

Programsko inženjerstvo

Ak. god. 2023./2024.

Darivanje krvi

Dokumentacija, Rev. 1

Grupa: *WeAreAdults*

Voditelj: *Iva Rengel*

Datum predaje: *<dan>. <mjesec>. <godina>.*

Nastavnik: *Vlado Sruk*

Sadržaj

1	Dnevnik promjena dokumentacije	3
2	Opis projektnog zadatka	5
2.1	Primjeri u \LaTeX u	9
3	Specifikacija programske potpore	12
3.1	Funkcionalni zahtjevi	12
3.1.1	Obrasci uporabe	14
3.1.2	Sekvencijski dijagrami	20
3.2	Ostali zahtjevi	21
4	Arhitektura i dizajn sustava	22
4.1	Baza podataka	24
4.1.1	Opis tablica	24
4.1.2	Dijagram baze podataka	25
4.2	Dijagram razreda	26
4.3	Dijagram stanja	27
4.4	Dijagram aktivnosti	28
4.5	Dijagram komponenti	29
5	Implementacija i korisničko sučelje	30
5.1	Korištene tehnologije i alati	30
5.2	Ispitivanje programskog rješenja	31
5.2.1	Ispitivanje komponenti	31
5.2.2	Ispitivanje sustava	31
5.3	Dijagram razmještaja	32
5.4	Upute za puštanje u pogon	33
6	Zaključak i budući rad	34
	Popis literature	35

Indeks slika i dijagrama 36

Dodatak: Prikaz aktivnosti grupe 37

1. Dnevnik promjena dokumentacije

Kontinuirano osvježavanje

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napravljen predložak. Dodani funkcionalni zahtjevi	Josipa Sever	30.10.2023.
0.2	Dopisane upute za povijest dokumentacije. Dodane reference.	*	24.08.2013.
0.5	Dodan <i>Use Case</i> dijagram i jedan sekvencijski dijagram, funkcionalni i nefunkcionalni zahtjevi i dodatak A	*	25.08.2013.
0.6	Arhitektura i dizajn sustava, algoritmi i strukture podataka	*	26.08.2013.
0.8	Povijest rada i trenutni status implementacije, Zaključci i plan daljnjeg rada	*	28.08.2013.
0.9	Opisi obrazaca uporabe	*	07.09.2013.
0.10	Preveden uvod	*	08.09.2013.
0.11	Sekvencijski dijagrami	*	09.09.2013.
0.12.1	Započeo dijagrame razreda	*	10.09.2013.
0.12.2	Nastavak dijagrama razreda	*	11.09.2013.
1.0	Verzija samo s bitnim dijelovima za 1. ciklus	*	11.09.2013.
1.1	Uređivanje teksta – funkcionalni i nefunkcionalni zahtjevi	* *	14.09.2013.

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

Rev.	Opis promjene/dodatka	Autori	Datum
1.2	Manje izmjene:Timer - Brojilo vremena	*	15.09.2013.
1.3	Popravljeni dijagrami obrazaca uporabe	*	15.09.2013.
1.5	Generalna revizija strukture dokumenta	*	19.09.2013.
1.5.1	Manja revizija (dijagram razmještaja)	*	20.09.2013.
2.0	Konačni tekst predloška dokumentacije	*	28.09.2013.

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

dio 1. revizije

Na osnovi projektnog zadatka detaljno opisati korisničke zahtjeve. Što jasnije opisati cilj projektnog zadatka, razraditi problematiku zadatka, dodati nove aspekte problema i potencijalnih rješenja. Očekuje se minimalno 3, a poželjno 4-5 stranica opisa. Teme koje treba dodatno razraditi u ovom poglavlju su:

- *potencijalna korist ovog projekta*
- *postojeća slična rješenja (istražiti i ukratko opisati razlike u odnosu na zadani zadatak). Dodajte slike koja predočavaju slična rješenja.*
- *skup korisnika koji bi mogao biti zainteresiran za ostvareno rješenje.*
- *mogućnost prilagodbe rješenja*
- *opseg projektnog zadatka*
- *moгуće nadogradnje projektnog zadatka*

Za pomoć pogledati reference navedene u poglavlju „Popis literature“, a po potrebi konzultirati sadržaj na internetu koji nudi dobre smjernice u tom pogledu.

Donori Krv

Naš zadatak svodi se na izradu programske potpore za web aplikaciju (trenutno nema ime) koja će služiti tome da svim donorima i zaposlenicima Hrvatskog Crvenog Križa olakša neke akcije vezane uz darivanje krvi.

Ukratko, koristeći našu aplikaciju, korisnici zainteresirani za davanje krvi moći će se prijaviti u sustav, pregledavati sve aktivne lokacije davanja krvi, prijavljivati se na njih, rezervirati termine i pregledavati neke osnovne informacije o aplikaciji i Hrvatskom Crvenom Križu te o podacima svog profila.

Zaposlenici Hrvatskog Crvenog Križa moći će pregledavati koliko ljudi će prisustvovati određenoj akciji i moći će upisivati sva darivanja koja će se prikazivati na našoj aplikaciji. Moći će organizirati i uređivati povremene akcije darivanja krvi koje se mogu održati na bilo kojoj lokaciji, i najvažnije, hitnom porukom moći će obavijestiti sve korisnike o nedostatku krvi na određenom području.

