

Programsko inženjerstvo

Ak. god. 2022./2023.

Fizikalna terapija

Dokumentacija, Rev. 1

Grupa: *PROGI_Tim_2022*

Voditelj: *Fran Ogrinšak*

Datum predaje: *18.11.2022.*

Nastavnik: *Miljenko Krhen*

Sadržaj

1 Dnevnik promjena dokumentacije	3
2 Opis projektnog zadatka	5
3 Specifikacija programske potpore	9
3.1 Funkcionalni zahtjevi	9
3.1.1 Obrasci uporabe	11
3.1.2 Sekvencijski dijagrami	19
3.2 Ostali zahtjevi	22
4 Arhitektura i dizajn sustava	24
4.1 Baza podataka	25
4.1.1 Opis tablica	26
4.1.2 Dijagram baze podataka	28
4.2 Dijagram razreda	29
4.3 Dijagram stanja	30
4.4 Dijagram aktivnosti	31
4.5 Dijagram komponenti	32
5 Implementacija i korisničko sučelje	33
5.1 Korištene tehnologije i alati	33
5.2 Ispitivanje programskog rješenja	34
5.2.1 Ispitivanje komponenti	34
5.2.2 Ispitivanje sustava	34
5.3 Dijagram razmještaja	35
5.4 Upute za puštanje u pogon	36
6 Zaključak i budući rad	37
Popis literature	38
Indeks slika i dijagrama	39

Dodatak: Prikaz aktivnosti grupe

40

1. Dnevnik promjena dokumentacije

Kontinuirano osvježavanje

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napravljen predložak.	Dominik Đurinić, Fran Hruza	21.10.2022.
0.2	Napisani funkcionalni i nefunkcionalni zahtjevi	Fran Hruza, Dominik Đurinić	21.10.2022.
0.3	Dodan <i>Use Case</i> dijagram	Fran Hruza	26.10.2022.
0.4	Napisan opis projektnog zadatka	Dominik Đurinić	7.11.2022.
0.5	Dodani sekvencijski dijagrami	Fran Hruza	7.11.2022.
0.6	Dodan dijagram razreda	Fran Hruza	15.11.2022.
0.7	Arhitektura i baza podataka (opis i ER dijagram)	Dominik Đurinić	16.11.2022.
1.0	Verzija samo s bitnim dijelovima za 1. ciklus	Fran Hruza, Dominik Đurinić	18.11.2022.

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

Rev.	Opis promjene/dodatka	Autori	Datum
------	-----------------------	--------	-------

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

Ideja projekta je izrada web aplikacije "Fizikalna terapija" za potrebe zdravstvene ustanove "Zdravljem do znanja".

Djelatnosti koje obavlja ova ustanova su fizikalna terapija i rehabilitacija bolesnika nakon lakših ili težih povreda. Ustanova broji ukupno 15 djelatnika, od kojih su 9 fizikalni terapeuti, 3 medicinske sestre, 2 administratora te voditelj ustanove. Broj djelatnika nije konačan već se može mijenjati.

Zahtjev zdravstvene ustanove je učinkovit informacijski sustav koji omogućuje izradu rasporeda bolesnika na terapije koje ustanova pruža, praćenje podataka o bolesnicima i njihovom napretku u liječenju. Također sustav bi trebao omogućiti praćenje rada svih djelatnika ustanove. Pristup navedenim mogućnostima imaju samo registrirani djelatnici ustanove. Registrirani korisnici usluga zdravstvene ustanove imaju mogućnost uvida u vlastiti termin te popis i opis usluga/tretmana koje dobivaju. Neregistrirani korisnici mogu vidjeti samo popis i opis svih usluga/tretmana koje ustanova pruža.

Razlikujemo četiri vrste korisnika informacijskog sustava:

- vlasnik sustava (administrator)
- voditelj ustanove
- medicinski djelatnici (fizioterapeuti i medicinske sestre)
- bolesnici (korisnici usluga)

Vlasnik sustava (administrator) upisuje sve podatke o uslugama/tretmanima te registrira medicinske djelatnike, tj. unosi podatke o medicinskim djelatnicima. Vlasnik sustava može, ali i ne mora biti zaposlenik ustanove.

Voditelj ustanove upisuje sve podatke o uslugama/tretmanima, registrira medicinske djelatnike te ima mogućnost pregleda aktivnosti u ustanovi: pregled aktivnosti djelatnika, broj pacijenata po djelatniku i broj pacijenata po usluzi/tretmanu te

zauzeće pojedinih uređaja. Sve aktivnosti može pregledavati u proizvoljnom vremenskom intervalu.

Medicinski djelatnik može biti fizioterapeut ili medicinska sestra. Nakon što je registriran od strane administratora ili voditelja ustanove, može se prijaviti u sustav i pristupiti funkcionalnostima sustava. Funkcionalnosti su: izrada rasporeda pacijenata, registracija pacijenata, evidencija dolaska pacijenta i obavljenosti terapije te evidencija stanja pacijenta na kraju terapije. Također može kao i voditelj ustanove pristupiti pregledu broja pacijenata po usluzi/tretmanu, zauzeću pojedinih uređaja te broju pacijenata, ali samo za one pacijente za koje je registriran kao voditelj za određeni dan.

Bolesnici se pri prvom dolasku u ustanovu registriraju od strane medicinskog djelatnika. Nakon registracije, bolesnik ima mogućnost prijave u sustav i pregleda vlastitog rasporeda tretmana, njihovog opisa te potvrde o dolaznosti na terapiju.

Sustav omogućava istovremeni rad svih korisnika sustava te podržava unos hrvatskih dijakritičkih znakova.

Za registraciju medicinskog djelatnika potrebni su sljedeći podaci:

- ime
- prezime
- stručna sprema
- specijalizacija (ukoliko postoji)
- adresa elektroničke pošte
- broj telefona
- nedostupnost medicinskog djelatnika (bolovanje, godišnji odmor i dr.)

Za registraciju bolesnika potrebni su sljedeći podaci:

- ime
- prezime
- broj telefona
- adresa elektroničke pošte (ukoliko postoji)
- dijagnoza (iz liste dijagnoza bolesti prema MKB)
- MBO (matični broj osiguranika)

- dodatno osiguranje (da/ne)
- ime i prezime liječnika koji je uputio bolesnika na terapiju (iz liste liječnika)
- lista odabranih tretmana/usluga (na koje je bolesnik upućen)

Pri prvom dolasku bolesnika u ustanovu, medicinski djelatnik koji ga zaprima izvršava registraciju bolesnika. Ukoliko je bolesnik već prije bio na nekom tretmanu, odnosno ima izvršenu registraciju, onda nije potrebna ponovna registracija već se podaci dohvaćaju iz baze podataka. Nakon registracije, isti medicinski djelatnik koji je zaprimio bolesnika izrađuje personalizirani raspored tretmana za bolesnika. Bitno je naglasiti sljedećih nekoliko uvjeta vezanih uz mogućnost izrade rasporeda.

1. Radno vrijeme ustanove je 8 sati dnevno u razdoblju od 9:00 do 17:00 sati, od ponedjeljka do petka. Unutar tog razdoblja se vrše medicinske usluge pacijentima. Subotom, nedjeljom, blagdanima i praznicima je ustanova zatvorena.
2. Ustanova posjeduje ograničeni broj resursa (uređaja i prostora) te je trajanje pojedine terapije vremenski unaprijed određeno. Svi uređaji mogu raditi paralelno i u isto vrijeme.
3. Svaki bolesnik ima terapiju u trajanju od 10 dana i to uvijek u isto vrijeme u danu. Ne postoji mogućnost nadoknade u slučaju ne dolaska bolesnika na predviđeni termin.

Potrebno je uskladiti izradu rasporeda bolesnika sa navedenim uvjetima, tj. poslovanjem zdravstvene ustanove.

Prilikom svakodnevnog dolaska bolesnika na terapiju, medicinski djelatnik evidentira bolesnikov dolazak. Sustav automatski registrira po danima medicinskog djelatnika koji je prihvatio pojedinog bolesnika.

