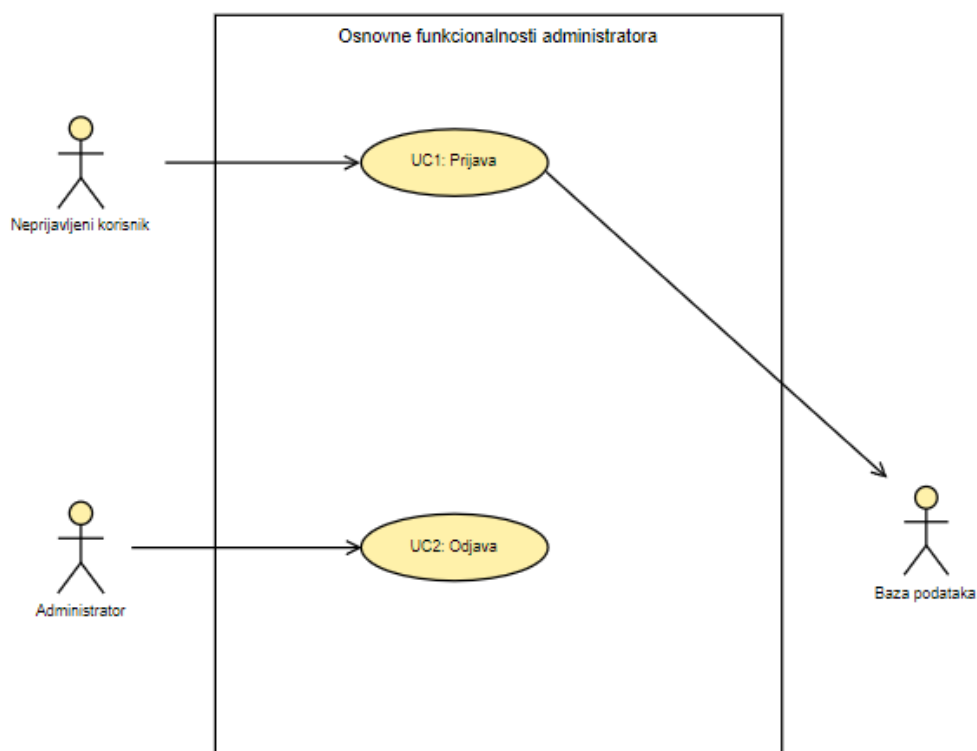
- **Glavni sudionik:** korisnički administrator
- **Cilj:** poslati e-mail s podacima korisnika prijevozniku
- **Sudionici:** baza podataka, aplikacija medicinskih usluga
- **Preduvjet:** administrator je prijavljen u sustav i ima ulogu korisničkog administratora
- **Opis osnovnog tijeka:**
  1. Plan o medicinskim uslugama je zaključan
  2. Aplikacija dohvaća e-mail adresu prijevoznika

3. Aplikacija unutar predloška za e-mail postavlja podatke o medicinskoj usluzi i pojedinostima
  4. Na e-mail adresu prijevoznika se šalje plan
- **Opis mogućih odstupanja:**
    - 3.a Podatci u aplikaciji ne kooperiraju sa podacima prijevoznika
      1. Sustav upozorava da je došlo do pogrešne korelacije podataka

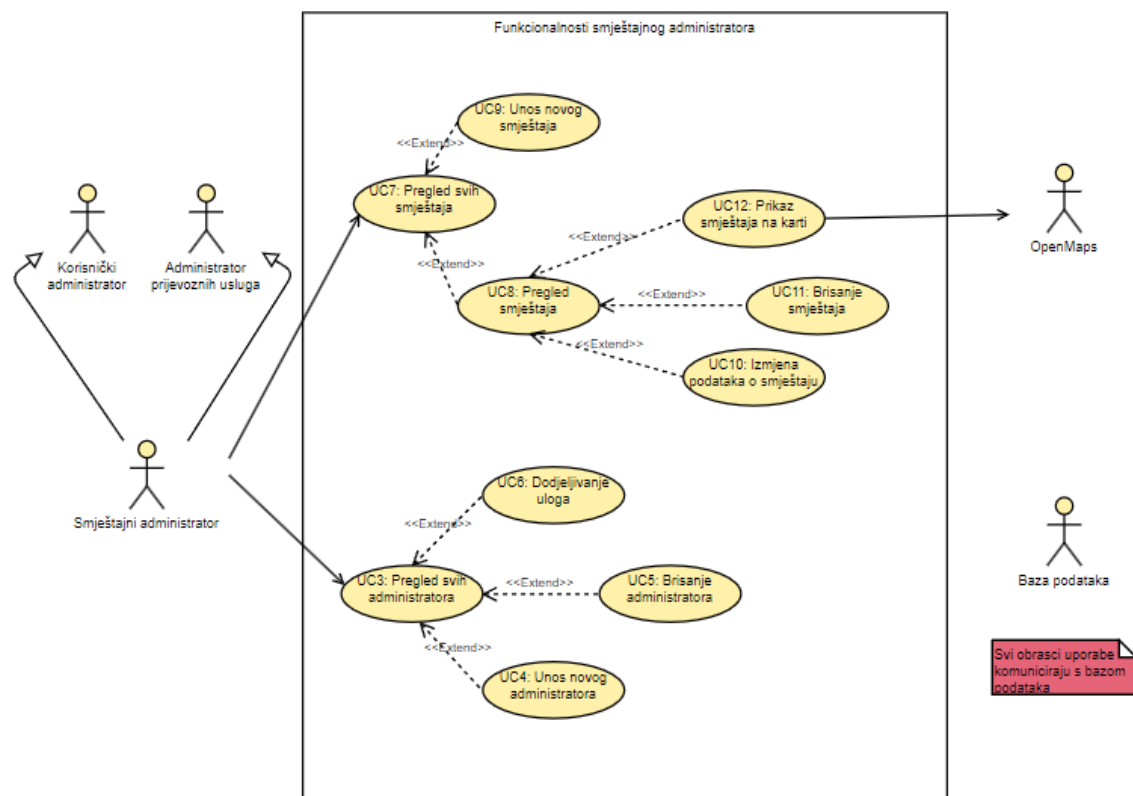
#### UC25 - slanje e-maila pacijentu

- **Glavni sudionik:** korisnički administrator
- **Cilj:** poslati e-mail s pojedinostima termina pacijentu
- **Sudionici:** baza podataka, aplikacija medicinskih usluga
- **Preduvjet:** administrator je prijavljen u sustav i ima ulogu korisničkog administratora
- **Opis osnovnog tijeka:**
  1. Plan o medicinskim uslugama je zaključan
  2. Dohvaća se e-mail adresa pacijenta
  3. Aplikacija unutar e-maila postavlja pojedinosti o terminu medicinske usluge - termin, prijevoznika, smještaj
  4. Na e-mail adresu pacijenta se šalje plan
- **Opis mogućih odstupanja:**
  - 3.a Podatci u aplikaciji ne kooperiraju sa pacijentovim podacima
    1. Sustav upozorava da je došlo do pogrešne korelacije podataka