Cilj ove aplikacije je modernizirati spremanje i prikaz podataka vezanih uz darivanje krvi, automatizirati i olakšati proces naručivanja, pa tako i skratiti vrijeme čekanja u redu, automatizirati proces izdaje potvrda nakon određenog broja darivanja i pružiti korisnicima informacije vezane uz moguća mjesta darivanja krvi pa tako i povećati volumen darovane krvi.

Lokacije na kojima je moguće dati krv sastoje se od dva tipa lokacija zasnovanih na stalnosti termina darivanja krvi:

1. Zdravstvene ustanove sa stalnim terminima
2. Lokacije privremenih akcija koje organizira Hrvatski Crveni Križ

U prvu kategoriju spadaju KBC Osijek, KBC Rijeka, KBC Split, OB Dubrovnik, OB Varaždin, OB Zadar i Hrvatski zavod za transfuzijsku medicinu Zagreb.

Korisnici koji će koristiti našu aplikaciju dijele se u dvije kategorije: nerigistrirani korisnik (gost) i registrirani korisnik. Nadalje, korisnik može registrirati dvije vrste računa: Donor krvi i Zaposlenik Hrvatskog Crvenog Križa (administrator, admin). Potreba za ove modele korištenja aplikacije proizlazi iz toga što dio aplikacije treba biti vidljiv i još neprijavljenim korisnicima te toga što prosječan donor krvi i zaposlenik Hrvatskog Crvenog Križa koriste aplikaciju na znatno različite načine.

- **Gost** - način korištenja aplikacije kada korisnik nije prijavljen u sustav. U ovom način moguće je prijaviti se ili registrirati novi korisnički račun. Osim toga, moguć je pregled svih lokacija vađenja krvi, i onih stalnih i trenutno aktivnih povremenih akcija, ali bez mogućnosti naručivanja termina. Također, gostu je dostupna stranica s osnovnim informacijama o aplikaciji i Hrvatskom Crvenom Križu.
- **Donor krvi** - kad se gost prijavi u sustav, jedna od mogućih opcija je napraviti račun kao donor krvi. U ovom načinu korištenja aplikacije korisnik može napraviti sve što i gost, s još dodatnim funkcionalnostima. Kod pregleda lokacija, prikazuje mu se gumb koji ga vodi na odabir i rezervaciju termina. Dostupna mu je i mogućnost pregleda vlastitog profila s brojnim informacijama koje ćemo detaljno navesti kasnije. Također, kad admin pošalje obavijest za hitnom akcijom, korisnik dobiva poruku na mail. Kad klikne na link u mailu ili pristupi aplikaciji nakon poziva, preusmjerit će se na stranicu na kojoj ima mogućnost rezervirati termin za hitnu akciju.
- **Administrator** - dalje u tekstu nazvan i Hrvatski Crveni Križ ili samo Crveni Križ. Također ima sve opcije iste kao i gost s još nekim dodatnim mogućnostima.

Admin verificira korisnika koji se registrira, nakon čega se korisnik može prijaviti u sustav. Također, dodaje, mijenja i arhivira akcije darivanja krvi. Može napraviti zapis svaki put kad netko daruje krv u svrhu praćenja povijesti darivanja krvi svakog korisnika. Može vidjeti koliko ljudi je rezerviralo termin za neku akciju.

Naša web aplikacija zamišljena je kao skupina raznih stranica koje će nam pomoći kod organizacije svih funkcionalnosti koje aplikacija treba imati. Registracija korisnika od gosta traži sljedeće informacije:

- Ime i prezime
- Spol
- Adresa
- Primarna zdravstvena ustanova
- Email adresa
- Datum rođenja
- MBO
- Krvna grupa

Jednom kad se osoba odluči registrirati, svi upisani podaci šalju se adminu na verifikaciju. Nakon što admin pregleda podatke i zaključi da izgledaju ispravno, potvrđuje podatke i njegov račun postaje aktivan. Korisnik dobiva mail da ima mogućnost prijaviti se u aplikaciju.

Kad se korisnik prijavi, prvo što vidi na stranici je prikaz svojeg profila. Na njemu su vidljive sve njegove osobne informacije, od kojih neke može i uređivati. Vidljive su mu sve aktivne rezervacije koje onda može otkazati ako je potrebno. Sljedeće što je vidljivo prijavljenom korisniku jest povijest svih termina kad je darovao krv i admin je to upisao u aplikaciju. I na kraju, vidljiva mu je lista svih potvrda koje može "osvojiti" ako dovoljan broj puta da krv.

Kretanjem kroz meni koji se nalazi na svakoj stranici aplikacije, korisnik se može prebaciti na stranicu za pregledavanje lokacija na kojima ima mogućnost prijaviti termin za davanje krvi. Sve lokacije na stranici također su vidljive na karti. Odabirom lokacije korisnik je preusmjeren na biranje termina i klikom na gumb potvrđuje rezervaciju.

Tijekom registracije gost ima mogućnost registrirati se kao zaposlenik Hrvatskog Crvenog Križa. Nakon što je verificiran, prijavom u sustav ovaj korisnik postaje admin. Može verificirati ostale goste koji se žele registrirati. Može stvoriti

novu akciju unoseći lokaciju, termine i trajanje akcije. Ako je nešto slučajno krivo unio, ima mogućnost urediti akciju.