Nakon izvršenog desetog dana terapije medicinski djelatnik obavlja razgovor s bolesnikom te u sustav unosi stanje bolesnika nakon terapije (napredak, zadovoljstvo, aktualni problem, izjava bolesnika). Također postoji mogućnost unosa zapažanja medicinskog djelatnika u sustav. Nakon razgovora izrađuje se dokument koji sadrži sve podatke o bolesniku, tretmanu/usluzi, broju dolazaka, imenima djelatnika koji su zaprimili bolesnika po danima te podaci uneseni prilikom završnog razgovora. Stvara se dokument u pdf formatu, koji se onda ispisuje te ga

bolesnik potpisuje. Dokument se pohranjuje u arhivu ustanove te se po želji bolesnika može poslati elektronička verzija dokumenta na adresu elektroničke pošte bolesnika. To je ujedno i završni faza tretmana/usluge.

Opasnost od Covid zaraze je još uvijek prisutan u svakodnevnom životu, stoga kao podsjetnik na zarazu sustav preuzima podatke s javno dostupnog servisa o broju zaraženih u Republici Hrvatskoj na taj dan i trend broja zaraza u određenom vremenskom intervalu. Svi navedeni podaci će se prikazati medicinskim djelatnicima prilikom prijave u sustav.

3. Specifikacija programske potpore

3.1 Funkcionalni zahtjevi

Dionici:

1. Voditelj ustanove
2. Bolesnici
3. Medicinski djelatnici
 - (a) Fizioterapeuti
 - (b) Medicinske sestre
4. Administrator (vlasnik sustava)
5. Razvojni tim

Aktori i njihovi funkcionalni zahtjevi:

1. Neregistrirani korisnik (inicijator) može:
 - (a) pregledati popis i opis usluga koje ustanova pruža
2. Registrirani bolesnik (inicijator) može:
 - (a) vidjeti svoj raspored te potvrdu dolaska
 - (b) pregledati popis i opis svojih usluga/tretmana
3. Administrator (inicijator) može:
 - (a) upisati sve podatke o uslugama/tretmanima
 - (b) registrirati podatke o medicinskim djelatnicima
4. Voditelj ustanove (inicijator) može:
 - (a) upisati sve podatke o uslugama/tretmanima
 - (b) registrirati podatke o medicinskim djelatnicima
 - (c) pregledati aktivnosti u ustanovi
 - i. pregled aktivnosti djelatnika
 - ii. broj pacijenata po djelatniku

iii. broj pacijenata po tretmanu/usluzi

5. Medicinski djelatnik (inicijator) može:

- (a) upisati podatke o pacijentu
- (b) raditi raspored pacijenata
- (c) evidentirati dolazak pacijenata
- (d) evidentirati stanje pacijenta nakon terapije
 - i. napredak pacijenta
 - ii. zadovoljstvo pacijenta uslugom
 - iii. izjava bolesnika
 - iv. vlastiti komentar i zapažanja

6. Baza podataka (sudionik):

- (a) pohranjuje podatke o korisnicima i njihovim ovlastima
- (b) pohranjuje podatke o uslugama koje ustanova nudi
- (c) pohranjuje raspored terapija

3.1.1 Obrasci uporabe

dio 1. revizije

Opis obrazaca uporabe

Funkcionalne zahtjeve razraditi u obliku obrazaca uporabe. Svaki obrazac je potrebno razraditi prema donjem predlošku. Ukoliko u nekom koraku može doći do odstupanja, potrebno je to odstupanje opisati i po mogućnosti ponuditi rješenje kojim bi se tijekom obrasca vratio na osnovni tijek.

UC1 - Registracija medicinskog djelatnika

- **Glavni sudionik:** Administrator, voditelj ustanove
- **Cilj:** Registracija i unos podataka medicinskog djelatnika
- **Sudionici:** Baza podataka, medicinski djelatnik
- **Preduvjet:** Prijavljenost u sustav
- **Opis osnovnog tijeka:**
 1. Administrator/Voditelj odabire opciju za registraciju medicinskog djelatnika
 2. Administrator/Voditelj upisuje potrebne podatke o djelatniku
 3. Administrator/Voditelj dobiva obavijest o uspješnoj registraciji djelatnika
- **Opis mogućih odstupanja:**
 - 2.a Unos podataka u nedozvoljenom formatu
 1. Sustav obavještava administratora/voditelja o krivom formatu podataka
 2. Administrator/Voditelj mijenja podatke u ispravan format ili odstupa od registracije

UC2 - Unos tretmana/usluga u ponudu

- **Glavni sudionik:** Administrator, voditelj ustanove
- **Cilj:** Ažurirati popis/opis tretmana/usluga koje ustanova nudi
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava voditelja/administratora u sustav
- **Opis osnovnog tijeka:**
 1. Administrator/Voditelj odabire opciju za ažuriranje ponude

2. Administrator/Voditelj odabire želi li stvoriti novi zapis ili odabire tretman/uslugu koju želi ažurirati
3. Administrator/Voditelj upisuje naziv i opis usluge

UC3 - Pregled aktivnosti u ustanovi

- **Glavni sudionik:** Voditelj ustanove
- **Cilj:** Pregled aktivnosti djelatnika, broj bolesnika po djelatniku te broj pacijenata po tretmanu/usluzi te zauzeće pojedinih uređaja
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava voditelja u sustav
- **Opis osnovnog tijeka:**
 1. Voditelj odabire funkcionalnost pregleda aktivnosti u ustanovi
 2. Voditelj odabire u kojem vremenskom intervalu želi informacije
 3. Sustav prikazuje tražene informacije

UC4 - Prijava medicinskog djelatnika

- **Glavni sudionik:** Medicinski djelatnik
- **Cilj:** Dobiti pristup funkcijama medicinskog djelatnika
- **Sudionici:** Baza podataka
- **Preduvjet:** Registracija medicinskog djelatnika
- **Opis osnovnog tijeka:**
 1. Medicinski djelatnik unosi svoj adresu elektroničke pošte i lozinku koju mu je dodijelio administrator/voditelj
 2. Potvrda o ispravnosti unesenih podataka
 3. Sustav prikazuje aktualne podatke o broju zaraženih Covidom-19 taj dan te trend broja zaraženih u određenom intervalu
 4. Pristup funkcijama medicinskog djelatnika
- **Opis mogućih odstupanja:**
 - 2.a Unos neispravne adrese elektroničke pošte/lozinke
 1. Sustav obavještava medicinskog djelatnika o neispravnom unosu podataka i vraća ga na stranicu za prijavu u sustav

UC5 - Registracija bolesnika

- **Glavni sudionik:** Medicinski djelatnik
- **Cilj:** Registracija i unos podataka bolesnika
- **Sudionici:** Baza podataka, Bolesnik

- **Preduvjet:** Prijava medicinskog djelatnika
 - **Opis osnovnog tijeka:**
 1. Medicinski djelatnik odabire opciju za registraciju bolesnika
 2. Medicinski djelatnik upisuje MBO bolesnika
 3. Sustav provjerava postoji li zapis o bolesniku s unesenim MBO-om (bolesnik je prethodno odradio jednu ili više terapija) i automatski registrira bolesnika
 4. Ako je bolesnik prvi put u ustanovi, medicinski djelatnik upisuje podatke
 - **Opis mogućih odstupanja:**
- 2.a Unos podataka u nedozvoljenom formatu
1. Sustav obavještava medicinskog djelatnika o krivom formatu podataka
 2. Medicinski djelatnik mijenja podatke u ispravan format ili odustaje od registracije

UC6 - Stvaranje rasporeda tretmana za bolesnika

- **Glavni sudionik:** Medicinski djelatnik
 - **Cilj:** Stvoriti raspored tretmana/usluga za bolesnika
 - **Sudionici:** Baza podataka
 - **Preduvjet:** Registracija bolesnika
 - **Opis osnovnog tijeka:**
 1. Medicinski djelatnik odabire bolesnika iz liste registriranih bolesnika
 2. Medicinski djelatnik unosi tretmane i njihov termin
 3. Sustav potvrđuje ispravnost podataka s obzirom na ograničenja
 - **Opis mogućih odstupanja:**
- 2.a Unos podataka koji su u koliziji s rasporedima drugih bolesnika
1. Sustav obavještava medicinskog djelatnika o zauzetom terminu te ga vraća na stranicu za izradu rasporeda