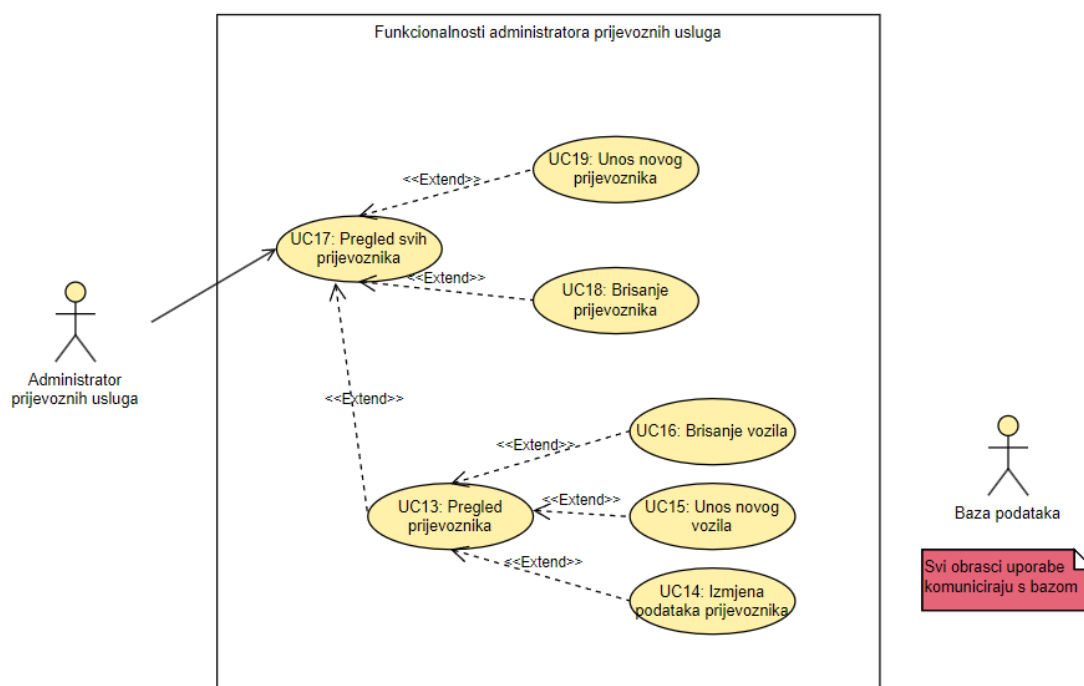
## Dijagrami obrazaca uporabe



Slika 3.1: Dijagram obrasca uporabe, osnovne funkcionalnosti administratora

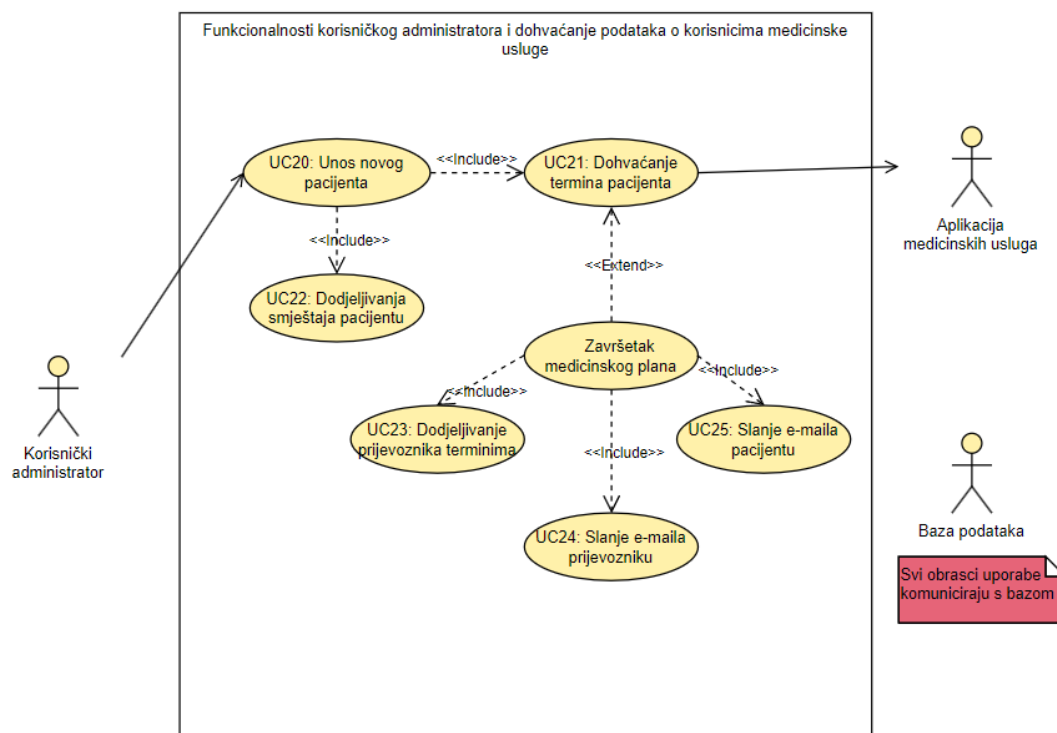


Slika 3.2: Dijagram obrasca uporabe, funkcionalnosti smještajnog administratora



Slika 3.3: Dijagram obrasca uporabe, funkcionalnosti administratora prijevoznih usluga



