

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

1	Dnevnik promjena dokumentacije	3
2	Opis projektnog zadatka	4
2.1	Primjeri u LaTeXu	8
3	Specifikacija programske potpore	11
3.1	Funkcionalni zahtjevi	11
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

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napravljen predložak.	Matošević	17.10.2019.
0.1	Napisan opis projekta te neka postojeća slična rješenja.	Boras	22.10.2019.

2. Opis projektnog zadatka

Cilj ovog projektnog zadatka je osmisлити i razviti programsku potporu za web-aplikaciju *"MojAutoServis"* koja će registriranim vlasnicima automobila uštediti vrijeme prilikom obavljanja redovitih i izvanrednih servisa vozila. Aplikacija će ih brzo i efikasno povezati sa auto-servisima, tj. sa serviserima, te tako eliminirati potrebu za dugim i nepotrebnim čekanjem u redovima i za ispunjavanjem papirologije.

Kada neregistrirani korisnik pokrene aplikaciju, ima opciju pregledavanja svih auto-servisa te su mu dostupne informacije o njima. Također ima mogućnost registracije. Postoje tri vrste korisnika:

- vlasnik automobila,
- auto-servis, i
- serviser.

Postojeći korisnik registrira se svojim jedinstvenim korisničkim imenom i lozinkom. Kako bi se registrirao, neregistrirani korisnik mora aplikaciji dati:

- svoje ime,
- svoje prezime,
- OIB, te
- adresu svoje e-pošte.

U slučaju da se radi o tvrtci, podatci koji su potrebni su:

- naziv,
- adresa, i
- OIB tvrtke.

Nakon registracije, korisnik unosi podatke o automobilu iz prometne dozvole, među kojima su obavezno:

- registracijska oznaka, te
- marka automobila

Korisnik može u bilo kojem trenutku dodati još automobila, te ih može imati neograničeno mnogo. Kako bi novi automobil bio dodan, uneseni podaci moraju biti potvrđeni u HUO registru. Ukoliko to želi, korisnik također ima mogućnost izbrisati automobil.

Kada korisnik želi poslati svoj automobil na servis, mora otvoriti radni nalog, prilikom čega odabire redoviti ili izvanredni servis. Nakon što je rad na automobilu obavljen, korisniku dolaze detalji servisa, te su mu vidljivi zajedno sa detaljima svih prijašnjih obavljenih servisa. Te informacije su dostupne sve dok se automobil ne obriše iz aplikacije.

Auto-servisi nakon registracije dodaju korisnički račun svoga administratora te servisere. Potom dodaju cjenik usluga i podatke o auto-dijelovima, te započinju s radom. Dostupna im je lista serviseri preko koje mogu upravljati svojim zaposlenicima.

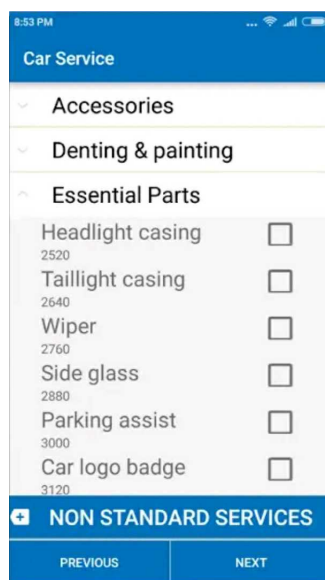
Serviseri primaju radne naloge poslane njihovoj tvrtci. Ako ih prihvate obavljaju servis, te u radni nalog dopisuju dodatne podatke*****. Naknadno, ako je na automobilu primjećena potreba za dodatnim popravcima, serviser daje preporuku za izvanredan servis. Serviseri, kao i obični korisnici, mogu dodavati podatke o vlastitim automobilima i sami otvarati radne naloga za njih.

Glavni administrator odobrava registracije autoservisa, ima upravu nad svim korisnicima, može brisati račune i ima pristup svim statističkim podacima koji se izračunavaju. **Treba dovršiti/dopuniti**

Primjeri sličnih rješenja

Dvije aplikacije koje su najbližnije ovoj su ShopBoss i GetAFix.