UC7 - Prihvat i evidencija dolaska bolesnika

- **Glavni sudionik:** Medicinski djelatnik
- **Cilj:** Prihvatiti i evidentirati dolazak bolesnika na terapiju
- **Sudionici:** Baza podataka, bolesnik
- **Preduvjet:** Prijava medicinskog djelatnika
- **Opis osnovnog tijeka:**

1. Medicinski djelatnik odabire funkcionalnost za evidenciju dolaska bolesnika
2. Medicinski djelatnik unosi ime i prezime te MBO bolesnika
3. Sustav potvrđuje ispravnost podataka
4. Sustav osvježava podatke u rasporedu bolesnika te podatke tko je pojedinog bolesnika taj dan prihvatio

- **Opis mogućih odstupanja:**

2.a Unos podataka koji su u koliziji s rasporedima drugih bolesnika

1. Sustav obavještava medicinskog djelatnika o neispravnosti podataka (bolesnik nije registriran ili nema termin taj dan)
2. Sustav medicinskog djelatnika vraća na stranicu za evidenciju dolaska bolesnika

UC8 - Evidencija kraja tretmana/usluge

- **Glavni sudionik:** Medicinski djelatnik
- **Cilj:** Stvoriti izvještaj iz informacija završnog razgovora
- **Sudionici:** Baza podataka, bolesnik
- **Preduvjet:** Prijava medicinskog djelatnika
- **Opis osnovnog tijeka:**
 1. Medicinski djelatnik odabire bolesnika za kojeg želi napraviti izvješće
 2. Medicinski djelatnik odabire funkcionalnost za izradu izvješća po završetku terapije
 3. Medicinski djelatnik unosi podatke iz intervjua te opcionalno svoje komentare i zapažanja
 4. Sustav stvara dokument pdf dokument
 5. Sustav nudi djelatniku mogućnost slanja dokumenta na adresu elektroničke pošte bolesnika

- **Opis mogućih odstupanja:**

2.a Odabrani bolesnik nije završio terapiju

1. Sustav obavještava medicinskog djelatnika da nije moguće stvoriti izvještaj za odabranog bolesnika
2. Sustav medicinskog djelatnika vraća na popis bolesnika

3.a Neispravan format podataka

1. Sustav obavještava medicinskog djelatnika o neispravnom formatu podataka
2. Sustav medicinskog djelatnika vraća na stranicu za izradu izvješća

4.a Ne postoji zapis o adresi elektroničke pošte bolesnika

1. Sustav nudi medicinskom djelatniku da upiše adresu elektroničke pošte bolesnika
2. Medicinski djelatnik upisuje adresu koju je dobio od bolesnika

UC9 - Pregled aktivnosti/informacija o uređajima i terminima

- **Glavni sudionik:** Medicinski djelatnik/Voditelj
- **Cilj:** Omogućiti djelatniku prikaz broja bolesnika po određenoj usluzi/tretmanu, zauzeće pojedinih uređaja i broj bolesnika za koje su taj dan voditelj oni registrirani kao voditelji za određeni dan.
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava medicinskog djelatnika
- **Opis osnovnog tijeka:**
 1. Medicinski djelatnik odabire opciju za pregled aktivnosti
 2. Sustav prikazuje broj bolesnika po određenoj usluzi/tretmanu, zauzeće pojedinih uređaja i broj bolesnika za koje je djelatnik voditelj taj dan

UC10 - Pregled usluga

- **Glavni sudionik:** Bolesnik
- **Cilj:** Pregledati popis i opis usluga ustanove
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Prikazuje se popis svih usluga koje ustanova nudi

UC11 - Prijava bolesnika u sustav

- **Glavni sudionik:** Bolesnik
- **Cilj:** Dobiti pristup korisničkom sučelju
- **Sudionici:** Baza podataka
- **Preduvjet:** Registracija bolesnika
- **Opis osnovnog tijeka:**
 1. Bolesnik unosi svoj MBO i lozinku koju stvara medicinski djelatnik
 2. Potvrda o ispravnosti unesenih podataka
 3. Pristup funkcijama prijavljenog bolesnika
- **Opis mogućih odstupanja:**
 - 2.a Neispravan MBO/lozinka

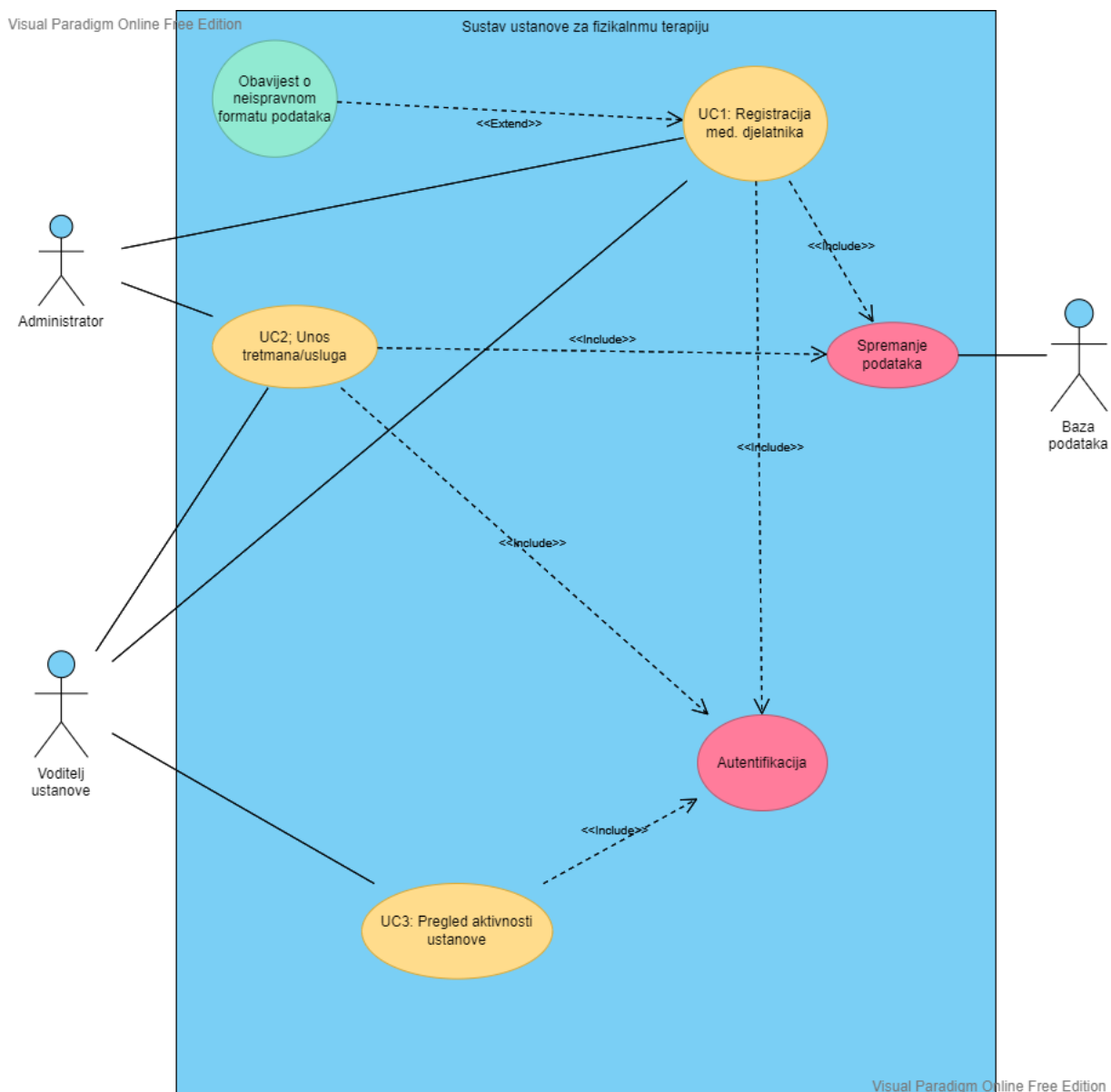
1. Sustav obavještava bolesnika o krivim podacima i vraća ga na stranicu za prijavu u sustav

