

Programsko inženjerstvo

Ak. god. 2020./2021.

Humanitarni šetaći pasa

Dokumentacija, Rev. 1

Grupa: *Cyfer*

Voditelj: *Jana Juroš*

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

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

Sadržaj

1	Dnevnik promjena dokumentacije	3
2	Opis projektnog zadatka	5
2.1	Motivacija i cilj	5
2.2	Postojeća slična rješenja	5
2.3	Opseg projektnog zadatka	7
2.4	Primjeri u \LaTeX u	9
3	Specifikacija programske potpore	12
3.1	Funkcionalni zahtjevi	12
3.1.1	Obrasci uporabe	13
3.1.2	Sekvencijski dijagrami	14
3.2	Ostali zahtjevi	15
4	Arhitektura i dizajn sustava	16
4.1	Baza podataka	16
4.1.1	Opis tablica	16
4.1.2	Dijagram baze podataka	17
4.2	Dijagram razreda	18
4.3	Dijagram stanja	19
4.4	Dijagram aktivnosti	20
4.5	Dijagram komponenti	21
5	Implementacija i korisničko sučelje	22
5.1	Korištene tehnologije i alati	22
5.2	Ispitivanje programskog rješenja	23
5.2.1	Ispitivanje komponenti	23
5.2.2	Ispitivanje sustava	23
5.3	Dijagram razmještaja	24
5.4	Upute za puštanje u pogon	25

6 Zaključak i budući rad	26
Popis literature	27
Indeks slika i dijagrama	28
Dodatak: Prikaz aktivnosti grupe	29

1. Dnevnik promjena dokumentacije

Kontinuirano osvježavanje

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napravljen predložak.	Juroš	18.10.2020.
0.2	Dodani opis projektnog zadatka	Juroš	23.10.2020.
0.5	Dodan <i>Use Case</i> dijagram i jedan sekvencijski dijagram, funkcionalni i nefunkcionalni zahtjevi i dodatak A	Ivošević	25.08.2013.
0.6	Arhitektura i dizajn sustava, algoritmi i strukture podataka	Grudenić	26.08.2013.
0.8	Povijest rada i trenutni status implementacije, Zaključci i plan daljnjeg rada	Ivošević	28.08.2013.
0.9	Opisi obrazaca uporabe	Jović	07.09.2013.
0.10	Preveden uvod	Jović	08.09.2013.
0.11	Sekvencijski dijagrami	Žužak	09.09.2013.
0.12.1	Započeo dijagrame razreda	Horvat	10.09.2013.
0.12.2	Nastavak dijagrama razreda	Horvat	11.09.2013.
1.0	Verzija samo s bitnim dijelovima za 1. ciklus	Ivošević	11.09.2013.
1.1	Uređivanje teksta – funkcionalni i nefunkcionalni zahtjevi	Grudenić Jović	14.09.2013.
1.2	Manje izmjene:Timer - Brojilo vremena	Grudenić	15.09.2013.
1.3	Popravljeni dijagrami obrazaca uporabe	Jović	15.09.2013.
1.5	Generalna revizija strukture dokumenta	Ivošević	19.09.2013.
1.5.1	Manja revizija (dijagram razmještaja)	Jović	20.09.2013.
2.0	Konačni tekst predložka dokumentacije	Ivošević	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

2.1 Motivacija i cilj

Cilj ovog projekta je stvoriti aplikaciju koja može na jednostavan i brz način spojiti građane i udruge za životinje kako bi se organizirale šetnje nezbrinutih pasa.

Naime, broj nezbrinutih životinja raste u Hrvatskoj te je procijenjeno da ima 10 000 napuštenih životinja. Skloništa i udruge za nezbrinute životinje spašavaju ranjene i nezbrinute životinje, te žele potaknuti građane na angažiranost, pomoć pri brizi za životinje i za njihove udomljavanje. Šetnja je psima primarna potreba, a skloništa nemaju dovoljno resursa za hranu, a kamoli za plaćanje šetnji. Pretpostavljamo da bi se ljudi rado uključili u šetanje pasa kad bi postojao način, pogotovo ako planiraju usvojiti jednoga. Naša aplikacija im to odsad može i omogućiti.