GetAFix je aplikacija za upravljanje auto-servisom. Za razliku od ove aplikacije, korisnika se traži da na slici automobila označi gdje postoje ogrebotine, te da procijeni stanje auta u raznim kategorijama (ulje, ...).



Slika 2.1: GetAFix

ShopBoss je software dizajniran za vlasnike automobila i auto-servise. Ima više mogućnosti od aplikacije MojAutoServis, te uz to ima mogućnost pristupa korisničkom računu sa bilo kojeg uređaja. U nju su također integrirani razni alati i web-stranice poput Carfax-a itd.

RO	STATUS	DATE	CUSTOMER	PHONE	VEHICLE	TOTAL	LICENSE	W+TK	TYPE
2987	INSPECTION	10/22/2019	TEST 123 TEST123	H: (860) 829-7377	2015 CHEVROLET SILVERADO 2500H	1,128.67	DEMOJC	ANOTHER TEST	
2986	INSPECTION	10/17/2019	TEST 123 TEST123	#377.2013	HYUNDAI SANTA FE SPORT 2.0	95.22	DEMOJC	ANOTHER TEST	
2985	INSPECTION	10/17/2019		H: (860) 829-7377	2011 JAGUAR XF PREMIUM	14.22	BKKA 198	DEMO	ANOTHER TEST
2983	INSPECTION	10/15/2019	ROCIDO SALAZAR	C: (480) 524-3361	2001 GMC SIERRA 1500 C3 EXT. C	358.41	JCJC	ANOTHER TEST	
2982	INSPECTION	10/15/2019		H: (860) 829-7377	2013 HYUNDAI ELANTRA GLS/UMIT	135.01	BJLM 182	DEMOJC	ANOTHER TEST
2981	INSPECTION	10/14/2019	KONRAD CURZE		2004 VOLKSWAGEN PASSAT GL SEDA	443.98	DEMOJPM	ANOTHER TEST	
2979	INSPECTION	10/10/2019	123 123		11 TOYOTA ARIUNER	14.22	DEMO	ANOTHER TEST	
2978	INSPECTION	10/10/2019	RICKY BOBBY	C: (410) 687-5309	2001 SATURN L200	645.05	GHL9219	DEMOJC	ANOTHER TEST
2977	INSPECTION	10/09/2019	JOHN BETA	C: (098) 765-4321	2009 FORD TAUROS SE AWD BLACK	90.30	DEMO	ANOTHER TEST	
2976	INSPECTION	10/09/2019	ROBOUITE GUILLMAN	C: (866) 555-1066	1965 VOLKSWAGEN BEETLE BLUE	14.22	DEMO	ANOTHER TEST	
2974	INSPECTION	10/09/2019	JOHN BETA	C: (098) 765-4321	2009 FORD TAUROS SE AWD BLACK	14.22	DEMO	ANOTHER TEST	
2973	INSPECTION	10/08/2019	ME		2006 VOLKSWAGEN NEW BEETLE 2.5	14.22	DEMOJD	ANOTHER TEST	
2972	INSPECTION	10/08/2019	KONRAD CURZE		2004 VOLKSWAGEN PASSAT GL SEDA	60.09	DEMO	ANOTHER TEST	
2971	INSPECTION	10/08/2019		H: (860) 829-7377	1993 OLDSMOBILE 88	14.22	DEMO	ANOTHER TEST	
2970	INSPECTION	10/08/2019	ROGAL DORN	C: (888) 555-1941	2006 AUDI A3 2.0T HATCHBACK 4	223.53	DEMOHM	ANOTHER TEST	
2968	INSPECTION	10/07/2019	CORVUS CORAX	H: (555) 555-5555	2004 VOLKSWAGEN PASSAT GL SEDA	182.22	DEMO	ANOTHER TEST	
2964	INSPECTION	10/05/2019		H: (860) 829-7377	2013 HYUNDAI ELANTRA GLS/UMIT	14.22	BJLM 182	DEMO	ANOTHER TEST
2963	INSPECTION	10/04/2019	BRIAN BARTLETT	H: (727) 417-3318	2012 GMC SIERRA 1500 SLT EXT.	211.78	AWBE24	DEMO	ANOTHER TEST
2962	INSPECTION	10/03/2019	FIRST TRY	H: (555) 555-555 Y	2004 ACURA RSX	310.42	DEMOJC	ANOTHER TEST	
2961	INSPECTION	10/03/2019	JOE MCNARSON	W: (917) 876-3211	2011 CHRYSLER TOWN & COUNTRY T	523.02	PLK-457U	DEMO	ANOTHER TEST
2959	INSPECTION	09/27/2019	JEFF CHAMBERS	C: (443) 000-0000	2003 FORD EXPLORER BLUE	42.48	3719CB8	DEMO	ANOTHER TEST
2958	INSPECTION	09/26/2019	SKYLINE		2003 FORD EXPLORER BLUE	738.16	DEMOJC	ANOTHER TEST	
2956	INSPECTION	09/25/2019	123			14.22	DEMO	ANOTHER TEST	
2955	INSPECTION	09/25/2019	CELIO ARIAS	H: (317) 395-8571	2013 HONDA CIVIC SI SEDAN 6-SP	194.12	DEMOJC	ANOTHER TEST	