UC12 - Pregled termina tretmana i evidencije dolaznosti

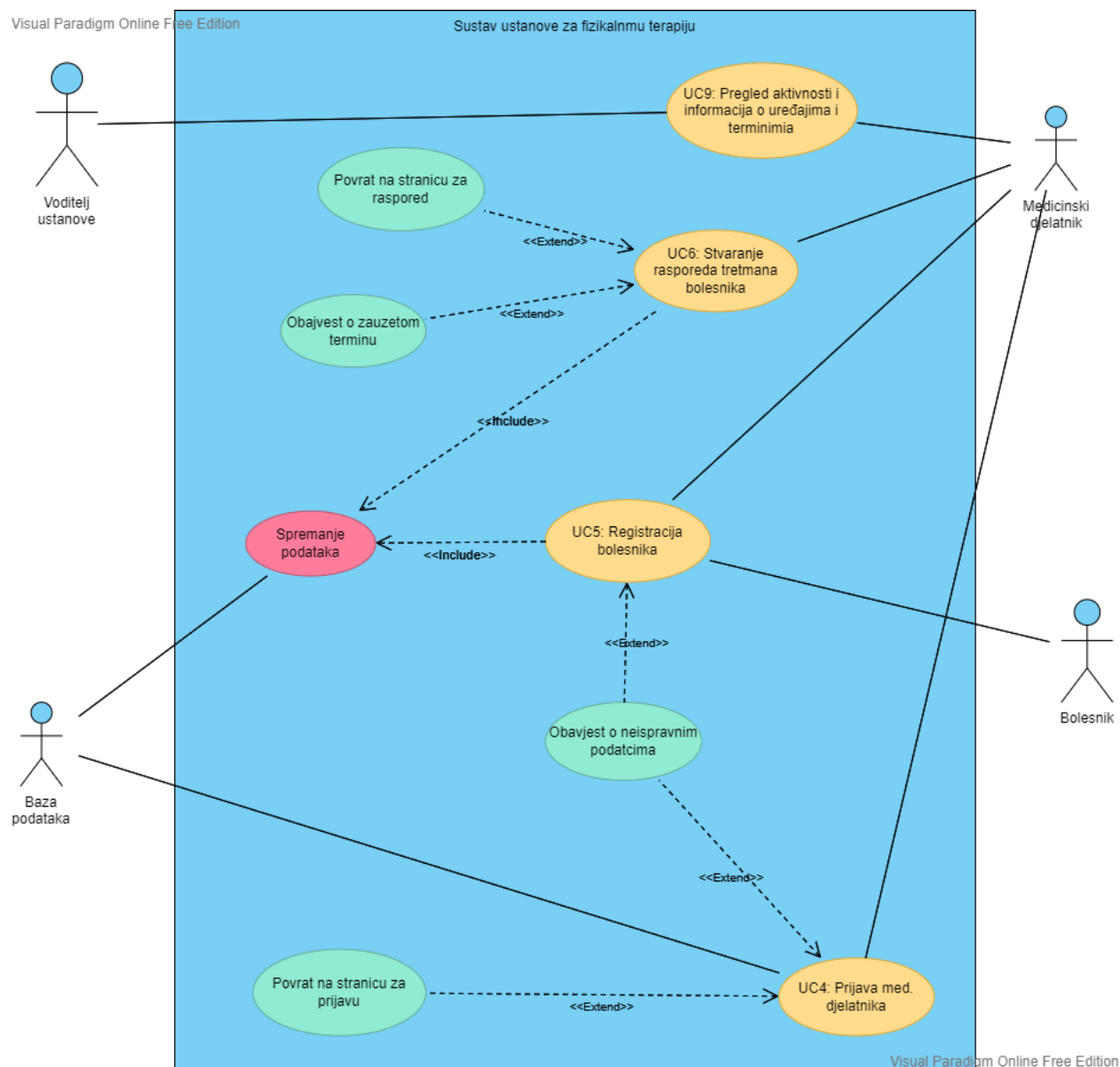
- **Glavni sudionik:** Bolesnik
- **Cilj:** Pregledati preostale tretmane i evidentirane dolaske
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava bolesnika
- **Opis osnovnog tijeka:**
 1. Bolesnik odabire opciju za prikaz termina i dolaznosti
 2. Sustav bolesnika preusmjerava na stranicu sa popisom termina tretmana te evidentiranim dolaskom

Dijagrami obrazaca uporabe

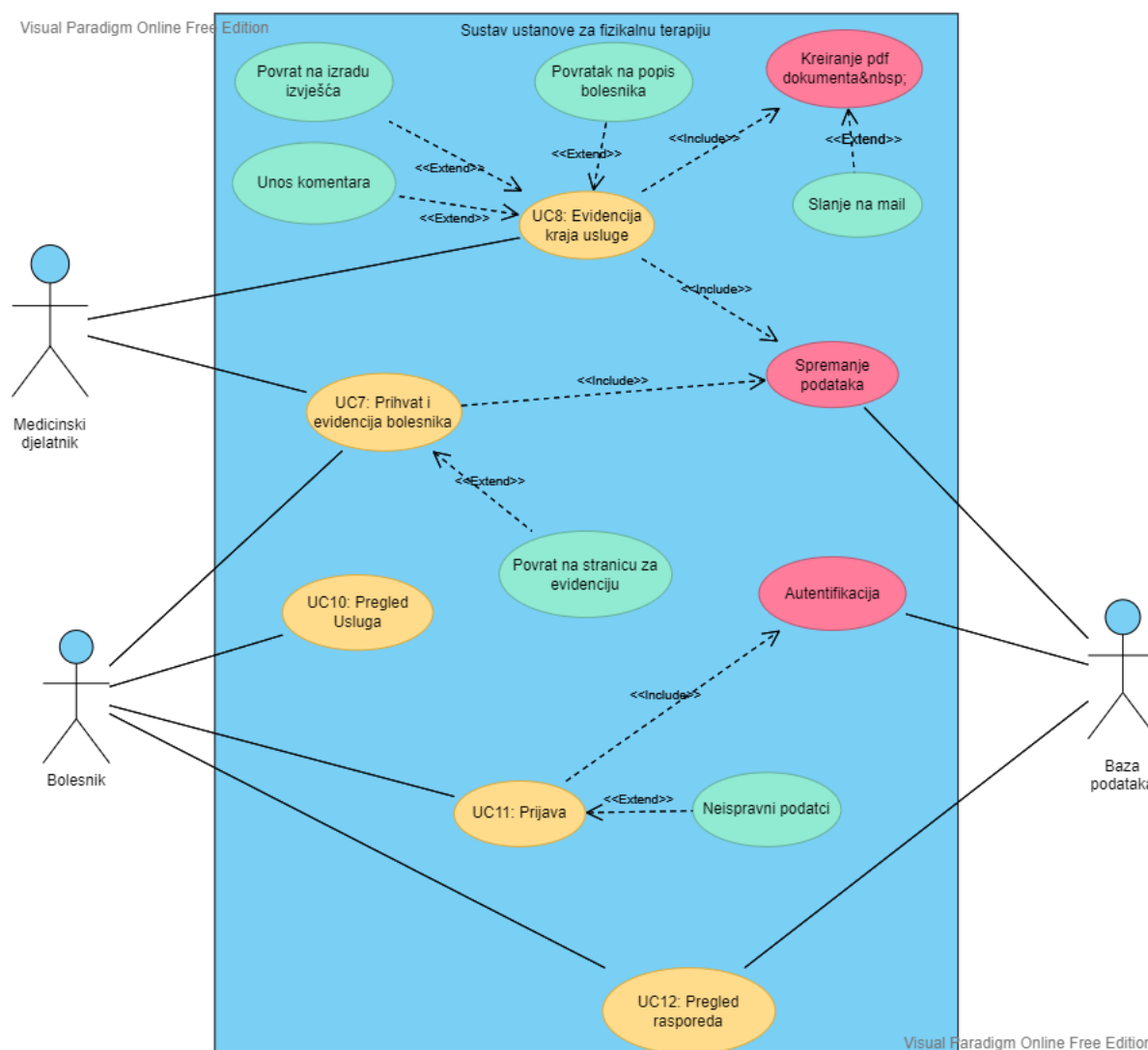
Prikazati odnos aktora i obrazaca uporabe odgovarajućim UML dijagramom. Nije nužno nacrtati sve na jednom dijagramu. Modelirati po razinama apstrakcije i skupovima srodnih funkcionalnosti.



