

Programsko inženjerstvo

Ak. god. 2023./2024.

Medicinska rehabilitacija

Dokumentacija, Rev. 1

Grupa: *Seven Smurfs*

Voditelj: *Dora Kašik*

Datum predaje: *17. 11. 2023.*

Nastavnik: *Miljenko Krhen*

Sadržaj

1. Dnevnik promjena dokumentacije	3
2. Opis projektnog zadatka „Medicinska rehabilitacija“	4
2.1 Uvod	4
2.2 Izgled stranica aplikacije	4
2.2.1 Početna stranica	4
2.2.2 Stranica za prijavu	4
2.2.3 Zaboravljena lozinka	5
2.2.4 Registracija	5
2.2.5 Pomoć	5
2.2.6 Kontakt	5
2.2.7 Početna stranica Admina	5
2.2.8 Početna stranica Korisnika	5
2.2.9 Naručivanje na termin	5
2.2.10 Početna stranica Djelatnika	5
2.2.11 Dodjela termina	5
2.3 Dodatno o aplikaciji	6
Ključne Funkcionalnosti	6
3. Specifikacija programske potpore	8
3.1 Funkcionalni zahtjevi	8
3.1.1 Obrasci uporabe	10
3.1.2 Sekvencijski dijagrami	19
3.2 Ostali zahtjevi	22
4. Arhitektura i dizajn sustava	23
Baza podataka: PostgreSQL u Docker kontejneru	23
Backend: Spring Boot	23
Frontend: React	23

Frontend Integracija: Tailwind CSS, Axios, React-Router-Dom, reCAPTCHA	23
Deployment: Render za backend i Netlify za frontend	24
4.1 Baza podataka	24
4.1.1 Opis tablica	25
4.1.2 Dijagram baze podataka	31
4.2 Dijagram razreda	32
4.3 Dijagram stanja	39
4.4 Dijagram aktivnosti	40
4.5 Dijagram komponenti	41
5. Implementacija i korisničko sučelje	42
5.1 Korištene tehnologije i alati	42
5.2 Ispitivanje programskog rješenja	43
5.2.1 Ispitivanje komponenti	43
5.2.2 Ispitivanje sustava	43
5.3 Dijagram razmještaja	44
5.4 Upute za puštanje u pogon	45
6. Zaključak i budući rad	46
Popis literature	47
Indeks slika i dijagrama	48
Dodatak: Prikaz aktivnosti grupe	49
Dnevnik sastajanja	49

1. Dnevnik promjena dokumentacije

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napravljen predložak	Marko Miletić	17.10.2023.
0.2	Dodani opis projektnog zadatka i funkcionalni zahtjevi	Marko Miletić	21.10.2023.
0.3	Kreiran dijagram arhitekture	Josip Begić	22.10.2023.
0.4	Početak rada na opisu baze podataka	Marko Miletić	27.10.2023.
0.5	Dodani opisi tablica baze podataka	Marko Miletić	02.11.2023.
0.6	Ažuriran dnevnik sastajanja	Marko Miletić	09.11.2023.
0.7	Ažurirana tablica aktivnosti	Marko Miletić	09.11.2023.
0.8	Kreiran dijagram baze podataka	Josip Begić	10.11.2023.
0.9	Kreiran dijagram komponenti	Josip Begić	10.11.2023.
1.0	Kreirani dijagrami razreda	Anton Vivoda	13.11.2023.
1.1	Dodani funkcionalni i ostali zahtjevi	Dora Kašik, Katarina Mikulić	14.11.2023.
1.2	Kreirani sekvencijski dijagrami i dijagrami obrazaca uporabe	Dora Kašik, Katarina Mikulić	14.11.2023.
1.3	Dodani svi dijagrami i opisi	Marko Miletić	14.11.2023.
1.4	Napisan opis arhitekture	Roko Krstičević	17.11.2023.
1.5	Ispravljene izmjene u bazi podataka i u izgledu stranice	Marko Miletić	17.11.2023.

2. Opis projektnog zadatka „Medicinska rehabilitacija“

2.1 Uvod

Cilj ovog projektnog zadatka je razviti programsku podršku za izradu web aplikacije „ReHub“ koja će olakšati prijavu i praćenje poboljšanja zdravstvenog stanja ljudi/bolesnika koji su imali lakše ili teže povrede nakon kojih je potrebno provesti fizikalnu terapiju i medicinsku rehabilitaciju. Razvijena kako bi omogućila transparentan, učinkovit, i siguran način za bolesnike da prate svoj put prema oporavku, „ReHub“ također pomaže zdravstvenim ustanovama bolje organizirati i pratiti rehabilitacijske procese.

2.2 Izgled stranica aplikacije

U nastavku se opisuju stranice web aplikacije te njihove funkcionalnosti.

2.2.1 Početna stranica

Na početnoj stranici korisnik može odabrati jednu od tri moguće opcije kako želi nastaviti rad u aplikaciji ili potražiti detaljnije informacije:

- Prijava
- Registracija
- Kontakt

2.2.2 Stranica za prijavu

Od korisnika se za uspješan nastavak rada u aplikaciji traži da unese valjano korisničko ime te ispravnu lozinku. Postoji i opcija „Zaboravljena lozinka“ kojom se korisniku na e-mail šalju upute za postavljanje nove lozinke u slučaju da je korisnik zaboravio lozinku.

2.2.3 Zaboravljena lozinka

Na ovoj stranici se od korisnika očekuje da unese e-mail adresu na koju želi dobiti daljnje upute o ponovnom postavljanju lozinke.

2.2.4 Registracija

Stranica sadrži polja u koje korisnik upisuje ime, prezime, e-mail adresu, broj telefona, OIB, lozinku, ponovno lozinku, MBO, datum rođenja te spol (opcionalno) ukoliko se želi uspješno registrirati.

2.2.5 Pomoć

Ova stranica sadrži odgovore na najčešće postavljena pitanja te sadrži detaljne upute kako se ispravno registrirati i prijaviti u aplikaciju.

2.2.6 Kontakt

Odabirom ove stranice korisnik može vidjeti e-mail adresu lokaciju te radno vrijeme organizacije.

2.2.7 Početna stranica Admina

Na početnoj stranici Admina mogu se vidjeti svi djelatnici i pacijenti u klinici. Gornji dio stranice sadrži opciju odjave te opciju biranja postavki.

2.2.8 Početna stranica Korisnika

Ovdje korisnik može vidjeti popis svojih termina te ima opciju kojom se naručuje na novi termin. Na vrhu stranice postoje opcije „Kontakt“, „Moj profil“ i „Odjava“.

2.2.9 Naručivanje na termin

Korisnik bira jednu od opcija „Prvi dolazak“ ili „Ponovna terapija“. Uz to navodi vrstu oboljenja, opis oboljenja, izdavatelja uputnice postupak liječenja i referencu na obavljenju terapiju.

2.2.10 Početna stranica Djelatnika

Djelatnik na svojoj početnoj stranici vidi popis prijave na termine, popis dostupnih i nedostupnih djelatnika, opciju za dodjeljivanje termina, popis slobodnih i nedostupnih soba i ordinacija te popis opreme.

2.2.11 Dodjela termina

Na ovoj stranici djelatnik ima pristup informacijama koje je korisnik naveo pri naručivanju

termina. Djelatnik odabire ordinaciju, termin i djelatnika koji će raditi na rehabilitaciji.

2.3 Dodatno o aplikaciji

Zamišljeno je da aplikacija bude što je više moguće lakša i intuitivnija za upotrebu, pošto računamo da bi pripadnici svih dobnih skupina mogli i htjeli koristiti ovu aplikaciju. U funkcionalnosti aplikacije vidimo veliku korist jer bi se njenom uporabom klijentima znatno olakšalo i ubrzalo naručivanje na medicinsku rehabilitaciju, a djelatnicima klinike bi se olakšala evidencija naručenih klijenata. Klinike bi također mogle doživjeti povećani interes klijenata zainteresiranih za njihove usluge. Uporabom aplikacije klijenti bi izbjegli dugotrajna čekanja u redovima bolnica i klinika. Aplikacija osigurava sigurnost i zaštitu osjetljivih zdravstvenih podataka te poboljšava komunikaciju između bolesnika, zdravstvenih ustanova i liječnika. Smatramo kako je ključno pratiti i analizirati statistiku rehabilitacije radi kontinuiranog poboljšanja procesa.

Ključne Funkcionalnosti

- *Registracija bolesnika:*

Bolesnici se registriraju putem aplikacije, pružajući svoje osobne podatke. Administrator ustanove provjerava i verificira podatke za osiguranje točnosti informacija.

- *Prijavljivanje na rehabilitaciju:*

Bolesnici unose detalje o svojim oboljenjima i zahtjevima za liječenjem. Liječnici su verificirani putem imenika liječnika.

- *Raspoređivanje termina:*

Djelatnici ustanove pregledavaju i odobravaju prijave bolesnika. Na temelju kapaciteta opreme, prostora i osoblja, dodjeljuju termine za rehabilitaciju.

- *Komunikacija s bolesnicima:*

Bolesnicima se automatski šalju informacije o njihovim terminima putem elektroničke pošte. Mogućnost direktnog kontakta putem elektroničke pošte za obavješćavanje o promjenama.

- *Sigurnost podataka:*

Visoka razina sigurnosti osigurava zaštitu osobnih podataka bolesnika i drugih osjetljivih informacija.

- *Pregled i upisivanje rezultata:*

Djelatnici ustanove imaju pristup svim podacima o bolesnicima i njihovim tretmanima.

Nakon svakog rehabilitacijskog zahvata, upisuju postignute rezultate.

- *Administrator:*

Sistemi administrator ima potpunu kontrolu nad aplikacijom.

Može definirati postavke, pristupne razine i konfiguracije sustava.

3. Specifikacija programske potpore

3.1 Funkcionalni zahtjevi

Dionici:

1. Pacijent
2. Zaposlenik (medicinsko osoblje)
3. Administrator
4. Razvojni tim
 - Dora Kašik (voditelj)
 - Josip Begić
 - Erik Greblo
 - Anton Vivoda
 - Roko Krstičević
 - Katarina Mikulić
 - Marko Miletić

Aktori i njihovi funkcionalni zahtjevi:

1. Pacijent (inicijator) može:
 - (a) Predati zahtjev za terapijom i dodjelom termina iste
 - i. Ako dolazi na ponovljenu terapiju, u zahtjevu prilaže referencu na prethodno obavljenju terapiju
 - (b) Izvršiti registraciju u sustav
2. Administrator (inicijator) može:
 - (a) Upravljači podacima o pacijentima i djelatnicima
 - (b) Upravljači podacima o dostupnoj opremi i prostorijama klinike
 - (c) Vršiti postupke za zaštitu podataka
 - (d) Dodavati nove djelatnike
3. Superadministrator (inicijator) može:
 - (a) Dodijeliti djelatniku administratorskih ovlasti
 - (b) Ukloniti administratorske ovlasti djelatnika
 - (c) Verificirati registraciju korisnika

- (d) Upravlјati podacima o pacijentima i djelatnicima
- (e) Upravlјati podacima o dostupnoj opremi i prostorijama klinike
- (f) Vršiti postupke za zaštitu podataka
- (g) Dodavati nove djelatnike

4. Zaposlenik (inicijator) može:

- (a) Dodijeliti termin pacijentu
 - i. Nakon dodjele termina šalje pacijentu potrebne podatke o terminu putem elektroničke pošte
- (b) Pristupiti pregledu aktivnih prijava pacijenata
- (c) Pristupiti podatku o dostupnim djelatnicima
- (d) Pristupiti podatku o trajanju pojedinog zahvata
- (e) Pristupiti podacima o dostupnoj opremi i prostorijama klinike
- (f) Registrirati dolazak pacijenta na terapiju i rezultate nakon obavljene terapije
- (g) Prihvatiti ili odbiti zahtjev pacijenta za terminom terapije

5. Baza podataka (sudionik):

- (a) Pohranjuje podatke o korisnicima i njihovim ovlastima
- (b) Pohranjuje podatke o terapijama
- (c) Pohranjuje podatke o čestim pitanjima
- (d) Pohranjuje podatke o prostorijama i opremi

