

Programsko inženjerstvo

Ak. god. 2020./2021.

Digitalizacija

Dokumentacija, Rev. <1 ili 2>

Grupa: *Kompletići*

Voditelj: *Martin Ante Rogošić*

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 Potencijalna korist ovog projekta	7
2.2 Slična rješenja	7
2.3 Mogućnost prilagodbe	7
2.4 Opseg projektnog zadatka	8
2.5 Moguće nadogradnje	8
2.6 Primjeri u \LaTeX u	9
3 Specifikacija programske potpore	12
3.1 Funkcionalni zahtjevi	12
3.1.1 Obrasci uporabe	14
3.1.2 Sekvencijski dijagrami	15
3.2 Ostali zahtjevi	16
4 Arhitektura i dizajn sustava	17
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
5 Implementacija i korisničko sučelje	23
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
6 Zaključak i budući rad	27
Popis literature	28
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.	*	1.11.2023.
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.
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 razmjesta)	*	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

Cilj ovog projekta je razvoj i implementacija aplikacije Digitaliziraj, koja će tvrtkama olakšati digitalizaciju dokumenata nužnih za poslovanje koristeći OCD (eng. Optical character recognition). Aplikacija će olakšati digitalizaciju dokumenata unutar organizacija, te osigurati da svaki radnik organizacije dobije dokumente za koje je zadužen. Time, aplikacija će povećati učinkovitost organizacije te ubrzati njeno djelovanje.

Neregistrirani korisnik mora napraviti korisnički račun kako bi mogao koristiti funkcionalnosti aplikacije. Za to su mu potrebni sljedeći podatci:

- *ime*
- *prezime*
- *email adresa*
- *lozinka*

Dodatno, ne registrirani korisnik mora odabrati na kojoj razini ovlasti želi stvoriti korisnički račun. Dostupne su 4 razine ovlasti:

- *zaposlenik*
- *revizor*
- *računovođa*

Razne razine iznad su navedene od najniže do najviše. U većini slučajeva viša razina ovlasti ima sve funkcionalnosti onih ispod sebe, osim u onim slučajevima kada struktura tvrtke te razne odgovornosti zaposlenika to traže drugačije. Detalji o funkcionalnostima svake razine ovlasti nalaze se u nastavku dokumenta. Dodatno postoji još razina ovlasti direktor, koja predstavlja razinu ovlasti vlasnika tvrtke. Aplikacija sama stvara jedan korisnički račun s tom razinom ovlasti te su podatci potrebni za prijavu u taj korisnički račun dani vlasniku tvrtke.

Zaposlenik ima najnižu razinu ovlasti unutar sustava. Korisnik s razinom ovlasti zaposlenik može učitavati slike u sustav te provjeriti jeli sustav točno odradio

konverziju slike u dokument. Jednom kad zaposlenik potvrdi da je pretvorba iz slike u dokument odrađena ispravno, generiran dokument se šalje jednom od revizora tvrtke na pregled. U svrhu ubrzanja procesa aplikacija omogućuje učitavanje do 50 slika istovremeno (ova funkcionalnost omogućena je i za sve više razine ovlasti). Pri tom zaposlenik i dalje mora svaku od konverzija potvrditi kao točnu prije nego što je dokument proslijeđen revizoru na pregled. Zaposlenik ima pristup povijesti svih dokumenata koje je skenirao.

Revizor je druga najniža razina ovlasti unutar sustava. Posao revizora jest kontrola rada zaposlenika te preusmjeravanje dokumenata odgovarajućem računovođi (ovisno o vrsti dokumenta). Kao takav revizor ima pristup povijesti svih dokumenata nad kojima je vršio kontrolu. U slučaju da revizor skenira dokument aplikacija će automatski detektirati računovođu kojem vrsta skeniranog dokumenta treba biti poslana, te će nakon što revizor potvrdi ispravnost konverzije taj dokument biti automatski poslan odgovarajućem računovođi.

Računovođa je najviša razina ovlasti u sustavu, izuzev direktora. Svaki računovođa je zadužen za obradu jedne od 3 vrste dokumenata koje aplikacija raspoznaje. To su računi, ponude i interni dokumenti. Računovođa ima pristup povijesti svih dokumenata one vrste za koju je zadužen. Posao računovođe je arhivirati dokumente. Pri arhiviranju dokumentu se dodjeljuje jedinstven broj arhiva. Računovođa također po potrebi može poslati dokument direktoru na potpisivanje. U tom slučaju direktor prima obavijest o tome. Nakon što direktor potpiše dokument računovođa dobiva obavijest da je dokument potpisan te ga onda može arhivirati. U slučaju da računovođa učitava sliku dokumenta one vrste za koju nije zadužen aplikacija će mu automatski ponuditi slanje dokumenta odgovarajućem računovođi.