Slika 3.1: UML obrazaca korištenja UC1 do UC3



Slika 3.2: UML obrazaca korištenja UC4 do UC6 te UC9



Slika 3.3: UML obrazaca korištenja UC7 do UC8 te od UC10 do UC12

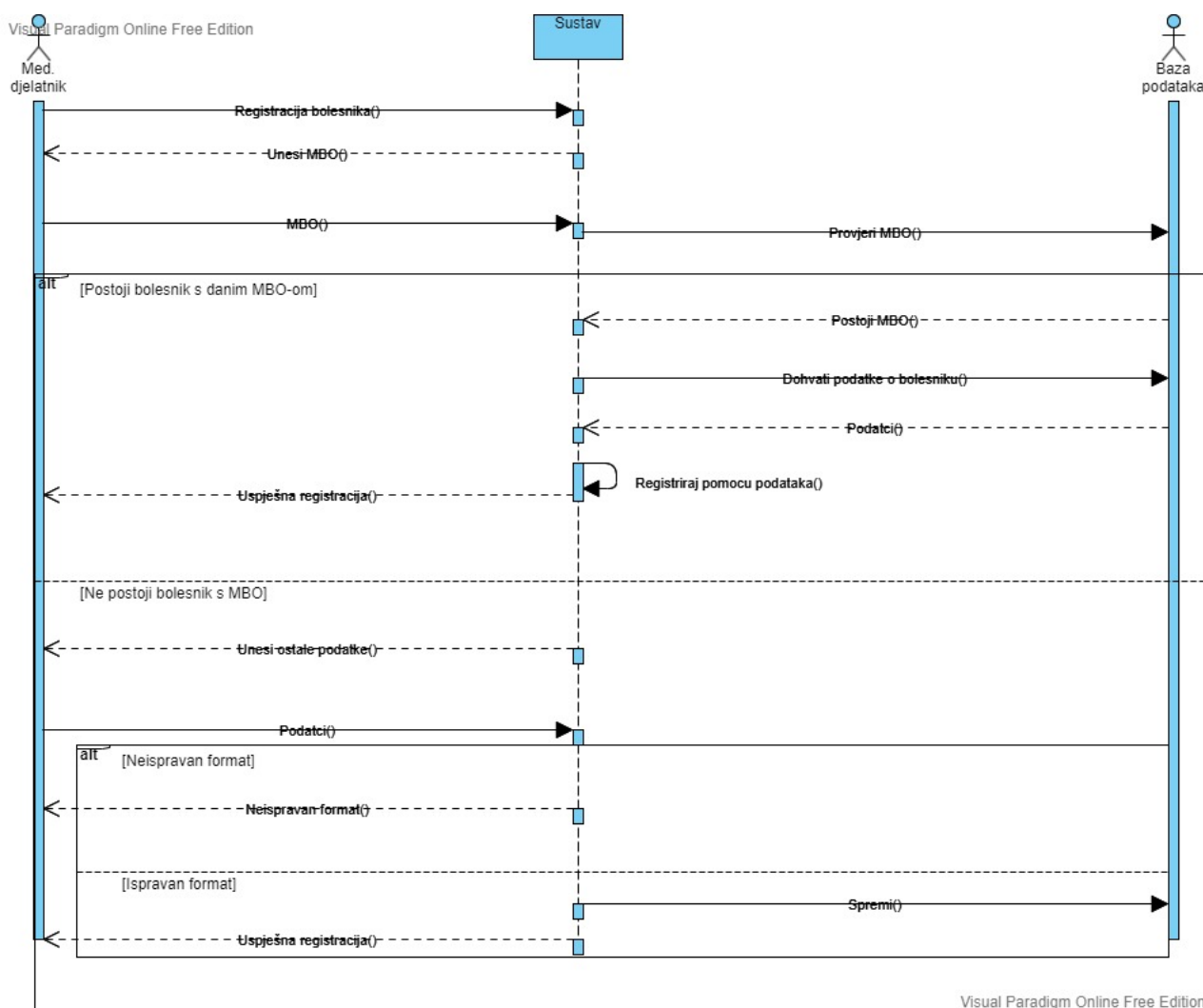
3.1.2 Sekvencijski dijagrami

dio 1. revizije

Nacrtati sekvencijske dijagrame koji modeliraju najvažnije dijelove sustava (max. 4 dijagrama). Ukoliko postoji nedoumica oko odabira, razjasniti s asistentom. Uz svaki dijagram napisati detaljni opis dijagrama.

Opis sekvencijskog dijagrama UC5

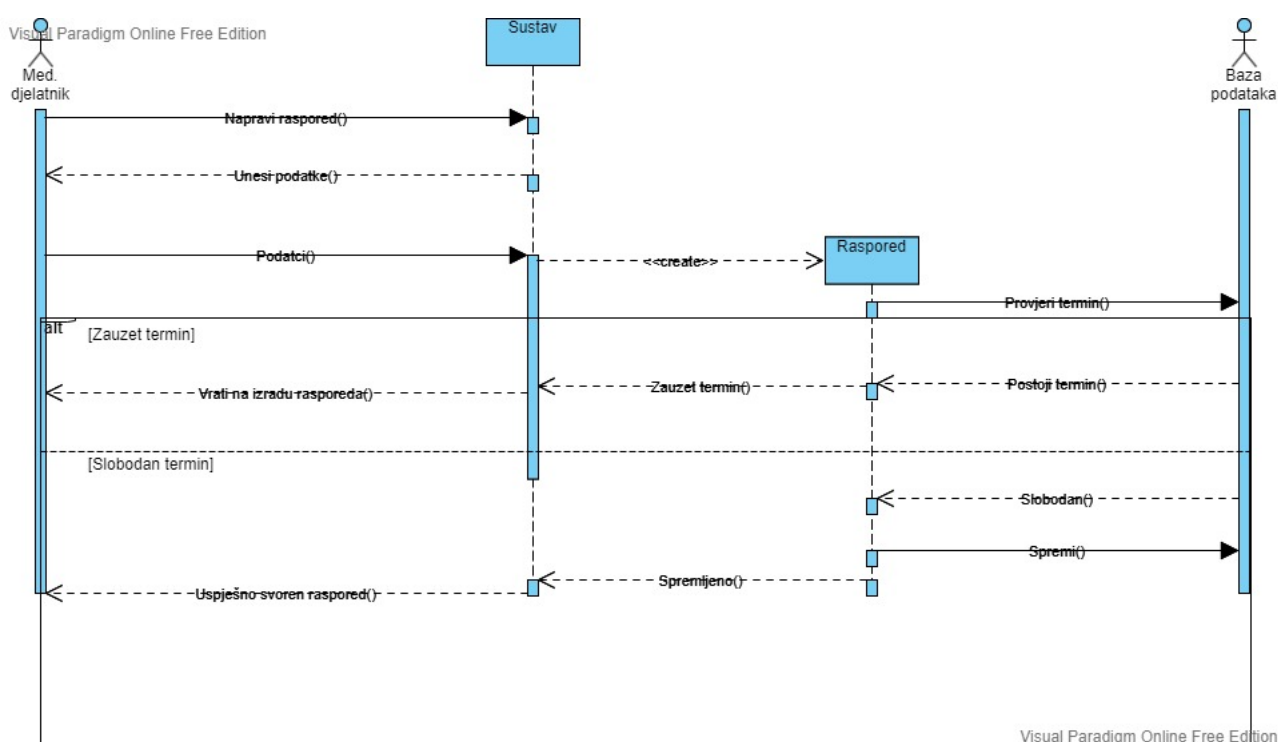
Medicinski djelatnik započinje registraciju bolesnika odabirom opcije na svom korisničkom sučelju. Sustav očekuje MBO bolesnika te po primitku MBO-a šalje upit bazi o postojanju zapisa bolesnika s unesenim MBO-om. Ako takav zapis postoji, sustav automatski iz njega registrira bolesnika te javlja medicinskom djelatniku da je registracija uspješna. Ako zapis ne postoji, sustav traži djelatnika podatke o bolesniku te mu javlja pogrešku ako je format podataka neispravan ili dojavljuje uspjeh registracije.