3.1.1 Obrasci uporabe

Opis obrazaca uporabe

UC1 – Registracija pacijenta

- Glavni sudionik: *Pacijent*
- Cilj: Registracija novog pacijenta u sustavu
- Sudionici: Baza podataka/Sustav
- Preduvjet: Bolesnik nema prethodno registriran račun
- Opis osnovnog tijeka:
 1. *Pacijent unosi osobne podatke, adresu elektroničke pošte i lozinku.*
 2. *Sustav provjerava ispravnost podataka.*
 3. *Sustav pohranjuje podatke i stvara korisnički račun.*
- Opis mogućih odstupanja: *Ako podaci nisu ispravni, sustav obavještava korisnika o potrebnim ispravkama.*

UC2 – Prijava pacijenta

- Glavni sudionik: *Pacijent*
- Cilj: Pristup stranici rehabilitacije nakon prijave
- Sudionici: Baza podataka
- Preduvjet: Pacijent je registriran i prijavljen u sustav.
- Opis osnovnog tijeka:
 1. *Pacijent šalje zahtjev za prijavom na sustav s korisničkim imenom i lozinkom.*
 2. *Sustav provjerava ispravnost podataka.*
 3. *Ako su podaci ispravni, pacijent je prijavljen u sustav.*
- Opis mogućih odstupanja: *Ako su uneseni neispravni podaci, sustav obavještava pacijenta i zahtijeva ispravke.*
- *Ako pacijent nije registriran, sustav obavještava korisnika da nema važeći račun.*

UC3 – Zahtjev za resetiranje lozinke

- Glavni sudionik: *Pacijent*
- Cilj: Prijava u sustav kako bi pristupio stranici rehabilitacije
- Sudionici: Baza podataka/Sustav
- Preduvjet: Korisnik ima registriran račun u sustavu
- Opis osnovnog tijeka:
 1. *Pacijent šalje zahtjev za poveznicu za resetiranje lozinke*
 2. *Sustav šalje poveznicu za resetiranje lozinke na korisnikovu adresu elektroničke pošte.*

- Opis mogućih odstupanja: *Ako se ne može pronaći korisnik s unesenom adresom e-pošte, sustav obavještava korisnika o neuspjelom zahtjevu.*

UC4 – Spremanje nove lozinke

- Glavni sudionik: *Pacijent*
- Cilj: *Spremiti novu lozinku nakon poveznice za resetiranje lozinke*
- Sudionici: *Baza podataka/Sustav*
- Preduvjet: *Korisnik je primio poveznicu za resetiranje lozinke i želi postaviti novu lozinku.*
- Opis osnovnog tijeka:
 1. *Pacijent šalje novu lozinku putem poveznice za resetiranje lozinke.*
 2. *Sustav sprema novu lozinku za korisnički račun.*

Opis mogućih odstupanja: *Ako poveznica za resetiranje lozinke nije ispravna ili je istekla, sustav obavještava korisnika o neuspjelom ažuriranju lozinke.*

UC5 – Dodavanje zaposlenika u bazu

- Glavni sudionik: *Administrator*
- Cilj: *Registrirati novog zaposlenika u sustavu*
- Sudionici: *Baza podataka/Sustav*
- Preduvjet: *Korisnik ima administratorske ovlasti i prijavljen je u sustav.*
- Opis osnovnog tijeka:
 1. *Administrator šalje zahtjev za registraciju zaposlenika s podacima o zaposleniku.*
 2. *Sustav provjerava ispravnost podataka.*
 3. *Sustav registrira zaposlenika u sustavu.*
- Opis mogućih odstupanja: *Ako podaci nisu ispravni, sustav obavještava administratora o potrebnim ispravkama.*

UC6 – Brisanje zaposlenika

- Glavni sudionik: *Administrator*
- Cilj: *Deaktivirati račun određenog zaposlenika u sustav*
- Sudionici: *Baza podataka/Sustav*
- Preduvjet: *Korisnik ima administratorske ovlasti i prijavljen je u sustav.*
- Opis osnovnog tijeka:
 1. *Administrator šalje zahtjev za brisanje zaposlenika s određenim identifikatorom.*
 2. *Sustav označava zaposlenika kao neaktivnog u bazi podataka.*

UC7 – Dodavanje učestalih pitanja (FAQ)

- Glavni sudionik: *Administrator*

- Cilj: Dodati novi FAQ u sustav
 - Sudionici: Baza podataka/Sustav
 - Preduvjet: Korisnik ima administratorske ovlasti i prijavljen je u sustav.
 - Opis osnovnog tijeka:
 1. Administrator šalje zahtjev za dodavanje novog FAQ-a s podacima o FAQ-u.
 2. Sustav provjerava ispravnost podataka.
 3. Sustav dodaje novi FAQ u sustav.
- Opis mogućih odstupanja: *Ako podaci nisu ispravni, sustav obavještava administratora o potrebnim ispravkama.*

UC8 – Prikaz svih učestalih pitanja (FAQ-a)

- Glavni sudionik: *Bilo koji korisnik*
- Cilj: Prikazati popis svih FAQ-ova u sustavu
- Sudionici: Baza podataka/Sustav
- Opis osnovnog tijeka:
 1. Korisnik šalje zahtjev za dohvaćanje svih FAQ-ova.
 2. Sustav dohvaća popis FAQ-ova iz baze podataka.
 3. Sustav vraća odgovor s popisom FAQ-ova.

UC9 – Brisanje učestalih pitanja (FAQ-a)

- Glavni sudionik: *Administrator*
- Cilj: Brisati određeni FAQ iz sustava
- Sudionici: Baza podataka/Sustav
- Preduvjet: Korisnik ima administratorske ovlasti i prijavljen je u sustav.
- Opis osnovnog tijeka:
 1. Administrator šalje zahtjev za brisanje FAQ-a s određenim identifikatorom.
 2. Sustav označava FAQ kao neaktivan u bazi podataka.

UC10 – Kreiranje sobe

- Glavni sudionik: *Zaposlenik*
 - Cilj: Kreirati novu sobu u sustavu
 - Sudionici: Baza podataka/Sustav
 - Preduvjet: Korisnik ima ovlasti zaposlenik i prijavljen je u sustav.
 - Opis osnovnog tijeka:
 1. Zaposlenik šalje zahtjev za kreiranje nove sobe s potrebnim podacima.
 2. Sustav provjerava ispravnost podataka.
- Opis mogućih odstupanja: *Ako podaci nisu ispravni, sustav obavještava administratora o potrebnim ispravkama.*

UC11 – Pregled svih soba

- Glavni sudionik: *Zaposlenik*
- Cilj: Prikazati sve sobe u sustavu
- Sudionici: Baza podataka/Sustav
- Preduvjet: Korisnik ima ovlasti zaposlenika i prijavljen je u sustav.
- Opis osnovnog tijeka:
 1. *Zaposlenik šalje zahtjev za dohvaćanje svih soba.*
 2. *Sustav dohvaća sve sobe iz baze podataka.*
 3. *Sustav vraća odgovor s popisom soba.*

UC12 – Pregled sobe po identifikacijskom broju

- Glavni sudionik: *Zaposlenik*
- Cilj: Pregledati pojedinu sobu u sustavu
- Sudionici: Baza podataka/Sustav
- Preduvjet: Korisnik ima ovlasti zaposlenika i prijavljen je u sustav.
- Opis osnovnog tijeka:
 1. *Zaposlenik šalje zahtjev za dohvat sobe s određenim identifikatorom.*
 2. *Sustav provjerava postoji li soba s navedenim identifikacijskim brojem.*
 3. *Ako postoji, sustav prikazuje podatke o sobi.*

Opis mogućih odstupanja: *Ako soba s navedenim identifikacijskim brojem ne postoji, sustav obavještava zaposlenika.*

UC13 – Brisanje sobe

- Glavni sudionik: *Zaposlenik*
- Cilj: Obrisati sobu iz sustava
- Sudionici: Baza podataka/Sustav
- Preduvjet: Korisnik ima ovlasti zaposlenika i prijavljen je u sustav.
- Opis osnovnog tijeka:
 1. *Zaposlenik šalje zahtjev za brisanje sobe s određenim identifikacijskim brojem.*
 2. *Sustav provjerava postoji li soba s navedenim identifikacijskim brojem.*
 3. *Ako postoji, sustav briše sobu.*

Opis mogućih odstupanja: *Ako podaci nisu ispravni, sustav obavještava administratora o potrebnim ispravkama.*

UC14 – Kreiranje opreme

- Glavni sudionik: *Zaposlenik*
- Cilj: Kreirati novu opremu u sustavu.

- Sudionici: Baza podataka/Sustav
 - Preduvjet: Korisnik ima ovlasti zaposlenika i prijavljen je u sustav.
 - Opis osnovnog tijeka:
 1. *Korisnik šalje zahtjev za kreiranje nove opreme s potrebnim podacima..*
 2. *Sustav provjerava ispravnost podataka..*
 3. *Sustav kreira novu opremu u sustav.*
- Opis mogućih odstupanja: *Ako podaci nisu ispravni, sustav obavještava zaposlenika o potrebnim ispravkama.*

UC15 – Pregled svih oprema

- Glavni sudionik: *Zaposlenik*
- Cilj: Prikazati sve sobe u sustavu
- Sudionici: Baza podataka/Sustav
- Preduvjet: Korisnik ima ovlasti zaposlenika i prijavljen je u sustav.
- Opis osnovnog tijeka:
 1. *Zaposlenik šalje zahtjev za dohvaćanje svih oprema.*
 2. *Sustav dohvaća svu opremu iz baze podataka.*
 3. *Sustav vraća odgovor s popisom opreme.*

UC16 – Pregled opreme po identifikacijskom broju

- Glavni sudionik: *Zaposlenik*
 - Cilj: Pregledati pojedinu opremu u sustavu
 - Sudionici: Baza podataka/Sustav
 - Preduvjet: Korisnik ima ovlasti zaposlenika i prijavljen je u sustav.
 - Opis osnovnog tijeka:
 1. *Zaposlenik šalje zahtjev za dohvat opreme s određenim identifikatorom.*
 2. *Sustav provjerava postoji li oprema s navedenim identifikacijskim brojem.*
 3. *Ako postoji, sustav prikazuje podatke o opremi.*
- Opis mogućih odstupanja: *Ako oprema s navedenim identifikacijskim brojem ne postoji, sustav obavještava zaposlenika.*

UC17 – Brisanje opreme

- Glavni sudionik: *Zaposlenik*
- Cilj: Obrisati opremu iz sustava
- Sudionici: Baza podataka/Sustav
- Preduvjet: Korisnik ima ovlasti zaposlenika i prijavljen je u sustav.
- Opis osnovnog tijeka:
 1. *Zaposlenik šalje zahtjev za brisanje opreme s određenim identifikacijskim brojem.*

2. Sustav sustav provjerava postoji li oprema s navedenim identifikacijskim brojem.
3. Ako postoji, sustav briše opremu.

UC18 – Prikaz svih pacijenata

- Glavni sudionik: *Administrator*
- Cilj: Prikazati popis svih pacijenata u sustavu
- Sudionici: Baza podataka/Sustav
- Preduvjet: Korisnik ima administratorske ovlasti i prijavljen je u sustav.
- Opis osnovnog tijeka:
 1. *Administrator šalje zahtjev za dohvaćanje svih pacijenata.*
 2. *Sustav dohvaća popis pacijenata iz baze podataka.*
 3. *Sustav vraća odgovor s popisom pacijenata.*

UC19 – Deaktivacija pacijentovog računa

- Glavni sudionik: *Pacijent, Zaposlenik, Admin*
- Cilj: Deaktivirati račun određenog pacijenta u sustavu
- Sudionici: Baza podataka/Sustav
- Preduvjet: Korisnik je prijavljen u sustav.
- Opis osnovnog tijeka:
 1. *Korisnik šalje zahtjev za deaktivaciju pacijentovog računa s određenim identifikatorom.*
 2. *Sustav označava pacijenta kao neaktivnog u bazi podataka.*