Direktor je najviša razina ovlasti unutar sustava, te predstavlja najodgovorniju osobu unutar organizacije koja koristi aplikaciju za svoje poslovanje. Direktor ima uvid u povijest svih dokumenata te u statistike svih zaposlenika. U slučaju da mu računovođa pošalje zahtjev za potpis može potpisati dokument. U slučaju da direktor sam učitava slike u sustav te vrši konverziju, odmah mu se nudi mogućnost potpisivanja dokumenta te aplikacija automatski određuje računovođu kojem treba proslijediti učitani dokument. Dodatno za promotivne svrhe, direktoru je omogućena objava dokumenata na sljedećim društvenim mrežama: (???treba dodati koje društvene

mreže jednom kad ustanovimo koje ćemo društvene vježbe koristiti)

2.1 Potencijalna korist ovog projekta

Svojom strukturom i funkcionalnošću aplikacija ostvaruje brz i učinkovit sustav digitalizacije i distribucije dokumenata unutar organizacije. Njenom primjenom može se osigurati učinkovito poslovanje te raspodjela odgovornosti među zaposlenicima što pospješuje rad organizacije u mnogim aspektima. S obzirom na njenu općenitost aplikacija bi mogla biti od interesa svim organizacijama koje se moraju baviti papirologijom. To uključuje: neprofitne humanitarne organizacije, tvrtke koje se natječu na tržištu, vladine agencije...

2.2 Slična rješenja

Na tržištu postoje razni sustavi za distribuciju dokumenata (???), također postoje i mnoge implementacije OCR-a.(???) No vrlo je malen broj alata koji integriraju te dvije tehnologije u jedan sustav. Time se zaobilazi potreba da se odvojeni alati koriste za digitalizaciju dokumenata i njihovu distribuciju. (Treba dodati imena i slike koje se traže)

2.3 Mogućnost prilagodbe

Implementacija sustava u velikoj mjeri ovisi o strukturi organizacije i njenim potrebama za distribuciju dokumenata. Dodatno ako je klasifikacija dokumenata koju organizacija koristi složena, potrebno je doraditi sustav automatskog prosljeđivanja dokumenata. Međutim implementacija dijelova sustava, koji se ne moraju mijenjati ovisno o potrebi klijenta, mogu se izvesti na način da je ta promjena relativno jednostavna.

2.4 Opseg projektnog zadatka

Konkretna implementacija izvedena u sklopu ovog projekta koristi relativno jednostavnu strukturu organizacije opisanu iznad, te razlikuje 4 različite vrste dokumenata. To je čini relativno jednostavnom za implementaciju no ujedno.

2.5 Moguće nadogradnje

Mnoge nadogradnje su moguće na sustav. Integracija s alatima za preuređivanje digitalnih dokumenata...

- *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.

2.6 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	

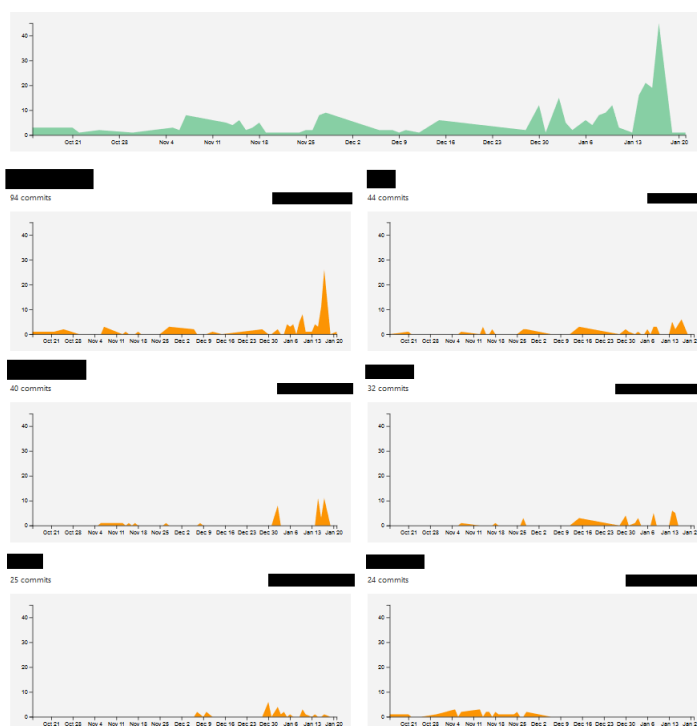
Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