2.2 Postojeća slična rješenja

Što se tiče postojećih rješenja iste tematike, ne postoji previše sličnosti s našim projektom. Postoje 3 web-stranice koje mogu spojiti šetače s vlasnicima pasa, no vlasnici plaćaju za te usluge te udruge nisu uključene. Prve dvije funkcioniraju na temelju oglasa - korisnik stavi oglas sa slikom svog kućnog ljubimca, vrijeme kad mu je potrebna usluga te koliko će platiti. Te stranice su:

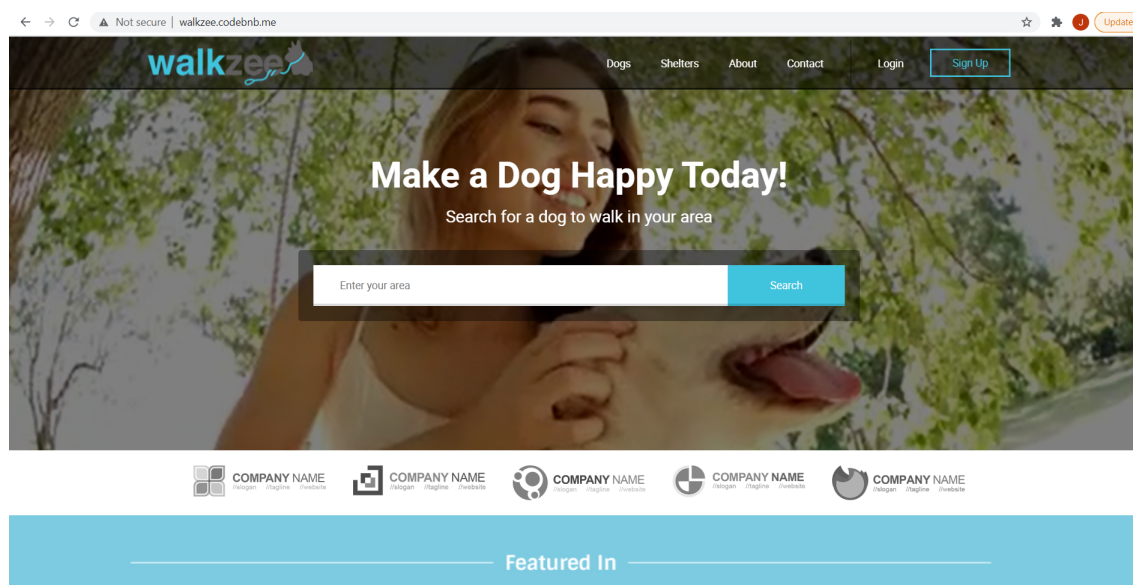
Njuškalo: – najveći online oglasnik u Hrvatskoj

Čuvalica: – Nacionalni portal za brigu o obitelji – briga za djecu, starije, ljubimce