UC20 – Kreiranje terapije za pacijenta

- Glavni sudionik: *Pacijent, Zaposlenik*
- Cilj: Dodati novu terapiju za određenog pacijenta.
- Sudionici: Baza podataka/Sustav
- Preduvjet: Korisnik je prijavljen u sustav.
- Opis osnovnog tijeka:
 1. *Korisnik šalje zahtjev za dodavanje nove terapije s podacima o terapiji*
 2. *Sustav provjerava ispravnost podataka.*
 3. *Sustav dodaje novu terapiju za određenog pacijenta u sustavu.*

Opis mogućih odstupanja: *Ako podaci nisu ispravni, sustav obavještava korisnika o potrebnim ispravkama.*

UC21 – Prikaz svih terapija za pacijenta

- Glavni sudionik: *Pacijent*
- Cilj: Prikazati popis svih terapija za određenog pacijenta
- Sudionici: Baza podataka/Sustav
- Preduvjet: Korisnik je prijavljen u sustav.
- Opis osnovnog tijeka:
 1. *Korisnik šalje zahtjev za dohvaćanje svih terapija za određenog pacijenta.*
 2. *Sustav dohvaća popis terapija iz baze podataka za određenog pacijenta.*
 3. *Sustav vraća odgovor s popisom terapija.*

UC22 – Unos rezultata terapije

- Glavni sudionik: *Zaposlenik*
 - Cilj: Unijeti rezultate terapije u sustav.
 - Sudionici: Baza podataka/Sustav
 - Preduvjet: Korisnik ima ovlasti zaposlenika i prijavljen je u sustav.
 - Opis osnovnog tijeka:
 1. *Zaposlenik šalje zahtjev za unos rezultata terapije s podacima o rezultatima.*
 2. *Sustav provjerava ispravnost podataka.*
 3. *Sustav unosi rezultate terapije u sustav.*
- Opis mogućih odstupanja: *Ako podaci nisu ispravni, sustav obavještava zaposlenika o potrebnim ispravkama.*

UC23 – Dodjela administratorskih ovlasti

- Glavni sudionik: *Superadministrator*
 - Cilj: Dodjela administratorskih ovlasti radi lakšeg upravljanja sustavom
 - Sudionici: Baza podataka/Sustav
 - Preduvjet: Korisnik ima ovlasti superadministratora i prijavljen je u sustav.
 - Opis osnovnog tijeka:
 1. *Superadministrator šalje zahtjev za dodjelu administratorskih ovlasti određenom korisniku.*
 2. *Sustav provjerava ispravnost zahtjeva i potvrđuje administratorske ovlasti korisniku.*
 3. *Sustav ažurira informacije o ovlastima korisnika u bazi podataka.*
- Opis mogućih odstupanja: *Ako podaci nisu ispravni ili korisnik već ima administratorske ovlasti, sustav obavještava superadministratora o potrebnim ispravkama ili prekoračenju ovlasti.*

UC24 – Uklanjanje administratorskih ovlasti

- Glavni sudionik: *Superadministrator*
- Cilj: Ukloniti administratorskih ovlasti korisniku u sustavu
- Sudionici: Baza podataka/Sustav
- Preduvjet: Korisnik ima ovlasti superadministratora i prijavljen je u sustav.
- Opis osnovnog tijeka:
 1. *Administrator šalje zahtjev za uklanjanje administratorskih ovlasti određenom korisniku.*
 2. *Sustav provjerava ispravnost zahtjeva i uklanja administratorske ovlasti korisniku.*
 3. *Sustav ažurira informacije o ovlastima korisnika u bazi podataka.*

Opis mogućih odstupanja: *Ako podaci nisu ispravni ili korisnik nema administratorske ovlasti, sustav obavještava administratora o potrebnim ispravkama ili nedostatku ovlasti za uklanjanje.*

UC25 – Prikaz svih zaposlenika

- Glavni sudionik: *Administrator*
- Cilj: Prikazati popis svih zaposlenika u sustavu
- Sudionici: Baza podataka/Sustav
- Preduvjet: Korisnik ima administratorske ovlasti i prijavljen je u sustav.
- Opis osnovnog tijeka:
 1. *Administrator šalje zahtjev za dohvaćanje svih djelatnika.*
 2. *Sustav dohvaća popis zaposlenika iz baze podataka.*
 3. *Sustav vraća odgovor s popisom zaposlenika.*

[illegible]

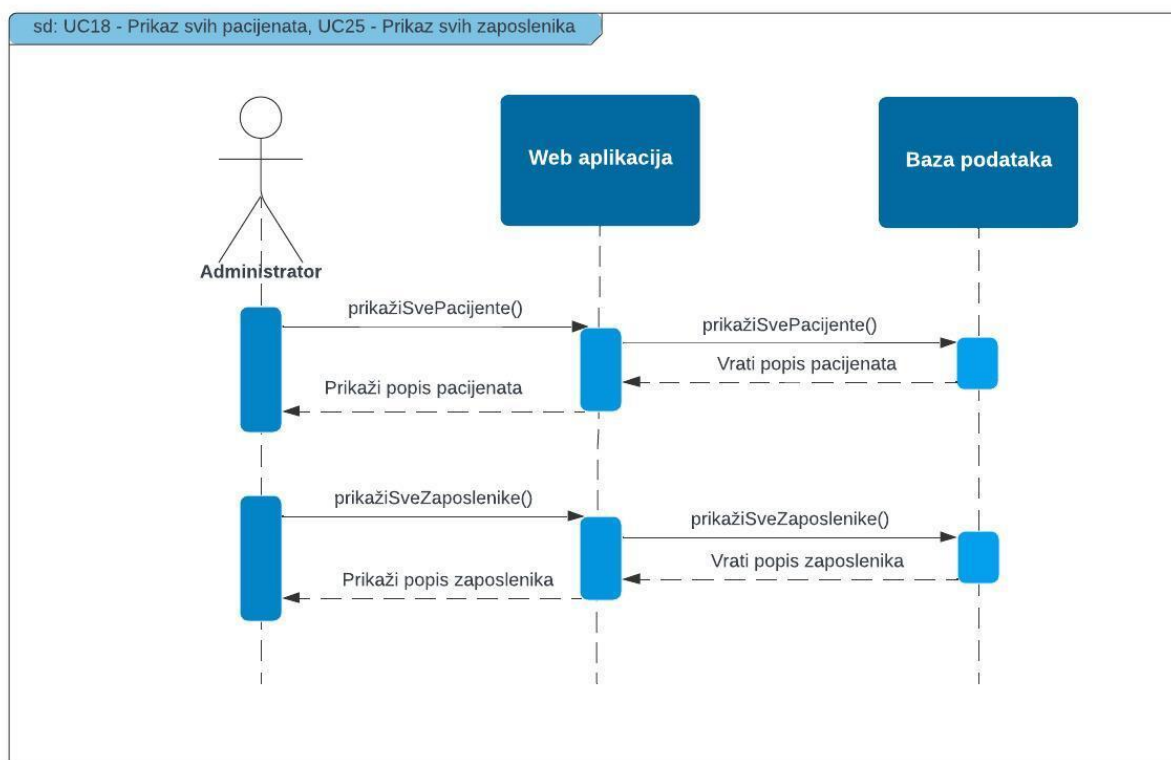
Slika 1 - Dijagram obrazaca uporabe

3.1.2 Sekvencijski dijagrami

U sljedećem dijelu prikazani su sekvencijski dijagrami i njihovi opisi.

3.1.2.1 Obrazac uporabe UC18 – Prikaz svih pacijenata i UC25 – Prikaz svih zaposlenika

Korisnik odabire opciju "Prikaži sve pacijente" ili "Prikaži sve zaposlenike" unutar web-aplikacije. Web-aplikacija iz baze podataka dohvaća sve relevantne informacije o pacijentima odnosno zaposlenicima. Ako baza podataka vrati informacije o pacijentima, web-aplikacija ih prikazuje korisniku, omogućujući mu pregled detalja poput imena, adrese, kontakt informacija itd. U suprotnom, ispisuje poruku "Nema dostupnih pacijenata". Ako korisnik odabere opciju "Prikaži sve zaposlenike", web-aplikacija prikazuje informacije o svim zaposlenicima, uključujući njihova imena, pozicije, kontakt informacije itd. U suprotnom, ispisuje poruku "Nema dostupnih zaposlenika".



Slika 2 - Sekvencijski dijagram za UC18/25

3.1.2.2 Obrazac uporabe UC8 – Prikaz svih učestalih pitanja, UC11 – Pregled svih soba i UC15 – Prikaz svih oprema

Korisnik odabire opciju "Učestala pitanja" unutar web-aplikacije. Web-aplikacija iz baze podataka dohvaća sve učestala pitanja. Ako postoje učestala pitanja, aplikacija ih prikazuje korisniku, pružajući odgovore i relevantne informacije. Ako nema učestalih pitanja, ispisuje poruku "Nema dostupnih učestalih pitanja."

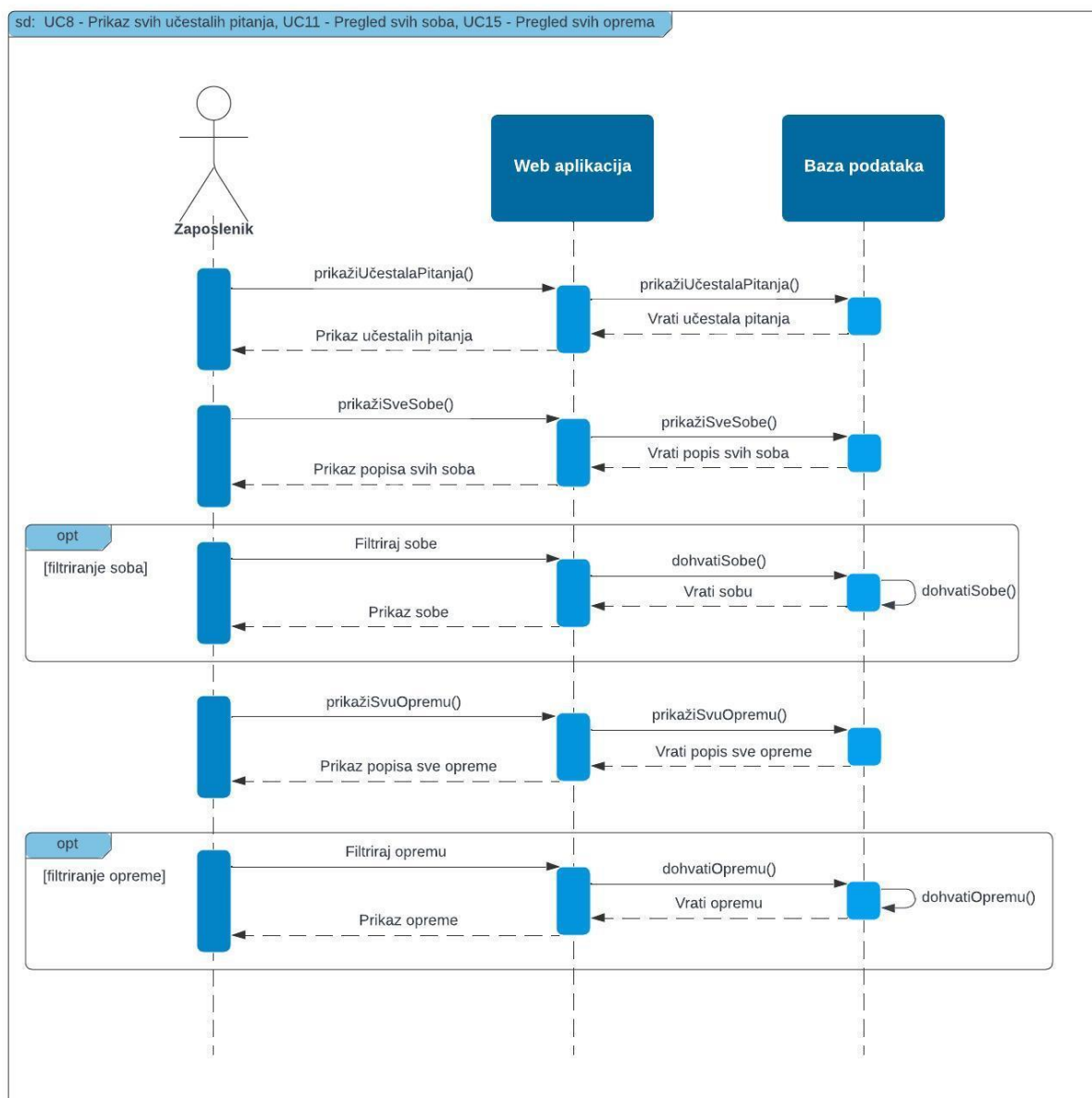
Dodatna funkcionalnost: Korisnik ima mogućnost filtriranja pitanja po kategorijama ili ključnim riječima.

