

# Programsko inženjerstvo

Ak. god. 2023./2024.

## DentAll

Dokumentacija, Rev. 0.1

Grupa: Potplaćeni

Voditelj: Luka Kokić

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

Nastavnik: <Ime i prezime nastavnika zaduženog za vašu grupu>

# Sadržaj

<b>1</b>	<b>Dnevnik promjena dokumentacije</b>	<b>3</b>
<b>2</b>	<b>Opis projektnog zadatka predložak</b>	<b>5</b>
2.1	Opis projektnog zadatka . . . . .	6
2.2	Primjeri teksta u L <sup>A</sup> T <sub>E</sub> Xu . . . . .	10
<b>3</b>	<b>Specifikacija programske potpore</b>	<b>13</b>
3.1	Funkcionalni zahtjevi . . . . .	13
3.1.1	Obrasci uporabe . . . . .	14
3.1.2	Sekvencijski dijagrami . . . . .	15
3.2	Ostali zahtjevi . . . . .	16
<b>4</b>	<b>Arhitektura i dizajn sustava</b>	<b>17</b>
4.1	Baza podataka . . . . .	17
4.1.1	Opis tablica . . . . .	17
4.1.2	Dijagram baze podataka . . . . .	18
4.2	Dijagram razreda . . . . .	19
4.3	Dijagram stanja . . . . .	20
4.4	Dijagram aktivnosti . . . . .	21
4.5	Dijagram komponenti . . . . .	22
<b>5</b>	<b>Implementacija i korisničko sučelje</b>	<b>23</b>
5.1	Korištene tehnologije i alati . . . . .	23
5.2	Ispitivanje programskog rješenja . . . . .	24
5.2.1	Ispitivanje komponenti . . . . .	24
5.2.2	Ispitivanje sustava . . . . .	24
5.3	Dijagram razmještaja . . . . .	25
5.4	Upute za puštanje u pogon . . . . .	26
<b>6</b>	<b>Zaključak i budući rad</b>	<b>27</b>
	<b>Popis literature</b>	<b>28</b>

**Indeks slika i dijagrama** 29

**Dodatak: Prikaz aktivnosti grupe** 30

# 1. Dnevnik promjena dokumentacije

## *Kontinuirano osvježavanje*

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napravljen predložak.	*	22.08.2013.
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.
1.2	Manje izmjene: Timer - Brojilo vremena	*	15.09.2013.

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

Rev.	Opis promjene/dodatka	Autori	Datum
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 predložak

### *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*
- *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.*

## 2.1 Opis projektnog zadatka

Cilj ovog projekta je razviti web aplikaciju „DentAll“ kojom će pružatelji usluga zdravstvenog turizma moći evidentirati i koordinirati lokalni smještaj i prijevoz korisnika usluga. Time bi se zdravstveni turizam učinio privlačnijim rješavanjem brige i cijene smještaja i prijevoza korisnika pri njihovom obitavanju u mjestu gdje koriste spomenute usluge. Uz to bi klijenti pružatelja usluga bili u mogućnosti unaprijed vidjeti geografsku sliku smještaja i okolice te moguće turističke opcije i kretanja tijekom privremenog obitavanja.

Aplikacija bi olakšala brige vlasnika usluga zdravstvenog turizma evidentiranjem, organizacijom i praćenjem smještaja i prijevoza klijenata uz programsku podršku interneta. Time se nastoji nadomjestiti nedostatak ovisnosti pružatelja zdravstvenih usluga o raznim i višestrukim pružateljima smještajnih usluga. Takvim pristupom kroz javnu i lagano dostupnu uslugu smanjio bi se broj posrednika u organizaciji i administraciji pružanja usluga zdravstvenog turizma. Također bi klijenti pronalazili sve potrebne informacije oko odabranog zdravstvenog turizma, tj. uz same zdravstvene usluge i o smještaju te prijevozu za svaki termin, čineći sam odabir prisustvovanja u inozemnim zdravstvenim uslugama jednostavniji i privlačniji.

