

Programsko inženjerstvo

Ak. god. 2023./2024.

Medicinska rehabilitacija

Dokumentacija, Rev.1

Grupa: *Proginator*

Voditelj: *EmaBadurina*

Datum predaje: 17. studenog 2023.

Nastavnik: Miljenko Krhen

Sadržaj

1 Dnevnik promjena dokumentacije	3
2 Opis projektnog zadatka	4
3 Specifikacija programske potpore	7
3.1 Funkcionalni zahtjevi	7
3.1.1 Obrasci uporabe	9
3.1.2 Sekvencijski dijagrami	21
3.2 Ostali zahtjevi	22
4 Arhitektura i dizajn sustava	23
4.1 Baza podataka	24
4.1.1 Opis tablica	25
4.1.2 Dijagram baze podataka	30
4.2 Dijagram razreda	30
4.3 Dijagram stanja	32
4.4 Dijagram aktivnosti	33
4.5 Dijagram komponenti	34
5 Implementacija i korisničko sučelje	35
5.1 Korištene tehnologije i alati	35
5.2 Ispitivanje programskog rješenja	36
5.2.1 Ispitivanje komponenti	36
5.2.2 Ispitivanje sustava	36
5.3 Dijagram razmještaja	37
5.4 Upute za puštanje u pogon	38
6 Zaključak i budući rad	39
Popis literature	40
Indeks slika i dijagrama	41

Dodatak: Prikaz aktivnosti grupe

42

1. Dnevnik promjena dokumentacije

Kontinuirano osvježavanje

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Dovršen opis projektnog zadatka	E.Badurina	13.11.2023.
0.2	Dovršen opis arhitekture	L.Lasović	13.11.2023.
0.2.1	Dodan dio obrazaca uporabe	L.Lasović, E.Badurina	13.11.2023.
0.3	Dovršeni obrasci uporabe	L.Lasović, E.Badurina	14.11.2023.
0.3.1	Dodani dijagrami obrazaca uporabe	A.Jakovčević	14.11.2023.
1.0	Verzija samo s bitnim dijelovima za 1. ciklus	*	11.09.2013.
1.1	*	* *	*
1.2	*	*	*
2.0	*	*	*

Moraju postojati glavne revizije dokumenata 1.0 i 2.0 na kraju prvog i drugog ciklusa. Između tih revizija mogu postojati manje revizije već prema tome kako se dokument bude nadopunjavao. Očekuje se da nakon svake značajnije promjene (dodatka, izmjene, uklanjanja dijelova teksta i popratnih grafičkih sadržaja) dokumenta se to zabilježi kao revizija. Npr., revizije unutar prvog ciklusa će imati oznake 0.1, 0.2, ..., 0.9, 0.10, 0.11.. sve do konačne revizije prvog ciklusa 1.0. U drugom ciklusu se nastavlja s revizijama 1.1, 1.2, itd.

2. Opis projektnog zadatka

Cilj ovog projekta je izgradnja web aplikacije koja će omogućiti ljudima/pacijentima lakše prijavljivanje na slobodne termine za medicinsku rehabilitaciju i fizikalnu terapiju te praćenje njihovog zdravstvenog napretka. Također će omogućiti zaposlenicima ustanove da odbijaju ili prihvataju termine ovisno o raspoloživosti prostorija, opreme i osoblja pritom imajući mogućnost uvida u pacijentove prošle terapije i napredak. Vrijeme provođenja rehabilitacije je svakim radnim danom od ponedjeljka do petka od 8:00 do 20:00 sati.

Aplikacija će razlikovati tri vrste korisnika:

- pacijenta
- liječnika - djelatnika zdravstvene ustanove
- administratora - djelatnika zdravstvene ustanove

Prilikom pokretanja web aplikacije svaki korisnik unosi svoju e-mail adresu i lozinku. Ovisno o vrsti korisnika biti će preusmjereni na različite stranice.

Pacijent se samostalno prijavljuje u sustav. U slučaju da još ne postoji imati će opciju registracije. Za registraciju mora unijeti:

- ime
- prezime
- e-mail adresu
- MBO - Matični Broj Osiguranika
- broj telefona
- lozinku

Prilikom registracije pomoću MBO-a provjerava se ako korisnik postoji u središnjem informacijskom sustavu zdravstvene zaštite. Nakon prijave ili registracije korisnik je preusmjeren na stranicu gdje se prikazuju njegovi termini i zahtjevi. U terminima se prikazuju protekli i budući termini, za protekle termine piše ako je pacijent došao te komentari djelatnika ustanove o napretku. Termin se dobiva nakon što se odobri zahtjev za jednim. Zahtjev se sastoji od:

- vremena predaje

- željenog termina
- vrste/tipa rehabilitacije
- liječnika koji je uputio pacijenta na terapiju
- reference na terapiju
- statusa

Status može biti odobreno, odbijeno, čeka na odobrenje ili isteklo u slučaju da nitko nije odobrio niti odbio termin do željenog vremena terapije. Klikom na opciju *naruči se* pacijent može napraviti novi zahtjev za terminom. Zahtjev popunjava unoseći vrstu rehabilitacije, liječnika koji ga je uputio, ukoliko se radi o ponovljenoj terapiji odabrati će i termin već obavljenog postupka terapije u ustanovi, također u komentar za liječnika dodaje opis svojih oboljenja. Status liječnika koji je izdao uputnicu provjerava se u imeniku liječnika. U odabiru termina biti će prikazano prvih nekoliko slobodnih termina za odabrani tip rehabilitacije i opcije da se unese željeni datum i vrijeme. Nakon odabira željenog termina pacijent predaje zahtjev. Pacijent će o odobrenju ili odbijanju zahtjeva, kao i svim mogućim promjenama biti obaviješten mailom.

Liječnik nakon prijavljivanja u sustav ima pregled svih pacijenata i njihovih podataka, klikom na prikaz terapije bit će mu prikazani svi termini odabranog pacijenta i detalji o njima, tu će imati opciju evidentirati dolazak pacijenta i zabilježiti komentare vezane uz napredak ili pregledati napredak i komentare iz prošlih termina. Također svaki djelatnik ustanove imati će mogućnost prihvatanja ili odbijanja termina ovisno o raspoloživosti prostorija i opreme. Liječnik koji prihvati zahtjev automatski će se dodati kao liječnik koji će tu terapiju izvoditi.