Korisnik odabire opciju "Pregled svih soba" unutar web-aplikacije. Web-aplikacija iz baze podataka dohvaća sve informacije o sobama. Ako postoje sobe, aplikacija ih prikazuje korisniku s detaljima poput broja sobe, vrste, dostupnih sadržaja itd. Ako nema dostupnih soba, ispisuje poruku "Nema dostupnih soba."

Dodatna funkcionalnost: Korisnik može filtrirati sobe prema vrsti, kapacitetu ili dostupnim sadržajima.

Korisnik odabire opciju "Prikaz svih oprema" unutar web-aplikacije. Web-aplikacija iz baze podataka dohvaća informacije o svim dostupnim opremama. Ako postoje informacije o opremama, aplikacija ih prikazuje korisniku, pružajući detalje o vrstama opreme, količini dostupnih itd. Ako nema dostupnih informacija o opremama, ispisuje poruku "Nema dostupne opreme."

Dodatna funkcionalnost: Korisnik može filtrirati opremu prema vrsti ili dostupnosti.



Slika 3 - Sekvencijski dijagram za UC8/11/15

3.2 Ostali zahtjevi

1. Sustav mora podržavati više različitih uloga korisnika.

- Sustav treba omogućiti definiranje različitih razina pristupa i funkcionalnosti ovisno o ulozi korisnika (Administrator, Liječnik, Pacijent, itd.).

2. Korisnički podaci moraju biti sigurno pohranjeni u sustavu.

- Sustav treba koristiti kriptiranje za pohranu i prijenos osjetljivih korisničkih podataka kako bi osigurao privatnost i sigurnost informacija.

3. Brza i pouzdana veza s bazom podataka.

- Sustav treba osigurati pouzdanu vezu s bazom podataka kako bi pristup podacima bio brz i otporan na vanjske pogreške.

4. Odgovarajuće vrijeme izvršavanja za upite prema bazi podataka.

- Izvršavanje upita koji pristupaju bazi podataka ne smije trajati dulje od definiranog vremena, kako bi se osigurala učinkovitost sustava.

5. Podrška za hrvatsku abecedu u korisničkom sučelju i pri unosu teksta.

- Korisničko sučelje i sustav trebaju podržavati hrvatsku abecedu pri unosu, prikazu i obradi tekstualnih podataka.

6. Web aplikacija implementirana korištenjem objektno-orijentiranih jezika.

- Sustav treba biti implementiran kao web aplikacija koristeći suvremene objektno-orijentirane jezike kako bi bio skalabilan i održiv.

7. Intuitivan i lagan za korištenje.

- Korisničko sučelje treba biti intuitivno i jednostavno za korištenje kako bi korisnici mogli efikasno koristiti sustav bez potrebe za dodatnim uputama.

8. Nadogradnja sustava bez narušavanja postojećih funkcionalnosti.

- Nadogradnje sustava trebaju biti provedene bez negativnog utjecaja na postojeće funkcionalnosti kako bi korisnici nastavili neometano koristiti sustav.

9. Zaštita od neispravnog korištenja korisničkog sučelja.

- Neispravno korištenje korisničkog sučelja ne smije narušiti funkcionalnosti i rad sustava. Sustav treba biti otporan na potencijalne greške ili zloupotrebu od strane korisnika.

4. Arhitektura i dizajn sustava

Rehub predstavlja sofisticirano arhitektonsko rješenje koje integrira PostgreSQL bazu podataka, Spring Boot za backend logiku, React za korisnički sučelje, te Tailwind CSS za moderno i responzivno oblikovanje. Sustav je implementiran na cloud platformama Render za backend i bazu te Netlify za frontend, koristeći njihove usluge za brzo, skalabilno, i pouzdan deploying.

Baza podataka: PostgreSQL

PostgreSQL, snažan sustav za upravljanje relacijskim bazama podataka, predstavlja ključnu komponentu u arhitekturi ove web aplikacije. Struktura baze podataka temeljito je dizajnirana kako bi zadovoljila specifične zahtjeve aplikacije. Kroz pažljivu integraciju s PostgreSQL-om, aplikacija ostvaruje brz i učinkovit pristup podacima, čime se osigurava optimalna performansa. Ova konfiguracija baze podataka predstavlja ključnu komponentu cjelokupnog arhitektonskog rješenja, pružajući čvrstu temeljnu strukturu za rad backend sustava.

Backend: Spring Boot

Spring Boot, kao Java-based framework, pruža snažan temelj za razvoj backend logike. Ova web aplikacija koristi Spring Boot za stvaranje RESTful API-ja koji komunicira s PostgreSQL bazom podataka. Ovo rješenje omogućuje efikasno upravljanje podacima i pruža mogućnost proširivosti sustava kroz modularnost i lakoću integracije.

Frontend: React

React, centralni JavaScript biblioteka, određuje arhitekturu frontend dijela ove aplikacije. Modularne React komponente omogućuju čist i jednostavan kod, prilagodljiv specifičnim zahtjevima sučelja. Ovaj pristup omogućuje stvaranje brzog, dinamičkog korisničkog sučelja koje se lako održava. Kroz React, osigurava se fluidna navigacija kroz aplikaciju, pridonoseći ukupnom intuitivnom korisničkom iskustvu.

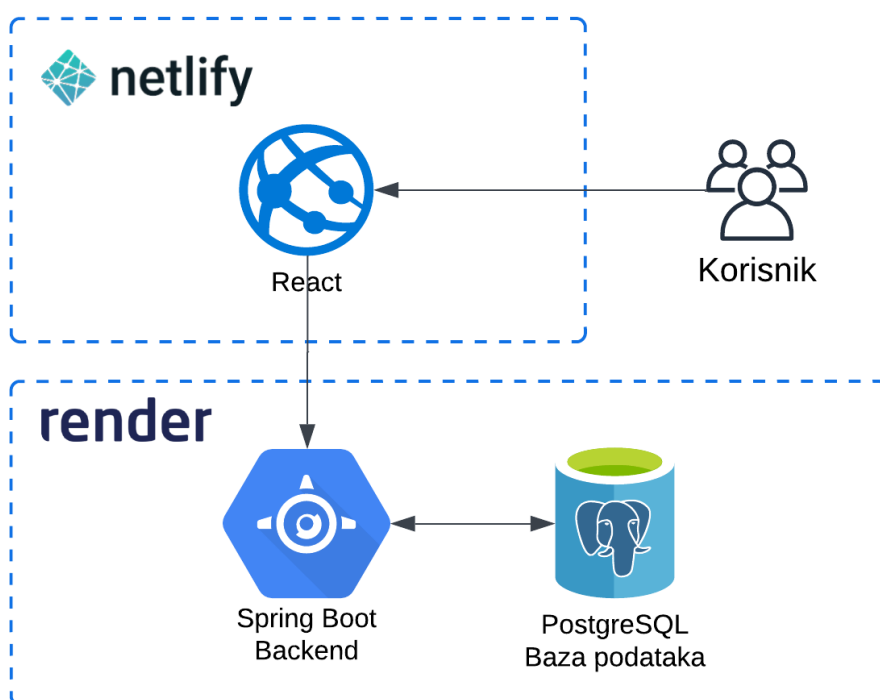
Frontend Integracija: Tailwind CSS, Axios, React-Router-Dom, reCAPTCHA

Frontend aplikacije unaprijeđen je integracijom Tailwind CSS za brzo oblikovanje sučelja, Axios za efikasnu komunikaciju s backendom, React-Router-Dom za fluidnu navigaciju te reCAPTCHA sustava radi dodatne sigurnosti. Tailwind CSS pridonosi estetskom izgledu, Axios omogućuje dinamičku interakciju s backendom, React-Router-Dom olakšava upravljanje rutama, dok reCAPTCHA sprječava zlonamjerne aktivnosti. Kroz ovu integraciju postiže se uravnoteženo frontend rješenje koje kombinira funkcionalnost, estetiku i sigurnost korisničkog iskustva

Deployment: Render za backend i Netlify za frontend

Backend i baze podataka su postavljeni na Render platformu za optimalno upravljanje resursima. Render osigurava automatsko skaliranje i brzu dostupnost aplikacije koristeći njihove cloud usluge. Na renderu je također podešeno kontinuirano postavljanje koje omogućuje automatsko ažuriranje softvera. Frontend je zasebno postavljen na Netlify platformi, koja pruža brzo i globalno dostupno CDN (Content Delivery Network).

Rehub predstavlja integrirano arhitektonsko rješenje koje kombinira najbolje prakse u razvoju aplikacija, pružajući stabilnost, skalabilnost i visoku razinu korisničkog iskustva. Korištenje navedenih tehnologija i servisa osigurava da aplikacija bude spremna za izazove suvremenog web razvoja.



Slika 4 - Dijagram arhitekture

4.1 Baza podataka

U bazi su stvorene relacije koje olakšavaju jednostavno rukovanje podacima, uključujući dodavanje, brisanje, izmjenu i dohvaćanje podataka. Svi entiteti i relacije su organizirani prema trećoj normalnoj formi kako bi se izbjegla redundancija podataka. Za bolje vizualiziranje strukture baze podataka, napravljen je relacijski dijagram koji je prikazan u nastavku.

Baza podataka se sastoji od tablica (relacija) koje su definirane svojim imenom i skupom atributa. Baza podataka ove aplikacije sastoji se od sljedećih entiteta:

- *rehub_user*
- *employee*
- *user_role*
- *role*
- *patient*
- *therapy*
- *therapy_result*
- *doctor*
- *appointment*
- *room*
- *equipment*
- *faq*
- *personal_data*
- *flyway_history_schema*

4.1.1 Opis tablica

Prva ćelija svake tablice označava njeno ime. U prvom stupcu navedeni su atributi tog entiteta, u drugom stupcu naveden je tip varijable, a u trećem opis svakog pojedinog atributa. Svjetlozelenom bojom označeni su primarni ključevi. Svjetlo plavom označeni su strani ključevi.

Rehub_user

Entitet označava korisnika aplikacije te sadrži attribute: id, username, password.

REHUB_USER	TIP	OPIS
ID	BIGSERIAL	JEDINSTVENI IDENTIFIKATOR KORISNIKA
USERNAME	VARCHAR	KORISNIČKO IME S KOJIM SE KORISNIK PRIJAVLJUJE
PASSWORD	VARCHAR	LOZINKA S KOJOM SE KORISNIK PRIJAVLJUJE

Employee

Entitet označava osobe zaposlene u klinici. Atributi ovoga entiteta su: id, first_name, last_name, phone_number, profession, date_of_birth, gender, created_at, last_modified_at, user_id.

EMPLOYEE	TIP	OPIS
ID	BIGSERIAL	JEDINSTVENI IDENTIFIKATOR ZAPOSLENIKA
FIRST_NAME	VARCHAR	IME ZAPOSLENIKA
LAST_NAME	VARCHAR	PREZIME ZAPOSLENIKA
PHONE_NUMBER	VARCHAR	BROJ MOBITELA ZAPOSLENIKA
PROFESSION	VARCHAR	STRUKA ZAPOSLENIKA
DATE_OF_BIRTH	DATE	DATUM ROĐENJA ZAPOSLENIKA
GENDER	VARCHAR	SPOL ZAPOSLENIKA
CREATED_AT	TIMESTAMP	TRENUK KREIRANJA
LAST_MODIFIED_AT	TIMESTAMP	TRENUK ZADNJE PROMJENE
USER_ID	VARCHAR	JEDINSTVENI IDENTIFIKATOR KORISNIKA

User_role

Entitet označava vrstu korisnika aplikacije, a njegovi atributi su: user_id, role_id.