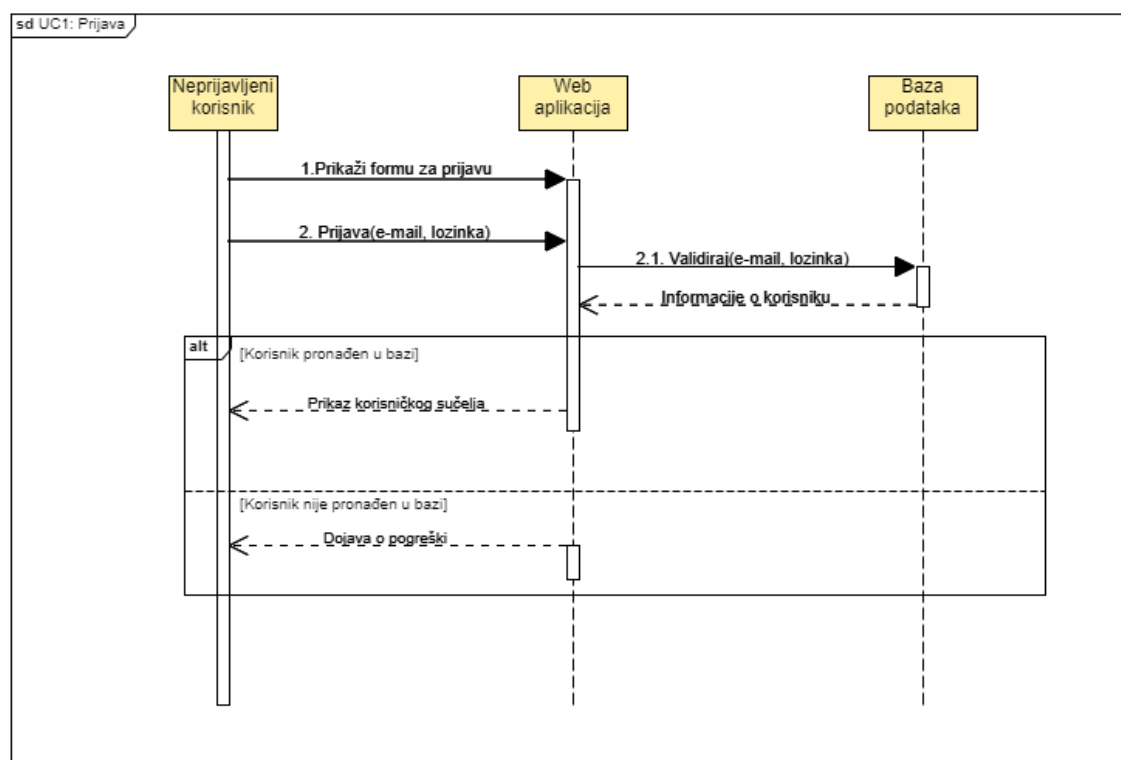


Slika 3.4: Dijagram obrasca uporabe, funkcionalnosti korisničkog administratora

### 3.1.2 Sekvencijski dijagrami

#### Prijava – obrazac uporabe UC1

Neregistrirani korisnik odabire gumb “Prijava se” te ga se preusmjeruje na stranicu s formom za prijavu. Ovdje upisuje svoj e-mail i lozinku te klikom na gumb “Nastavi” uneseni podaci se šalju na validaciju. Ukoliko se podaci podudaraju s onima pohranjenima u bazi, dostavljaju se informacije o administratoru te token za određene uloge koje administrator ima. Ovisno o tokenu, administratoru postaje dostupno korisničko sučelje prilagođeno njegovoj ulozi ili ulogama. U suprotnom, ako uneseni podaci ne odgovaraju niti jednom administratoru u bazi, korisnik prima obavijest o pogreški kako bi znao da nije registriran u sustavu.



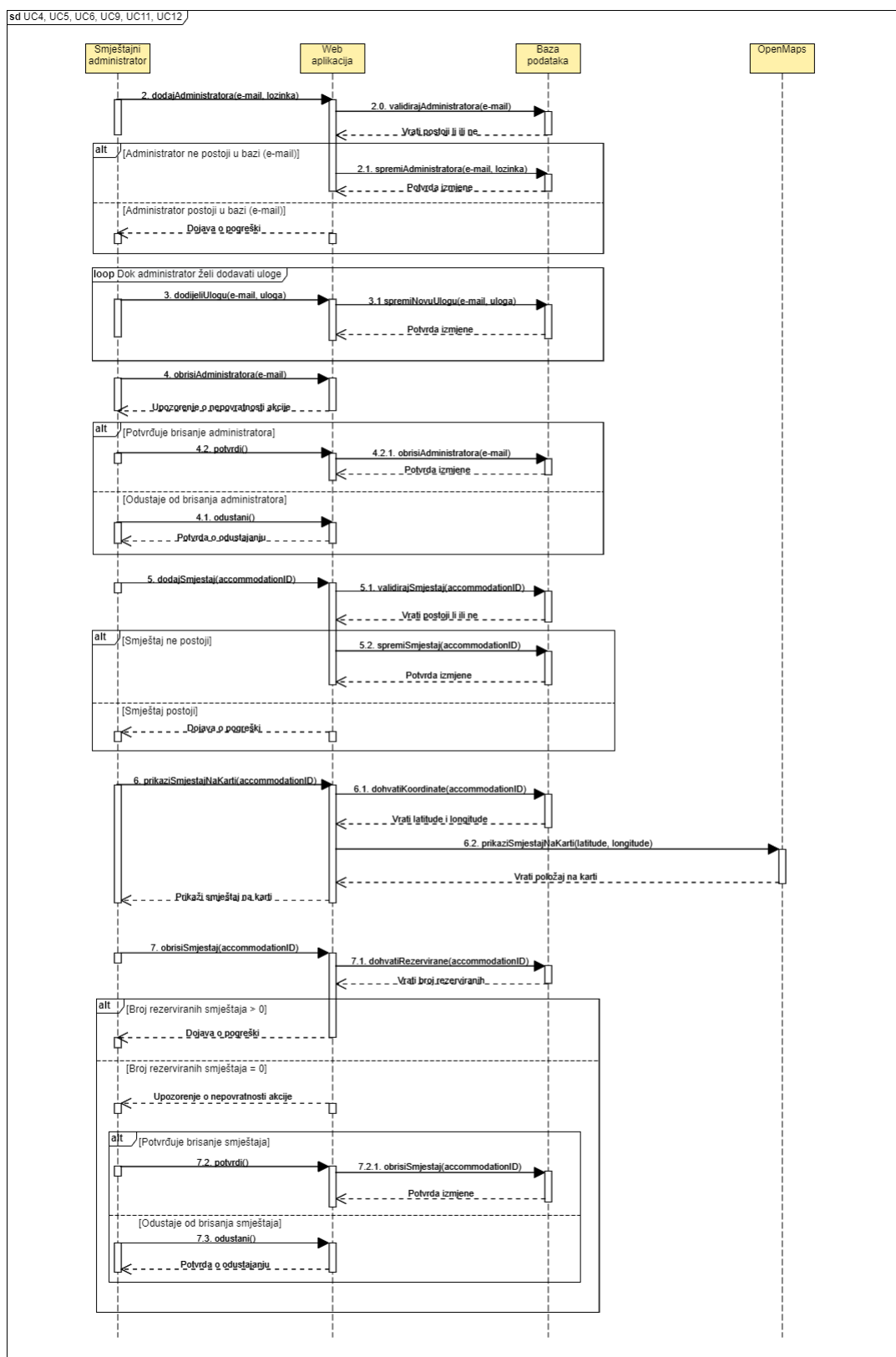
Slika 3.5: Sekvencijski dijagram za UC1

**Stvaranje, brisanje, dodjeljivanje uloga administratorima – obrazac uporabe UC4, UC5, UC6**

Smještajni administrator ima pristup listi svih administratora u bazi. On je ovlašten unositi nove administratore ukoliko oni već ne postoje u bazi, identificirajući ih prema e-mail adresi. U slučaju da administrator već postoji, sustav generira poruku o pogreški. Nadalje, smještajni administrator ima ovlasti dodjeljivanja uloga drugim administratorima. Također, ima mogućnost uklanjanja drugih administratora, pri čemu mu se šalje upozorenje o nepovratnosti akcije. U tom trenutku, smještajni administrator ima opciju potvrde ili odustajanja. Ukoliko potvrdi brisanje, izmjene se pohranjuju u bazu.