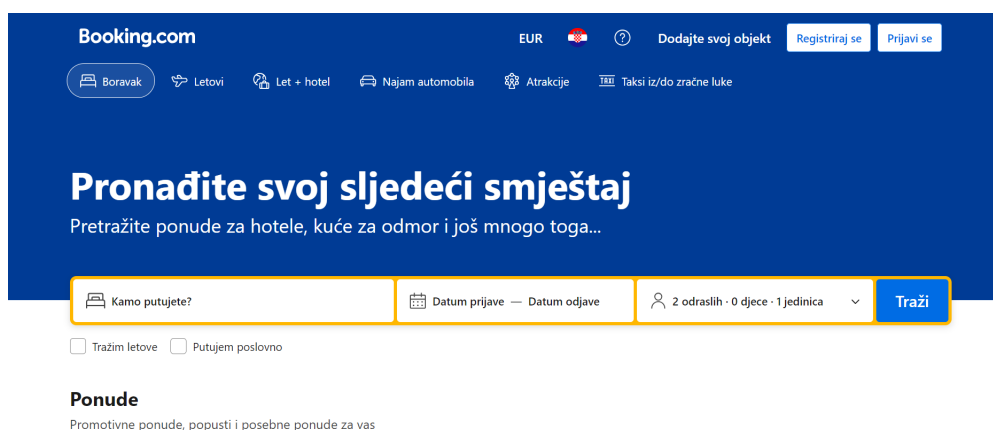
Administrator ima pregled svih pacijenata i djelatnika. Uz ovlasti koje imaju liječnici administrator pri zaposlenju novog liječnika izrađuje korisnički račun za njega s inicijalnom lozinkom. Također nakon prestanka radnog odnosa administrator može ukloniti tog liječnika. Administrator definira sve što je potrebno za ispravan rad sustava, dakle može mijenjati dostupnost opreme i unositi novu opremu, mijenjati dostupnost prostorija i unositi nove. Također ako dođe do promjena u dostupnosti prostorija ili opreme, liječnik će imati mogućnost obavješćavanja pacijenta o otkazanom terminu te mogućnost izdavanja novog termina.

—

Ovim projektom smanjio bi se opseg posla djelatnika ustanove i olakšao proces prijave na rehabilitaciju pacijentima, umjesto prijavljivanja u živo i rješavanja papirologije pacijentima će biti omogućeno prijavljivanje, a djelatnicima odbijanje ili

prihvaćanje termina rehabilitacije u samo par klikova.

Odgovarajući primjer slične aplikacije na istu temu nije pronađen, moguće, zbog zatvorenosti ovakvog tipa aplikacije prema javnosti, ali primjer sa sličnim funkcijama iako ne u istu svrhu je booking.com (2.1). Na početnoj stranici *booking*-a nailazimo na opcije prijave i registracije, iako za početak korištenja same stranice to nije nužno kao kod ove aplikacije. Korisnik *booking*-a rezervira apartman, dok korisnik aplikacije za rehabilitaciju rezervira svoj termin za terapiju. Kao što korisnik na *booking*-u unosi grad u kojem želi rezervirati apartman, korisnik u ovoj aplikaciji odabire tip rehabilitacije za koju se želi prijaviti. U oba slučaja kod prijave na željeni termin ili rezervacije apartmana korisnik dobiva neke ponudene termine/opcije ili može samostalno odabrati vrijeme željenog termina/rezervacije.



Slika 2.1: Primjer slične aplikacije

Aplikacija će se moći proširiti da djelatnici uopće ne moraju prihvaćati ili odbijati prijave, već će to sustav raditi automatski na temelju raspoloživih podataka o opremi, prostorijama, djelatnicima i već zauzetim terminima. Svi termini, prostori, oprema i djelatnici vidljivi su administratoru koji im može mijenjati status za određeno vrijeme, na primjer kada liječnik ode na godišnji odmor može promijeniti njegov status u neaktivan za to razdoblje ili kada cijela ustanova ima sastanak onemogućiti sve termine za vrijeme tog sastanka. Ostala bi mogućnost upućivanja maila ili telefonskog poziva pacijentu u slučaju ikakvih promjena.

3. Specifikacija programske potpore

3.1 Funkcionalni zahtjevi

Dionici:

1. Pacijent
2. Djelatnik
 - Liječnik
 - Administrator
3. Razvojni tim

Pacijenti i djelatnici zajedničkim imenom su korisnici.

Aktori i njihovi funkcionalni zahtjevi:

1. Neregistrirani korisnik (inicijator) može:
 - (a) registrirati se u sustav, unijeti potrebne podatke(ime, prezime, e-mail adresu, MBO, broj telefona, lozinku)
2. Neprijavljeni korisnik (inicijator) može:
 - (a) prijaviti se u sustav koristeći svoju e-mail adresu i lozinku
3. Pacijent-prijavljeni korisnik (inicijator) može:
 - (a) prikazati i mijenjati svoje osobne podatke i lozinku
 - (b) vidjeti svoje zahtjeve za terminom terapije (vrijeme predaje zahtjeva, vrijeme željenog termina, tip rehabilitacije, liječnika koji ga je uputio na rehabilitaciju, referencu na prošlu terapiju i status)
 - (c) vidjeti svoje termine (vrijeme termina, prostoriju, tip rehabilitacije, liječnika, napomene i ishod termina)
 - (d) filtrirati svoje termine
 - (e) prikazati nalaz (detalji i komentari) s termina odrađene terapije
 - (f) naručiti se na terapiju, unijeti tip rehabilitacije, liječnika koji ga je uputio na terapiju, opis oboljenja i zahtijevani postupak liječenja

- (g) odabrati termin za neku terapiju između priloženih ili unijeti datum i odabrati neki na temelju toga ponuđeni termin

4. Djelatnik-prijavljeni korisnik (inicijator) može:

- (a) prikazati i mijenjati svoje osobne podatke i lozinku
- (b) vidjeti popis svih pacijenata (ime, prezime, MBO, e-mail adresa, broj telefona)
- (c) prikazati termine pojedinog pacijenta i prikazati evidenciju termina pojedinog pacijenta
- (d) evidentirati dolazak pacijenta na termin, dodati komentar o odrađenom terminu i vidjeti informacije o terapiji
- (e) prihvatiti/odbiti zahtjev za terminom na temelju raspoloživih prostorija i opreme

5. Administrator - prijavljeni korisnik (inicijator) može:

- (a) vidjeti popis svih djelatnika (ime, prezime, OIB, e-mail adresa)
- (b) urediti podatke djelatnika
- (c) ukloniti djelatnika
- (d) dodati/registirati novog djelatnika (ime, prezime, e-mail adresa, OIB, lozinka)
- (e) mijenjati raspoloživost soba i opreme
- (f) dodavati opremu i sobe
- (g) otkazivati i mijenjati termine, te o tome obavještavati pacijenta

6. Baza podataka (sudionik) može:

- (a) pohranjuje podatke i ovlasti korisnika
- (b) pohranjuje podatke o opremi i prostorijama
- (c) pohranjuje podatke o terapijama i terminima

3.1.1 Obrasci uporabe

Opis obrazaca uporabe

UC1 - Prijava u sustav

- **Glavni sudionik:** Korisnik
- **Cilj:** Prijaviti se u sustav
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je registriran u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik unosi svoju e-mail adresu i lozinku
 2. Provjera ispravnosti podataka
 3. Prikaz početne stranice ovisno o korisniku
- **Opis mogućih odstupanja:**
 - 2.a Neispravna e-mail adresa ili lozinka
 1. obavijest o neispravnosti unesenih podataka

UC2 - Registracija

- **Glavni sudionik:** Neregistrirani korisnik
- **Cilj:** Registrirati se u sustav
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Korisnik unosi svoje podatke (ime, prezime, e-mail adresu, MBO, lozinka)
 2. Provjera ispravnosti podataka verifikacijom iz baze podataka sustava zdravstvene zaštite
 3. prikaz početne stranice
- **Opis mogućih odstupanja:**
 - 2.a Korisnik ne postoji u sustavu zdravstvene zaštite
 1. Sustav obavještava korisnika o neispravnosti podataka
 2. Korisnik mijenja podatke ili odustaje od registracije