Trenutna postojeća rješenja uključuju korištenje hotela i motela, što čini cijeli proces zdravstvenog turizma skupljim i kompliciranijim. Dugoročno bi pružateljima bilo isplativije iznajmljivati vlastite nekretnine u sklopu pružanja cjelokupnih usluga. Također je u trenutnim rješenjima prijevoz ostavljen na odgovornost klijenata, što otežava korištenje cijele usluge i njezinu privlačnost. Spajanjem oba problema u jedan sustav bi učinilo uslugu zdravstvenog turizma puno efektivnijom i jeftinijom, a time i više privlačnom mogućim budućim klijentima.

Postojeće vrste konkurentnih rješenja:

- javno dostupne informacije o načinima sudjelovanja u zdravstvenom turizmu i pripadnim članovima
- udruge sa svrhom spajanja svih dionika pružanja usluga zdravstvenog turizma

Primjer za usluge pružanja informacija je web stranica „MedicalTourism.com“ koja pruža korisnicima podatke o pružateljima zdravstvenih usluga, smještaja i mogućih tretmana koje spomenute usluge pružaju. Također sadrži i mnoge druge informacije kao vodiče za destinacije, usporedbe cijena i slično. No ipak je spomenuta web aplikacija napravljena za pružanje laganog pristupa svim potrebnim informacijama za moguće klijente, pružatelje zdravstvenih usluga, pružatelje smještaja te osiguravajuća društva na jednom mjestu te se time ne sukobljava do konkurentske razine sa svrhom kreiranja projektne aplikacije.

Dok je primjer druge vrste rješenja, čak konkurentnije ideji projekta, udruga Medical Tourism Association, koja spaja sve potrebne članove zdravstvenog turizma kroz program članstva. Udruga operira po cijelom svijetu i služi svrhu omogućavanja pružanja usluga zdravstvenog turizma kroz povezivanje potrebnih članova te svrhu informiranja javnosti o mogućnostima korištenja tih usluga. Također podržava edukaciju budućih pružatelja sa sveukupnim ciljem povećanja učestalosti zdravstvenog turizma u svijetu. Ovdje opisan projekt ipak sadrži određene funkcionalnosti koje nedostaju navedenoj konkurenciji, kao automatiziran proces organizacije i administracije, te lagano dostupni pregled informacija o obitavanju za klijente zdravstvenih usluga.



Slika 2.1: Mogućnosti povezivanja klijenata i pružatelja koje nudi web stranica "MedicalTourism.com"



Ciljani klijenti opisanih usluga su pružatelji usluga zdravstvenog turizma po cijelom svijetu, uključujući Hrvatsku i slične države sa jeftinom cijenom boravka. Optimalno bi bilo započeti pružanje usluga aplikacije najprije Europskim državama i okolici, a zatim, uz dovoljnu uspješnost i isplativost projekta, raširiti dostupnost usluge ostatku svijeta. Ciljani klijenti bi također bili i pružatelji lokalnih prijevoznih usluga, od privatnih firmi do javnih taksija, koje bi aplikacija spajala sa pružateljima zdravstvenih usluga za dogovor o prijevozu njihovih klijenata. Također bi, u manjoj perspektivi, evidencija većine korisnika zdravstvenog turizma na jednom mjestu olakšala statističke analize razvoja zdravstvenog turizma po cijelom svijetu.

U aplikaciji postoje četiri uloge:

- smještajni administrator
- administrator prijevoznih usluga
- korisnički administrator
- obični korisnik (pacijent)

Jedan korisnik može imati više administratorskih uloga, dok uloga smještajnog administratora sadrži najveće ovlasti kojima mogu definirati druge korisnike i dodjeljivati uloge. Svaka uloga administratora sadrži određene posebne mogućnosti dodavanja i promjene informacija.

Smještajni administratori upravljaju podacima o smještaju; unose podatke o raspoloživom smještaju te definiraju smještajne kapacitete i unose ili brišu osnovne podatke o smještaju. Podatci smještaja se sastoje od tipa stana, kategorije opremljenosti, adrese i vremenskog perioda dostupnosti za korištenje. Uz svaki smještaj je dostupan grafički prikaz geografskog položaja uz Google Maps usluge.

Administratori prijevoznih usluga upravljaju podacima o prijevoznicima. To uključuje osnovne osobne i kontaktne podatke prijevoznika, vrstu i kapacitet prijevoznog sredstva te radno vrijeme raspoloživosti. Osnovni podatci prijevoznika se ne mogu mijenjati.