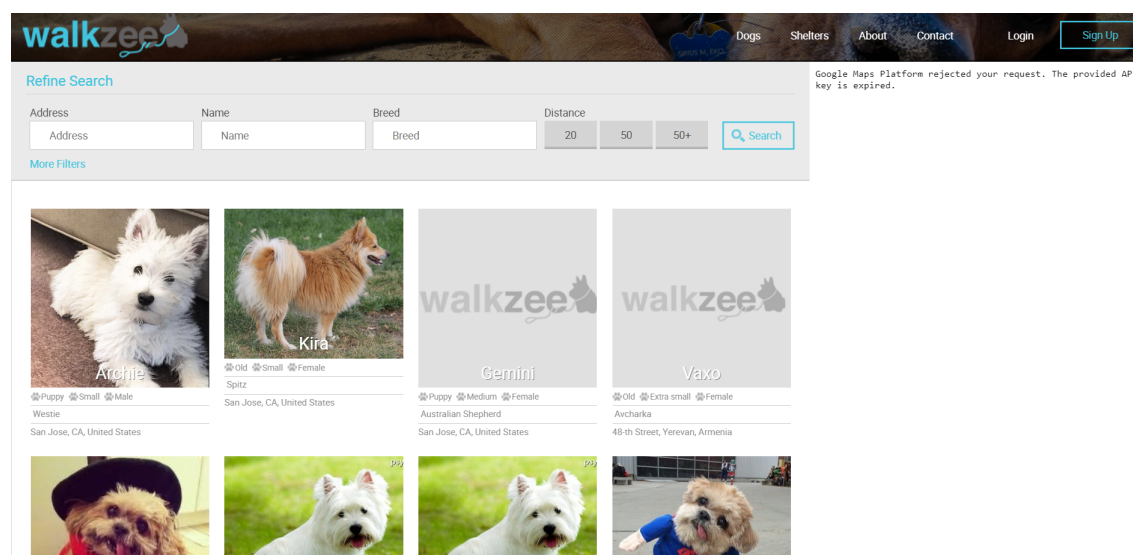
Također, postoji web-stranica obrta **PETS STEP** koji nudi uslugu čuvanja i šetnje pasa. Na stranici imaju kontakt i cjenik.

Kao što možemo vidjeti, u Hrvatskoj ne postoji usluga slična našoj, no u svijetu ih možemo pronaći. Najbližnji primjer bi bila web platforma **Walkzee** – “*1st free online platform connecting shelter dogs in need of a walk to dog lovers looking for a walking buddy!*” Walkzee su stvorili Cristina i Charlie Saunders 2015. te je ideja

identična kao ideja iza našeg projekta. Nažalost, njihova stranica nije zaživjela te nema mnogo udruga niti pasa na stranici. Naime, ima samo 10 stranica te 14 pasa. Osim toga, stranica izgleda nedovršeno pa ne znamo je li ikada uopće funkcionirala. Nadamo se da njihov neuspjeh leži u lošoj egzekuciji, a ne u lošoj ideji odnosno indiferentnosti javnosti prema napuštenim životinjama.



Slika 2.1: Walkzee – naslovna stranica



Slika 2.2: Walkzee – pregled pasa

2.3 Opseg projektnog zadatka

Aplikaciju će koristiti registrirane udruge za životinje, registrirani građani i javni posjetitelji koji nisu registrirani i imaju mogućnost pristupa naslovnoj stranici aplikacije i detaljima profila udruge. Udruge i građani se mogu registrirati, pri čemu građani sljedeće podatke:

- ime
- prezime
- adresa e-pošte

a udruge još dodatno:

- naziv udruge
- OIB udruge

Glavni zadatak aplikacije je povezati udruge za životinje s građanima koji imaju želju i vrijeme za šetanje pasa, te time povećati izgled udovoljavanja pasa i psihološkog efekta dobrobiti socijalizacije za psa i za čovjeka. Javni posjetitelj može doći u aplikaciju i pregledati sve udruge na naslovnoj stranici, zatim otići na detalje profila pojedine udruge, a osim toga ima i uvid u rang listu registriranih šetača. Rang lista prikazuje poredak šetača s obzirom na broj šetnji, broj pasa, te duljinu šetnje koju su odradili u prethodnih mjesec dana.

Svaka registrirana udruga može kreirati vlastiti profil koji će se prikazivati na javnoj stranici. U sklopu profila udruge neke od bitnih informacija su:

- slike i opisi pasa
- raspoloživost pasa za određeni vremenski period (datum i vrijeme)
- lokacija gdje se nalaze
- za kakve šetnje su psi predodređeni (skupne ili individualne šetnje).