UC3 - pregled osobnih podataka

- **Glavni sudionik:** Korisnik
- **Cilj:** Pregledati osobne podatke

- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik mora biti prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju "Profil"
 2. Sustav prikazuje korisnikove osobne podatke

UC4 - Promjena osobnih podataka

- **Glavni sudionik:** Korisnik
- **Cilj:** Promijeniti osobne podatke
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik mora biti prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju "Uredi"
 2. Korisnik unosi/mijenja podatke
 3. Korisnik odabire opciju "Spremi promjene"

UC5 - Promjena lozinke

- **Glavni sudionik:** Korisnik
- **Cilj:** Promijeniti lozinku
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju "Promijeni lozinku"
 2. Korisnik potvrđuje staru lozinku
 3. Korisnik unosi novu lozinku
 4. Korisnik potvrđuje novu lozinku
 5. Sustav korisnika prebacuje na prijavu
 6. Korisnik se prijavljuje s novom lozinkom
- **Opis mogućih odstupanja:**
 - 2.a korisnik unosi neispravnu staru lozinku
 1. Sustav upozorava korisnika o neispravnosti lozinke
 2. Korisnik ponovno unosi lozinku ili odustaje od promjene
 - 3.a Nova lozinka je jednaka staroj
 1. Sustav upozorava korisnika da su mu lozinke iste
 2. Korisnik mijenja novu lozinku ili odustaje od promjene

UC6 - Kreiranje terapije

- **Glavni sudionik:** Pacijent
- **Cilj:** Kreiranje procesa terapije
- **Sudionici:** Baza podataka
- **Preduvjet:** Pacijent je prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Pacijent odabire opciju "Kreiraj terapiju"
 2. Otvara se stranica za prijavljivanje terapije
 3. Pacijent unosi ime i prezime liječnika, opis oboljenja i zahtijevani postupak liječenja, te odabire vrstu terapije
 4. Pacijent odabire opciju "Pošalji"
 5. Provjera ispravnosti podataka u imeniku liječnika
 6. Prikaz stranice za odabir termina
- **Opis mogućih odstupanja:**
 - 3.a Liječnik ne postoji u imeniku liječnika
 1. Sustav obavještava pacijenta o neispravnim podacima (ime i prezime liječnika)
 2. Pacijent mijenja podatke ili odustaje od kreiranja terapije

UC7 - Odabir termina

- **Glavni sudionik:** Pacijent
- **Cilj:** Poslati zahtjev za željenim terminom
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Pacijent odabire terapiju za koju se prijavljuje na termin
 2. Pacijent unosi datum i na temelju njega bira ponuđene termine
 3. Pacijent opcionalno dodaje napomenu za liječnika
 4. Pacijent šalje zahtjev klikom na "Pošalji"
- **Opis mogućih odstupanja:**
 - 2.a Pacijentu ne odgovara niti jedan termin
 1. Pacijent odustaje od prijave na termin

UC8 - Prikaz nalaza termina

- **Glavni sudionik:** Pacijent
- **Cilj:** Pacijent vidi svoje nalaze
- **Sudionici:** Baza podataka

- **Preduvjet:** Prijavljen u sustav, postoje prošli termini
- **Opis osnovnog tijeka:**
 1. Pacijent odabire opciju prikaži nalaz
 2. Otvara se skočni prozor s informacijama i komentarom liječnika (nalazom)
 3. Pacijent odabire opciju zatvori u skočnom prozoru

UC9 - Prikaz pacijenta

- **Glavni sudionik:** Djelatnik
- **Cilj:** Prikazati termine pojedinog pacijenta
- **Sudionici:** Baza podataka
- **Preduvjet:** Djelatnik je prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Djelatnik pretražuje pacijenta upisom imena i prezimena u tražilicu
 2. Djelatnik odabire opciju "Prikaži termine" pored željenog pacijenta
 3. Sustav otvara tablicu termina odabranog pacijenta
 4. Djelatnik evidentira(UC10)/prikazuje evidenciju(UC8)
 5. Djelatnik odabire opciju povratka na popis pacijenata
- **Opis mogućih odstupanja:**
 - 1.a Upisani pacijent ne postoji u sustavu
 1. Prikaz teksta o ne postojanju pacijenta
 2. Djelatnik mijenja upisano ime i prezime ili odustaje od pretraživanja
 - 3.a Odabrani pacijent još nije bio ni na jednoj terapiji
 1. Prikaz teksta o ne postojanju termina za odabranog pacijenta
 2. Djelatnik odabire opciju povratka na popis pacijenata

UC10 - Evidencija dolaska

- **Glavni sudionik:** Djelatnik
- **Cilj:** Evidentirati dolazak pacijenta na termin
- **Sudionici:** Baza podataka
- **Preduvjet:**
 - Djelatnik je prijavljen u sustav
 - Pacijent mora imati termin koji nije evidentiran
- **Opis osnovnog tijeka:**
 1. Djelatnik odabire opciju evidentiraj pored termina odabranog pacijenta
 2. Otvara se obrazac za evidenciju dolaska na termin

3. Djelatnik evidentira dolazak odabirom jedne od tri ponuđene opcije(nije došao, nije uspješno odradio terapiju ili uspješno je odradio terapiju)
 4. Djelatnik unosi komentar o odrađenoj terapiji
 5. Djelatnik predaje evidenciju
 6. povratak na popis termina odabranog pacijenta
- **Opis mogućih odstupanja:**
 - 3.a Djelatnik odustaje od evidentiranja termina
 1. Djelatnik odabire opciju povratka na popis termina odabranog pacijenta

UC11 - Prihvaćanje zahtjeva za terminom

- **Glavni sudionik:** Djelatnik
- **Cilj:** Prihvatiti/odbiti zahtjev za terminom na temelju
- **Sudionici:** Baza podataka
- **Preduvjet:**
 - Djelatnik je prijavljen u sustav
 - Moraju postojati ne evidentirani zahtjevi
- **Opis osnovnog tijeka:**
 1. Djelatnik odabire opciju "Zahtjevi"
 2. Djelatnik odabire opciju evidentiraj pored zahtjeva
 3. Djelatniku se prikazuju moguće vrste opreme
 4. Djelatnik odabire vrstu opreme
 5. Prikaz slobodnih prostorija s odabranom vrstom opreme
 6. Djelatnik odabire prostoriju i opcionalno piše komentar za pacijenta
 7. Djelatnik potvrđuje termin
 8. Sustav šalje e-mail pacijentu čiji je termin potvrđen sa svim potrebnim informacijama
 9. Prikazuje se popis zahtjeva za terminom
- **Opis mogućih odstupanja:**
 - 5.a Nema slobodnih soba za tu vrstu opreme
 - (a) Djelatnik odbija zahtjev za terminom
 - (b) Iz baze podataka dohvaćaju se slijedeći slobodni termini od željenog datuma za tog pacijenta
 - (c) Otvara se prozor za dodjeljivanje novog termina s podacima o odabranom pacijentu i vrsti opreme
 - (d) Djelatnik odabire prvi slijedeći slobodni termin i sobu

- (e) Djelatnik dodaje komentar
- (f) Djelatnik potvrđuje novi termin
- (g) Sustav šalje pacijentu e-mail o dodjeljenom terminu
- (h) Prikazuje se popis zahtjeva za terminom

