

Programsko inženjerstvo

Ak. god. 2023./2024.

Medicinska rehabilitacija

Dokumentacija, Rev. 1

Grupa: Teletabisi

Voditelj: Benjamin Gregov

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

Nastavnik: Miljenko Krhen

Sadržaj

1 Dnevnik promjena dokumentacije	3
2 Opis projektnog zadatka	5
3 Specifikacija programske potpore	10
3.1 Funkcionalni zahtjevi	10
3.1.1 Obrasci uporabe	12
3.1.2 Sekvencijski dijagrami	13
3.2 Ostali zahtjevi	14
4 Arhitektura i dizajn sustava	15
4.1 Baza podataka	15
4.1.1 Opis tablica	15
4.1.2 Dijagram baze podataka	16
4.2 Dijagram razreda	17
4.3 Dijagram stanja	18
4.4 Dijagram aktivnosti	19
4.5 Dijagram komponenti	20
5 Implementacija i korisničko sučelje	21
5.1 Korištene tehnologije i alati	21
5.2 Ispitivanje programskog rješenja	22
5.2.1 Ispitivanje komponenti	22
5.2.2 Ispitivanje sustava	22
5.3 Dijagram razmještaja	23
5.4 Upute za puštanje u pogon	24
6 Zaključak i budući rad	25
Popis literature	26
Indeks slika i dijagrama	27

Dodatak: Prikaz aktivnosti grupe

28

1. Dnevnik promjena dokumentacije

Kontinuirano osvježavanje

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napravljen predložak. Napisan opis projektnog zadatka i funkcionalni zahtjevi	Neven Pralas	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

Uvod

Cilj projekta "Medicinska rehabilitacija" je razvoj programske podrške za stvaranje istoimene web aplikacije koja će omogućiti korisnicima jednostavno naručivanje za fizičnu terapiju i medicinsku rehabilitaciju. Korisnici u našem slučaju su bolesnici koji su se ozlijedili te trebaju stručnu pomoć. Zadatak nam je omogućiti intuitivno korisničko sučelje kako bi olakšali korištenje aplikacije bolesnicima, ali i zdravstvenim djelatnicima.

Glavna motivacija bila nam je modernizacija hrvatskog zdravstva. Vjerujemo kako pacijentima nije drago zvati mobitelom svoje doktore znajući da su pretrpani poslom, a također im je dosta čekanje na odgovor zdravstvenog djelatnika kada im se pošalje mail. S druge strane nije ni zdravstvenim djelatnicima lako odgovarati na sve te silne poruke. Ovom aplikacijom bismo omogućili pacijentima da biraju dostupan termin koji njima najbolje odgovara, a doktore poštedili dodatnog posla.

Između ostalog ova bi aplikacija omogućivala liječnicima praćenje poboljšanja zdravstvenog stanja pacijenata. Ako se pacijenti prvi puta naručuju za terapiju potrebna je registracija, a svaki sljedeći put je nužna prijava u sustav. Za registraciju potrebni su **sljedeći podaci**:

- ime i prezime
- lozinka
- e-mail adresa
- OIB

Administrator gornje podatke mora verificirati iz središnjeg informacijskog sustava zdravstvene zaštite. Bolesnici prilikom prijave moraju navesti vrstu i opis svojih oboljenja, zahtijevani postupak liječenja te liječnika koji ih je uputio na rehabilitaciju. Nakon toga moraju odabrati neki od dostupnih termina. Ako se ne radi o prvom dolasku u ustanovu zahtijeva se i prikaz reference na već obavljeni postupak u toj ustanovi. Zdravstveni djelatnici dodjeljuju termine tako da vode brigu o ukupnom kapacitetu opreme/uređaja, kapacitetu prostorija, broju dostupnih djelatnika te trajanju svakog zahvata. Nakon dobivenog termina bolesniku dolazi e-pošta sa terminom te ostalim dodatnim informacijama. U slučaju neočekivanih promjena djelatnik može kontaktirati

bolesnika putem e-pošte. Radno vrijeme ustanove u kojoj se provodi rehabilitacija je svakim radnim danom od 8 do 20 sati. Ustanova ima pravo odrediti određene slobodne dane na temelju blagdana ili praznika.