Slika 3.4: Sekvencijski dijagram UC5

Opis sekvencijskog dijagrama UC6

Medicinski djelatnik započinje izradu rasporeda odabirom opcije za izradu rasporeda iz korisničkog sučelja. Sustav očekuje podatke o tretmanima i terminima. Po primitku podataka, sustav stvara objekt raspored i provjerava jesu li termini unesenih tretmana zauzeti. Ako su termini zauzeti sustav šalje poruku medicinskom djelatniku poruku o zauzetom terminu te ga vraća na sučelje za izradu rasporeda. Ako su svi termini slobodni, raspored se sprema u bazu podataka te se sustav dojavljuje poruku o uspješnom stvaranju rasporeda te ga vraća na početnu stranicu.

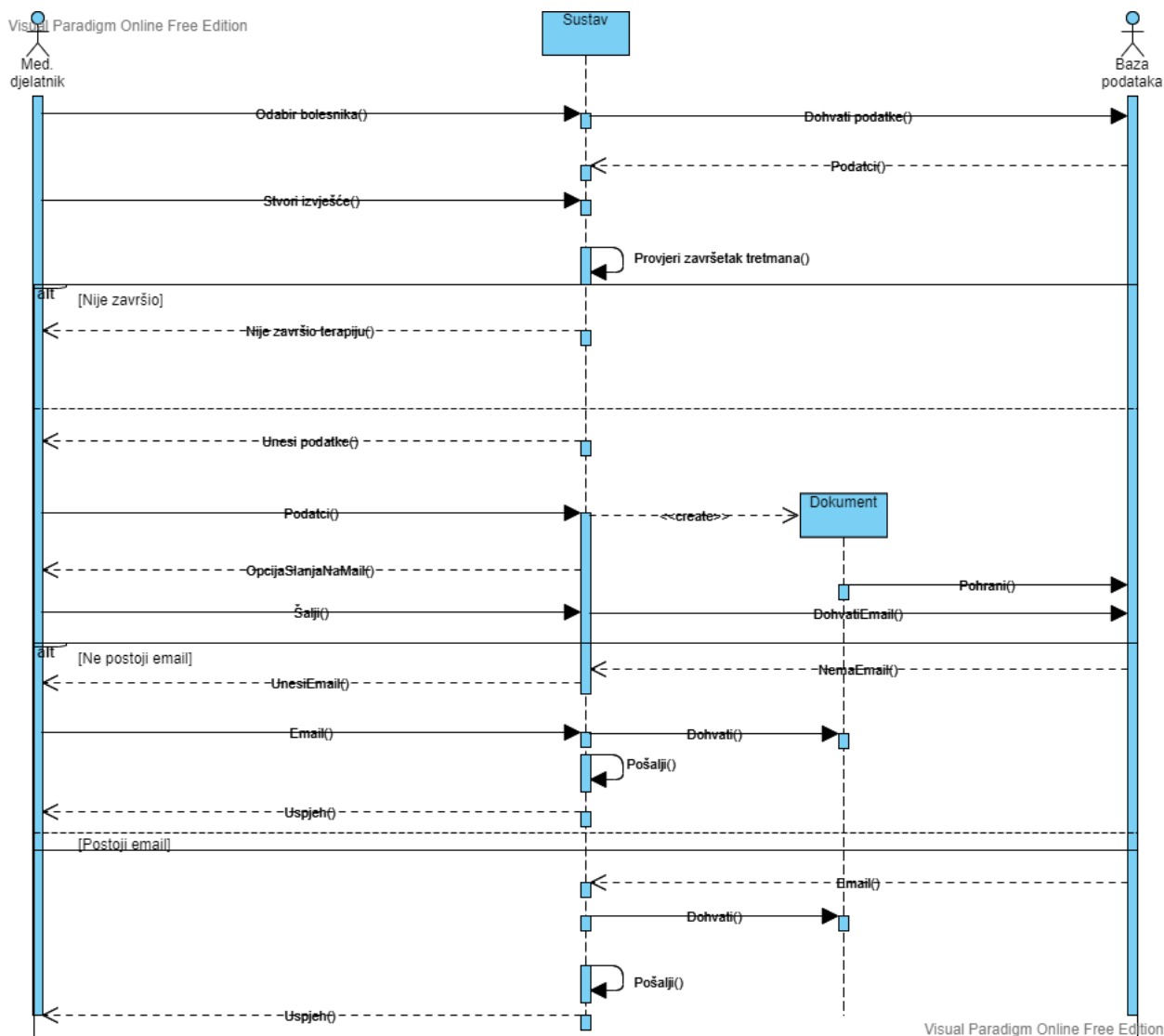


Slika 3.5: Sekvencijski dijagram UC6

Opis sekvencijskog dijagrama UC8

Kada medicinski djelatnik želi izraditi izvješće za završetak bolesnikova tretmana, odabire bolesnika za kojeg želi izraditi izvješće. Sustav dohvaća podatke o bolesniku te djelatnik odabire funkcionalnost izrade izvješća iz korisničkog sučelja. Sustav provjerava je li bolesnik gotov s propisanim tretmanom te ako nije, sustava obavještava djelatnika te ga vraća na početnu stranicu. U suprotnom sustav traži podatke od djelatnika te nakon što dobije podatke stvara dokument i

pohranjuje ga. Nakon što uspješno generira dokument, sustav o tome obavještava djelatnika i nudi mu opciju slanja dokumenta na adresu elektroničke pošte bolesnika. U slučaju da zapis o adresi elektroničke pošte bolesnika ne postoji, sustav od djelatnika traži da ju unese te šalje dokument. Ako zapis postoji, sustav dokument šalje na tu adresu.



Slika 3.6: Sekvencijski dijagram UC8

3.2 Ostali zahtjevi

dio 1. revizije

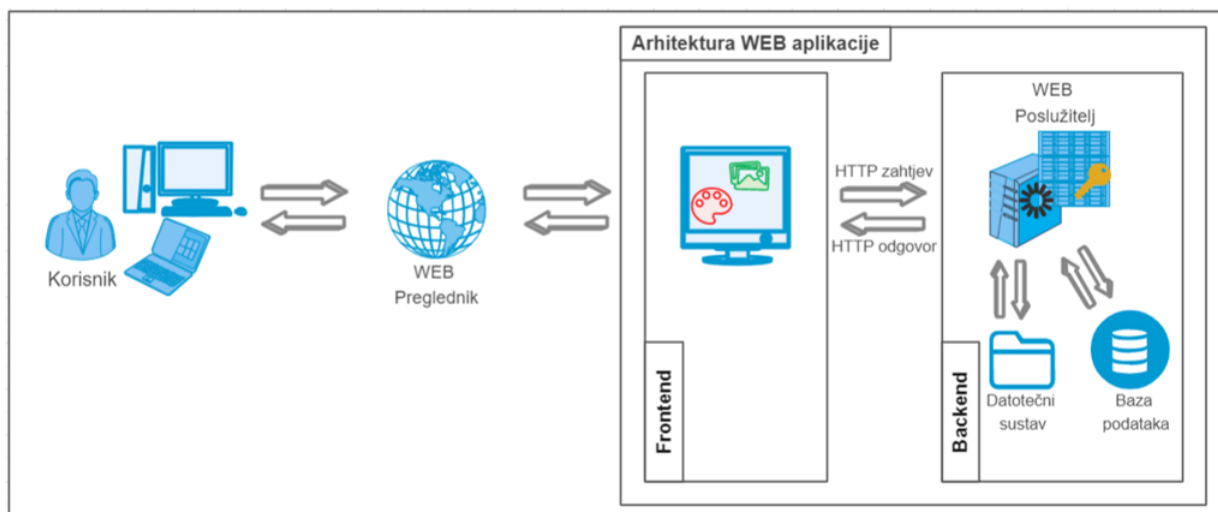
*Nefunkcionalni zahtjevi i zahtjevi domene primjene dopunjuju funkcionalne zahtjeve. Oni opisuju **kako se sustav treba ponašati** i koja **ograničenja** treba poštivati (performanse, korisničko iskustvo, pouzdanost, standardi kvalitete, sigurnost...). Primjeri takvih zahtjeva u Vašem projektu mogu biti: 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.*

