

# Programsko inženjerstvo

Ak. god. 2021./2022.

## TrueBlood

Dokumentacija, Rev. 1

Grupa: *MEGATRON*

Voditelj: *Toni Ivanković*

Datum predaje: *<dan>*. 11. 2021.

Nastavnik: *Ivan Lovrić*

# Sadržaj

<b>1 Dnevnik promjena dokumentacije</b>	<b>2</b>
<b>2 Opis projektnog zadatka</b>	<b>3</b>
2.1 Primjeri u $\text{\LaTeX}$ u . . . . .	6
<b>3 Specifikacija programske potpore</b>	<b>9</b>
3.1 Funkcionalni zahtjevi . . . . .	9
3.1.1 Obrasci uporabe . . . . .	12
3.1.2 Sekvencijski dijagrami . . . . .	13
3.2 Ostali zahtjevi . . . . .	14
<b>4 Arhitektura i dizajn sustava</b>	<b>15</b>
4.1 Baza podataka . . . . .	16
4.1.1 Opis tablica . . . . .	16
4.1.2 Dijagram baze podataka . . . . .	16
4.2 Dijagram razreda . . . . .	17
<b>Popis literature</b>	<b>18</b>
<b>Indeks slika i dijagrama</b>	<b>19</b>
<b>Dodatak: Prikaz aktivnosti grupe</b>	<b>20</b>

# 1. Dnevnik promjena dokumentacije

## *Kontinuirano osvježavanje*

Table 1.1:

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napravljen predložak.	Toni	28.10.2021.
0.2	Započeto pisanje opisa projektnog zadatka	Toni	3.11.2021.

*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 projekta TrueBlood je napraviti programsku potporu te korisničko sučelje za sustav dobrovoljnog darivanja krvi. Web aplikacija nudi mogućnost stvaranja korisničkih računa za darivatelje (donore), evidenciju pokušaja doniranja krvi, evidenciju trenutnih zaliha krvi, i mnoge druge funkcionalnosti koje su opisane u nastavku. Svrha aplikacije je obogatiti iskustvo darivanja te potaknuti ljude da na jednostavniji način pristupe procesu i dobrim djelom pomognu opterećenom zdravstvenom sustavu. Aplikacija velik dio funkcionalnosti radi automatski, pa se time rasterećuje osoblje i ubrzava i poboljšava cjelokupni proces.

Dio funkcionalnosti koje razvijamo u ovom projektu već postoji u sustavu Hrvatskog zavoda za transfuzijsku medicinu (<https://www.hztm.hr>), gdje je također moguće vidjeti trenutne zalihe krvi u odnosu na optimalne razine, a sustav ispisuje i upozorenje o nedostatku pojedinih krvnih grupa. Ipak, sustav HZTM-a ne nudi funkcionalnost opisanu u našem zadatku, gdje se donori mogu sami prijaviti u sustav te imati uvid u svoje prethodne pokušaje darivanja, a ne nudi ni slanje potvrda mailom ni naknadno ispisivanje potvrda. Ipak, sustav HZTM-a poslužio nam je kao dobra baza iz koje smo mogli vidjeti dosadašnju realizaciju bar dijela sustava.



Slika 2.1: Postojeća stranica HZTM-a

Aplikacija se izrađuje u razvojnom okviru *Java Spring Boot* s poslužiteljske strane, a u *Reactu* sa klijentske strane. Navedeni razvojni okviri nude sve funkcionalnosti potrebne za izradu opisanog sustava, što će biti detaljnije opisano u poglavlju ??.

### 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.*
- *moгуćnost prilagodbe rješenja*
- *opseg projektnog zadatka*
- *могуће 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.*

## 2.1 Primjeri u L<sup>A</sup>T<sub>E</sub>Xu

*Ovo potpoglavlje izbrisati.*

U nastavku se nalaze različiti primjeri kako koristiti osnovne funkcionalnosti L<sup>A</sup>T<sub>E</sub>Xa 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<sup>A</sup>T<sub>E</sub>Xu - [https://www.fer.unizg.hr/\\_download/repository/LaTeX-upute.pdf](https://www.fer.unizg.hr/_download/repository/LaTeX-upute.pdf)
- L<sup>A</sup>T<sub>E</sub>X 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: # \$ % & { } \_ | < > ^ ~ \

Table 2.1:

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	

Table 2.2: 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 2.2: Primjer slike s potpisom





Slika 2.3: Primjer slike s potpisom 2

Referenciranje slike 2.3 u tekstu.