USER_ROLE	TIP	OPIS
USER_ID	BIGSERIAL	JEDINSTVENI IDENTIFIKATOR KORISNIKA
ROLE_ID	BIGSERIAL	JEDINSTVENI IDENTIFIKATOR VRSTE KORISNIKA

Role

Entitet označava ulogu i sastoji se od atributa: id, name.

ROLE	TIP	OPIS
ID	BIGSERIAL	JEDINSTVENI IDENTIFIKATOR ULOGE
NAME	VARCHAR	IME ULOGE

Patient

Entitet predstavlja pacijenta koji je korisnik aplikacije. Atributi su: id, first_name, last_name, gender, phone_number, date_of_birth, created_at, last_modified_at, user_id.

PATIENT	TIP	OPIS
ID	BIGSERIAL	JEDINSTVENI IDENTIFIKATOR PACIJENTA
FIRST_NAME	VARCHAR	IME PACIJENTA
LAST_NAME	VARCHAR	PREZIME PACIJENTA
GENDER	VARCHAR	SPOL PACIJENTA
PHONE_NUMBER	VARCHAR	BROJ MOBITELA PACIJENTA
DATE_OF_BIRTH	DATE	DATUM ROĐENJA PACIJENTA
CREATED_AT	TIMESTAMP	TRENUTAK KREIRANJA
LAST_MODIFIED_AT	TIMESTAMP	TRENUTAK ZADNJE PROMJENE
USER_ID	INT	JEDINSTVENI IDENTIFIKATOR KORISNIKA

Therapy

Entitet opisuje terapiju dodijeljenu pacijentu. Njeni atributi su: id, type, request, status, ref_id, created_at, last_modified_at, patient_id, room_id, doctor_id, appointment_id, therapy_result_id.

THERAPY	TIP	OPIS
ID	BIGSERIAL	JEDINSTVENI IDENTIFIKATOR TERAPIJE
TYPE	VARCHAR	VRSTA TERAPIJE
REQUEST	VARCHAR	ZAHTJEV ZA TERAPIJOM
STATUS	VARCHAR	STATUS TERAPIJE
REF_ID	INT	JEDINSTVENI IDENTIFIKATOR
CREATED_AD	TIMESTAMP	TRENUK KREIRANJA
LAST_MODIFIED_AT	TIMESTAMP	TRENUK ZADNJE PROMJENE
PATIENT_ID	BIGSERIAL	JEDINSTVENI IDENTIFIKATOR PACIJENTA
ROOM_ID	BIGSERIAL	JEDINSTVENI IDENTIFIKATOR SOBE
DOCTOR_ID	BIGSERIAL	JEDINSTVENI IDENTIFIKATOR DOKTORA
APPOINTMENT_ID	BIGSERIAL	JEDINSTVENI IDENTIFIKATOR TERMINA
THERAPY_RESULT_ID	BIGSERIAL	JEDINSTVENI IDENTIFIKATOR REZULTATA TERAPIJE

Doctor

Entitet predstavlja doktora koji je korisnik aplikacije. Atributi su: id, first_name, last_name, pin, phone_number, speciality.

DOCTOR	TIP	OPIS
ID	BIGSERIAL	JEDINSTVENI IDENTIFIKATOR DOKTORA
FIRST_NAME	VARCHAR	IME DOKTORA
LAST_NAME	VARCHAR	PREZIME DOKTORA
PIN	VARCHAR	PIN DOKTORA
PHONE_NUMBER	VARCHAR	BROJ MOBITELA DOKTORA
SPECIALITY	VARCHAR	DOKTOROVA SPECIJALIZACIJA

Appointment

Entitet predstavlja termin na koji se pacijent naručio, a atributi su: id, start_at, end_at.

APPOINTMENT	TIP	OPIS
ID	BIGSERIAL	JEDINSTVENI IDENTIFIKATOR TERMINA
START_AT	TIMESTAMP	POČETAK TERMINA
END_AT	TIMESTAMP	KRAJ TERMINA

Therapy_result

Ovaj entitet predstavlja rezultat terapije, a definiran je sljedećim atributima: id, status, result.

THERAPY_RESULT	TIP	OPIS
ID	BIGSERIAL	JEDINSTVENI IDENTIFIKATOR REZULTATA TERAPIJE
STATUS	VARCHAR	STATUS TERAPIJE
RESULT	VARCHAR	KONAČNI REZULTAT TERAPIJE

Room

Entitet predstavlja sobu u kojoj se odvija terapija, a njeni atributi su: id, label, capacity, status, special_message.

ROOM	TIP	OPIS
ID	BIGSERIAL	JEDINSTVENI IDENTIFIKATOR SOBE
LABEL	VARCHAR	OZNAKA SOBE
CAPACITY	INT	KAPACITET SOBE
STATUS	VARCHAR	STATUS O KORIŠTENJU SOBE
SPECIAL_MESSAGE	VARCHAR	NAPOMENA

Equipment

Ovaj entitet se odnosi na opremu koja se koristi prilikom terapije, a opisuju ju: id, name, status, special_message, room_id.

EQUIPMENT	TIP	OPIS
ID	BIGSERIAL	JEDINSTVENI IDENTIFIKATOR OPREME
NAME	VARCHAR	IME OPREME
STATUS	VARCHAR	STATUS O KORIŠTENJU OPREME
SPECIAL_MESSAGE	VARCHAR	NAPOMENA
ROOM_ID	INT	JEDINSTVENI IDENTIFIKATOR SOBE

Personal_data

Ovaj entitet sadrži osobne podatke o korisniku aplikacije, a atributi su mu: id, first_name, last_name, pin, phin i date_of_birth.

PERSONAL_DATA	TIP	OPIS
ID	<i>BIGSERIAL</i>	JEDINSTVENI IDENTIFIKATOR KORISNIKA
FIRST_NAME	VARCHAR	IME KORISNIKA
LAST_NAME	VARCHAR	PREZIME KORISNIKA
PIN	VARCHAR	OIB KORISNIKA
PHIN	VARCHAR	MATIČNI BROJ KORISNIKA
DATE_OF_BIRTH	DATE	DATUM ROĐENJA KORISNIKA

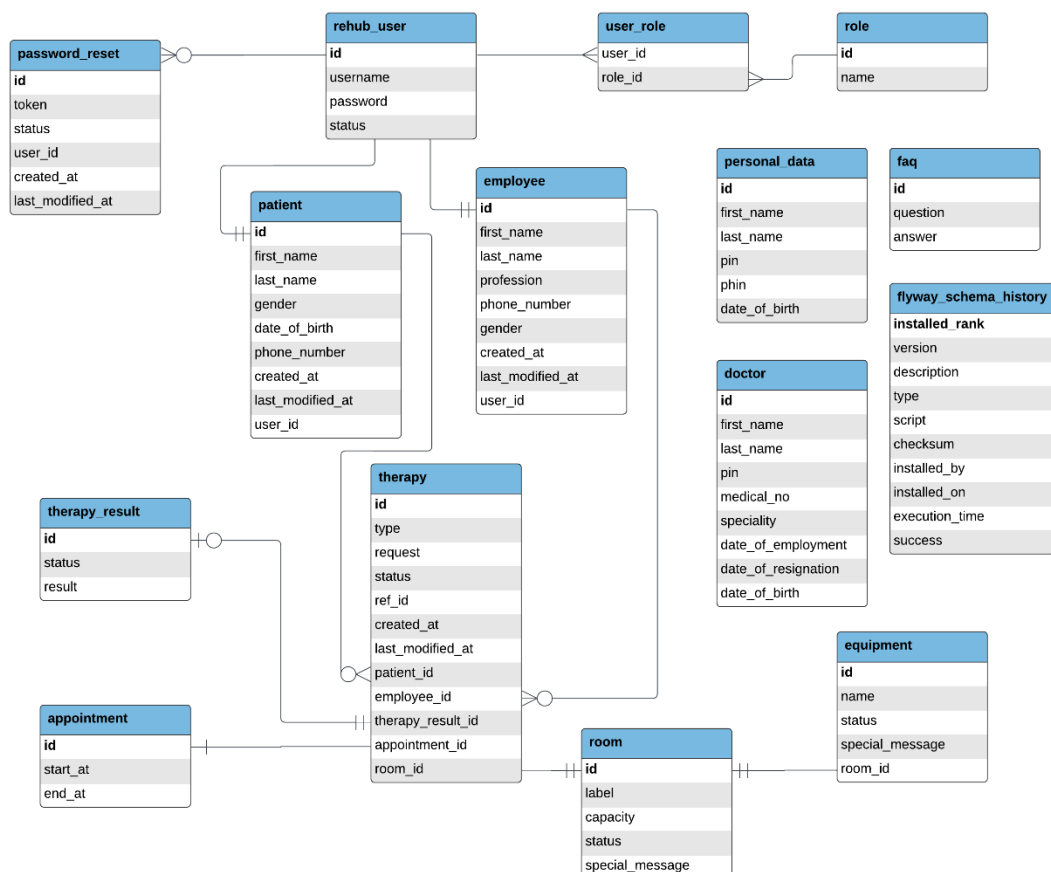
Faq

Entitet se odnosi na često postavljena pitanja i sastoji se od sljedećih atributa: id, question te answer.

FAQ	TIP	OPIS
ID	<i>BIGSERIAL</i>	JEDINSTVENI IDENTIFIKATOR PITANJA I ODGOVORA
QUESTION	VARCHAR	PITANJE

ANSWER	VARCHAR	ODGOVOR
--------	---------	---------

4.1.2 Dijagram baze podataka



Slika 5 - Dijagram baze podataka

4.2 Dijagram razreda

Dijagramom razreda prikazujemo razrede u sustavu, njihove attribute i metode te veze između razreda koji se nasljeđuju ili međusobno komuniciraju. U nastavku slijede dijagrami čiji razredi imaju sličnu funkcionalnost i razinu apstrakcije

EntityClassDiagram - prikazuje razrede koji pripadaju sloju Model. Svaki od razreda se preslikava u odgovarajuću tablicu u bazi

RepositoryClassDiagram – prikazuje razrede koji pripadaju sloju Repository. Ovaj sloj komunicira sa bazom podataka i sa slojem Service.

ServiceClassDiagram – prikazuje razrede koji pripadaju sloju Service. Ovaj sloj komunicira sa slojem Repository i Controller.

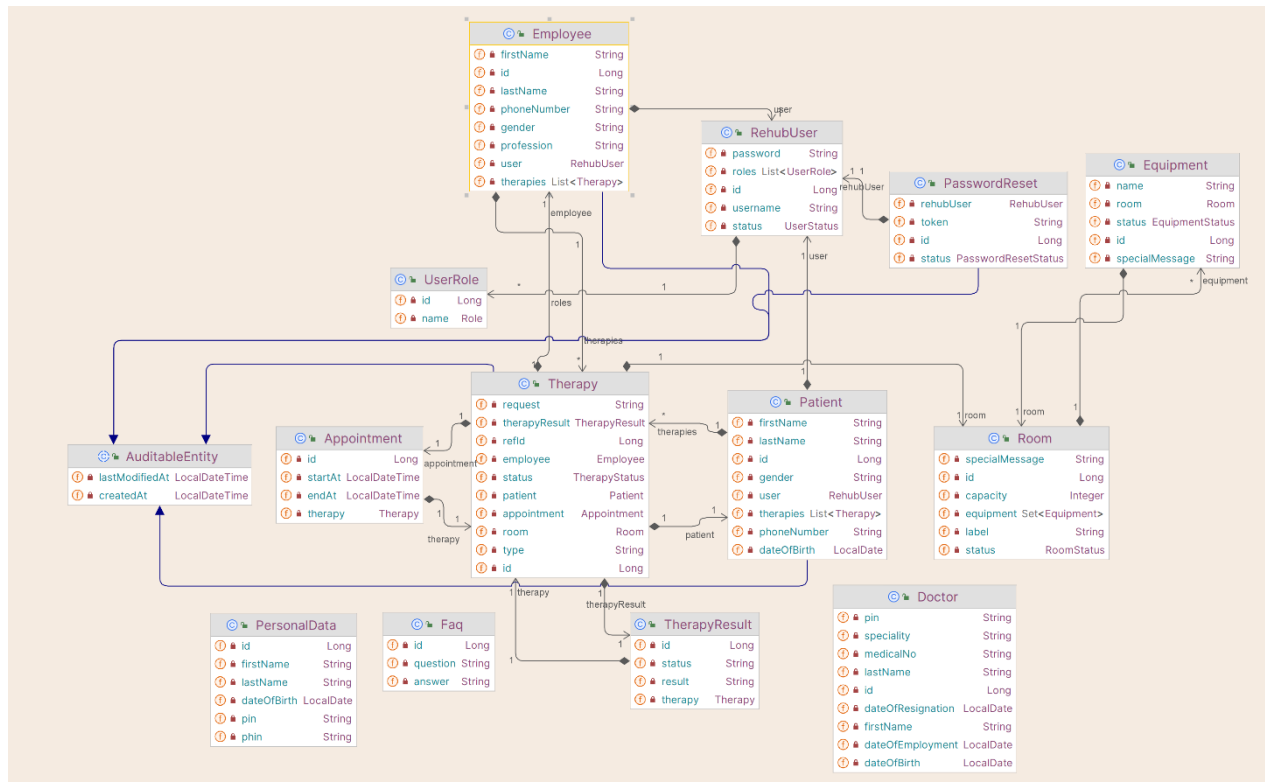
ControllerClassDiagram – prikazuje razrede sloja Controller. Ovaj sloj komunicira sa slojem Service i s frontend dijelom aplikacije.