Akcije koje admin stvara imaju dva stanja:

- aktivne - akcija je stvorena i još nije istekla. Vidljiva je svim korisnicima i admin ju može uređivati ako je potrebno.
- arhivirane - akcija je "istekla" i više nije vidljiva korisnicima. Admin ju više ne može uređivati, ali ju može pregledavati.

Sve akcije počinju kao aktivne. Kad prođe zadnji dan akcije, ona se automatski arhivira. Ako iz nekog razloga admin treba zatvoriti akciju prije dana isteka, ponuđena mu je mogućnost to napraviti.

Admin također u slučaju nestanka zaliha neke krvne grupe može poslati korisnicima obavijest za hitnu akciju darivanja krvi. Upisom krvne grupe i lokacije u odgovarajuće prozore prvo šalje svim korisnicima u krugu od 20km poruku na mail i modal obavijest u aplikaciji. Korisnici imaju mogućnost potvrditi svoj dolazak i rezervirati termin za navedenu hitnu akciju. Ako se u određenom vremenskom roku ne prijavi dovoljno ljudi, aplikacija će poslati poruku svim ostalim korisnicima u regiji.

Moguće nadogradnje koje bi daljnje poboljšale ovaj sustav su pregled statistike za zaposlenike Hrvatskog Crvenog Križa. To uključuje izradu posebne stranice gdje bi administratorima bilo prikazano koliko ljudi se prijavilo za koju akciju i u kojim terminima, koji su omjeri krvnih grupa prijavljenih ljudi, koliko ljudi se odazvalo na neku hitnu akciju, koliko krvi je sveukupno sakupljeno kroz različite vremenske periode i slično.

2.1 Primjeri u L^AT_EXu

Ovo potpoglavlje izbrisati.

U nastavku se nalaze različiti primjeri kako koristiti osnovne funkcionalnosti L^AT_EXa koje su potrebne za izradu dokumentacije. Za dodatnu pomoć obratiti se asistentu na projektu ili potražiti upute na sljedećim web sjedištima:

- Upute za izradu diplomskog rada u L^AT_EXu - https://www.fer.unizg.hr/_download/repository/LaTeX-upute.pdf
- L^AT_EX projekt - <https://www.latex-project.org/help/>
- StackExchange za Tex - <https://tex.stackexchange.com/>

podcrtani tekst, **podebljani tekst**, *nagnuti tekst*

primjer primjer primjer primjer primjer primjer

- primjer
- primjer
- primjer
 1. primjer
 - 1.a primjer
 - b primjer
 2. primjer

primjer url-a: <https://www.fer.unizg.hr/predmet/proinz/projekt>

posebni znakovi: # \$ % & { } _ | < > ^ ~ \

naslov unutar tablice		
IDKorisnik	INT	Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod
korisnickoIme	VARCHAR	

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

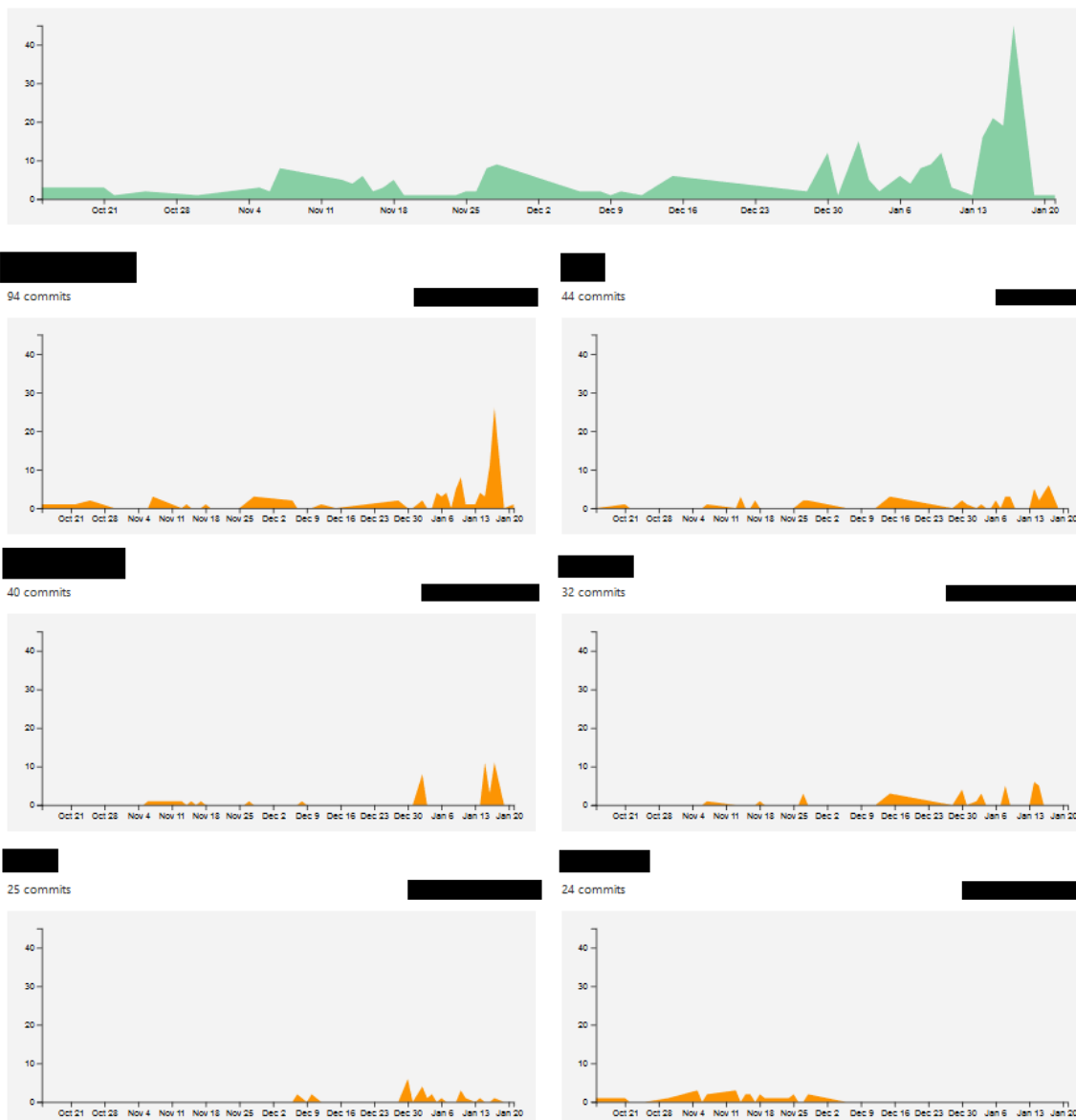
naslov unutar tablice		
email	VARCHAR	
ime	VARCHAR	
primjer	VARCHAR	