UC12 - Registracija djelatnika

- **Glavni sudionik:** Administrator
- **Cilj:** Registrirati novog liječnika
- **Sudionici:** Baza podataka
- **Preduvjet:**
 - Djelatnik je prijavljen u sustav
 - Djelatnik ima ovlasti administratora
- **Opis osnovnog tijeka:**
 1. Djelatnik odabire opciju "Dodaj djelatnika"
 2. Prikazuje se stranica za registraciju novog djelatnika
 3. Administrator unosi podatke o djelatniku(ime, prezime, OIB, e-mail adresu i inicijalnu lozinku)
 4. Administrator odabire opciju "Dodaj račun"
 5. Podatci se unose u bazu podataka
- **Opis mogućih odstupanja:**
 - 3.a Podatci ne zadovoljavaju željeni format
 1. Sustav upozorava administratora o neispravnom formatu podataka
 2. Administrator mijenja podatke ili odustaje od registracije djelatnika

UC13 - Uređivanje podataka djelatnika

- **Glavni sudionik:** Administrator
- **Cilj:** Promijeniti podatke djelatnika
- **Sudionici:** Baza podataka
- **Preduvjet:**
 - Djelatnik je prijavljen u sustav
 - Djelatnik ima ovlasti administratora
 - Postoji barem jedan liječnik/djelatnik
- **Opis osnovnog tijeka:**
 1. Administrator odabire opciju "Uredi" pokraj imena djelatnika
 2. Prikazuje se stranica s podacima odabranog djelatnika

3. Administrator mijenja podatke o djelatniku(ime, prezime, OIB, e-mail adresu)
 4. Administrator odabire opciju "Spremi"
 5. Podatci se mijenjaju u bazi podataka
 6. Prikazuje se popis djelatnika
- **Opis mogućih odstupanja:**
 - 3.a Podatci ne zadovoljavaju željeni format
 1. Sustav upozorava administratora o neispravnom formatu podataka
 2. Administrator mijenja podatke ili odustaje od promjene podataka
 - 4.a Administrator ne želi promijeniti podatke
 1. Administrator odabire opciju "Odustani"
 2. Podatci se ne mijenjaju u bazi podataka
 3. Prikazuje se popis djelatnika

UC14 - Promjena ovlasti djelatnika

- **Glavni sudionik:** Administrator
- **Cilj:** Promijeniti ovlasti djelatnika
- **Sudionici:** Baza podataka
- **Preduvjet:**
 - Djelatnik je prijavljen u sustav
 - Djelatnik ima ovlasti administratora
 - Postoji barem jedan liječnik/djelatnik
- **Opis osnovnog tijeka:**
 1. Administrator odabire opciju "Uredi" pokraj imena djelatnika
 2. Prikazuje se stranica s podacima odabranog djelatnika
 3. Administrator odabire opciju "Promijeni ovlasti"
 4. Otvara se skočni prozor
 5. Administrator pridjeljuje/uklanja administratorske ovlasti djelatnika
 6. Administrator odabire opciju "Spremi"
 7. Otvara se skočni prozor s upozorenjem i zahtjevom za unos lozinke administratora
 8. Administrator unosi lozinku
 9. Administrator odabire opcije "potvrdi"
 10. Sustav potvrđuje lozinku i sprema promjene u bazu podataka
 11. Vraća se na podatke o djelatniku
- **Opis mogućih odstupanja:**

10.a Administrator je unio neispravnu lozinku

1. Sustav upozorava administratora o unosu neispravne lozinke
2. Administrator ponovno unosi lozinku ili odustaje od promjene ovlasti

UC15 - Uklanjanje djelatnika

- **Glavni sudionik:** Administrator
- **Cilj:** Ukloniti djelatnika
- **Sudionici:** Baza podataka
- **Preduvjet:**
 - Djelatnik je prijavljen u sustav
 - Djelatnik ima ovlasti administratora
 - Postoji barem jedan liječnik/djelatnik
- **Opis osnovnog tijeka:**
 1. Administrator odabire opciju "Ukloni" pokraj imena djelatnika
 2. Otvara se skočni prozor s upozorenjem i zahtjevom za unos lozinke administratora
 3. Administrator unosi lozinku
 4. Administrator odabire opciju "Potvrdi"
 5. Sustav potvrđuje lozinku
 6. U bazi podataka se mijenja vrijednost atributa jeAktivan odabranog djelatnika
 7. prikazuje se popis djelatnika bez upravo uklonjenog djelatnika
- **Opis mogućih odstupanja:**
 - 5.a Administrator je unio neispravnu lozinku
 1. Sustav upozorava administratora o unosu neispravne lozinke
 2. Administrator ponovno unosi lozinku ili odustaje od uklanjanja djelatnika

UC16 - Unos opreme

- **Glavni sudionik:** Administrator
- **Cilj:** Unos opreme
- **Sudionici:** Baza podataka
- **Preduvjet:**
 - Djelatnik je prijavljen u sustav
 - Djelatnik ima ovlasti administratora

- **Opis osnovnog tijeka:**

1. Administrator odabire opciju "Inventar"
2. Otvara se stranica s popisom opreme
3. Administrator odabire opciju "Dodaj opremu"
4. Otvara se skočni prozor s prostorom za unos podataka o opremi
5. Unosi podatke o opremi (ime i opis)
6. Odabire prostoriju u kojoj će se oprema nalaziti
7. Administrator odabire opciju "Spremi"
8. Podatci se spremaju u bazu podataka i generira se jedinstveni identifikator opreme
9. Prikazuje se stranica s popisom opreme

- **Opis mogućih odstupanja:**

- 6.a Administrator želi odustati od unosa opreme
 1. Administrator odabire opciju "odustani"

UC17 - Promjena dostupnosti opreme

- **Glavni sudionik:** Administrator

- **Cilj:** Promjeniti dostupnost opreme

- **Sudionici:** Baza podataka

- **Preduvjet:**

- Djelatnik je prijavljen u sustav
- Djelatnik ima ovlasti administratora

- **Opis osnovnog tijeka:**

1. Administrator odabire opciju "Inventar"
2. Otvara se stranica s popisom opreme
3. Administrator pretražuje opremu
4. Administrator mijenja dostupnost opreme odabirom opcije "Dostupno"/"Nedostupno"
5. Promjena dostupnosti odabrane opreme u bazi podataka

UC18 - Unos prostorija

- **Glavni sudionik:** Administrator

- **Cilj:** Unijeti nove prostorije

- **Sudionici:** Baza podataka

- **Preduvjet:**

- Djelatnik je prijavljen u sustav
- Djelatnik ima ovlasti administratora

- **Opis osnovnog tijeka:**

1. Administrator odabire opciju "Prostorije"
2. Otvara se stranica s popisom prostorija
3. Administrator odabire opciju "Dodaj prostoriju"
4. Prikazuje se skočni prozor s prostorom za unos podataka o prostoriji
5. Administrator unosi broj i kapacitet prostorije
6. Administrator odabire opciju "spremi"
7. Podatci se spremaju u bazu podataka
8. Prikazuje se stranica s popisom opreme

- **Opis mogućih odstupanja:**

- 6.a Administrator želi odustati od unosa prostorije
 1. Administrator odabire opciju "Odustani"