Svi kreirani detalji profila udruge vidljivi su na javnoj aplikaciji. Posjetitelj dolaskom na naslovnu stranicu može odabrati detalje profila udruge i dobiti uvid u profile pasa, statistike o šetanjima svih pasa, lokaciju, te mogućnost prijave za

šetanje pasa. Statistika o šetnjama pruža informacije koji psi su češće bili u šetnji od ostalih, te time koji psi imaju veću potrebu za šetnjom.

Ukoliko posjetitelj odluči pripomoći udruzi i priključiti se šetnji pasa, ima opciju registracije. Nakon toga građanin može odabrati psa/e, odabrati željeni termin šetnje i prijaviti se za šetača. Termin šetnje se odabire u obliku datuma i vremena. Nakon uspješnog „rezerviranja“ psa za šetnju, termin za odabranog psa vidljiv je na kalendaru registriranim građanima. Građani imaju opciju skinuti raspored za odabrani dan, tjedan ili mjesec u PDF obliku.

Građanin ima opciju prijave u vlastiti profil i pregleda vlastitih rasporeda šetanja, vlastitih statistika šetnji, zajedno s mogućnošću označavanja statistike šetanja kao javnih; kako bi podaci građana dospjeli na rang listu na javnoj stranici. Svi registrirani korisnici imaju mogućnost mijenjanja podataka u svom profilu.

- *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.*
- *mogućnost prilagodbe rješenja*
- *opseg projektnog zadatka*
- *moguće 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.

2.4 Primjeri u L^AT_EXu

Ovo potpoglavlje izbrisati.

U nastavku se nalaze različiti primjeri kako koristiti osnovne funkcionalnosti L^AT_EXa 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^AT_EXu - https://www.fer.unizg.hr/_download/repository/LaTeX-upute.pdf
- L^AT_EX 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

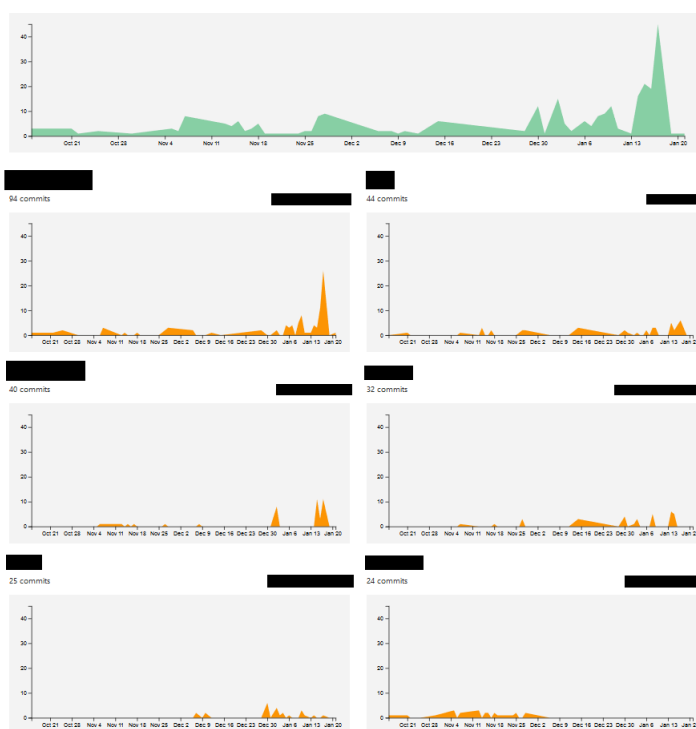
primjer url-a: <https://www.fer.unizg.hr/predmet/proinz/projekt>