Korisnički administratori definiraju korisnike medicinskih usluga uz unos njihovih osnovnih podataka. Podatci korisnika se sastoje od imena, prezimena, kontaktnih podataka, vremena dolaska i odlaska u/iz zemlje te preferencije o veličini i kvaliteti smještaja. Detalji njihovih tretmana se preuzimaju iz vanjske aplikacije o evidenciji medicinskih usluga.

Aplikacija periodički provjerava status unesenih podataka i pridjeljuje adekvatni smještaj upisanim korisnicima te po zaključenju plana medicinskih usluga određenog korisnika im pridjeljuje prijevoznike od raspoloživih za svaki od termina. Sa završetkom spomenutog zaključenja sustav šalje poruku elektroničke pošte korisniku medicinskih usluga sa svim potrebnim informacijama te također šalje poruke elektroničke pošte prijevoznicima pridijeljenima terminima korisnika sa svim njima potrebnim informacijama.

Moguće dodatne funkcionalnosti za aplikaciju koje se mogu nadodati nakon izvršavanja jezgrenih funkcionalnosti su:

- prijavljivanje samog korisnika medicinskih usluga u aplikaciju čime mogu provjeriti osobne podatke, smještaj, prijevoznike i termine
- proširenje aplikacije u dodatnu svrhu prikaza općenitih podataka o medicinskim turizmu za privlačenje dodatnih klijenata

## 2.2 Primjeri teksta 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: # \$ % & { } \_ | < > ^ ~ \

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

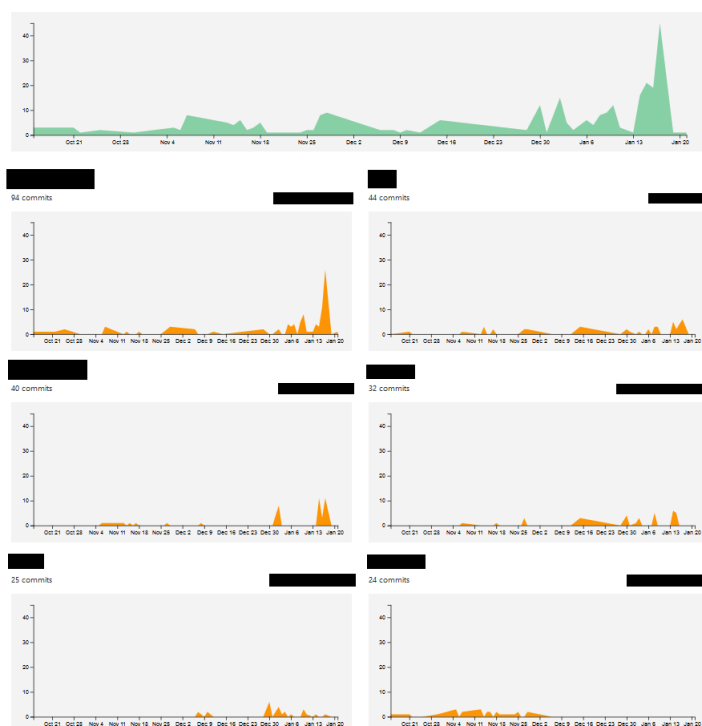
Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