UC19 - Promjena dostupnosti prostorije

- **Glavni sudionik:** Administrator

- **Cilj:** Promjeniti dostupnost prostorije

- **Sudionici:** Baza podataka

- **Preduvjet:**

- Djelatnik je prijavljen u sustav
- Djelatnik ima ovlasti administratora

- **Opis osnovnog tijeka:**

1. Administrator odabire opciju "Prostorije"
2. Otvara se stranica s popisom prostorija
3. Administrator pretražuje prostorije
4. Administrator mijenja dostupnost željene prostorije odabirom opcije "Dostupno"/"Nedostupno" koja se nalazi pored podataka o prostoriji
5. Promjena dostupnosti odabrane prostorije u bazi podataka

UC20 - Obavijest o promjeni termina

- **Glavni sudionik:** Administrator

- **Cilj:** Unijeti podatke o novom terminu i obavijestiti pacijenta o tome putem e-maila

- **Sudionici:** Baza podataka

- **Preduvjet:**

- Djelatnik je prijavljen u sustav
- Djelatnik ima ovlasti administratora

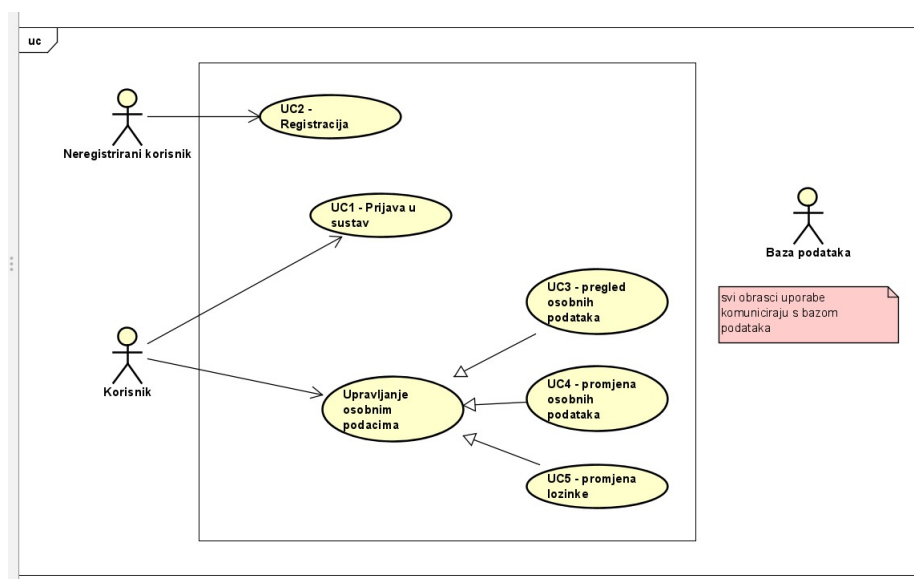
- **Opis osnovnog tijeka:**

1. Administrator odabire opciju "Promjena termina"
2. Otvara se stranica s prostorom za podatke o pacijentu i terminu
3. Administrator pretražuje i odabire pacijenta
4. Administrator unosi podatke o novom terminu
5. Administrator dodaje komentar
6. Administrator odabire opciju "Pošalji"
7. Podaci o terminu se spremaju u bazu podataka
8. Odabranom pacijentu se šalje e-mail o promjenama i informacijama o novom terminu

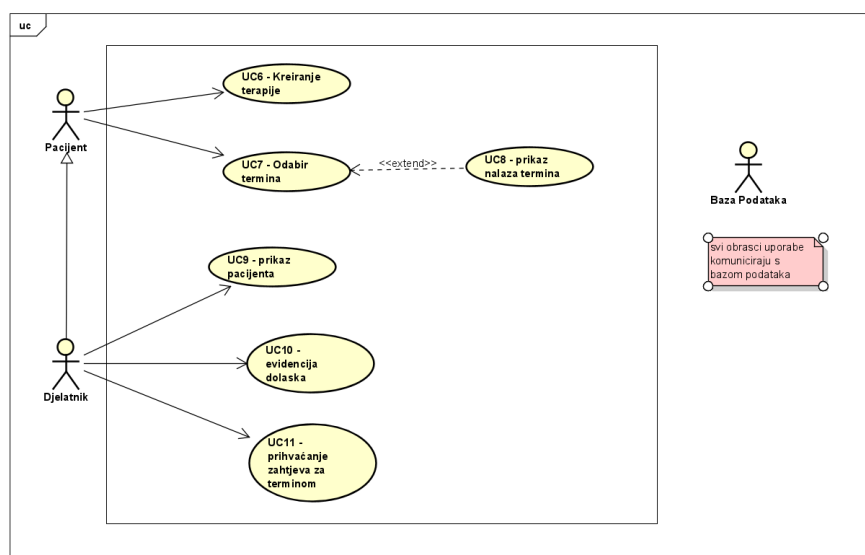
- **Opis mogućih odstupanja:**

- 3.a Administrator unosi nepostojećeg pacijenta/krivo uneseno ime/prezime/MBO pacijenta
 1. Sustav upozorava administratora o nepostojanju pacijenta u sustavu
 2. Administrator mijenja svoj unos ili odustaje od promjene termina