Tablica 2.1: Naslov s referencom izvan tablice

IDKorisnik	INT	Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod
korisnickoIme	VARCHAR	
email	VARCHAR	
ime	VARCHAR	
primjer	VARCHAR	



Slika 2.1: Primjer slike s potpisom



Slika 2.2: Primjer slike s potpisom 2

Referenciranje slike 2.2 u tekstu.

3. Specifikacija programske potpore

3.1 Funkcionalni zahtjevi

dio 1. revizije

Navesti **dionike** koji imaju **interes u ovom sustavu** ili **su nositelji odgovornosti**. To su prije svega korisnici, ali i administratori sustava, naručitelji, razvojni tim.

Navesti **aktore** koji izravno **koriste** ili **komuniciraju sa sustavom**. Oni mogu imati inicijatorsku ulogu, tj. započinju određene procese u sustavu ili samo sudioničku ulogu, tj. obavljaju određeni posao. Za svakog aktora navesti funkcionalne zahtjeve koji se na njega odnose.

Dionici:

1. Neregistrirani korisnik (gost)
2. Registrirani korisnik
 - (a) donor krvi
 - (b) zaposlenik Hrvatskog Crvenog Križa (admin)
3. Razvojni tim

Aktori i njihovi funkcionalni zahtjevi:

1. Korisnik(inicijator) može:
 - (a) pregledavati stranicu s općim informacijama
 - (b) pregledavati stranicu s lokacijama na kojima može darivati krv
 - (c) prijaviti se u sustav
 - (d) registrirati se u sustav
2. Registrirani korisnik(inicijator) može:
 - (a) pregledavati vlastiti profil

- i. pregledavati i uređivati osobne informacije
- ii. pregledavati prijavljene rezervacije termina i otkazati ih
- iii. pregledavati povijest darivanja krvi
- iv. pregledavati dobivene potvrde
- (b) rezervirati termin za akciju po izboru
- (c) odjaviti se

3. Zaposlenik Crvenog Križa ili admin(inicijator) može:

- (a) dodati nove akcije
- (b) urediti postojeće akcije
 - i. urediti podatke o akciji
 - ii. prijevremeno arhivirati akciju
- (c) verificirati korisnika
- (d) bilježiti instance davanja krvi
- (e) postaviti hitnu akciju za određenu krvnu grupu

4. Baza podataka(sudionik) može:

- (a) pohranjivati podatke o donorima, rezerviranim terminima i povijesti darivanja krvi
- (b) pohranjivati podatke o akcijama i zdravstvenim ustanovama

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 - Prijava korisnika

- **Glavni sudionik:** Registrirani korisnik
- **Cilj:** Prijaviti se u korisnički račun radi pristupa stranici
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je registriran
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju prijava
 2. Unosi tražene podatke (korisničko ime i lozinku)
 3. Pristup korisničkim funkcijama
- **Opis mogućih odstupanja:**
 - 2.a Pogrešno korisničko ime ili lozinka
 1. Sustav obavještava korisnika o pogrešci i vraća ga na stranicu za prijavu kao neregistriranog korisnika

UC2 - Pregled akcija darivanja krvi

- **Glavni sudionik:** Neregistrirani korisnik
- **Cilj:** Pregledati sve dostupne lokacije i termine darivanja krvi
- **Sudionici:** Baza podataka
- **Opis osnovnog tijeka:**
 1. Korisniku se na karti prikazuju lokacije na kojima je moguće darivati krv (bolnice i akcije Crvenog Križa)
 2. Odabir jedne od prikazanih lokacija
 3. Prikaz adrese i vremena za moguću rezervaciju