1. Dijagnoza se odabire iz liste dijagnoza bolesti prema važećoj MKB (Međunarodna klasifikacija bolesti i srodnih zdravstvenih problema)
2. Radno vrijeme ustanove je od 9:00 do 17:00 od ponedjeljka do petka, termini tretmana određeni ovim vremenom
3. Ograničen broj uređaja
4. Omogućen unos hrvatskih dijakritičkih znakova (UTF-8 encoding)
5. Sustav izveden u objektnoj paradigmi

4. Arhitektura i dizajn sustava

dio 1. revizije

Arhitekturu sustava definiramo kao struktura sustava koja sadrži elemente programa i njihova izvana vidljiva obilježja te odnose među njima. Arhitekturu sustava dijelimo u nekoliko podsustava i prikazujemo njihovu međusobnu komunikaciju.



Slika 4.1: Arhitektura sustava

Arhitektura našeg sustava odnosno web aplikacije sastoji se od dva osnovna dijela:

- Frontend
- Backend

Frontend predstavlja sve što krajnji korisnik vidi (prezentacija sadržaja, uređenje te raspored elemenata web aplikacije) i sve čime vrši interakciju (unos, odabir, obrazac, gumbi, i dr.).

Backend predstavlja logiku web aplikacije odnosno kako web aplikacija funkcionira te kako pristupa podacima iz baze podataka i datotečnog sustava.

Frontend komunicira s backendom preko API-a. API (engl. *Application Programming Interface*) predstavlja način na koji dva ili više računalnih programa međusobno komuniciraju. Kod web aplikacija se ta komunikacija temelji na HTTP zahtjevima i HTTP odgovorima.

U backendu dijelu razlikujemo dva podsustava: web poslužitelj i baza podataka.

Web poslužitelj je temeljna komponenta za rad web aplikacije. Omogućuje komunikaciju klijenta (web preglednika) s web aplikacijom. Komunikacija je ostvarena već spomenutim HTTP (engl. *Hyper Text Transfer Protocol*) protokolom.

Web preglednik je posrednik u komunikaciji između korisnika i web poslužitelja. Omogućuje korisniku prikaz web stranice sa svim njenim elementima.

Baza podataka je skup međusobno povezanih podataka koji su tako organizirani da omoguće lakši pristup, pohranu i izmjenu podataka. Web poslužitelj komunicira s bazom podataka tako što šalje zahtjeve za podacima i dobiva podatke ukoliko postoje u bazi podataka.

Frontend je ostvaren u programskom jeziku JavaScript u radnom okviru React, backend u programskom jeziku PHP u radnom okviru Symfony, a baza podataka pomoću Heroku PSQl baze. Razvojna okruženja koja koristimo su Visual Studio Code i IntelliJ IDEA.

Oblikovanju arhitekture pristupili smo oblikovanjem od vrha prema dnu (engl. *Top-down design*). Krenuli smo od osnovnih koncepata i funkcionalnosti web aplikacije do krajnjih detalja izvedbe.

4.1 Baza podataka

dio 1. revizije

Baza podataka je skup međusobno povezanih podataka koji su pohranjeni i precizno organizirani kako bi omogućili lakše upravljanje podacima (pohrana, dohvat i izmjena podataka). Ona predstavlja sliku stvarnog organizacijskog sustava te

strukturu olakšava modeliranje stvarnog svijeta. Relacija (tablica) je objekt baze podataka. Definiramo ju kao imenovanu dvodimenzionalnu tablicu čije imenovane stupce nazivamo atributi, a retke nazivamo n-torkama. U ovoj web aplikaciji izrađena je relacijska baza podataka pomoću Heroku PSQl baze i sastoji se od sljedećih entiteta:

- User
- Staff
- Patient
- Appointment

4.1.1 Opis tablica

User Entitet sadrži osnovne informacije o korisniku web aplikacije. Attribute koje koristi su: id, name, surname, password, phone_number, email i roles. Entitet je u vezi *One-to-Zero-or-One(1:0..1)* s entitetom Staff pomoću atributa id te je u vezi *One-to-Zero-or-One(1:0..1)* s entitetom Patient pomoću atributa id.

user		
id	INT	Jedinstveni identifikator korisnika
name	VARCHAR	Ime korisnika
surname	VARCHAR	Prezime korisnika
password	VARCHAR	Lozinka korisnika
phone_number	VARCHAR	Broj telefona korisnika
email	VARCHAR	Adresa elektroničke pošte
roles	JSON	Uloga/razina ovlasti korisnika

Staff Entitet sadrži informacije o medicinskim djelatnicima zdravstvene ustanove. Attribute koje koristi su: id, user_id_id, qualifications, specialization i availability. Entitet je u vezi *Zero-or-One-to-One(0..1:1)* s entitetom User pomoću atributa user_id_id te je u vezi *One-or-Many-to-One-or-Many(1..N:1..N)* s entitetom Patient pomoću atributa id.

staff		
id	INT	Jedinstveni identifikator djelatnika
user_id_id	INT	Jedinstveni identifikator korisnika
qualifications	VARCHAR	Stručna sprema djelatnika
specialization	VARCHAR	Specijalizacija djelatnika
availability	BOOLEAN	Dostupnost/nedostupnost djelatnika

Patient Entitet sadrži informacije o bolesnicima, tj. korisnicima usluga zdravstvene ustanove. Attribute koje koristi su: id, user_id_id, doctor_id_id, diagnosis, insurance_id, is_insured i services. Entitet je u vezi *Zero-or-One-to-One*(0..1:1) s entitetom User pomoću atributa user_id_id, u vezi je *One-or-Many-to-One-or-Many*(1..N:1..N) s entitetom Staff pomoću atributa doctor_id_id te je u vezi *One-to-One-or-Many*(1:1..N) s entitetom Appointment pomoću atributa id.

patient		
id	INT	Jedinstveni identifikator pacijenta
user_id_id	INT	Jedinstveni identifikator korisnika
doctor_id_id	INT	Jedinstveni identifikator medicinskog djelatnika
diagnosis	VARCHAR	Dijagnoza (povijest bolesti) pacijenta
insurance_id	VARCHAR	MBO (matični broj osiguranika)
is_insured	BOOLEAN	Posjeduje li pacijent dodatno osiguranje
services	JSON	Tretmani na koje je pacijent upućen

Appointment Entitet sadrži informacije o terminu tretmana za pacijenta. Attribute koje koristi su: id, patient_id_id, service, date i machine. Entitet je u vezi *One-or-Many-to-One*(1..N:1) s entitetom Patient pomoću atributa patient_id_id.