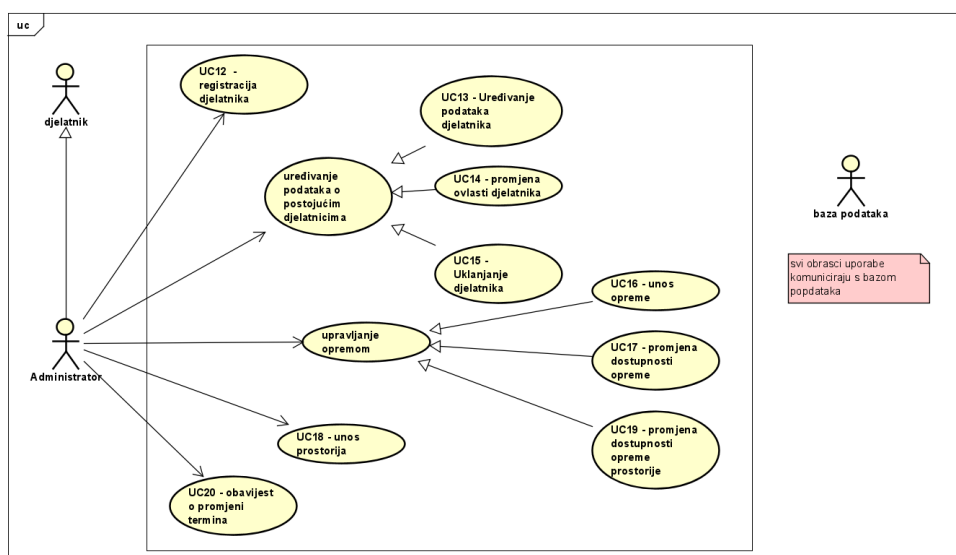
Dijagrami obrazaca uporabe



Slika 3.1: Dijagram obrasca uporabe - Korisnik i neregistrirani korisnik



Slika 3.2: Dijagram obrasca uporabe - Djelatnik i pacijent



Slika 3.3: Dijagram obrasca uporabe - Administrator

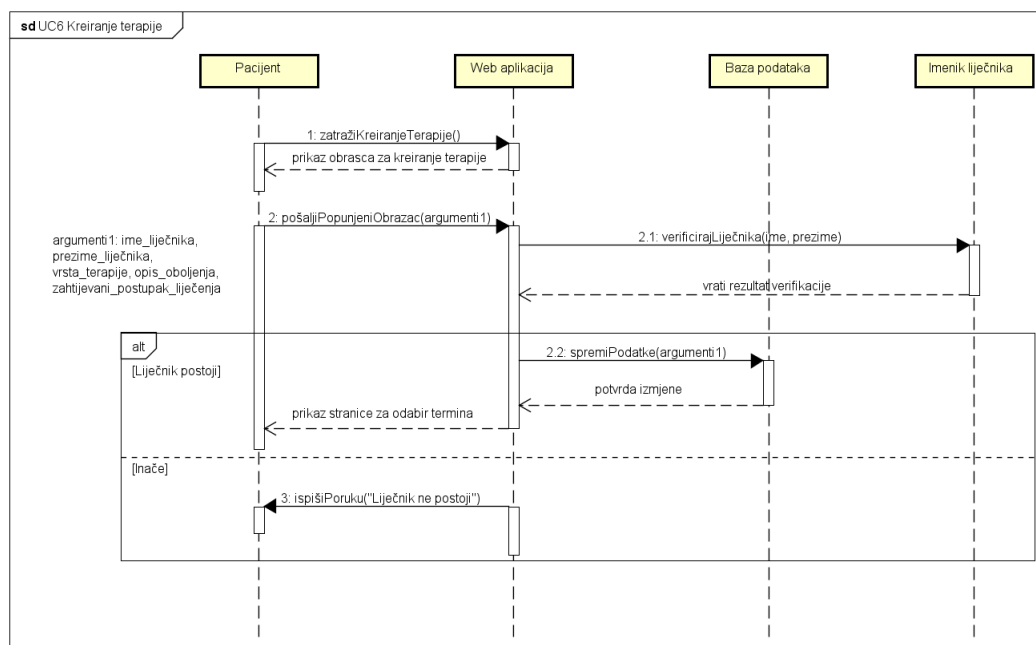
3.1.2 Sekvencijski dijagrami

UC6 - Kreiranje terapije

Pacijent šalje zahtjev poslužitelju za naručivanje na terapiju. Web aplikacija mu prikazuje obrazac koji mora ispuniti kako i se naručio na terapiju. Pacijent unosi potrebne podatke (ime i prezime liječnika koji ga je uputio na rehabilitaciju, vrstu terapije, opis oboljenja i zahtjevani postupak liječenja). Pacijent šalje popunjeni obrazac. Web aplikacija na temelju primljenih podataka provjerava krebilitet liječnika u imeniku liječnika. Ovisno o rezultatu promjene web aplikacija će učiniti jednu od dvije stvari. Ako liječnik postoji u imeniku liječnika web-aplikacija sprema sve podatke u bazu podataka i prikazuje stranicu za odabir termina. U slučaju da liječnik ne postoji u imeniku liječnika aplikacija ispisuje poruku "Liječnik ne postoji"...

UC6 sekvencijski dijagram nije točan, kako ga poboljšati?

Koje bi još obrasce uporabe trebalo odabrati za napraviti sekvencijske dijagrame?



Slika 3.4: Sekvencijski dijagram za UC6

3.2 Ostali zahtjevi

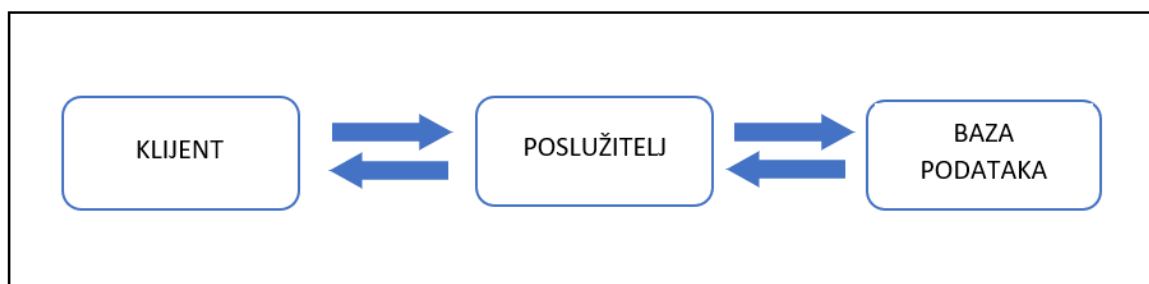
dio 1. revizije

Nefunkcionalni zahtjevi i zahtjevi domene primjene dopunjuju funkcionalne zahtjeve. Oni opisuju **kako se sustav treba ponašati** i koja **ograničenja** treba poštivati (performanse, korisničko iskustvo, pouzdanost, standardi kvalitete, sigurnost...). Primjeri takvih zahtjeva u Vašem projektu mogu biti: podržani jezici korisničkog sučelja, vrijeme odziva, najveći mogući podržani broj korisnika, podržane web/mobilne platforme, razina zaštite (protokoli komunikacije, kriptiranje...)... Svaki takav zahtjev potrebno je navesti u jednoj ili dvije rečenice.

4. Arhitektura i dizajn sustava

Sustav se može podijeliti na podsustave:

- Klijent
- Poslužitelj
- Baza podataka



Slika 4.1: Organizacija sustava

Klijent Klijent je web preglednik pomoću kojeg korisnici pristupaju našoj web aplikaciji. Često korišteni web preglednici su: Google Chrome, Apple Safari, Mozilla Firefox. Kada korisnik pristupa web aplikaciji, web preglednik šalje HTTP (engl. *Hypertext Transfer Protocol*) zahtjeve za preuzimanje statičkih datoteka web poslužitelju. Statičke datoteke mogu biti HTML, CSS i JavaScript datoteke koje sadrže React kod. Nakon preuzimanja datoteka, preglednik ih koristi za izgradnju i prikaz korisničkog sučelja te izvršavanje funkcija unutar aplikacije.

Poslužitelj Poslužitelj je Python kod pisan unutar web mikrookvira Flask. On služi kao posrednik između klijenta (korisničkog sučelja) i baze podataka. U Python kodu (Flask aplikaciji) definirani su RESTful API-ji (kao rute ili krajnje točke) koji omogućuju klijentima da šalju HTTP zahtjeve za određene podatke. Prilikom obrade tih zahtjeva, Python kod šalje upite bazi podataka kako bi dohvatio, promijenio ili dodao tražene podatke.

Baza podataka Kada baza podataka zaprimi upite od poslužitelja, ona ih izvršava i vraća rezultate poslužitelju. Vraćeni rezultati mogu biti u obliku potvrde o izvršenom upitu ili u obliku podataka koje je poslužitelj zatražio od baze podataka. Baza podataka ostaje pasivna sve dok ne zaprimi nove upite od poslužitelja.

Aplikacije je organizirana po MVC (*engl. Model-View-Controller*, hrv. Model-Pogled-Nadglednik) obrascu.