**Stvaranje, brisanje i prikaz smještaja na karti – obrasci uporabe UC9, UC11, UC12**

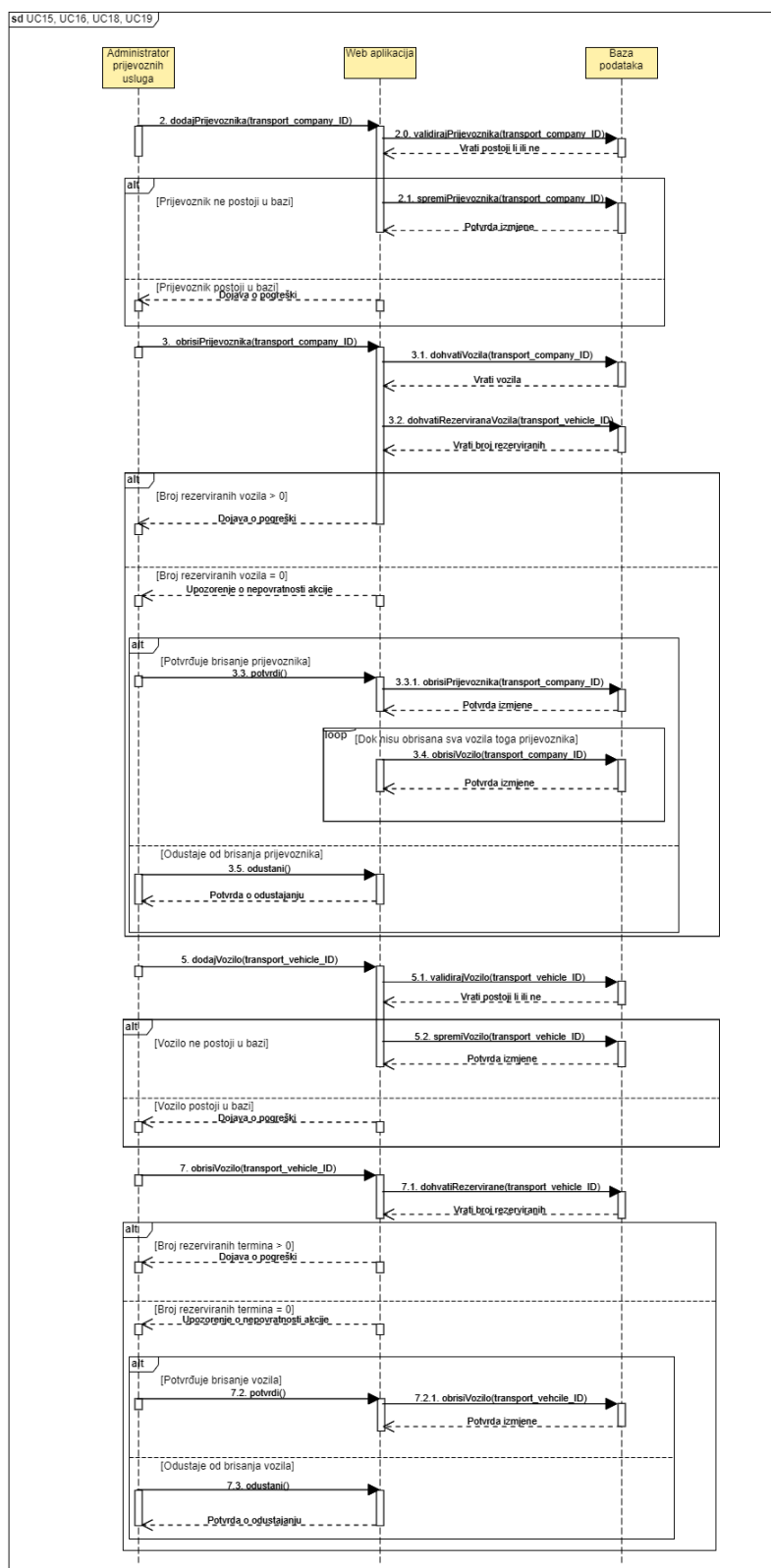
Smještajni administrator ima ovlasti i nad smještajem. Može dodavati nove smještaje, a ukoliko takav smještaj već ne postoji, podaci se pohranjuju u bazu. Nadalje, smještajni administrator ima mogućnost pregledavanja lokacije smještaja na karti putem koordinata (latitude i longitude). U postupku brisanja smještaja, sustav provjerava da li je određeni smještaj rezerviran. Ako nije, administratoru se šalje upozorenje o nepovratnosti akcije te ima opciju potvrde ili odustajanja. Ukoliko potvrdi, izmjene se pohranjuju u bazu.



Slika 3.6: Sekvencijski dijagram za UC4, UC5, UC6, UC9, UC11, UC12

**Stvaranje i brisanje prijevoznika, stvaranje i brisanje vozila – obrasci uporabe UC15, UC16, UC18, UC19**

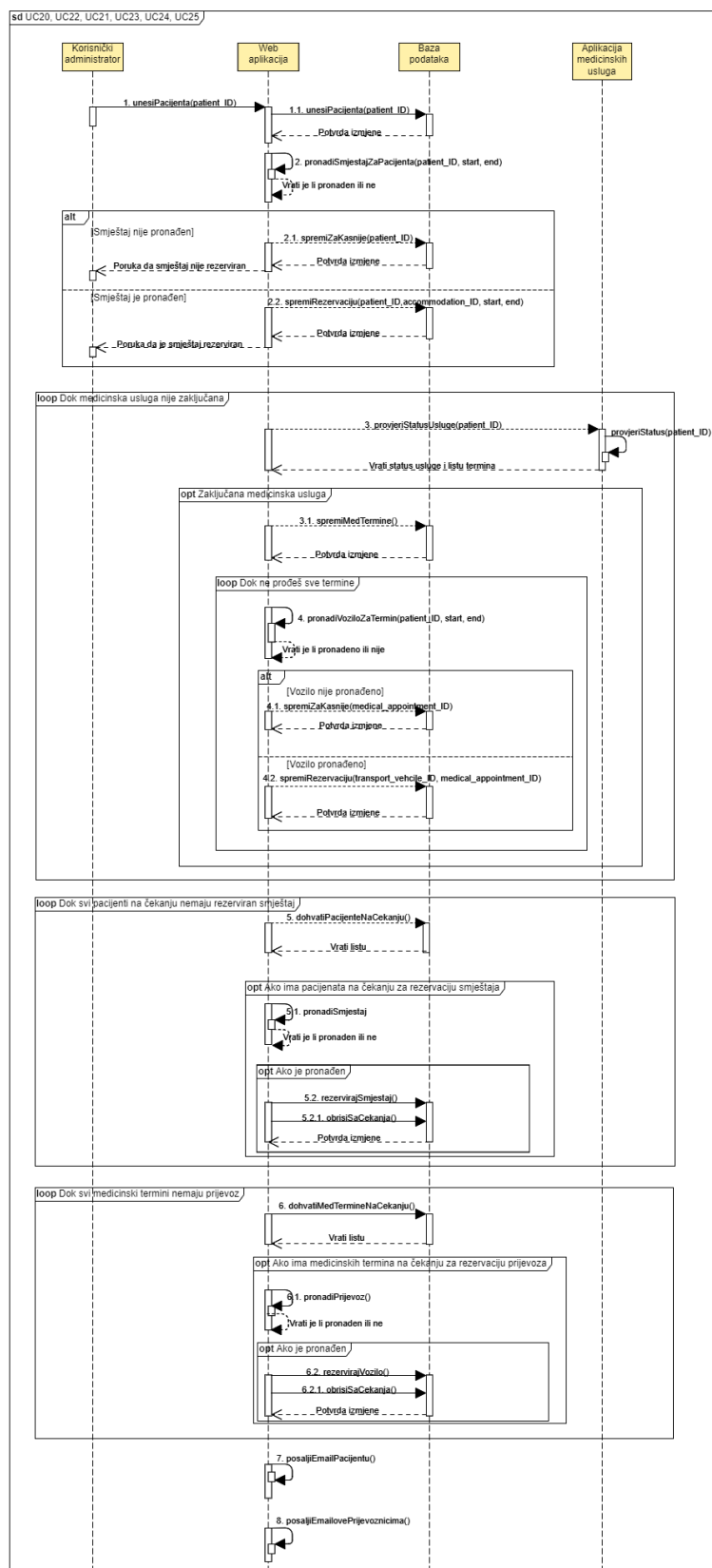
Administrator prijevoznih usluga ima ovlasti dodavanja novih prijevoznika, pod uvjetom da već nisu prisutni u sustavu. Kada želi obrisati prijevoznika, prvo se provjerava jesu li vozila toga prijevoznika rezervirana u nekom terminu. Ukoliko jesu, šalje se poruka pogreške, a ako nisu administratoru se šalje upozorenje o nepovratnosti akcije te ima mogućnost potvrditi svoj izbor. Ako potvrdi, briše se prijevoznik i sva vozila toga prijevoznika. Dodavanjem novog vozila provjerava se postoji li već isto vozilo, ako ne postoji vozilo se dodaje u bazu. Brisanjem vozila provjerava se je li vozilo rezervirano u nekom terminu. Ukoliko jest, šalje se poruka o pogreški, a ako nije administratoru se šalje upozorenje o nepovratnosti te ima mogućnost potvrditi svoj izbor. Potvrđivanjem akcije briše se odabrano vozilo.



