

# Oblikovanje programske potpore

Ak. god. 2019./2020.

## Moj AutoServis

Dokumentacija, Rev. 0.1

Grupa: *InfiniTeam*

Voditelj: *Patrik Matošević*

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

Nastavnik: *Nikolina Frid*

# Sadržaj

<b>1 Dnevnik promjena dokumentacije</b>	<b>3</b>
<b>2 Opis projektnog zadatka</b>	<b>4</b>
2.1 Primjeri u LaTeXu . . . . .	5
<b>3 Specifikacija programske potpore</b>	<b>8</b>
3.1 Funkcionalni zahtjevi . . . . .	8
3.1.1 Obrasci uporabe . . . . .	11
3.1.2 Sekvencijski dijagrami . . . . .	12
3.2 Ostali zahtjevi . . . . .	13
<b>4 Arhitektura i dizajn sustava</b>	<b>14</b>
4.1 Baza podataka . . . . .	14
4.1.1 Opis tablica . . . . .	14
4.1.2 Dijagram baze podataka . . . . .	15
4.2 Dijagram razreda . . . . .	16
4.3 Dijagram stanja . . . . .	17
4.4 Dijagram aktivnosti . . . . .	18
4.5 Dijagram komponenti . . . . .	19
<b>5 Implementacija i korisničko sučelje</b>	<b>20</b>
5.1 Korištene tehnologije i alati . . . . .	20
5.2 Ispitivanje programskog rješenja . . . . .	21
5.2.1 Ispitivanje komponenti . . . . .	21
5.2.2 Ispitivanje sustava . . . . .	21
5.3 Dijagram razmještaja . . . . .	22
5.4 Upute za puštanje u pogon . . . . .	23
<b>6 Zaključak i budući rad</b>	<b>24</b>
<b>Popis literature</b>	<b>25</b>

<b>Indeks slika i dijagrama</b>	<b>26</b>
<b>Dodatak: Prikaz aktivnosti grupe</b>	<b>27</b>

# 1. Dnevnik promjena dokumentacije

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napravljen predložak.	Matošević	17.10.2019.
0.2	Napisani funkcionalni i ostali zahtjevi	Has, Prpić	23.10.2019.
0.3	Uređeni i dopunjeni funkcionalni zahtjevi	Prpić	25.10.2019.
0.4	Funkcionalni zahtjevi uređeni u skladu s obrascima uporabe	Has, Prpić	4.11.2019.

## 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.*
- *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 Primjeri u LaTeXu

*Ovo potpoglavlje izbrisati.*

U nastavku se nalaze različiti primjeri kako koristiti osnovne funkcionalnosti LaTeXa 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 LaTeXu - [https://www.fer.unizg.hr/\\_download/repository/LaTeX-upute.pdf](https://www.fer.unizg.hr/_download/repository/LaTeX-upute.pdf)
- LaTeX 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
  2. primjer