EnumClassDiagram – prikazuje enumeracije koje koristi sloj Model.

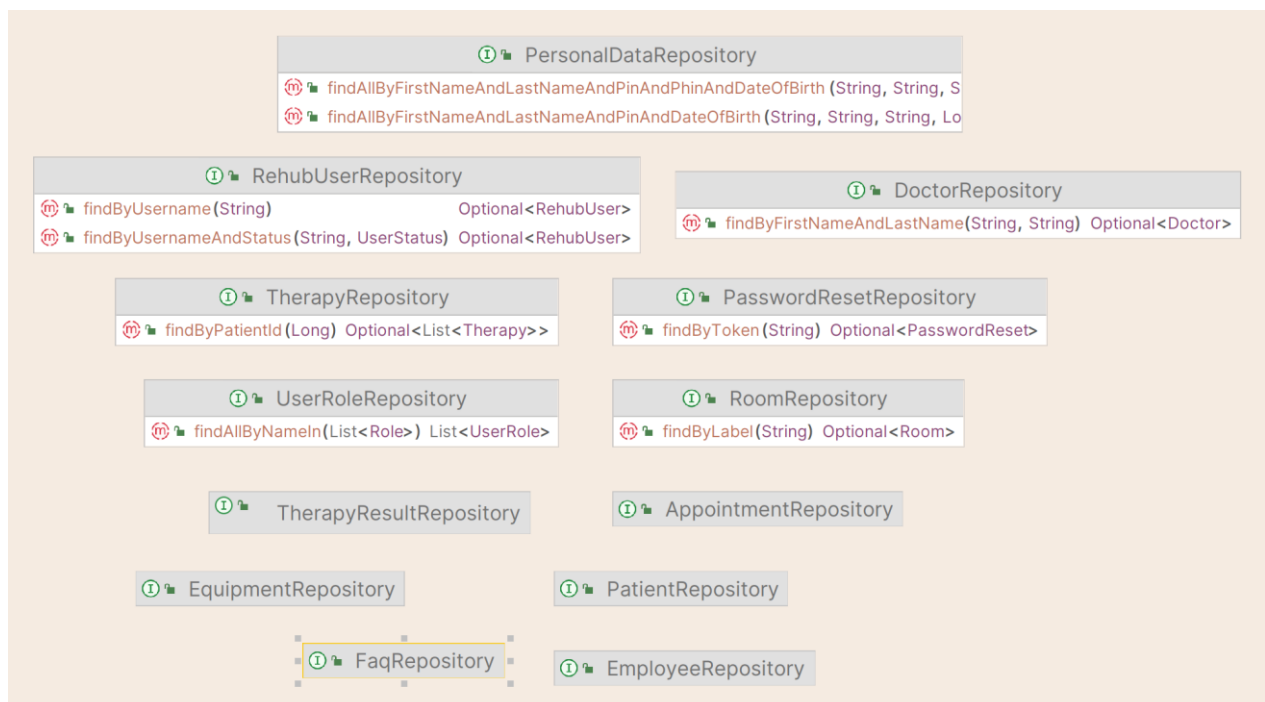
ConfigClassDiagram – prikazuje razrede sloja Config. Ovaj sloj služi za sigurnosne potrebe i komuniciranje putem e-maila.

RequestClassDiagram – prikazuje razrede sloja Request. Ovaj sloj komunicira sa slojem Controller u svrhu validacije Post Request-a.

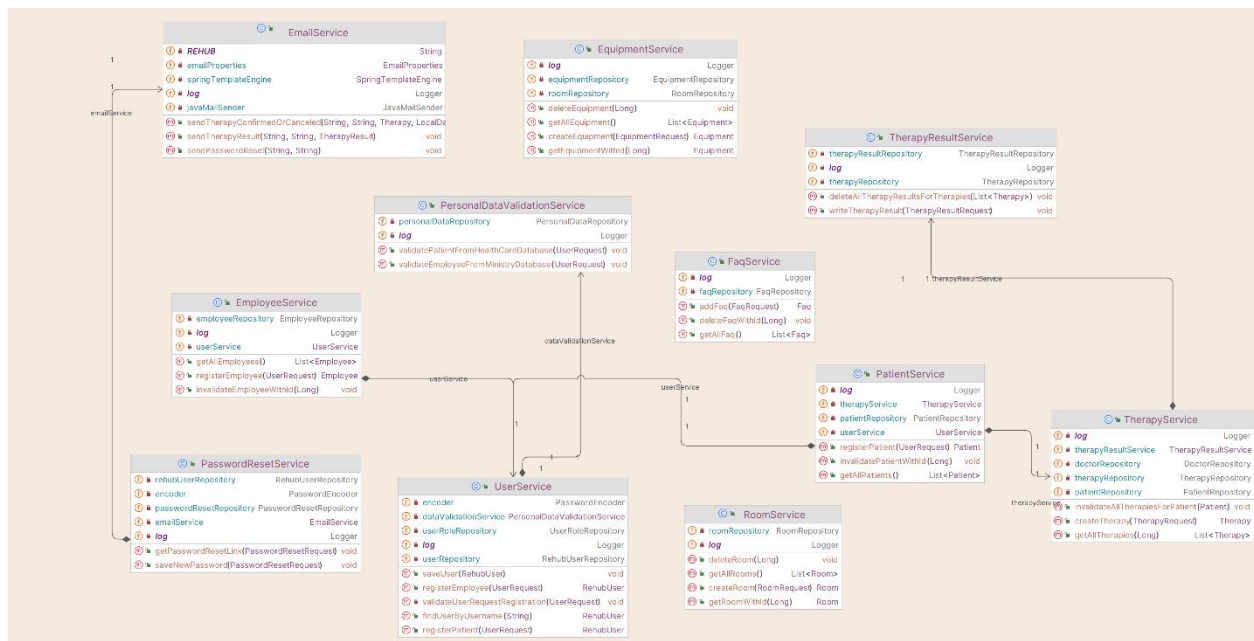
ResponseClassDiagram – prikazuje razrede sloja Response. Ovaj sloj komunicira sa slojem Controller u svrhu vraćanja Response poruke.



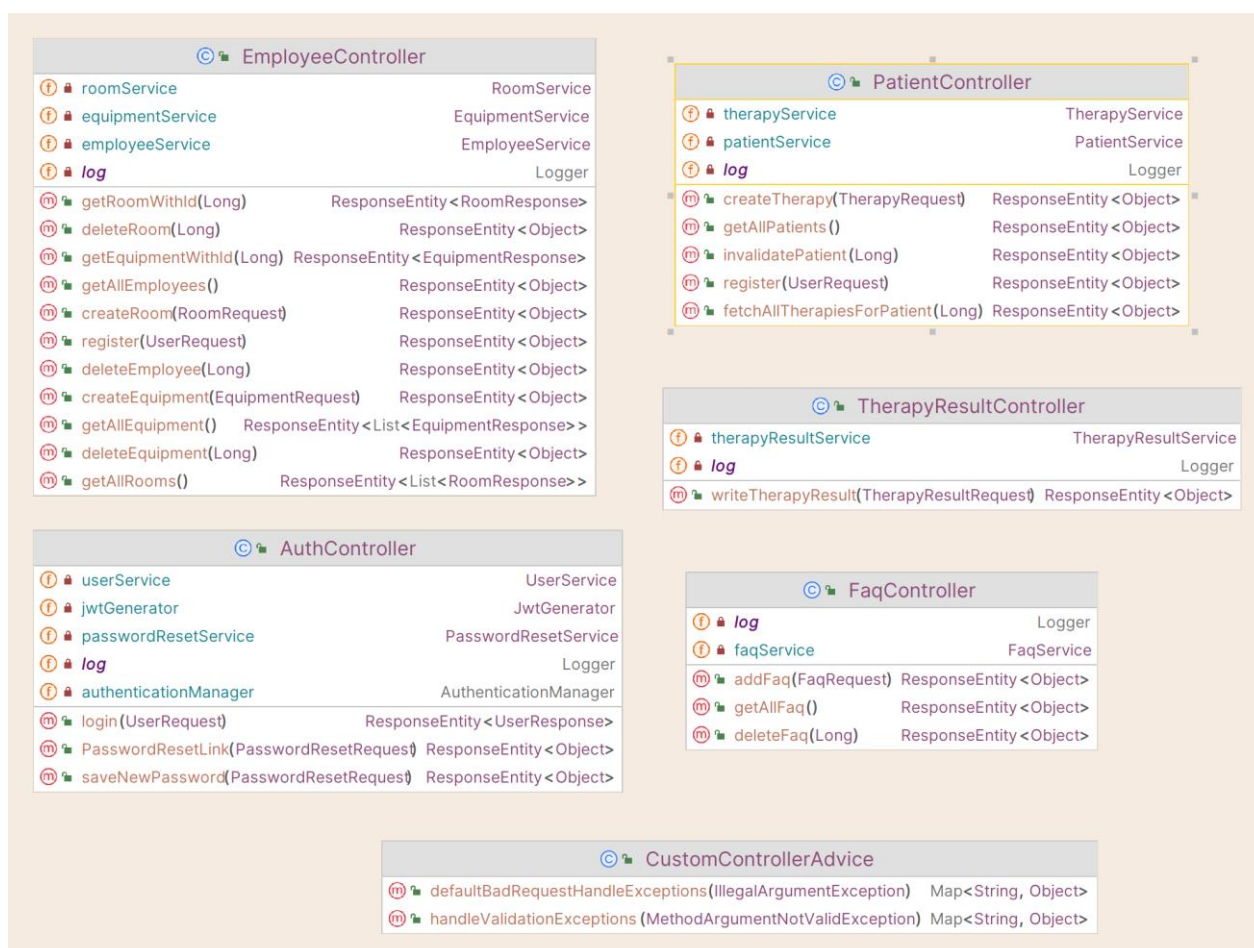
Slika 6 - Dijagram razreda (Model)