UC3 - Pregled informacija

- **Glavni sudionik:** Neregistrirani korisnik

- **Cilj:** Pregled općih informacija o Crvenom Križu i akcijama darivanja krvi
- **Sudionici:** -
- **Opis osnovnog tijeka:**
 1. Odabir "O nama" u navigacijskoj traci
 2. Prikaz informacija o Crvenom Križu, njegovoj djelatnosti i akcijama darivanja krvi koje organiziraju

UC4 - Registracija

- **Glavni sudionik:** Neregistrirani korisnik
- **Cilj:** Registracija novog korisnika
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik nije registriran
- **Opis osnovnog tijeka:**
 1. Neregistrirani korisnik odabire opciju za registraciju
 2. Aplikacija ga preusmjerava na stranicu za registraciju
 3. Ispunjava obrazac
 4. Aplikacija provjerava unesene podatke i stvara korisnički račun te šalje poveznicu za verifikaciju mail adrese
 5. Korisnik aktivira račun verifikacijom mail adrese
- **Opis mogućih odstupanja:**
 - 3.a Korisnik odustaje od registracije
 1. Povratak na početnu stranicu
 - 4.a Neki od unesenih podataka ne zadovoljavaju kriterije obrasca
 1. Sustav javlja grešku
 - 4.b Korisnik pokušava registrirati račun mail adresom koja već ima registrirani račun na web aplikaciji
 1. Sustav javlja grešku

UC5 - Rezervacija termina

- **Glavni sudionik:** Donor krvi
- **Cilj:** Rezervacija termina za darivanje krvi
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav, nalazi se na stranici "Karta" s označenim mogućim lokacijama
- **Opis osnovnog tijeka:**
 1. Korisnik odabire željenu lokaciju

2. Aplikacija izbacuje prozor s dostupnim terminima za rezervaciju
 3. Korisnik odabire termin
 4. Aplikacija provjerava podobnost korisnika i potvrđuje rezervaciju te vraća poruku potvrde
- **Opis mogućih odstupanja:**
 - 2.a Na odabranoj lokaciji nema dostupnih termina
 1. Sustav javlja grešku
 - 4.a Korisnik pokušava rezervirati termin u kojem zbog zdravstvenih razloga ne može darivati krv
 1. Sustav javlja grešku

UC6 - Pregled profila

- **Glavni sudionik:** Registrirani korisnik
- **Cilj:** Pregled vlastitog profila i informacija
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik odabire karticu "Moj profil"
 2. Aplikacija ga preusmjerava na stranicu profila i prikazuje korisničke podatke

UC7 - Odjava korisnika

- **Glavni sudionik:** Registrirani korisnik
- **Cilj:** Odjava iz aplikacije nakon završetka rada
- **Preduvjet:** Korisnik je prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Odabir opcije "Odjavi se"
 2. Povratak na početni zaslon kao gost

UC8 - Otkazivanje termina

- **Glavni sudionik:** Donor krvi
- **Cilj:** Otkazivanje rezerviranog termina
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav i ima rezerviran termin
- **Opis osnovnog tijeka:**
 1. Korisnik otvara profilnu stranicu

2. Pregledava svoje nadolazeće rezervacije
 3. Odabire termin koji želi otkazati i opciju "Otkazi termin"
 4. Od korisnika se traži potvrda
 5. Korisnik potvrđuje akciju
 6. Termin se briše iz baze podataka i uklanja s donorovog popisa rezervacija
- **Opis mogućih odstupanja:**
 - 5.a Korisnik odustaje od akcije
 1. Rezervacija se ne briše i korisnik se vraća na profilnu stranicu

UC9 - Uređivanje osobnih podataka

- **Glavni sudionik:** Registrirani korisnik
- **Cilj:** Promjena netočnih osobnih podataka
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik otvara profilnu stranicu
 2. Odabir opcije "Uređivanje podataka"
 3. Promjena podataka u otvorenom prozoru za uređivanje
 4. Spremanje promjena i ažuriranje baze podataka
- **Opis mogućih odstupanja:**
 - 4.a Korisnik odustaje od promjena
 1. Promjene se ne spremaju i korisnik se vraća na profilnu stranicu
 - 4.b Korisnik ne sprema promjene
 1. Sustav obavještava korisnika da promjene nisu spremljene
 2. Korisnik sprema promjene i vraća se na profilnu stranicu

UC10 - Dodavanje akcija

- **Glavni sudionik:** Administrator
- **Cilj:** Dodavanje akcije Crvenog Križa za darivanje krvi
- **Sudionici:** Baza podataka
- **Preduvjet:** Administrator je prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Administrator odredi lokacije i vrijeme održavanja akcije
 2. Potvrđuje svoje odabir
- **Opis mogućih odstupanja:**

2.a Administrator nije potvrdio odabir

1. Sustav upozorava da promjene nisu spremljene

UC11 - Uređivanje akcija

- **Glavni sudionik:** Administrator
- **Cilj:** Uređivanje podataka aktivne akcije
- **Sudionici:** Baza podataka
- **Preduvjet:** Administrator je prijavljen, akcija je aktivna
- **Opis osnovnog tijeka:**
 1. Administrator odabire akciju koju želi urediti
 2. Uređuje željene podatke
 3. Potvrđuje promjene
- **Opis mogućih odstupanja:**
 - 3.a Administrator nije potvrdio promjene
 1. Sustav upozorava da promjene nisu spremljene