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

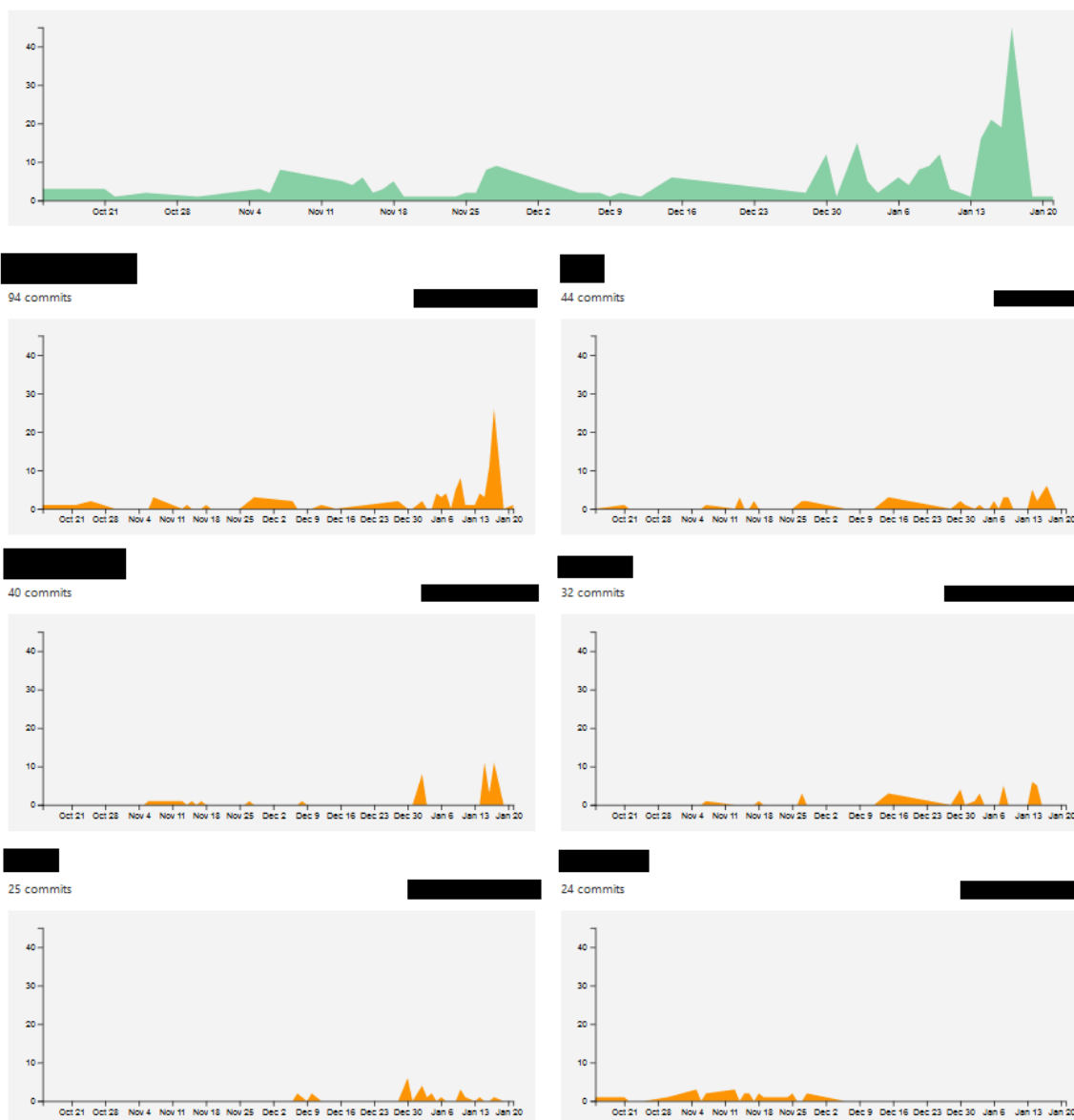
naslov unutar tablice		
IDKorisnik	INT	Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod
korisnickoIme	VARCHAR	
email	VARCHAR	
ime	VARCHAR	
primjer	VARCHAR	

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

Tablica 2.3: Naslov ispod tablice.



Slika 2.1: Primjer slike s potpisom



Slika 2.2: Primjer slike s potpisom 2



## 3. Specifikacija programske potpore

### 3.1 Funkcionalni zahtjevi

Dionici:

1. Posjetitelj
2. Registrirani korisnik
3. Vlasnik automobila
4. Vlasnik autoservisa
5. Serviser
6. Administrator

Aktori i njihovi funkcionalni zahtjevi:

1. Posjetitelj (inicijator) može:
  - (a) Registracija vlasnika automobila
  - (b) Registracija vlasnika autoservisa
  - (c) Pregled autoservisa
2. Registrirani korisnik (inicijator) može:
  - (a) Upravljanje korisničkim računom
    - i. Pregled korisničkih podataka
    - ii. Promjena korisničkih podataka
  - (b) Prijava u sustav
  - (c) Odjava
3. Vlasnik automobila (inicijator) može:
  - (a) Pregled autoservisa
  - (b) Upravljanje automobilima
    - i. Dodavanje automobila
    - ii. Pregled statističkih podataka za automobil
    - iii. Brisanje automobila

- (c) Upravljanje servisnim nalogima
  - i. Otvaranje novog servisnog naloga
  - ii. Pregled statusa servisnog naloga