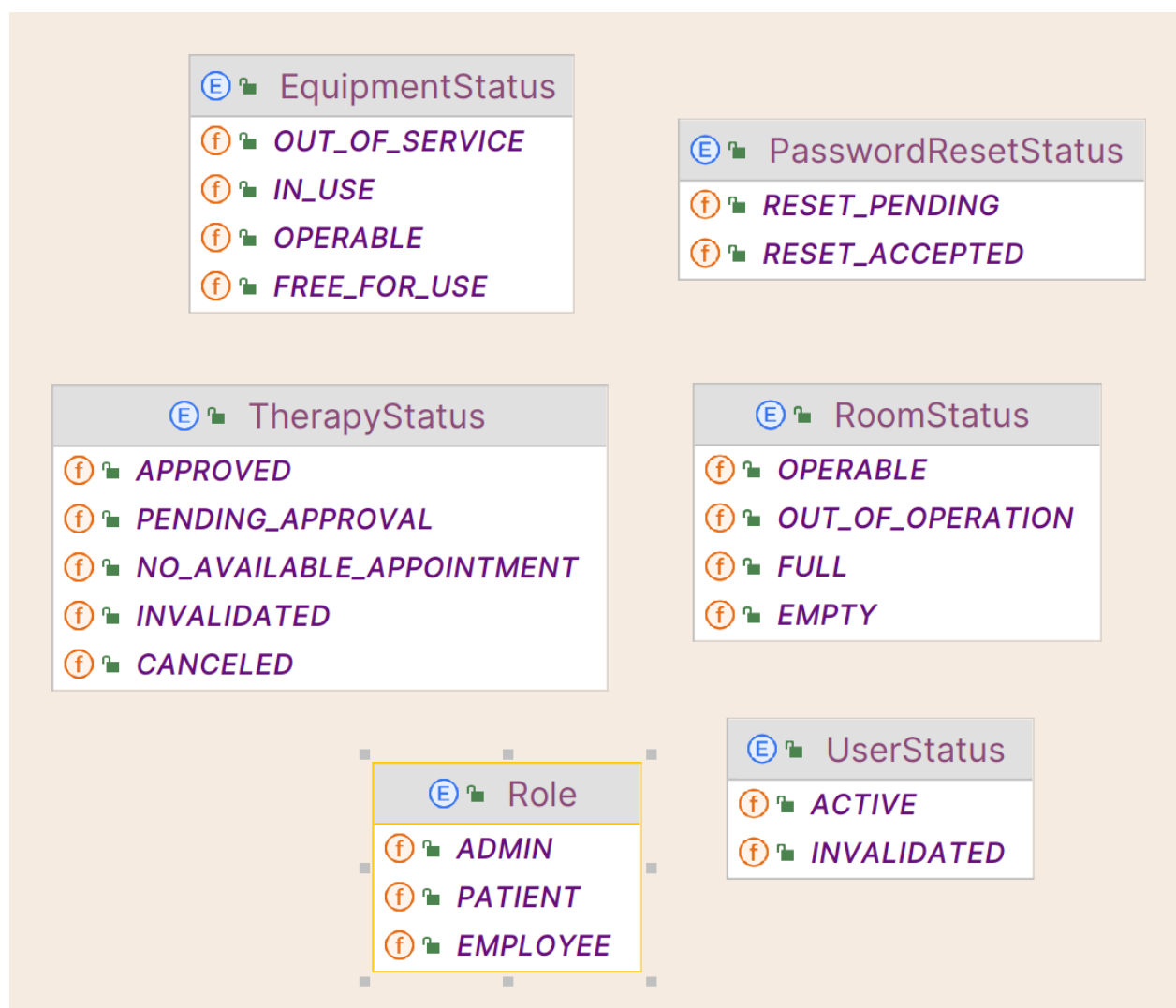
Slika 7 - Dijagram razreda (Repository)



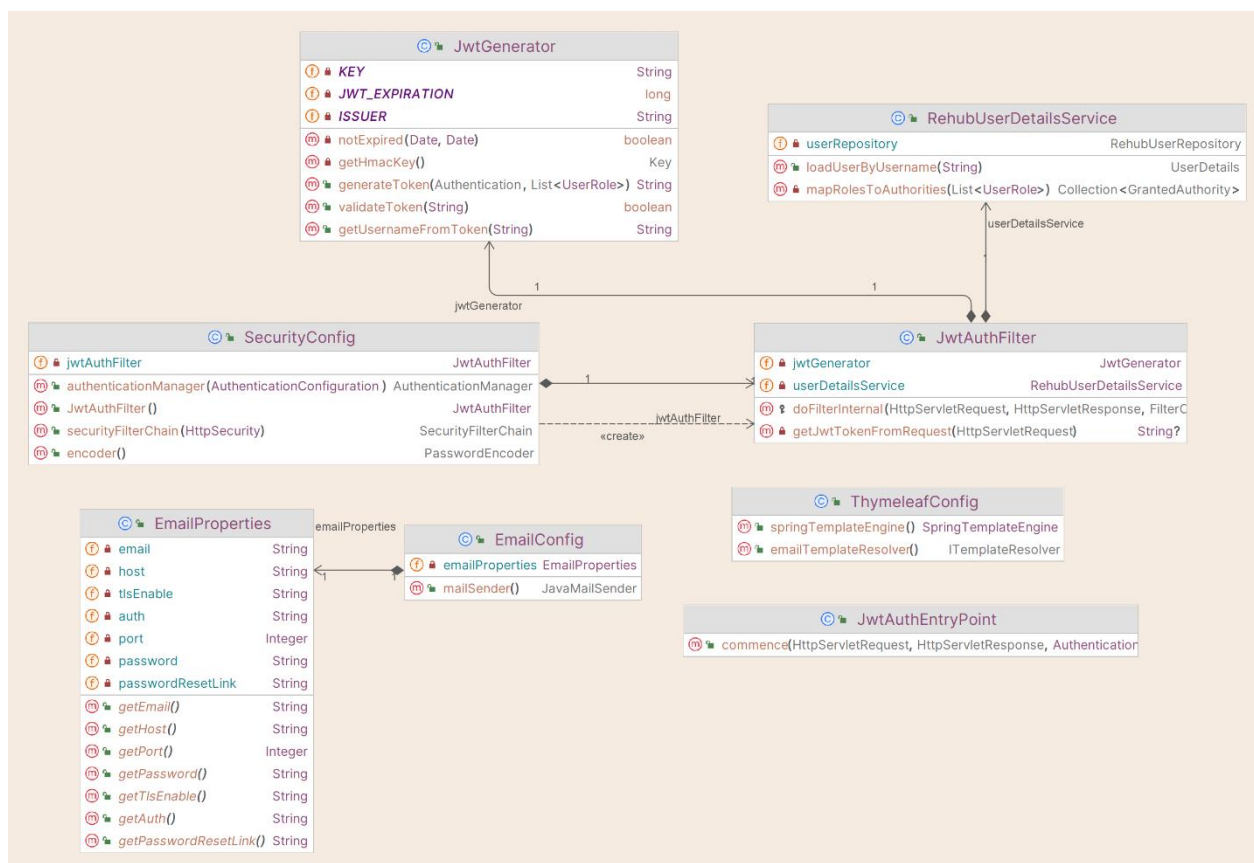
Slika 8 - Dijagram razreda (Service)



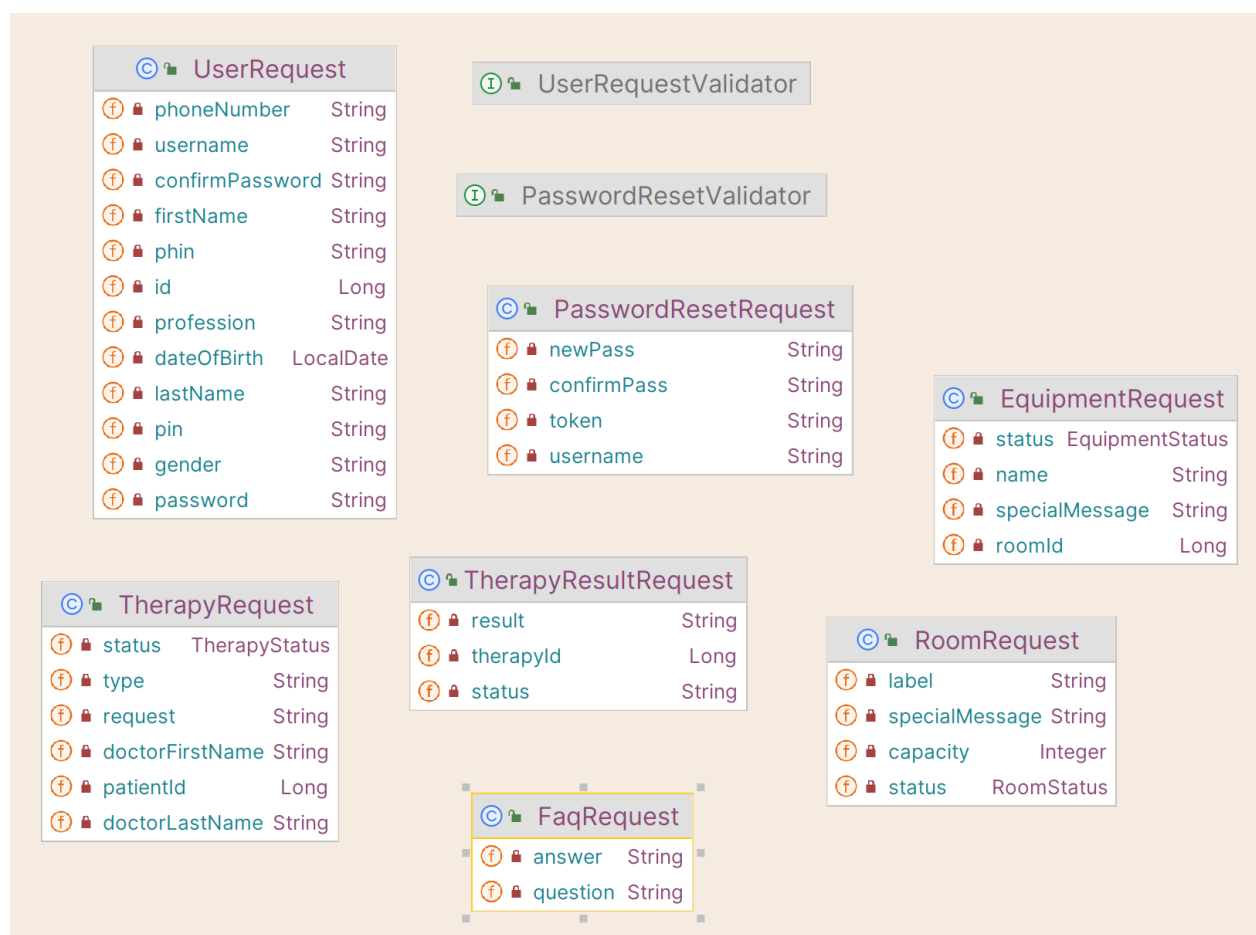
Slika 9 - Dijagram razreda (Controller)



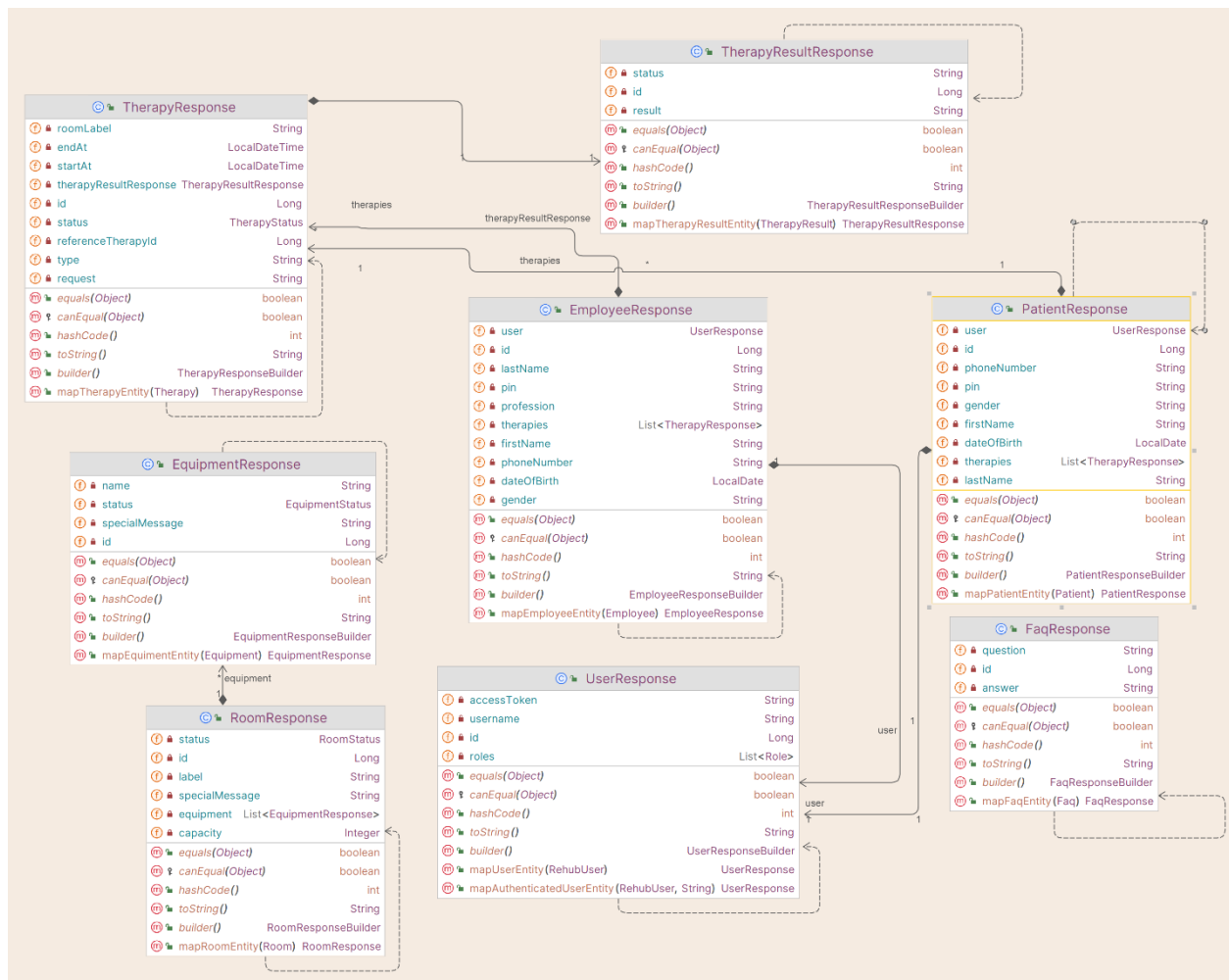
Slika 10 – Dijagram enumeracija (Model)