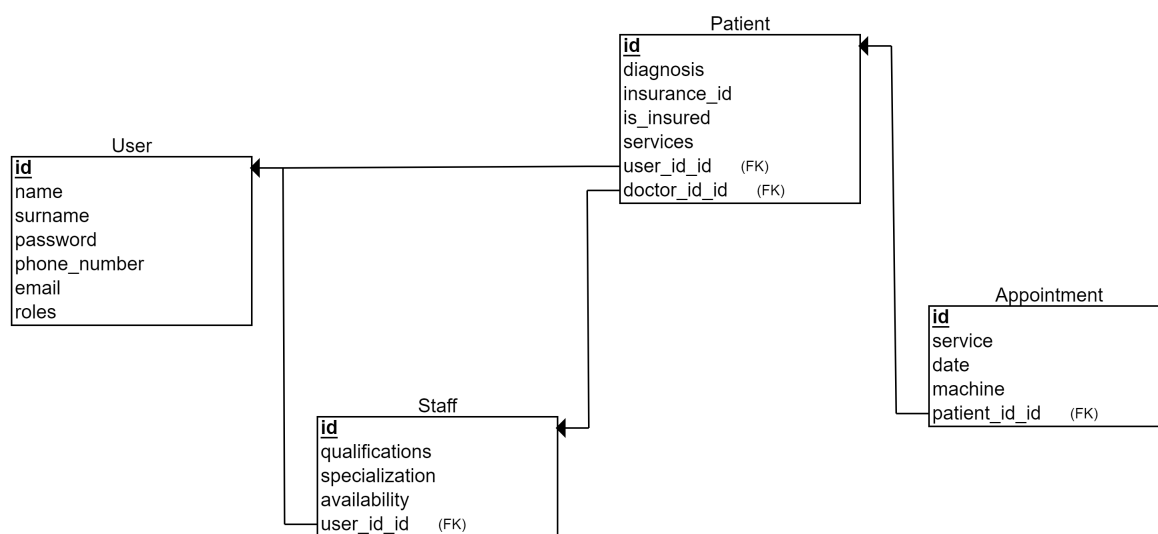
appointment		
id	INT	Jedinstveni identifikator termina tretmana
patient_id_id	INT	Jedinstveni identifikator pacijenta

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

appointment		
service	VARCHAR	Usluge/tretmani pacijenta
date	TIMESTAMP	Datum i vrijeme termina tretmana
machine	VARCHAR	Uređaj potreban za tretman

4.1.2 Dijagram baze podataka

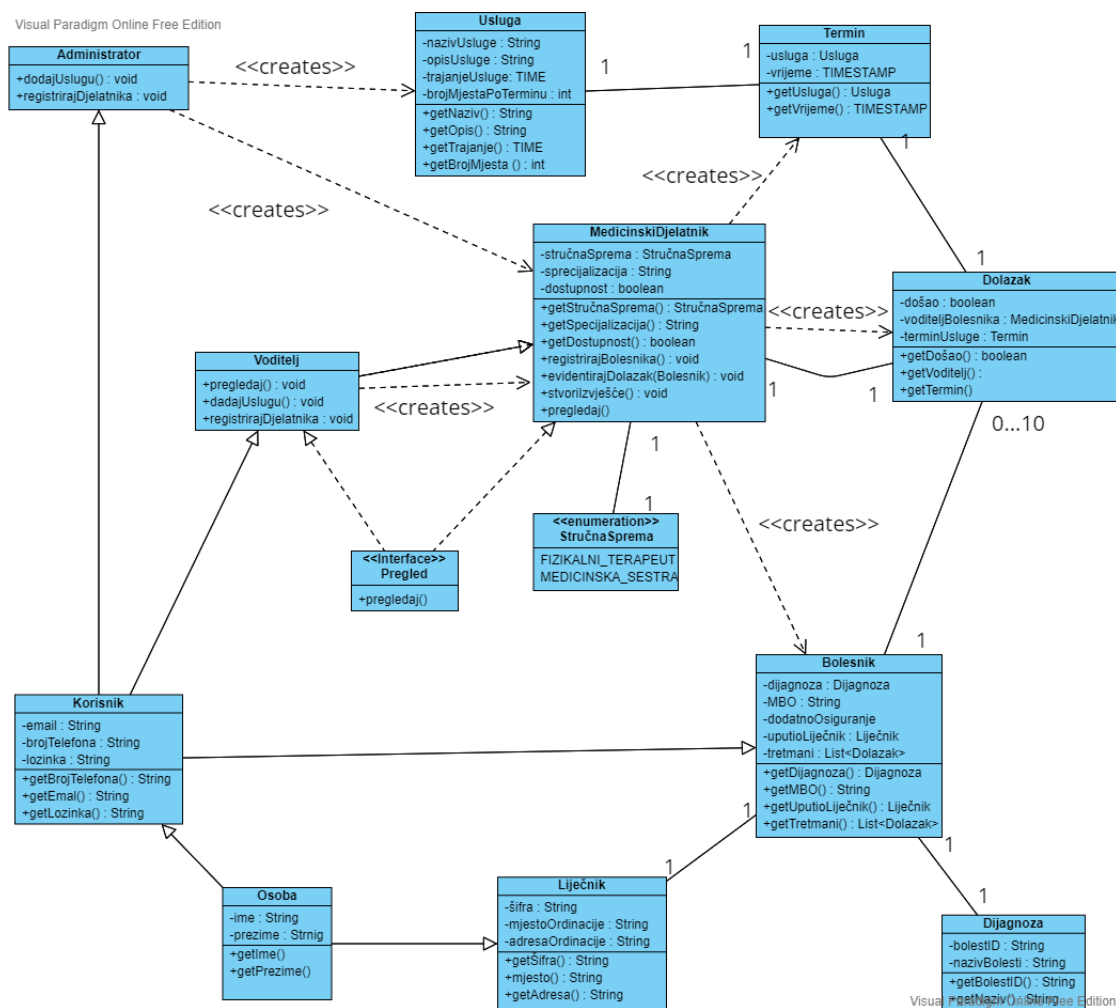


Slika 4.2: E-R dijagram baze podataka

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



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

3.1	UML obrazaca korištenja UC1 do UC3	17
3.2	UML obrazaca korištenja UC4 do UC6 te UC9	18
3.3	UML obrazaca korištenja UC7 do UC8 te od UC10 do UC12	19
3.4	Sekvencijski dijagram UC5	20
3.5	Sekvencijski dijagram UC6	21
3.6	Sekvencijski dijagram UC8	22
4.1	Arhitektura sustava	24
4.2	E-R dijagram baze podataka	28
4.3	Dijagram razreda	29

Dodatak: Prikaz aktivnosti grupe

Dnevnik sastajanja

Kontinuirano osvježavanje

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

1. sastanak

- Datum: u ovom formatu: 20. listopada 2022.
- Prisustvovali: D.Đurinić, L.Kurtin, P.Pašić, F.Hruza, M.Bakač, F.Ogrinšak
- Teme sastanka:
 - Upoznavanje članova tima
 - Raspodjela u članova u timove prema vrsti posla (frontend, backend, dokumentacija)
 - Odabir tehnologija i alata koji će se koristiti na projektu

2. sastanak

- Datum: u ovom formatu: 10. studenoga 2022.
- Prisustvovali: L.Kurtin, P.Pašić, F.Hruza, M.Bakač, F.Ogrinšak
- Teme sastanka:
 - Razrađeni daljnji planovi izrade dokumentacije
 - Raspodjela zadataka za izradu frontend dijela aplikacije
 - Inicijalizirana grana na GitLab-u za frontend

3. sastanak

- Datum: u ovom formatu: 17. studenoga 2022.
- Prisustvovali: D.Đurinić, L.Kurtin, P.Pašić, F.Hruza, F.Ogrinšak, M.J.Kocijan
- Teme sastanka:
 - Pregled napravljenog posla
 - Diskusija daljnjeg razvoja aplikacije
 - Diskusija potrebnih preinaka dokumentacije kako bi bila spremna za rok predaje 1.revizije

Tablica aktivnosti

Kontinuirano osvježavanje

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

	Ime Prezime voditelja	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime
Upravljanje projektom							
Opis projektnog zadatka							
Funkcionalni zahtjevi							
Opis pojedinih obrazaca							
Dijagram obrazaca							
Sekvencijski dijagrami							
Opis ostalih zahtjeva							
Arhitektura i dizajn sustava							
Baza podataka							
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

	Ime Prezime voditelja	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime
Upute za puštanje u pogon							
Dnevnik sastajanja							
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.