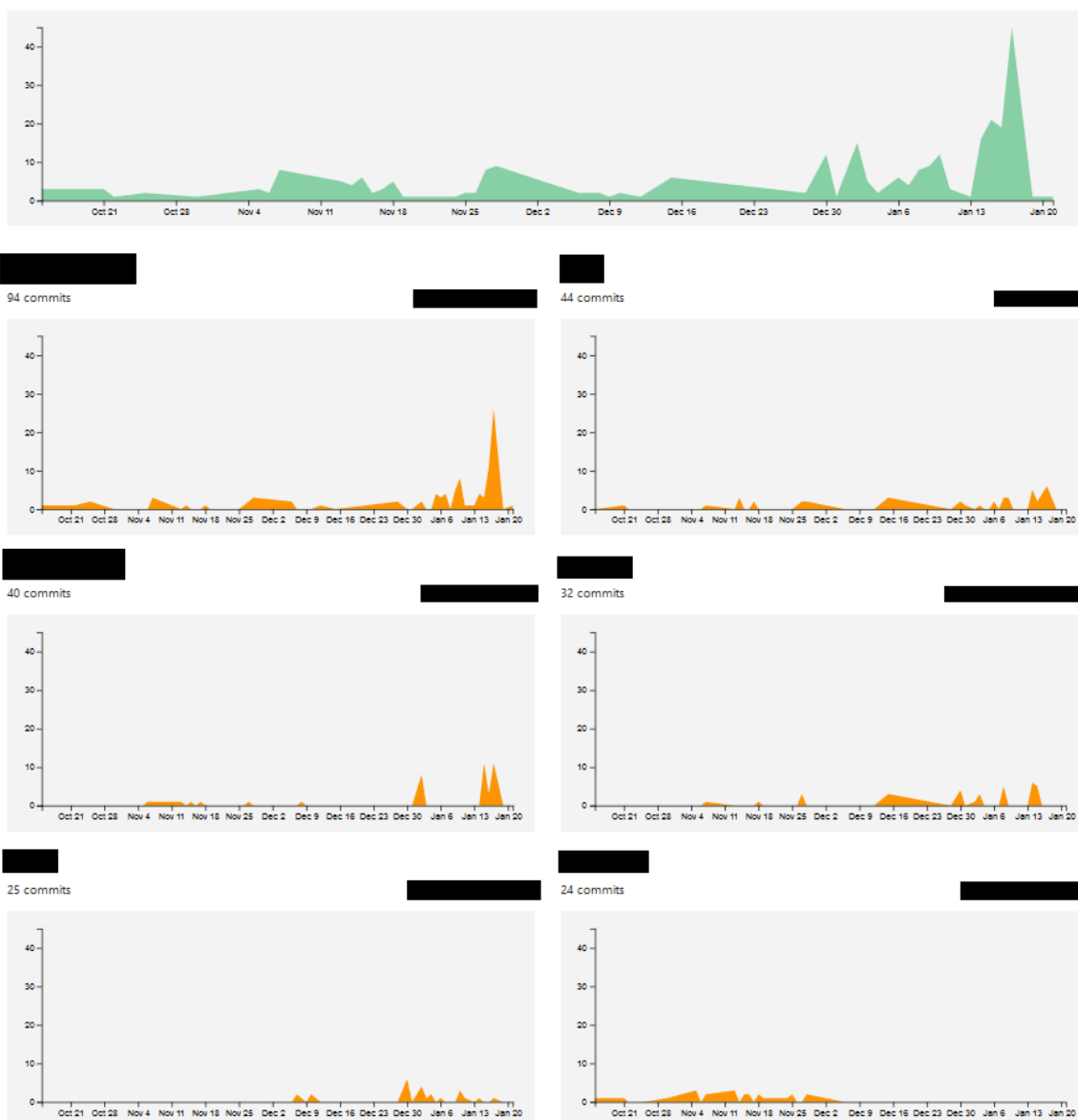
naslov unutar tablice		
ime	VARCHAR	
primjer	VARCHAR	

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 2.2: 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

#### *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. Dionik 1
2. Dionik 2
3. ...

#### **Aktori i njihovi funkcionalni zahtjevi:**

1. Aktor 1 (inicijator) može:
  - (a) funkcionalnost 1
  - (b) funkcionalnost 2
    - i. podfunkcionalnost 1
    - ii. podfunkcionalnost 2
  - (c) funkcionalnost 3
2. Aktor 2 (sudionik) može:
  - (a) funkcionalnost 1
  - (b) funkcionalnost 2

### 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:

- 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)

### 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 Selenium<sup>1</sup>. 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.*

---

<sup>1</sup><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	Mogućnosti povezivanja klijenata i pružatelja koje nudi web stranica "MedicalTourism.com" . . . . .	7
2.2	Primjer slike s potpisom . . . . .	11
2.3	Primjer slike s potpisom 2 . . . . .	12

# 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: 2. studenoga 2023.
- Prisustvovali: I.Prezime, I.Prezime
- Teme sastanka:
  - opis prve teme
  - opis druge teme

#### 2. sastanak

- Datum: u ovom formatu: 2. studenoga 2023.
- 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.*

	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](https://gitlab.com) stranice, u izborniku Repository, pritiskom na stavku Contributors.*