4. Vlasnik autoservisa (inicijator) može:

- (a) Prihvaćanje radnog naloga
- (b) Upravljanje rezervnim dijelovima
  - i. Dodavanje rezervnog dijela
  - ii. Uređivanje rezervnog dijela
  - iii. Brisanje rezervnog dijela
- (c) Uređivanje podataka autoservisa
- (d) Upravljanje serviserima
  - i. Dodavanje serviseri
  - ii. Uklanjanje serviseri

5. Serviser (inicijator) može:

- (a) Rad na radnom nalogu
  - i. Uređivanje naloga redovnog servisa
  - ii. Zatvaranje radnog naloga
  - iii. Rad na nalogu izvanrednog servisa
    - A. Dodavanje ugrađenog rezervnog dijela
    - B. Brisanje ugrađenog rezervnog dijela
    - C. Dodavanje usluge
    - D. Brisanje usluge

6. Administrator (inicijator) može:

- (a) Administracija korisnika
  - i. Pregled podataka korisničkog računa
  - ii. Brisanje korisničkog računa
- (b) Administracija autoservisa
  - i. Pregled podataka autoservisa
  - ii. Brisanje autoservisa

7. HUO registar (sudionik) može:

- (a) Provjera registracije i dohvat broja šasije

8. Baza podataka (sudionik) može:

- (a) Komunikacija sa svim dijelovima sustava
- (b) Pohrana podataka i ovlasti

### 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*

- Aplikacija mora biti pregledna te mora imati laku dostupnost raznim funkcionalnostima i sadržajima
- Mogućnost istovremenog pristupa aplikaciji od strane više korisnika
- U slučaju neispravnog korištenja korisničkog sučelja sustav mora prikazati odgovarajuću poruku upozorenja
- Aplikacija mora podržavati hrvatsku abecedu pri unosu podataka
- Sustav treba imati pristup HUUO registru
- U slučaju prijednog određenog broja kilometara od posljednjeg redovnog servisa, aplikacija treba prikazati poruku preporuke redovnog servisa nakon prijave u sustav
- Sustav koristi HRK kao valutu za izračun i prikaz cijena
- Izvršavanje dijelova aplikacije mora se odvijati u razumnom roku

## 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 tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam

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

1. Oblikovanje programske potpore, FER ZEMRIS, <http://www.fer.hr/predmet/opp>
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 . . . . .	6
2.2	Primjer slike s potpisom 2 . . . . .	7

# Dodatak: Prikaz aktivnosti grupe

## Dnevnik sastajanja

### 1. sastanak

- Datum: 09.10.2019.
- Prisustvovali: P.Matošević, K.Boras, D.V.Cvitković, D.Facković, M.Has, N.Kolarec, J.Prpić
- Teme sastanka:
  - sastanak s asistentom
  - analiza zadanog zadatka i mogućih alata i tehnologija
  - rasprava potencijalnih ostalih tema

### 2. sastanak

- Datum: 17.10.2019.
- Prisustvovali: P.Matošević, K.Boras, D.V.Cvitković, D.Facković, N.Kolarec, J.Prpić
- Teme sastanka:
  - postavljanje okoline potrebne za nastavak rada na projektu
  - upoznavanje sa Gitom i kontrolerima u Spring Boot-u
  - rasprava oko organizacije posla
  - diskusija o iskustvu članova tima u pojedinim tehnologijama
  - detaljna analiza zadatka

### 3. sastanak

- Datum: 23.10.2019.
- Prisustvovali: M.Has, J.Prpić
- Teme sastanka:
  - postavljanje funkcionalnih zahtjeva
  - postavljanje ostalih zahtjeva

## Tablica aktivnosti

	Patrik Matošević	Katarina Boras	Daria Vanesa Cvitković	Dora Facković	Mislav Has	Nina Kolarec	Juraj Prpić
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							
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>							

	<b>Patrik Matošević</b>	<b>Katarina Boras</b>	<b>Daria Vanesa Cvitković</b>	<b>Dora Facković</b>	<b>Mislav Has</b>	<b>Nina Kolarec</b>	<b>Juraj Prpić</b>
<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.*