## 3. Specifikacija programske potpore

### 3.1 Funkcionalni zahtjevi

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.

#### Primarni dionici:

1. Naručitelj projekta (zavod za javno zdravstvo, Crveni križ...)
2. Razvojni tim

#### Sekundarni dionici:

1. Donori
2. Djelatnici banke krvi
3. Administratori sustava
4. Svi korisnici interneta koji koriste javnu stranicu sustava radi pregleda trenutne razine krvi

#### Aktori i njihovi funkcionalni zahtjevi:

1. Administrator (inicijator) može:
  - (a) Administrirati korisničke račune:
    - i. Dodati račun djelatnika
    - ii. Deaktivirati bilo koji račun u sustavu
    - iii. Definirati optimalne granice razina krvi
2. Djelatnik banke (inicijator) može:
  - (a) Prikupiti podatke o donoru
    - i. Prikupiti matične i kontakt podatke

- ii. Prikupiti zdravstvene podatke (krvna grupa)
  - (b) Stvoriti korisnički račun donora
  - (c) Nadopuniti podatke postojećeg računa donora (uključujući i zdravstvene podatke)
  - (d) Evidentirati pokušaj doniranja
    - i. Zabilježiti trenutne zdravstvene podatke o donoru
    - ii. Zabilježiti uspješnost pokušaja doniranja
  - (e) Evidentirati trajno odbijanje donora (postojanje + razlog)
  - (f) Povećati zalihi krvi u sustavu uspješnim doniranjem
  - (g) Evidentirati slanje krvi u vanjsku instituciju (bolnicu)
3. Djelatnik banke (sudionik) može:
- (a) Primiti obavijest o prekoračenju bilo koje od optimalnih granica zaliha krvi
  - (b) Primiti aktivacijski link radi aktivacije računa
4. Donor (inicijator) može:
- (a) Kreirati svoj korisnički račun prije prvog darivanja
  - (b) Nadopuniti podatke računa (osim zdravstvenih podataka - krvne grupe i trajnog odbijanja)
  - (c) Pregledati povijest darivanja krvi
  - (d) Preuzeti PDF potvrdu o postojećem doniranju krvi
5. Donor (sudionik) može:
- (a) Primiti aktivacijski link radi aktivacije računa
  - (b) Dobiti generirani donorID
  - (c) Primiti upozorenje o prekoračenju donje optimalne granice zaliha krvi (ako donor nije trajno odbijen)
  - (d) Pri spajanju u sustav dobiti informaciju o trenutnim zalihama krvi svoje krvne grupe (ako donor nije trajno odbijen)
  - (e) Primiti obavijest o isteku perioda zabrane darivanja u trajanju od 3 do 4 mjeseca od zadnjeg darivanja
  - (f) Primiti e-mail s PDF potvrdom o darivanju (nakon darivanja i nakon vlastitog iniciranja izdavanja potvrde)
6. Korisnik javnog weba (inicijator) može:

- (a) Kreirati novi korisnički račun donora
- (b) Pregledati trenutno stanje zaliha krvi svake krvne grupe

### 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.*

#### UC<broj obrasca> -<ime obrasca>

- **Glavni sudionik:** <sudionik>
- **Cilj:** <cilj>
- **Sudionici:** <sudionici>
- **Preduvjet:** <preduvjet>
- **Opis osnovnog tijeka:**
  1. <opis korak jedan>
  2. <opis korak dva>
  3. <opis korak tri>
  4. <opis korak četiri>
  5. <opis korak pet>
- **Opis mogućih odstupanja:**
  - 2.a <opis mogućeg scenarija odstupanja u koraku 2>
    1. <opis rješenja mogućeg scenarija korak 1>
    2. <opis rješenja mogućeg scenarija korak 2>
  - 2.b <opis mogućeg scenarija odstupanja u koraku 2>
  - 3.a <opis mogućeg scenarija odstupanja u koraku 3>

#### **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.*

## 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

### *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:*

Sustav...