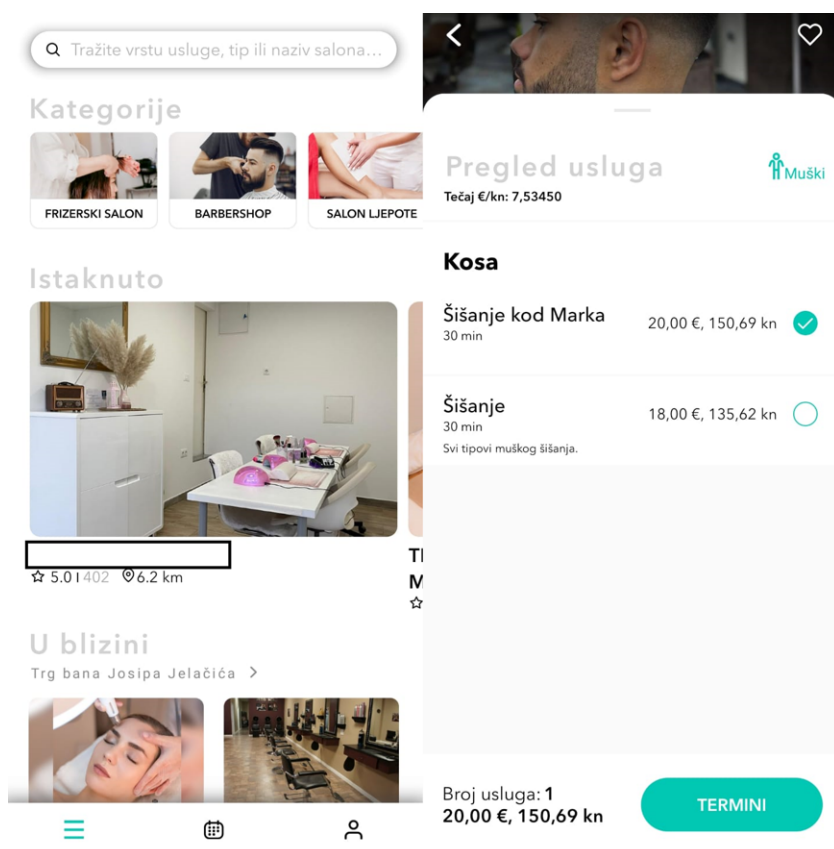
Primjer sličnog rješenja

Primjer sličnog rješenja je aplikacija "Buker". To je aplikacija koja omogućuje naručivanje za termin u određenim salonima (npr. frizerskim ili kozmetičkim).

Registracija je omogućena putem telefonskog broja gdje korisnik mora verificirati svoje podatke ili još jednostavnije putem google računa. Za registraciju u ovaj sustav potrebno je ispuniti svoje ime, prezime te e-mail adresu. Naravno ovdje se neće tražiti puno informacija kao u našem sustavu niti se očekuje toliko stroga kontrola istinitosti podataka.

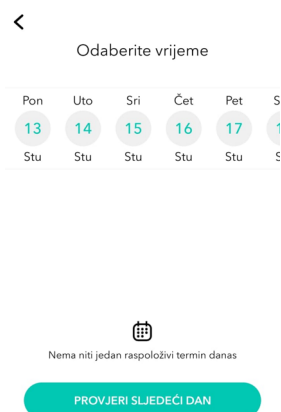
Slika 2.1: Registracija u sustav "Buker"

Zatim imamo mogućnost odabrati kategoriju te u pojedinoj kategoriji određeni salon gdje dobijemo prikaz njihovih usluga. Označimo onu uslugu koju želimo te kliknemo na dugme "Termini".



Slika 2.2: Odabir kategorije i usluge

Na kraju nam preostaje odabir dostupnih termina. Naravno ovdje slobodni termini ovise isključivo o radnom vremenu salona, zauzeću od nekog drugog klijenta te trajanju samog zahvata. Kod naše aplikacije termini ovise dodatno o kapacitetu opreme/uređaja, kapacitetu prostorija i broju dostupnih djelatnika.



Slika 2.3: Odabir termina

Dodatno pojašnjenje

Korisnici sustava su:

- *pacijenti*
- *zdravstveni djelatnici*
- *administrator*

Svaki od te tri skupine korisnika ima različita dopuštenja. Djelatnici ustanove mogu vidjeti sve bolesnike i tretmane koji su im dodijeljeni. Nakon svakog obavljenog postupka djelatnik u sustav registrira pacijenta, a nakon završenog ciklusa zapisuje postignute rezultate. Pacijenti mogu vidjeti samo svoje tretmane. Administrator upravlja čitavim sustavom. Njemu su dostupni svi podaci i može definirati sve što je potrebno za ispravan rad sustava.

Za unesene podatke prilikom registracije u aplikaciju osigurat ćemo provjeru konzistentnosti s podacima iz baze podataka. Nakon toga korisnik postaje pacijent, izabire datum te ima mogućnost otkazati termin 1 dan prije početka tog termina.