UC12 - Arhiviranje akcija

- **Glavni sudionik:** Administrator
- **Cilj:** Arhivirati aktivnu akciju
- **Sudionici:** Baza podataka
- **Preduvjet:** Administrator je prijavljen, akcija je aktivna
- **Opis osnovnog tijeka:**
 1. Administrator odabire akciju koju želi arhivirati
 2. Potvrđuje odabir
- **Opis mogućih odstupanja:**
 - 2.a Administrator nije potvrdio odabir
 1. Sustav upozorava da promjene nisu spremljene

UC13 - Verifikacija kritičnih podataka

- **Glavni sudionik:** Korisnik, Administrator
- **Cilj:** Verificirati podatke novog korisnika pri registraciji
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik se registrirao, administrator je prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Administrator pristupa popisu registriranih korisnika

2. Za neverificirane korisnike sustav provjerava ima li osoba barem 18 godina, postoji li njezin MBO u bazi i, ako postoji, odgovara li ime i prezime toj osobi
 3. Pri uspješnoj provjeri omogućava se gumb "Verificiraj"
 4. Administrator klikom na gumb verificira korisnika
- **Opis mogućih odstupanja:**
 - 4.a Korisnik ne zadovoljava kriterije
 1. Administrator klikne na gumb "Odbij"
 2. Korisnik se briše s liste neverificiranih korisnika i iz baze podataka

UC14 - Evidentiranje darivanja krvi

- **Glavni sudionik:** Donor krvi, Administrator
- **Cilj:** Dodati darivanje krvi u povijest darivanja i u bazu podataka
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je darovao krv, Administrator je prijavljen
- **Opis osnovnog tijeka:**
 1. Administrator pristupa bazi podataka
 2. Odabire unos novog zapisa
 3. Popunjava obrazac koji traži ime, prezime, MBO, datum i krvnu grupu osobe
 4. Obrazac se validira i podaci se unose u Bazu
- **Opis mogućih odstupanja:**
 - 4.a Neki od traženih podataka nije upisan, unesen je datum iz budućnosti, nije unesena odgovarajuća krvna grupa ili je MBO u pogrešnom formatu
 1. Sustav javlja grešku