Slika 3.7: Sekvencijski dijagram za UC15, UC16, UC18, UC19

**Unos pacijenta, dodjeljivanje smještaja, dohvaćanje termina, dodjeljivanje prijevoza terminima, slanje e-maila – obrasci uporabe UC20, UC21, UC22, UC23, UC24, UC25**

Korisnički administrator unosi podatke o pacijentu te se on unosi u bazu. Odmah se traži smještaj za termin pacijenta, ako se pronađe smještaj se rezervira u tome vremenu, a ako ne pacijenta se stavlja na čekanje. Nakon unosa pacijenta, moramo provjeravati je li aplikacija medicinskih usluga zaključala medicinsku uslugu. Kada je medicinska usluga zaključana spremamo medicinske termine te svima moramo pokušati dodijeliti prijevoznika, to jest, vozilo u vremenu termina usluge. Ako je vozilo pronađeno, rezervira se, a ako nije termin se stavlja na čekanje. Kada se prođu svi termini, moramo dodijeliti smještaj pacijentu ako mu još nije dodijeljen te nakon toga za termine koji nemaju rezerviran prijevoz pronaći i rezervirati vozilo. Nakon rezerviranih prijevoza i smještaja pacijentu se šalje mail o svim rezervacijama, a prijevoznicima kontaktni podaci korisnika te podaci o rezervaciji smještaja.



Slika 3.8: Sekvencijski dijagram za UC20, UC21, UC22, UC23, UC24, UC25



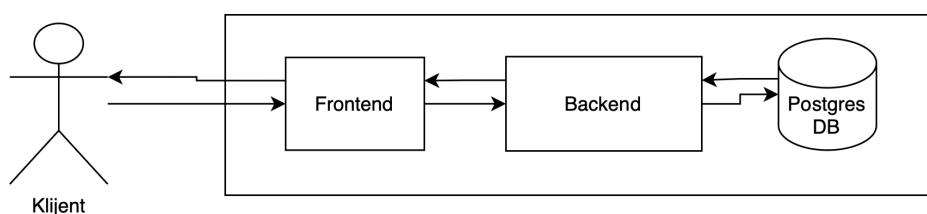
## 3.2 Ostali zahtjevi

- Aplikacija mora osigurati visoku razinu pouzdanosti u evidenciji i koordinaciji smještaja i prijevoza, uz minimalnu grešku u podacima.
- Sučelje treba biti intuitivno i prilagođeno korisnicima koji možda nisu upoznati s aplikacijama sličnog sadržaja.
- Vrijeme odziva aplikacije na korisničke zahtjeve treba biti brzo kako bi se osiguralo učinkovito upravljanje rezervacijama.
- Aplikacija treba podržavati veliki broj korisnika i smještajnih kapaciteta, uz mogućnost lakoće nadogradnje za dodatne kapacitete.
- Sigurnosni protokoli za zaštitu osobnih podataka korisnika i osjetljivih informacija o rezervacijama su neophodni.
- Aplikacija treba biti kompatibilna s postojećim sustavima za administraciju zdravstvenih usluga.
- Aplikacija treba osigurati efikasno upravljanje uslugama kako bi se zadovoljile potrebe i očekivanja klijenata.
- Aplikacija mora omogućiti efikasno praćenje i upravljanje rezervacijama smještaja i prijevoza, s detaljnim izvještajima o statusu i dostupnosti.
- Aplikacija treba biti dizajnirana s fokusom na efikasnost i jednostavnost navigacije, omogućujući brzu i laku upotrebu bez nepotrebnih vizualnih distrakcija.

## 4. Arhitektura i dizajn sustava

Arhitektura se može podijeliti na tri glavna dijela:

- *REST API poslužitelj (Spring Boot)*
- *Klijentska aplikacija (React)*
- *Baza podataka*



Slika 4.1: Arhitektura sustava

Korisnik interagira s klijentskom aplikacijom putem web preglednika. Ova aplikacija je odvojena od backend sustava te koristi React uz typescript za stvaranje dinamičkog korisničkog sučelja.

REST API poslužitelj izgrađen je koristeći Spring Boot, koji omogućuje izradu visoko efikasnih i skalabilnih RESTful servisa. Komunikacija između klijenta i servera odvija se putem HTTP protokola, gdje frontend šalje zahtjeve backendu, a backend odgovara s potrebnim podacima u JSON formatu.

Klijentska aplikacija je odgovorna za prikazivanje podataka korisniku i obradu korisničkih interakcija. Ona komunicira s backendom putem REST API-ja za dohvaćanje i slanje podataka.

Postgres baza podataka čuva sve podatke potrebne za aplikaciju. Spring Boot aplikacija komunicira s bazom podataka koristeći Spring Data JPA za upravljanje podacima.

Odlučili smo se za korištenje IntelliJ-a kao razvojnog okruženja za backend, a za frontend koristimo Visual Studio Code. Za razliku od tradicionalnog MVC koncepta, naš sustav je podijeljen na odvojene slojeve gdje backend (REST API poslužitelj) i frontend (klijentska aplikacija) rade neovisno jedan o drugome. Uspoređujući

s tradicionalnim MVC konceptom, mogli bismo reći da je View (V u MVC) na frontendu, a Model i Controller (M i C u MVC) na backendu. Ovo omogućuje fleksibilnost i lakše skaliranje svakog dijela aplikacije zasebno.

## 4.1 Baza podataka

Za potrebe naše aplikacije koristiti ćemo relacijsku bazu podataka kako bismo lakše oblikovali stvarni svijet. Baza nam je potrebna za metodičku pohranu podataka te njihovo brzo dohvaćanje. Naša baza podataka se sastoji od sljedećih entiteta:

- Patient
- Accommodation
- AccommodationOrder
- AccommodationBooking
- TransportCompany
- TransportVehicle
- TransportBooking
- MedicalAppointment
- AdminRole
- AdminRoles
- Admin

### 4.1.1 Opis tablica

**Patient** Ova entitet sadrži sve informacije o korisniku usluga našeg sustava. Kako su svi korisnici pacijenti odlučili smo ih imenovati tako. Entitet sadrži attribute: `patient_ID`, `first_name`, `last_name`, `PIN`, `email` i `phone_number`. Ovaj entitet je u vezi *One-to-Many* s entitetom `AccommodationOrder` preko atributa `patient_ID` i u *One-to-Many* s entitetom `AccommodationBooking` preko atributa `patient_ID`.

Patient		
patient_ID	VARCHAR	primarni ključ tablice
first_name	VARCHAR	ime pacijenta
last_name	VARCHAR	prezime pacijenta

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

Patient		
PIN	VARCHAR	Personal Identification Number, kao OIB u hrvatskoj
email	VARCHAR	pacijentov email
phone_number	VARCHAR	pacijentov telefonski broj

**Accommodation** Ovaj entitet sadržava potrebne podatke o nekom smještaju koji je smještajni administrator unio. Entitet sadrži attribute: accommodation\_ID, type, category, address, availability\_start, availability\_end i location. Ovaj entitet je u *One-to-Many* vezi s entitetom AccommodationBooking preko atributa accommodation\_ID.