Slika 2.2: ShopBoss

Uz ove aplikacije, postoje mnoge koje su usmjerene samo na vlasnike automobila, pružajući im jednostavno i efikasno praćenje stanja njihovih vozila, te one koje su usmjerene samo na auto-servise i pomažu im u organizaciji i provođenju servisa. Ova aplikacija ta će dva smjera spojiti u jedan, omogućavajući i vlasnicima automobila i auto-servisima da komuniciraju i organiziraju servise automobila.

Moguće nadogradnje projektnog zadatka

Jedna od mogućih nadogradnji koje bi učinile ovu web-stranicu pristupačnijom je prilagodba iste za korištenje na mobilnim uređajima. Ovo bi bilo izrazito pogodno za slučajeve kada je korisniku potreban izvandredni servis zbog kvara koji se dogodio tijekom vožnje, npr. nakon sudara. Web-stranica bi mogla biti dalje nadograđena da nudi uslugu vučne službe,

Još jedan potencijalni dodatak projektnom zadatku je mogućnost korisnika koji dodaje novi automobil da vidi podatke o svim obavljenim servisima nad tim automobilom dok nije bio u njegovom vlasništvu.

autopraonica? ponuda rezervnih dijelova i njihove ugradnje rating
hhhh

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
- 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

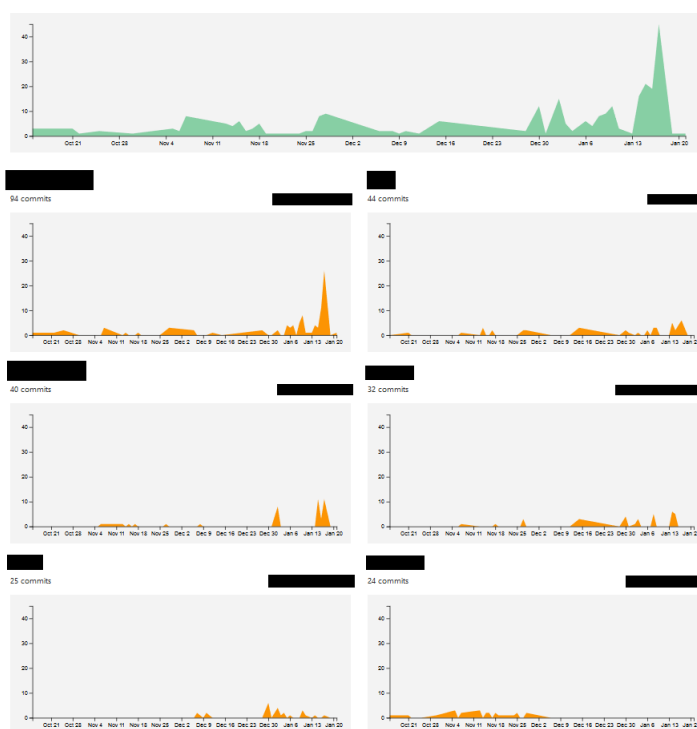
- primjer
 - primjer
 - primjer
1. primjer
 2. primjer