- *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)*

Arhitektura se može podijeliti na tri podsustava:

- Web poslužitelj
- Web aplikacija
- Baza podataka

Web poslužitelj osnova je rada web aplikacije. Njegova primarna zadaća je komunikacija klijenta s aplikacijom. Komunikacija se odvija preko HTTP (engl. Hyper Text Transfer Protocol) protokola, što je protokol u prijenosu informacija na webu. Poslužitelj je onaj koji pokreće web aplikaciju te joj prosljeđuje zahtjev.

Korisnik koristi web aplikaciju za obrađivanje željenih zahtjeva. Web aplikacija obrađuje zahtjev te ovisno o zahtjevu, pristupa bazi podataka nakon čega preko poslužitelja vraća korisniku odgovor u obliku HTML dokumenta vidljivog u web pregledniku.

Programski jezik koji smo odabrali za izradu naše web aplikacije je Java zajedno sa SpringBoot radnim okvirom te programski jezik Javascript sa React libraryjem. Odabrano razvojno okruženje je IntelliJ. Sustav se temelji na MVC (Model-View-Controller) obrazcu.



MVC se sastoji od tri cjeline:

- **Model** - (Model) podaci i poslovna logika aplikacije
- **Pogled** (View) - prikaz predhodno modeliranih podataka
- **Upravitelj** (Controller) - upravlja zahtjevima korisnika

Funkcionalnosti aplikacije se izvršavaju slanjem upita na endpoint-e. To su

## 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*

Table 4.1:

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

# Popis literature

## *Kontinuirano osvježavanje*

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

1. Programsko inženjerstvo, FER ZEMRIS, <http://www.fer.hr/predmet/proinz>
2. I. Sommerville, "Software engineering", 8th ed, Addison Wesley, 2007.
3. T.C.Lethbridge, R.Langaniere, "Object-Oriented Software Engineering", 2nd ed. McGraw-Hill, 2005.
4. I. Marsic, Software engineering book", Department of Electrical and Computer Engineering, Rutgers University, <http://www.ece.rutgers.edu/~marsic/books/SE>
5. The Unified Modeling Language, <https://www.uml-diagrams.org/>
6. Astah Community, <http://astah.net/editions/uml-new>
7. <https://web.archive.org/web/20120729161926/http://st-www.cs.illinois.edu/users/smarch/sdocs/mvc.html>
8. <https://www.w3.org/TR/wsdl.html>

# Indeks slika i dijagrama

2.1	Postojeća stranica HZTM-a . . . . .	4
2.2	Primjer slike s potpisom . . . . .	7
2.3	Primjer slike s potpisom 2 . . . . .	8

# 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: 11. studenoga 2021.
- Prisustvovali: I.Prezime, I.Prezime
- Teme sastanka:
  - opis prve teme
  - opis druge teme

#### 2. sastanak

- Datum: u ovom formatu: 11. studenoga 2021.
- Prisustvovali: I.Prezime, I.Prezime
- 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.*

Table 4.2:

	Toni Ivanković	Ana Gršković	Borna Mahović	Jakov Matošić	Dora Nevidal	Marko Opačić	Luka Vukota
Upravljanje projektom	5						
Opis projektnog zadatka	1						
Funkcionalni zahtjevi	6						
Opis pojedinih obrazaca	6						
Dijagram obrazaca	7						
Sekvencijski dijagrami	2.5						
Opis ostalih zahtjeva	1.5						
Arhitektura i dizajn sustava							
Baza podataka							
Dijagram razreda							
Dijagram stanja							
Dijagram aktivnosti							
Dijagram komponenti							
Korištene tehnologije i alati	2						
Ispitivanje programskog rješenja							

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

	<b>Toni Ivanković</b>	<b>Ana Gršković</b>	<b>Borna Mahović</b>	<b>Jakov Matošić</b>	<b>Dora Nevidal</b>	<b>Marko Opačić</b>	<b>Luka Vukota</b>
Dijagram razmještaja							
Upute za puštanje u pogon							
Dnevnik sastajanja	1.5						
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>	2						

## Dijagrami pregleda promjena

### *dio 2. revizije*

*Prenijeti dijagram pregleda promjena nad datotekama projekta. Potrebno je na kraju projekta generirane grafove s gitlaba prenijeti u ovo poglavlje dokumentacije. Dijagrami za vlastiti projekt se mogu preuzeti s [gitlab.com](https://gitlab.com) stranice, u izborniku Repository, pritiskom na stavku Contributors.*