Općenito, *Model* definira podatke, njihovu strukturu i operacije koje se mogu izvršavati nad tim podacima. *Pogled* predstavlja prikaz podataka korisniku (npr. korisničko sučelje). *Nadglednik* predstavlja posrednike između Modela i Nadglednika. Oni obrađuju zahtjeve korisnika, vrše operacije nad Modelom i ažuriraju Pogled. Organizacijom prema ovom obrascu olakšava se proširivanje i održavanje aplikacije.

Naša aplikacija ne sadrži doslovno sve tri navedene komponente. Ovdje Python objedinjuje Model i dio logike Nadglednika. Komponente u React-u odgovaraju Pogledu te je i ovdje sadržan dio logike Nadglednika.

4.1 Baza podataka

Za rješavanje projektnog zadatka odabrana je relacijska baza podataka. Implementirali smo je korištenjem sustava PostgreSQL. To je besplatan sustav za upravljanje bazom podataka otvorenog koda. Objekti relacijske baze podataka su relacije. Neformalno, i u PostgreSQL-u, to su dvodimenzionalne imenovane tablice gdje imenovani stupac predstavlja atribut, a redak zapis relacije. Izbor ovakve baze podataka omogućio nam je lakše strukturiranje podatka i definiranje veza između njih, normalizaciju, skalabilnost te sigurnost kod pohrane i dohvata podataka. Zbog potreba naše aplikacije, baza podataka sadrži entitete:

- Korisnik
- Djelatnik
- Uloga
- Pacijent
- Terapija
- VrstaTerapije
- Termin
- Status
- Oprema
- ZauzetaOprema
- Soba
- Uredaj
- VrstaUredaja

4.1.1 Opis tablica

Korisnik:

Entitet Korisnik sadrži sve bitne informacije o korisniku aplikacije. Entitet sadrži attribute: idKorisnika, ime, prezime, datumRodnja, adresa, email, lozinka. Korisnik može biti ili djelatnik zdravstvene ustanove ili pacijent koji se želi naručiti na terapiju u zdravstvenoj ustanovi.

Korisnik		
idKorisnika	INT	jedinstveni identifikator korisnika
ime	VARCHAR(50)	ime korisnika
prezime	VARCHAR(50)	prezime korisnika
datumRodnja	DATE	datum rođenja korisnika
adresa	VARCHAR(100)	adresa mjesta stanovanja korisnika
email	VARCHAR(50)	e-mail adresa korisnika
lozinka	VARCHAR(50)	hash lozinke za prijavu u aplikaciju

Djelatnik:

Entitet Djelatnik je ekskluzivna specijalizacija entiteta Korisnik. Entitet sadrži attribute: idKorisnika, aktivan, idUloge.

Djelatnik		
idKorisnika	INT	jedinstveni identifikator korisnika (korisnik.idKorisnika)
aktivan	CHAR(2)	označava radni odnos liječnika i zdravstvene ustanove
idUloge	INT	jedinstveni identifikator uloge (uloga.idUloge)

Uloga:

Entitet Uloga služi za razlikovanje djelatnika zdravstvene ustanove. U aplikaciji razlikujemo liječnike od administratora. Entitet sadrži attribute: idUloge, imeUloge.

Uloga		
idUloge	INT	jedinstveni identifikator uloge
imeUloge	VARCHAR(50)	naziv uloge (liječnik ili administrator)

Pacijent:

Entitet Pacijent je ekskluzivna specijalizacija entiteta Korisnik. Entitet sadrži atribute: idKorisnika, MBO.

Pacijent		
idKorisnika	INT	jedinstveni identifikator korisnika (korisnik.idKorisnika)
MBO	CHAR(9)	matični broj osiguranika (pacijenta)

Terapija:

Entitet Terapija sadrži sve bitne informacije o terapiji na koju je pacijent naručen. Entitet sadrži atribute: idTerapije, idLijecnika, opisOboljenja, zahtPostLijec, datumPoc, datumZavrs, idPacijenta, idDjelatnika, idVrste.

Terapija		
idTerapije	INT	jedinstveni identifikator terapije
idLijecnika	INT	jedinstveni identifikator liječnika
opisOboljenja	VARCHAR(200)	opis oboljenja pacijenta
zahtPostLijec	VARCHAR(200)	zahtjev za postupkom liječenja (terapijom)
datumPoc	DATE	datum početka terapije
datumZavrs	DATE	datum završetka terapije
idPacijenta	INT	jedinstveni identifikator pacijenta (korisnik.idKorisnika)
idDjelatnika	INT	jedinstveni identifikator djelatnika (korisnik.idKorisnika)
idVrste	INT	jedinstveni identifikator vrste uređaja (vrstaTerapije.idVrste)

VrstaTerapije:

Entitet VrstaTerapije sadrži sve bitne informacije o vrsti terapije na koju je pacijent naručen. Entitet sadrži attribute: idVrste, imeVrste, opisVrste.

VrstaTerapije		
idVrste	INT	jedinstveni identifikator vrste terapije
imeVrste	VARCHAR(50)	naziv vrste terapije
opisVrste	VARCHAR(200)	opis vrste terapije

Termin:

Entitet Termin sadrži sve bitne informacije o terminu terapije na koji je pacijent naručen. Entitet Termin ne postoji bez entiteta vlasnika, entiteta Terapija. (Termin je egzistencijalno slab entitet.)

Termin		
idTermina	INT	jedinstveni identifikator termina terapije
idTerapije	INT	jedinstveni identifikator terapije (terapija.idTerapije)
od	DATETIME	datum i vrijeme početka termina
do	DATETIME	datum i vrijeme završetka termina
komentar	VARCHAR(200)	komentar liječnika o napretku terapije
brSobe	VARCHAR(10)	jedinstveni identifikator sobe (soba.brSobe)
idStatus	INT	jedinstveni identifikator statusa (status.idStatus)
idKorisnika	INT	jedinstveni identifikator korisnika (korisnik.idKorisnika)

Soba:

Entitet Soba sadrži sve bitne informacije o sobi u kojoj se provodi neka terapija. Entitet sadrži attribute: brSobe, kapacitet.

Soba		
brSobe	VRACHAR(10)	jedinstveni identifikator sobe
kapacitet	INT	kapacitet sobe, broj pacijenata koji istovremeno mogu biti na terapiji u nekoj sobi

Uredaj:

Entitet Uredaj sadrži sve bitne informacije o uređaju koji se koristi u nekoj terapiji. Entitet sadrži attribute: idUredaja, brSobe, idVrste.

Uredaj		
idUredaja	INT	jedinstveni identifikator uređaja
brSobe	VARCHAR(10)	jedinstveni identifikator sobe (soba.brSobe)
idVrste	INT	jedinstveni identifikator vrste uređaja (vrstaUredaja.idVrste)

VrstaUredaja:

Entitet VrstaUredaja sadrži sve bitne informacije o vrsti uređaja koja se primjenjuje u nekoj terapiji. Entitet sadrži attribute: idVrste, imeVrste, opisVrste.

VrstaUredaja		
idVrste	INT	jedinstveni identifikator vrste uređaja
imeVrste	VARCHAR(50)	naziv vrste uređaja
opisVrste	VARCHAR(200)	opis vrste uređaja

Oprema:

Entitet Oprema sadrži sve bitne informacije o opremi koja je na raspolaganju u zdravstvenoj ustanovi. Entitet sadrži attribute: idOpreme, imeOpreme, opisOpreme.