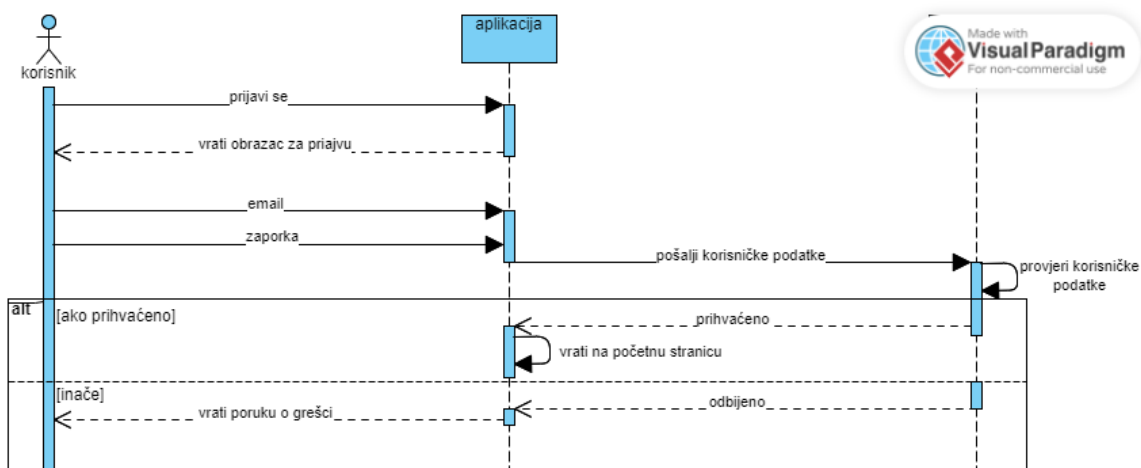
Dijagrami obrazaca uporabe

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

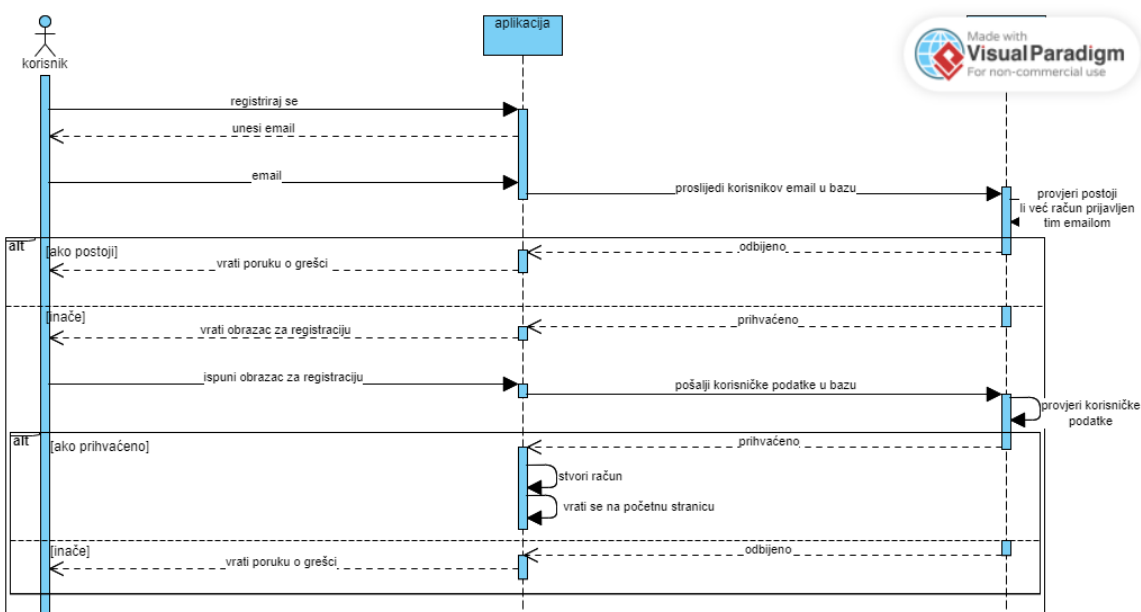
3.1.2 Sekvencijski dijagrami

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.



Slika 3.1: Sekvencijski dijagram - prijava



Slika 3.2: Sekvencijski dijagram - registracija

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.*

- implementacija jednostavnog i intuitivnog korisničkog sučelja
- responzivan dizajn za mogućnost korištenja aplikacije na mobilnim uređajima i računalima
- sigurnost podataka u bazi i u komuniciranju s bazom
- osiguravanje rada više korisnika u stvarnom vremenu
- neispravno korištenje aplikacije ne smije narušiti rad sustava
- dizajn treba biti konzistentan kroz sve stranice aplikacije

4. Arhitektura i dizajn sustava

dio 1. revizije

Potrebno je opisati stil arhitekture te identificirati: podsustave, preslikavanje na radnu platformu, spremišta podataka, mrežne protokole, globalni upravljački tok i sklopovsko-programске zahtjeve. Po točkama razraditi i popratiti odgovarajućim skicama:

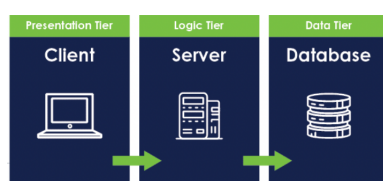
- izbor arhitekture temeljem principa oblikovanja pokazanih na predavanjima (objasniti zašto ste baš odabrali takvu arhitekturu)
- organizaciju sustava s najviše razine apstrakcije (npr. klijent-poslužitelj, baza podataka, datotečni sustav, grafičko sučelje)
- organizaciju aplikacije (npr. slojevi frontend i backend, MVC arhitektura)
- **Proces odabira arhitekture**

U našem je zadatku cilj web aplikacije olakšati darivateljima krvi i Crvenom Križu cijeli proces darivanja krvi: od organiziranja akcija do pronalaženja istih i rezerviranja termina. Specifikacijom dionika i definiranjem funkcionalnih zahtjeva zaključili smo kako naša aplikacija mora imati tri razine: razinu baze podataka, razinu web-aplikacije i razinu klijenta. Tako navedenim razinama korespondiraju tri sloja: sloj pristupa podacima, sloj aplikacijske logike i sloj korisničkog sučelja. Razmatrajući potrebna svojstva, poput smanjivanja nepotrebnih međuovisnosti između modula radi jednostavnosti pri budućim promjenama, hijerarhijske organizacije razina koja omogućava pristup viših razina nižima preko programskog sučelja i grupiranje dijelova programa koji pristupaju i mijenjaju iste podatke te procedura koje se izvode slijedno radi lakšeg snalaženja, odabrali smo višeslojnu arhitekturu temeljenu na klijent-poslužitelj odnosu, najbližnju MVC stilu arhitekture.

- **Višeslojna arhitektura i MVC stil**

Odnos klijent-poslužitelj opisuje međusobni odnos programa koji sudjeluju u aplikaciji, to jest njihovo funkcioniranje prije, tijekom i po završetku komunikacije klijenta i poslužitelja. Aplikacija sadrži tri osnovna sloja:

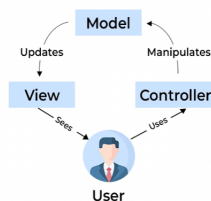
1. Klijentsku komponentu - sloj s korisničkim sučeljem, predstavljena web preglednikom, služi kao sučelje između klijenta i poslužitelja, omogućava klijentu slanje HTTP zahtjeva i primanje HTTP odgovora, prevoditelj koji web stranicu pisanu u kodu prikazuje u klijentu razumljivom obliku
2. Poslužiteljsku komponentu - sloj aplikacijske logike, predstavlja web poslužitelj, centar za razmjenu informacija i pružanje usluga, služi za pohranu, obradu i dostavu web stranica klijentu, pokreće web aplikaciju i prosljeđuje joj klijentove zahtjeve, po potrebi pristupa bazi podataka i vraća klijentu odgovor vidljiv u web pregledniku
3. Baza podataka - sloj za pohranu podataka



Slika 4.1: Troslojna arhitektura

Prednost klijent-server arhitekture je mogućnost odvojenog oblikovanja slojeva, što također omogućava “separation of concerns”, to jest svaki sloj brine samo za svoju zadaću i ne brine o funkcionalnostima drugih slojeva, a njihova se međuovisnost ostvaruje komunikacijom pomoću sučelja. Ovako razrađenoj arhitekturi, najbliži je Model-View-Controller stil. Osnovna je karakteristika ovog stila razdvajanje briga nezavisnim razvojem pojedinih dijelova aplikacije, što omogućuje jednostavnije i učinkovitije testiranje i izmjene dijelova sustava radi njihove dorade te skalabilnost i fleksibilnost cijele aplikacije razvijene na ovakvom principu. Korisničko sučelje je odvojeno od ostatka sustava, a povezanost elemenata se ostvaruje kroz tri sloja: sloja “View” na klijentskoj strani, te slojeva “Model” i “Controller” na poslužiteljskoj strani.