Accommodation		
accommodation_ID	VARCHAR	primarni ključ tablice
type	VARCHAR	versta smještaja
category	VARCHAR	kategorija smještaja
address	VARCHAR	adresa smještaja
availability_start	DATETIME	datum i vrijeme od kada je smještaj dostupan
availability_end	DATETIME	datum i vrijeme do kada je smještaj dosupan
location	POINT	kordinate smještaja

**AccommodationOrder** Ovaj entitet se koristi za pohranjivanje pacijentovih zahtjeva o traženom smještaju. U slučaju da traženi smještaj nije odmah dostupan zahtjev se sprema kako bih se kasnije mogao opet pogledati. Entitet sadrži attribute: accommodation\_order\_ID, arrival\_datetime, departure\_datetime, accommodation\_type, accommodation\_category i patient\_ID. Ovaj entitet je u *Many-to-One* vezi s entitetom Patient preko patient\_ID atributa.

AccommodationOrder		
accommodation_order_ID	VARCHAR	primarni ključ tablice
arrival_datetime	DATETIME	vrijeme dolaska pacijenta u državu, od tada mu treba smještaj
departure_datetime	DATETIME	vrijeme odlaska pacijenta iz države, do tada treba smještaj
accommodation_type	VARCHAR	željeni tip smještaja koji pacijent traži
accommodation_category	VARCHAR	željena kategorija smještaja koju pacijent traži
patient_ID	STRING	ID pacijenta koji je napravio ovaj zahtjev

**AccommodationBooking** Ovaj vezni entitet se koristi za pohranjivanje informacije koji pacijent je kada u kojem smještaju. Entitet sadrži attribute: accommodation\_booking\_ID, start\_datetime, end\_datetime, accommodation\_ID, patient\_ID. Ovaj entitet je u *Many-to-One* vezi s entitetom Accommodation preko atributa accommodation\_ID. U *Many-to-One* vezi s entitetom Patient preko atributa patient\_ID. I u *One-to-Many* s entitetom TransportBooking preko accommodation\_booking\_ID.

AccommodationBooking		
accommodation_booking_ID	VARCHAR	primarni ključ tablice
start_datetime	DATETIME	datum i vrijeme od kada je pacijent u smještaju
end_datetime	DATETIME	datum i vrijeme do kada je pacijent u smještaju
accommodation_ID	VARCHAR	smještaj u koji je pacijent smješten
patient_ID	VARCHAR	pacijent koji ostaje u smještaju

**TransportCompany** Ovaj entitet sadržava informacije o transportnoj firmi. Entitet sadrži attribute: transport\_company\_ID, name, phone\_number, email. Ovaj

entitet je u *One-to-Many* vezi s entitetom TransportVehicle preko atributa transport\_company\_ID.

TransportCompany		
transport_company_ID	VARCHAR	primarni ključ tablice
name	VARCHAR	ime firme
phone_number	VARCHAR	telefonski broj firme
email	VARCHAR	email firme

**TransportVehicle** Ovaj entitet sadržava informacije o nekom transportnom vozilu. Entitet sadrži attribute: transport\_comapny\_ID, name, phone\_number, email. Ovaj entitet je u *Many-to-One* vezi s entitetom TransportCompany preko atributa transport\_company\_ID. I u *One-to-Many* vezi s entitetom TransportBooking preko atributa transport\_vehicle\_ID.

TransportVehicle		
transport_vehicle_ID	VARCHAR	primarni ključ tablice
type	VARCHAR	vrsta vozila
capacity	INT	kapacitet vozila
transport_company_ID	VARCHAR	transportna firma kojoj pripada ovo vozilo

**TransportBooking** Ovo je slabi vezni entitet koji označava prijevoz pacijenta. Ujedinjuje tri entiteta iz kojih saznajem kojeg pacijenta treba voziti, od kuda će biti transportiran, koje vozilo će se koristiti i do koje klinike će se transportirati. Entitet sadrži attribute: transport\_vehicle\_ID, accommodation\_booking\_ID, medical\_appointment\_ID. Entitet je u *Many-to-One* vezi s entitetom TransportVehicle preko atributa transport\_vehicle\_ID. U *Many-to-One* vezi s entitetom AccommodationBooking preko atributa accomodaton\_booking\_ID. I u *Many-to-One* vezi s entitetom MedicalAppointment preko atributa medical\_appointment\_ID.

TransportBooking		
transport_vehicle_ID	VARCHAR	vozilo koje obavlja ovaj projevovoz
accommodation_booking_ID	VARCHAR	poveznica na smještaj i pacijenta koje treba prevesti iz tog smještaja
medical_appointment_ID	VARCHAR	poveznica na medicinski tretman koji nam daje informaciju gdje pacijenta treba voziti

**MedicalAppointment** Ovaj entitet korisimo za spremanje podataka o medicinskim tretmanima nekog pacijenta. Informacije o ovom ćemo dobiti iz medicinskog sustava, ali ako nije moguće odmah napraviti TransportBooking zbog npr. nedostatka vozila, želimo pospremiti informacije o tretmanu. Entitet sadrži attribute: medical\_appointment\_ID, PIN, clinic\_address, start\_datetime, end\_datetime. Entitet je u *One-to-One* vezi s entitetom TransportBooking preko atributa medical\_appointment\_ID.

MedicalAppointment		
medical_appointment_ID	VARCHAR	primarni ključ tablice
PIN	VARCHAR	Personal Identification Number pacijenta za kojeg je ovo medicinski tretman
clinic_address	VARCHAR	adresa klinike u kojoj je tretman
start_datetime	DATETIME	datum i vrijeme početka tretmana
end_datetime	DATETIME	datum i vrijeme kraja tretmana

**AdminRole** Ovaj entitet označava role admina. Sadrži attribute, admin\_role\_ID i name. Entitet je u *One-to-Many* vezi s entitetom AdminRoles preko atributa admin\_role\_ID.

AdminRole		
admin_role_ID	VARCHAR	primarni ključ tablice
name	VARCHAR	deskriptivni naziv role

**Admin** Ovaj entitet sadrži podatke o adminima. Entitet sadrži atribute: admin\_ID, email, first\_name, last\_name. Entitet je u *One-to-Many* vezi s entitetom AdminRoles preko atributa admin\_ID.

Admin		
admin_ID	VARCHAR	primarni ključ tablice
email	VARCHAR	email admina
first_name	VARCHAR	ime admina
last_name	VARCHAR	prezime admina

**AdminRoles** Ovaj vezni entitet služi kao spoj nekog admina s njegovom rolom. Atributi entiteta su: admin\_role\_ID i admin\_ID. Entitet je u *Many-to-One* vezi s entitetom AdminRole preko atributa admin\_role\_ID. I entitet je u *Many-to-One* vezi s entitetom Admin preko atributa admin\_ID.