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

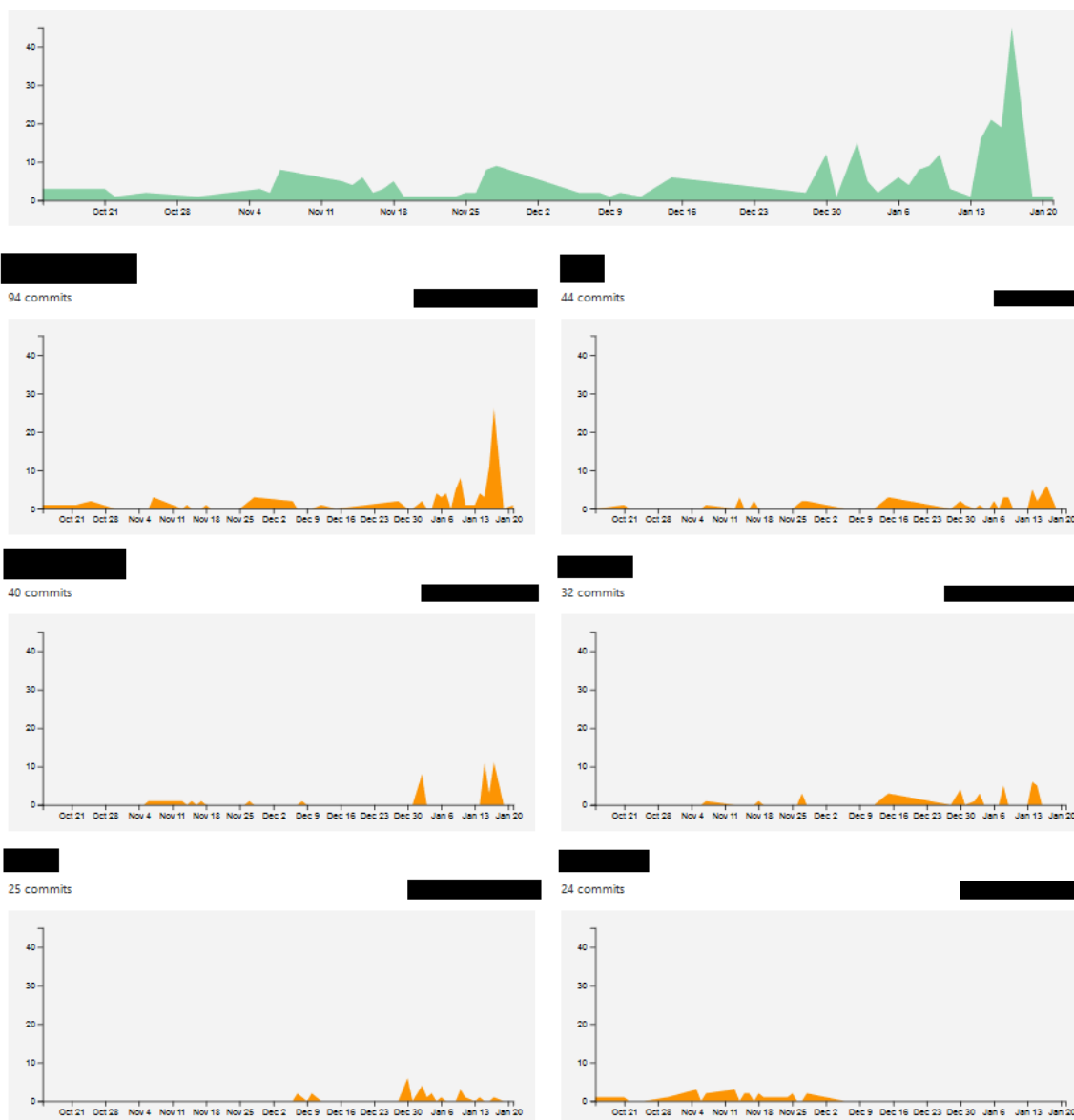
naslov unutar tablice		
IDKorisnik	INT	Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod
korisnickoIme	VARCHAR	
email	VARCHAR	
ime	VARCHAR	
primjer	VARCHAR	

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 2.3: Primjer slike s potpisom



Slika 2.4: Primjer slike s potpisom 2

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 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 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

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	GetAFix	6
2.2	ShopBoss	6
2.3	Primjer slike s potpisom	9
2.4	Primjer slike s potpisom 2	10

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

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>							

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