Slika 11 - Dijagram razreda (Config)



Slika 12 - Dijagram razreda (Request)

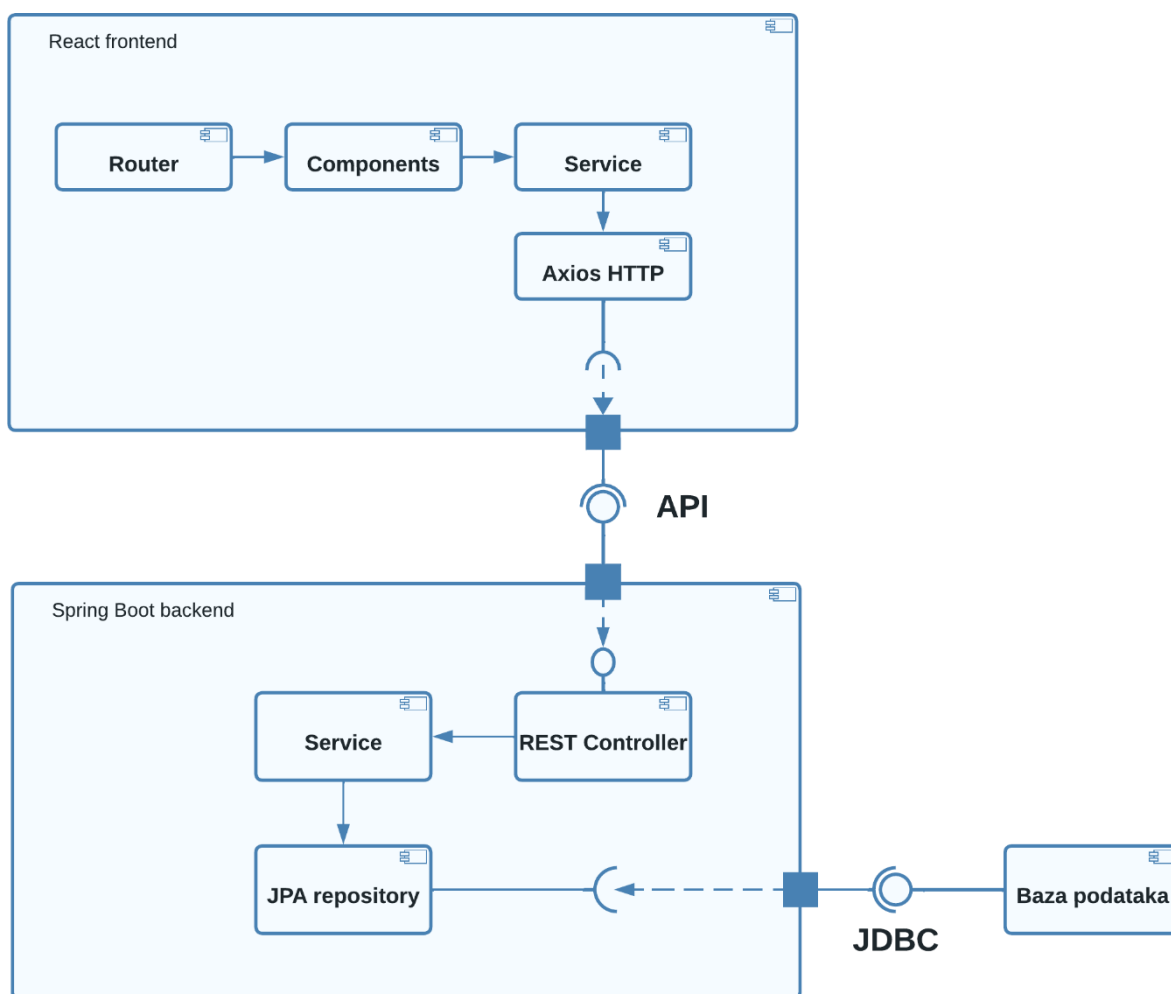


Slika 13 - Dijagram razreda (Response)

4.3 Dijagram stanja

4.4 Dijagram aktivnosti

4.5 Dijagram komponenti



Slika 14 - Dijagram komponenti

5. Implementacija i korisničko sučelje

5.1 Korištene tehnologije i alati

5.2 Ispitivanje programskog rješenja

5.2.1 Ispitivanje komponenti

5.2.2 Ispitivanje sustava

¹<https://www.seleniumhq.org/>

5.3 Dijagram razmještaja

5.4 Upute za puštanje u pogon

6. Zaključak i budući rad

Popis literature

1. Programsko inženjerstvo, FER ZEMRIS, <http://www.fer.hr/predmet/proinz>
2. The Unified Modeling Language, <https://www.uml-diagrams.org/>
3. Astah Community, <http://astah.net/editions/uml-new>
4. Baeldung, <https://www.baeldung.com>
5. Docker Docs, <https://docs.docker.com>
6. React, <https://react.dev/>

Indeks slika i dijagrama

Slika 1 - Dijagram obrazaca uporabe	18
Slika 2 - Sekvencijski dijagram za UC18/25.....	19
Slika 3 - Sekvencijski dijagram za UC8/11/15.....	21
Slika 4 - Dijagram arhitekture.....	24
Slika 5 - Dijagram baze podataka	31
Slika 6 - Dijagram razreda (Model).....	33
Slika 7 - Dijagram razreda (Repository)	33
Slika 8 - Dijagram razreda (Service).....	34
Slika 9 - Dijagram razreda (Controller)	34
Slika 10 - Dijagram enumeracija (Model).....	35
Slika 11 - Dijagram razreda (Config).....	36
Slika 12 - Dijagram razreda (Request)	37
Slika 13 - Dijagram razreda (Response)	38
Slika 14 - Dijagram komponenti	41

Dodatak: Prikaz aktivnosti grupe

Dnevnik sastajanja

1. sastanak

- Datum: 16. listopada 2023.
- Prisustvovali: Kašik, Begić, Vivoda, Greblo, Miletić
- Teme sastanka:
 - Upoznavanje s kolegama

2. sastanak

- Datum: 18. listopada 2023.
- Prisustvovali: Kašik, Begić, Krstičević, Mikulić, Vivoda, Greblo, Miletić
- Teme sastanka:
 - Upoznavanje s projektnim zadatkom
 - Podjela zadataka
 - Dogovor oko načina daljnje komunikacije

3. sastanak

- Datum: 19. listopada 2023.
- Prisustvovali: Kašik, Krstičević
- Teme sastanka:
 - Dogovor oko daljnje raspodjele posla

4. sastanak

- Datum: 19. listopada 2023.
- Prisustvovali: Begić, Greblo, Vivoda
- Teme sastanka:
 - Dogovor oko daljnje raspodjele posla

5. sastanak

- Datum: 23. listopada 2023.
- Prisustvovali: Kašik, Krstičević
- Teme sastanka:
 - Dogovor oko daljnje raspodjele posla

6. sastanak

- Datum: 24. listopada 2023.
- Prisustvovali: Begić, Mikulić, Vivoda, Greblo, Miletić

- Teme sastanka:
 - Dogovor oko daljnje raspodjele posla
7. sastanak
- Datum: 25. listopada 2023.
 - Prisustvovali: Kašik, Begić
 - Teme sastanka:
 - Laboratorijska vježba
8. sastanak
- Datum: 26. listopada 2023.
 - Prisustvovali: Kašik, Begić, Krstičević, Mikulić, Vivoda, Greblo, Miletić
 - Teme sastanka:
 - Upoznavanje s pokretanjem backenda
 - Predstavljanje dizajna aplikacije
9. sastanak
- Datum: 30. listopada 2023.
 - Prisustvovali: Kašik, Krstičević
 - Teme sastanka:
 - Dogovor oko daljnje raspodjele posla
10. sastanak
- Datum: 31. listopada 2023.
 - Prisustvovali: Begić, Mikulić, Vivoda, Greblo, Miletić
 - Teme sastanka:
 - Dogovor oko daljnje raspodjele posla
11. sastanak
- Datum: 6. studeni 2023.
 - Prisustvovali: Kašik, Krstičević
 - Teme sastanka:
 - Dogovor oko daljnje raspodjele posla
12. sastanak
- Datum: 7. studeni 2023.
 - Prisustvovali: Begić, Mikulić, Vivoda, Greblo, Miletić
 - Teme sastanka:
 - Dogovor oko daljnje raspodjele posla
13. sastanak
- Datum: 8. studeni 2023.
 - Prisustvovali: Kašik, Begić, Krstičević, Mikulić, Vivoda, Greblo, Miletić
 - Teme sastanka:
 - Dogovor oko daljnje raspodjele posla

- Dogovor vezan uz izradu dokumentacije
- Laboratorijska vježba

14. sastanak

- Datum: 13. studeni 2023.
- Prisustvovali: Kašik, Krstičević
- Teme sastanka:
 - Dogovor oko daljnje raspodjele posla

15. sastanak

- Datum: 14. studeni 2023.
- Prisustvovali: Begić, Mikulić, Vivoda, Greblo, Miletić
- Teme sastanka:
 - Dogovor oko daljnje raspodjele posla

16. sastanak

- Datum: 15. studeni 2023.
- Prisustvovali: Kašik, Begić, Krstičević, Mikulić, Vivoda, Greblo, Miletić
- Teme sastanka:
 - Dogovor oko daljnje raspodjele posla
 - Završni pregled dosad napravljenog posla

Tablica aktivnosti

	Dora Kašik	Josip Begić	Roko Krstičević	Katarina Mikulić	Anton Vivoda	Erik Greblo	Marko Miletić
Upravljanje projektom	6	5					2
Opis projektnog zadatka							7
Funkcionalni zahtjevi	7			9			
Opis pojedinih obrazaca	9			8			4
Dijagram obrazaca	6			9			3
Sekvencijski dijagrami			5		2	7	

Opis ostalih zahtjeva		4	5				
Arhitektura i dizajn sustava	4	4	6				4
Baza podataka		16		4	11	13	7
Dijagram razreda		2			9		3
Dijagram stanja		3				6	
Dijagram aktivnosti					4	4	
Dijagram komponenti			3		4		
Korištene tehnologije i alati			5				
Ispitivanje programskog rješenja			6				

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

	Dora Kašik	Josip Begić	Roko Krstičević	Katarina Mikulić	Anton Vivoda	Erik Greblo	Marko Miletić
Dijagram razmještaja							
Upute za puštanje u pogon							
Dnevnik sastajanja							
Zaključak i budući rad							
Popis literature							

Dijagrami pregleda promjena