posebni znakovi: # \$ % & { } _ | < > ^ ~ \

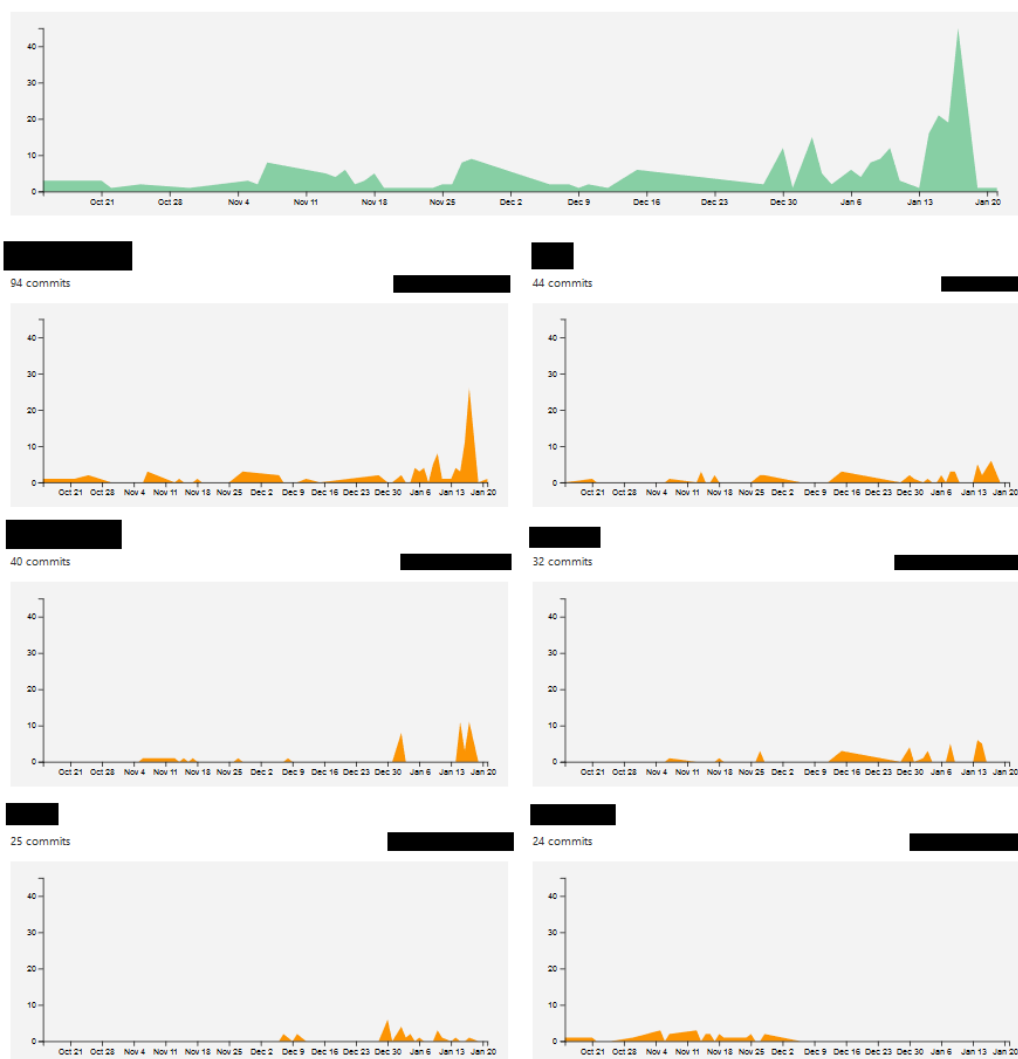
naslov unutar tablice		
IDKorisnik	INT	Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod
korisnickoIme	VARCHAR	
email	VARCHAR	
ime	VARCHAR	
primjer	VARCHAR	

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

Referenciranje slike 2.4 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 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

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	Walkzee – naslovna stranica	6
2.2	Walkzee – pregled pasa	6
2.3	Primjer slike s potpisom	10
2.4	Primjer slike s potpisom 2	11

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: 29. listopada 2020.
- Prisustvovali: I.Prezime, I.Prezime
- Teme sastanka:
 - opis prve teme
 - opis druge teme

2. sastanak

- Datum: u ovom formatu: 29. listopada 2020.
- 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							
Upute za puštanje u pogon							
Dnevnik sastajanja							
Zaključak i budući rad							
Popis literature							

	Ime Prezime voditelja	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime
<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.