Opreme, odnosno uređaji koje posjeduje ustanova se kategoriziraju. Spomenuli smo da slobodni termini ovise o ukupnom kapacitetu opreme/uređaja. Naravno da nije isto ako je čovjeku oštećena ruka ili noga. Tu se radi o drugoj opremi koju će zdravstveni djelatnik morati upotrijebiti za liječenja pacijenta. Dakle, korisnik kojemu je ozljeđena ruka neće zauzimati opremu korisniku kojemu je ozljeđena noga. Očekujemo da će broj oprema/uređaja za određeni dio tijela ipak biti veći od jedan, no svakako će biti ograničen.

Korisnik se u sustav prijavljuje upisivanjem e-mail adrese te svoje lozinke koja mora zadovoljavati određena pravila. Lozinka mora imati minimalno 8 znakova, od čega mora biti minimalno jedno veliko slovo te jedna znamenka. Ako korisnik prilikom registracije i smišljanja lozinke nije zadovoljio navedena pravila, dobit će upozorenje što mu nedostaje za ispravnu lozinku.

*Aplikacija će imati ***n*** stranica. Te stranice su redom: Naslovna stranica, stranica za registraciju, stranica za prijavu, **stranica za promjenu lozinke**, stranica za odabir termina, stranica za administratora gdje administrator održava sustav i potvrđuje registracije, stranica gdje djelatnik potvrđuje termine, **"help page" koja objašnjava funkcionalnost aplikacije** i stranica u kojoj se provjerava status pacijenta (odnosno njihovu povijest sa tom ustanovom).*

Opseg projekta

Na projektu radi 7 osoba, studenata FER-a. Projekt se radi u edukativne svrhe provjere znanja studenata u sklopu predmeta "Programsko inženjerstvo". Vremensko ograničenje za prvu verziju projekta je 7 tjedana, a druga verzija (odnosno cijeli projekt) mora biti isporučen u roku od 14 tjedana od početka rada. Ukupan broj potrošenog vremena otići će najviše na sami rad oko projekta, ali značajan dio vremena otići će i na sastanke uživo gdje će tim redovito raspraviti o tjednom napretku.

3. Specifikacija programske potpore

3.1 Funkcionalni zahtjevi

Dionici:

1. Vlasnik (naručitelj)
2. Pacijenti
3. Zdravstveni djelatnici
4. Administrator
5. Razvojni tim

Aktori i njihovi funkcionalni zahtjevi:

1. Neregistrirani/neprijavljeni korisnik (inicijator) može:
 - (a) se registrirati u sustav kao pacijent te tako stvoriti novi korisnički račun za koji su mu potrebni ime, prezime, e-mail adresa, OIB i lozinka
 - (b) pregledati naslovnu stranicu i stranicu koja objašnjava funkcionalnost aplikacije
2. Pacijent (inicijator) može:
 - (a) prijaviti se u sustav
 - (b) odabrati željeni termin dolazaka, a ukoliko se radi o ponovljenoj terapiji i referencu na već obavljeni postupak terapije u ustanovi
 - (c) navesti opis svog oboljenja te odabrati liječnika koji će ga liječiti
 - (d) otkazati zakazani termin u roku 24 sata prije samoga termina te potvrditi ili odbiti pomaknuti termin zbog novonastale okolnosti
 - (e) primati elektroničku poštu o zakazanom terminu i novonastalim okolnostima
3. Zdravstveni djelatnik (inicijator) može:
 - (a) prijaviti se u sustav
 - (b) pregledavati prijave i dodjeljivati pacijentima termine

- (c) direktno kontaktirati pacijenta putem elektroničke pošte u slučaju neočekivanih promjena
- (d) vidjeti sve pacijente i sve tretmane koji su im dodijeljeni
- (e) nakon obavljenog postupka rehabilitacije registrirati dolazak pacijenta
- (f) nakon završenog ciklusa upisati postignute rezultate s pacijentom

4. Administrator (inicijator) može:

- (a) verificirati podatke iz središnjeg informacijskog sustava zdravstvene zaštite
- (b) pristupiti popisu korisnika, zdravstvenih djelatnika te njihovim osobnim podacima
- (c) stvoriti zdravstvenog djelatnika u sustavu

5. Baza podataka (sudionik) može:

- (a) pohraniti podatke o korisnicima i njihovim dopuštjenjima
- (b) pohraniti podatke o terminima (slobodnim i zauzetim)
- (c) pohraniti podatke o opremama/uslugama (slobodnim i zauzetim)

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 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	Registracija u sustav "Buker"	6
2.2	Odabir kategorije i usluge	7
2.3	Odabir termina	7

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

2. sastanak

- Datum: u ovom formatu: 5. 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 stranice, u izborniku Repository, pritiskom na stavku Contributors.