1. *Model (model)* - predstavlja podatke i logiku vezanu za njih, služi za oblikovanje podataka i prilagodbu komunikaciji s bazom
2. *View (pogled)* - komunikacija s korisnikom i prikaz podataka koje je dobio od upravitelja
3. *Controller (upravitelj)* - povezuje modele i poglede, na temelju primljenih korisničkih zahtjeva i podataka izvršava potrebne funkcije i formira odgovore



Slika 4.2: MVC stil arhitekture

Web aplikacija je podijeljena na front-end i back-end. Front-end čini prezentacijski dio aplikacije, to jest ono što korisnik vidi u web pregledniku. Za izradu aplikacije na klijentskoj strani odabrali smo radni okvir React, koji je zasnovan na JavaScriptu i pruža učinkovit razvoj korisničkog sučelja koristeći već gotove web komponente. Back-end čini dio sustava u kojem se obrađuju korisnički zahtjevi i vrše potrebne akcije, to jest poslužiteljska strana klijent-server arhitekture. Za njegovu smo izradu odabrali programski jezik Python i radni okvir FastAPI. I za razvoj klijentske i za razvoj poslužiteljske strane odabrali smo Visual Studio Code kao radno okruženje.

4.1 Baza podataka

dio 1. revizije

Potrebno je opisati koju vrstu i implementaciju baze podataka ste odabrali, glavne komponente od kojih se sastoji i slično.

4.1.1 Opis tablica

Svaku tablicu je potrebno opisati po zadanom predlošku. Lijevo se nalazi točno ime varijable u bazi podataka, u sredini se nalazi tip podataka, a desno se nalazi opis varijable. Svjetlozelenom bojom označite primarni ključ. Svjetlo plavom označite strani ključ

korisnik - ime tablice		
IDKorisnik	INT	Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

korisnik - ime tablice		
korisnickoIme	VARCHAR	
email	VARCHAR	
ime	VARCHAR	
primjer	VARCHAR	

4.1.2 Dijagram baze podataka

U ovom potpoglavlju potrebno je umetnuti dijagram baze podataka. Primarni i strani ključevi moraju biti označeni, a tablice povezane. Bazu podataka je potrebno normalizirati. Podsjetite se kolegija "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 slike s potpisom	10
2.2	Primjer slike s potpisom 2	11
3.1	Sekvencijski dijagram - prijava	20
3.2	Sekvencijski dijagram - registracija	20
4.1	Troslojna arhitektura	23
4.2	MVC stil arhitekture	24

Dodatak: Prikaz aktivnosti grupe

Dnevnik sastajanja

Kontinuirano osvježavanje

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

1. sastanak

- Datum: 21. listopada 2023.
- Prisustvovali: I. Rengel, T. Lovreković, R. Klečar, N. Ivandić, M. Hrnčić, J. Sever
- Teme sastanka:
 - inicijalna rasprava o implementaciji projekta
 - podjela poslova
 - dogovor oko korištenih tehnologija

2. sastanak

- Datum: 24. listopada 2023.
- Prisustvovali: I. Rengel, T. Lovreković, R. Klečar, N. Ivandić, M. Hrnčić, J. Sever, Z. Cvitanović
- Teme sastanka:
 - opis korisnika i njihovih funkcija
 - dogovor oko potrebnih UML dijagrama obrasca
 - funkcionalni zahtjevi
 - razrada stranice profila korisnika
 - podjela poslova

3. sastanak

- Datum: 26. listopada 2023.
- Prisustvovali: I. Rengel, T. Lovreković, R. Klečar, N. Ivandić, M. Hrnčić, J. Sever, Z. Cvitanović
- Teme sastanka:
 - poboljšavanje raspodjele korisnika

- baza podataka
- koji su nam sekvencijski dijagrami potrebni
- raspodjela poslova

4. sastanak

- Datum: 31. listopada 2023.
- Prisustvovali: I. Rengel, T. Lovreković, N. Ivandić, J. Sever, M. Hrnčić
- Teme sastanka:
 - rješavanje problema gita
 - rasprava o profesorovim komentarima
 - raspodjela poslova

5. sastanak

- Datum: u ovom formatu: 9. studenoga 2023.
- Prisustvovali: I. Rengel, T. Lovreković, R. Klečar, N. Ivandić, M. Hrnčić, J. Sever, Z. Cvitanović
- Teme sastanka:
 - opis prve teme
 - opis druge teme

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							

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

	Ime Prezime voditelja	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime
Dijagram razmještaja							
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.