AdminRoles		
admin_role_ID	VARCHAR	poveznica na rolu ovog admina
admin_ID	VARCHAR	poveznica na kojeg admina se odnosi ova rola



## 4.1.2 Dijagram baze podataka

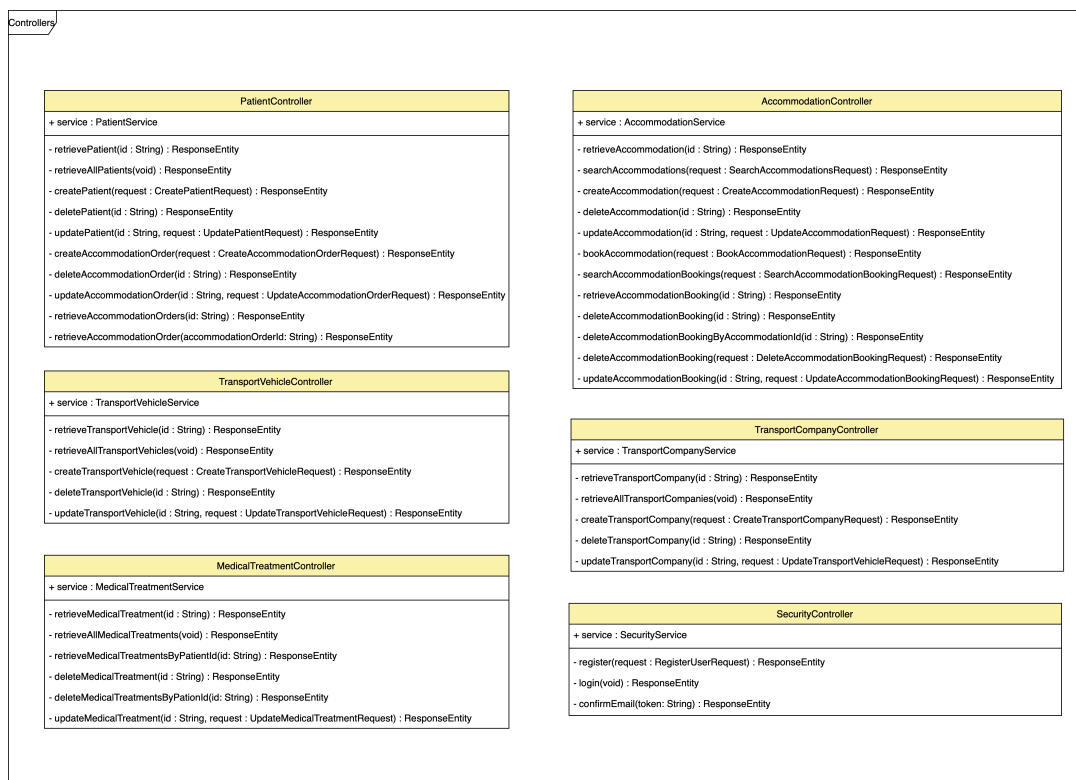


Slika 4.2: ER Dijagram baze podataka

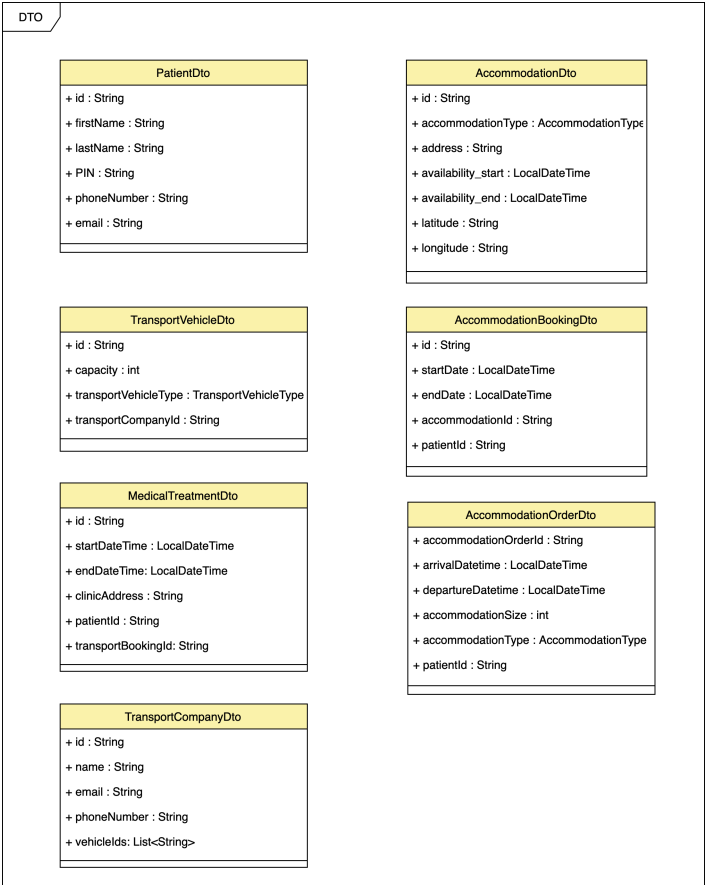
## 4.2 Dijagram razreda

Na slikama 4.3 do 4.6 prikazani su ključni elementi backend dijela arhitekture DentAll aplikacije.

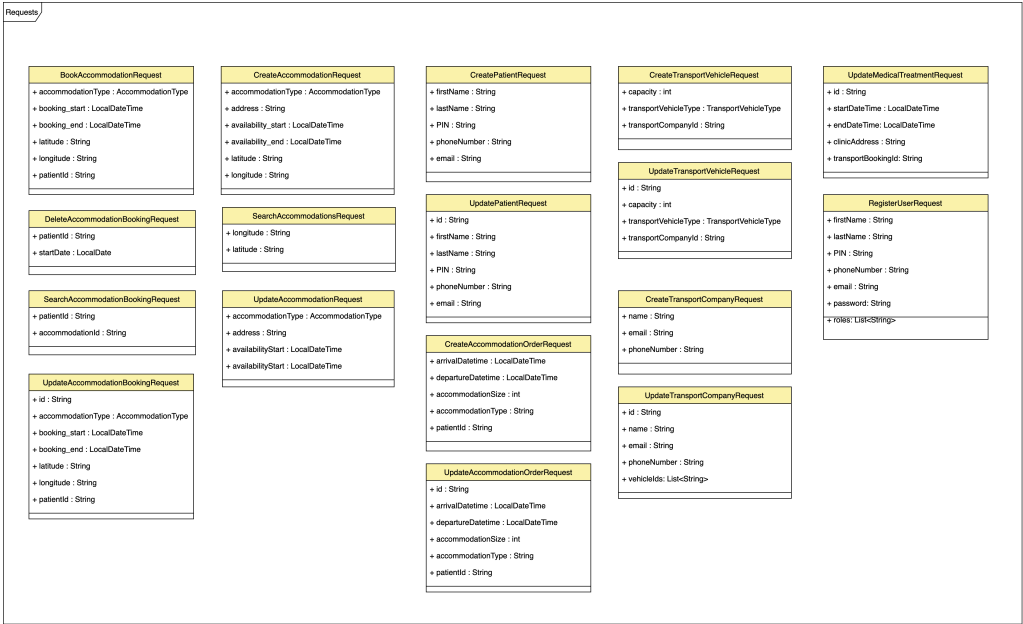
Prva slika 4.3 prikazuje dijagram razreda svih Controller razreda koji se koriste u našoj aplikaciji. Slika 4.4 fokusira se na dijagram razreda s DTO razredima koje služe za prijenos podataka između Controllera i Modela. Svaka DTO klasa odgovara određenom razredu iz Modela, osiguravajući učinkovit prijenos podataka unutar aplikacije. Na slici 4.5 nalaze se Requests razredi koje predstavljaju zahtjeve od klijenta prema Controller razredima. Ovi razredi sadrže sve potrebne informacije za izvođenje funkcionalnosti poput CREATE i UPDATE. Obuhvaćajući sve domain razrede iz modela DentAll aplikacije, dijagram razreda sa slike 4.6 prikazuje relacije i međusobnu povezanost između razreda unutar modela.