Oprema		
idOpreme	INT	jedinstveni identifikator opreme

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

Oprema		
imeOpreme	VARCHAR(50)	naziv opreme (npr. štuke)
opisOpreme	VARCHAR(200)	opis namjene opreme

ZauzetaOprema:

Entitet ZauzetaOprema sadrži sve bitne informacije o alokaciji opreme po terminima. Entitet sadrži atribute: idTermina, idTerapije, idOpreme.

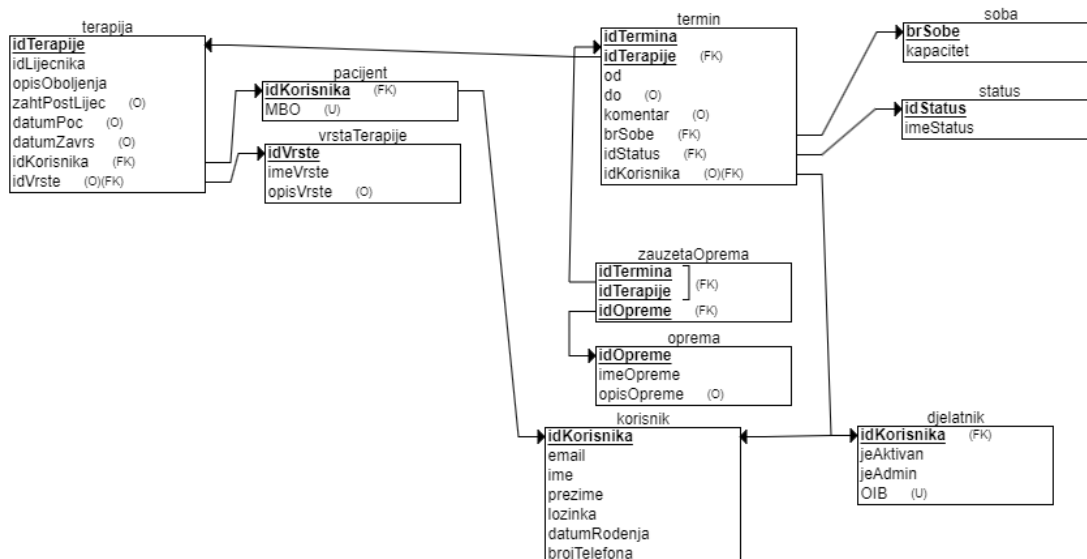
ZauzetaOprema		
idTermina	INT	jedinstveni identifikator termina (termin.idTermina)
idTerapije	INT	jedinstveni identifikator terapije (terapija.idTerapije)
idOpreme	INT	jedinstveni identifikator opreme (oprema.idOpreme)

Status:

Entitet Status sadrži sve bitne informacije o statusu termina terapija koji je dodijeljen nekom pacijentu. Entitet sadrži atribute: idStatus, imeStatus.

Status		
idStatus	INT	jedinstveni identifikator uređaja
imeStatus	VARCHAR(50)	naziv statusa (npr. u tijeku, završen...)

4.1.2 Dijagram baze podataka



Slika 4.2: Relacijska shema baze podataka

4.2 Dijagram razreda

Potrebno je priložiti dijagram razreda s pripadajućim opisom. Zbog preglednosti je moguće dijagram razlomiti na više njih, ali moraju biti grupirani prema sličnim razinama apstrakcije i srodnim funkcionalnostima.

dio 1. revizije

Prilikom prve predaje projekta, potrebno je priložiti potpuno razrađen dijagram razreda vezan uz **generičku funkcionalnost** sustava. Ostale funkcionalnosti trebaju biti idejno razrađene u dijagramu sa sljedećim komponentama: nazivi razreda, nazivi metoda i vrste pristupa metodama (npr. javni, zaštićeni), nazivi atributa razreda, veze i odnosi između razreda.

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 Selenium¹. 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.

¹<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>

Indeks slika i dijagrama

2.1	Primjer slične aplikacije	6
3.1	Dijagram obrasca uporabe - Korisnik i neregistrirani korisnik	19
3.2	Dijagram obrasca uporabe - Djelatnik i pacijent	20
3.3	Dijagram obrasca uporabe - Administrator	20
3.4	Sekvencijski dijagram za UC6	21
4.1	Organizacija sustava	23
4.2	Relacijska shema baze podataka	30

Dodatak: Prikaz aktivnosti grupe

Dnevnik sastajanja

1. sastanak

- Datum: 18. listopada 2023.
- Prisustvovali: svi
- Teme sastanka:
 - razjašnjavanje dobivene teme projekta - Medicinska Rehabilitacija
 - odabir tehnologija koje će biti korištene u izradi projekta

2. sastanak

- Datum: 23. listopada 2023.
- Prisustvovali: E.Badurina, L.Lasović, V.Rohak, K.Vrdoljak, F.Zlutar
- Teme sastanka:
 - razjašnjavanje teme projekta
 - zapisivanje konkretnih pitanja vezano uz projekt

3. sastanak

- Datum: 25. listopada 2023.
- Prisustvovali: svi
- Teme sastanka:
 - podjela uloga
 - razrada baze podataka

4. sastanak

- Datum: 6. studenoga 2023.
- Prisustvovali: svi
- Teme sastanka:
 - komentiranje predložka UI-dizajna
 - razrada i dovršavanje baze podataka
 - razjašnjavanje dodatnih nejasnoća, pitanja za laboratorijsku vježbu

5. sastanak

- Datum: 8. studenoga 2023.

- Prisustvovali: L.Akmačić, E.Badurina, L.Lasović
- Teme sastanka:
 - komentar do sad napravljenog

6. sastanak

- Datum: 15. studenoga 2023.
- Prisustvovali: -
- Teme sastanka:
 - revizija do sada napravljenog
 - Baza podataka
 - Funkcionalnost prijave i registracije

Tablica aktivnosti

Kontinuirano osvježavanje

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

	Ema Badurina	Lovro Akmačić	Andrea Jakovčević	Lucija Lasović	Vlatko Rohak	Karlo Vrdoljak	Fran Zlatar
Upravljanje projektom							
Opis projektnog zadatka	4	2	2	2	2	2	2
Funkcionalni zahtjevi	5			5			
Opis pojedinih obrazaca	5			5			
Dijagram obrazaca			3				
Sekvencijski dijagrami				2			
Opis ostalih zahtjeva							
Arhitektura i dizajn sustava				2			
Baza podataka	2	2	2	6	2	3	3
Dijagram razreda							
Dijagram stanja							
Dijagram aktivnosti							
Dijagram komponenti							
Korištene tehnologije i alati							
Ispitivanje programskog rješenja							
Dijagram razmještaja							

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

	Ema Badurina	Lovro Akmačić	Andrea Jakovčević	Lucija Lasović	Vlatko Rohak	Karlo Vrdoljak	Fran Zlatar
Upute za puštanje u pogon							
Dnevnik sastajanja	0.2						
Zaključak i budući rad							
Popis literature							
<i>Dodatne stavke kako ste podijelili izradu aplikacije</i>							
<i>npr. izrada početne stranice</i>							
<i>izrada baze podataka</i>							
<i>spajanje s bazom podataka</i>							
<i>back end</i>							

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 stranice, u izborniku Repository, pritiskom na stavku Contributors.