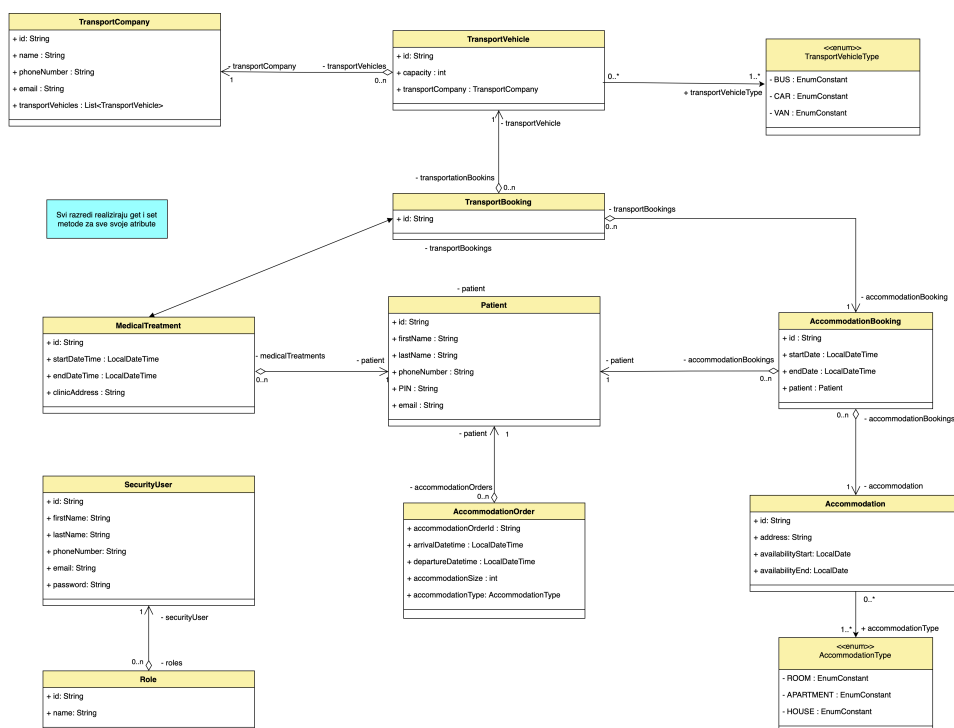
Slika 4.3: Dijagram razreda - dio Controllers



Slika 4.4: Dijagram razreda - dio DTO



Slika 4.5: Dijagram razreda - dio Requests



Slika 4.6: Dijagram razreda - dio 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

## *Kontinuirano osvježavanje*

*Popisati sve reference i literaturu koja je pomogla pri ostvarivanju projekta.*

1. Programsko inženjerstvo, FER ZEMRIS, <http://www.fer.hr/predmet/proinz>
2. I. Sommerville, "Software engineering", 8th ed, Addison Wesley, 2007.
3. T.C.Lethbridge, R.Langaniere, "Object-Oriented Software Engineering", 2nd ed. McGraw-Hill, 2005.
4. I. Marsic, Software engineering book", Department of Electrical and Computer Engineering, Rutgers University, <http://www.ece.rutgers.edu/~marsic/books/SE>
5. The Unified Modeling Language, <https://www.uml-diagrams.org/>
6. Astah Community, <http://astah.net/editions/uml-new>
7. Visual Paradigm Online, <https://online.visual-paradigm.com/>
8. DBDiagrams, <https://dbdiagram.io>

# Indeks slika i dijagrama

3.1	Dijagram obrasca uporabe, osnovne funkcionalnosti administratora	21
3.2	Dijagram obrasca uporabe, funkcionalnosti smještajnog administratora . . . . .	22
3.3	Dijagram obrasca uporabe, funkcionalnosti administratora prijevoznih usluga . . . . .	23
3.4	Dijagram obrasca uporabe, funkcionalnosti korisničkog administratora . . . . .	24
3.5	Sekvencijski dijagram za UC1 . . . . .	25
3.6	Sekvencijski dijagram za UC4, UC5, UC6, UC9, UC11, UC12 . . . . .	27
3.7	Sekvencijski dijagram za UC15, UC16, UC18, UC19 . . . . .	29
3.8	Sekvencijski dijagram za UC20, UC21, UC22, UC23, UC24, UC25 . . . . .	31
4.1	Arhitektura sustava . . . . .	33
4.2	ER Dijagram baze podataka . . . . .	40
4.3	Dijagram razreda - dio Controllers . . . . .	41
4.4	Dijagram razreda - dio DTO . . . . .	42
4.5	Dijagram razreda - dio Requests . . . . .	42
4.6	Dijagram razreda - dio Model . . . . .	43

# Dodatak: Prikaz aktivnosti grupe

## Dnevnik sastajanja

### *Kontinuirano osvježavanje*

*U ovom dijelu potrebno je redovito osvježavati dnevnik sastajanja prema predlošku.*

#### 1. sastanak

- Datum: 25. listopada 2023.
- Prisustvovali: Filip Buljan, Roko Gligora, Neven Lukić, Karla Pišonić, Karla Šmuk
- Teme sastanka:
  - dogovor o korištenim tehnologijama
  - dogovor podjele zadataka
  - komentiranje osnovnih zahtijeva projekta

#### 2. sastanak

- Datum: 1. studenoga 2023.
- Prisustvovali: Filip Buljan, Roko Gligora, Neven Lukić, Maksim Madžar, Karla Pišonić, Karla Šmuk
- Teme sastanka:
  - korištene i organizacija GitHuba
  - podjela zadataka
  - pregled modela baze podataka
  - pregled dizajna korisničkog sučelja

#### 3. sastanak

- Datum: 8. studenoga 2023.
- Prisustvovali: Filip Buljan, Roko Gligora, Neven Lukić, Maksim Madžar, Karla Pišonić, Karla Šmuk
- Teme sastanka:
  - podjela zadataka
  - dogovor oko backend funkcionalnosti

- razgovor o idejama za puštanje u pogon

#### 4. sastanak

- Datum: 15. studenoga 2023.
- Prisustvovali: Filip Buljan, Roko Gligora, Neven Lukić, Maksim Madžar, Karla Pišonić, Ante Vujčić, Karla Šmuk
- Teme sastanka:
  - podjela zadataka za dovršetak prve revizije
  - razgovor o puštanje u pogon



## Tablica aktivnosti

### Kontinuirano osvježavanje

*Napomena: Doprinosi u aktivnostima treba navesti u satima po članovima grupe po aktivnosti.*

	Neven Lukić	Filip Buljan	Roko Gligora	Maksim Madžar	Karla Pišonić	Karla Šmuk	Ante Vujčić
Upravljanje projektom	6						
Opis projektnog zadatka						5	
Funkcionalni zahtjevi						3	
Opis pojedinih obrazaca	2				4		
Dijagram obrazaca						5	
Sekvencijski dijagrami						10	
Opis ostalih zahtjeva		1					
Arhitektura i dizajn sustava	3						
Baza podataka	7			2			
Dijagram razreda			6	3			
Dijagram stanja							
Dijagram aktivnosti							
Dijagram komponenti							
Korištene tehnologije i alati							
Ispitivanje programskog rješenja							
Dijagram razmještaja							
Upute za puštanje u pogon							

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

	Neven Lukić	Filip Buljan	Roko Gligora	Maksim Madžar	Karla Pišonić	Karla Šmuk	Ante Vujčić
Dnevnik sastajanja	1						
Zaključak i budući rad							
Popis literature							
Backend		6	22	20			
Frontend	4		2				
Puštanje u pogon				1			
Istraživanje informacija/tehnologija		10				4	

## 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.*