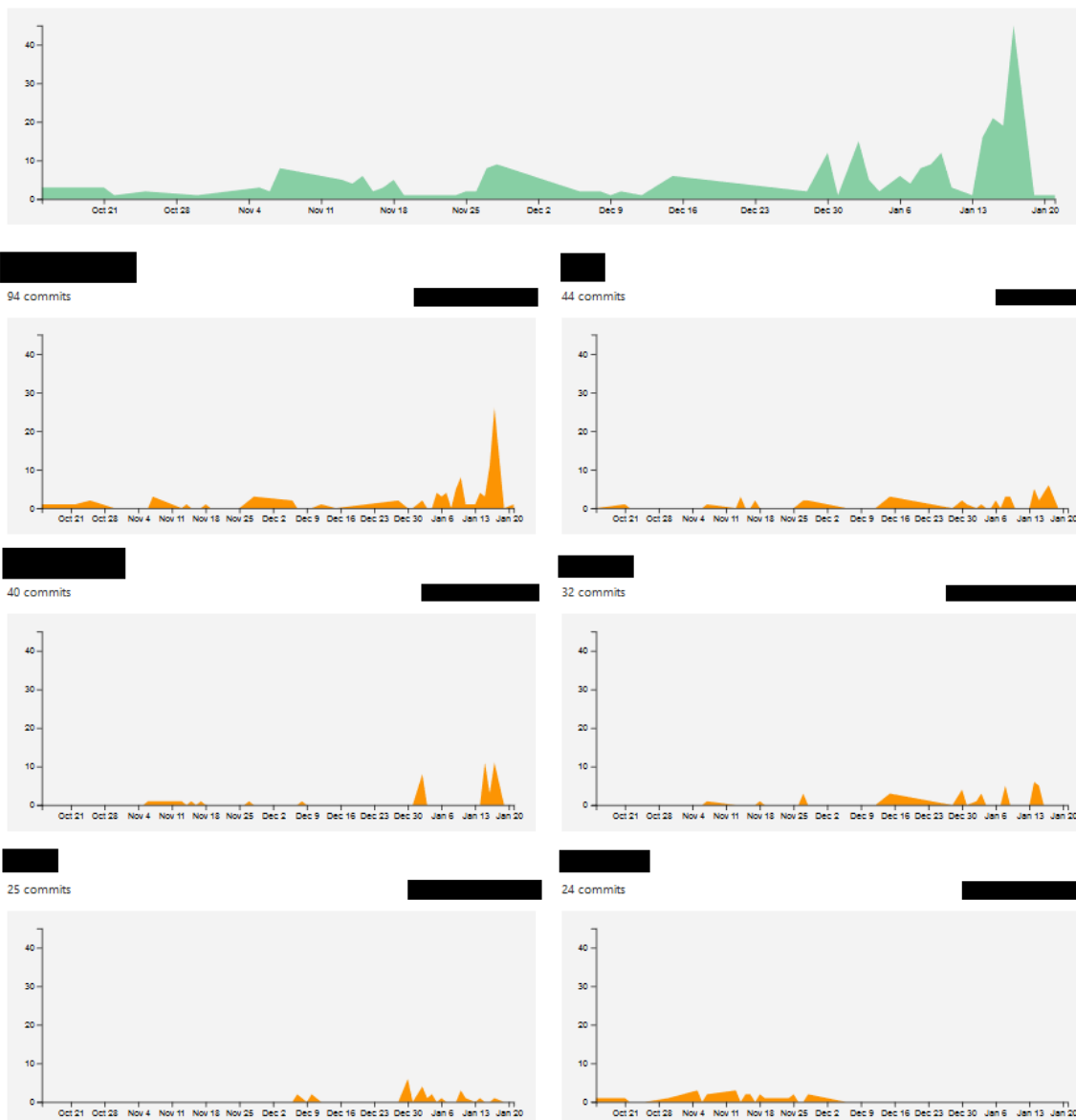
naslov unutar tablice		
ime	VARCHAR	
primjer	VARCHAR	

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 2.1: Primjer slike s potpisom



Slika 2.2: Primjer slike s potpisom 2

Referenciranje slike 2.2 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. Zaposlenici
2. Revizori
3. Računovođe
4. Direktor
5. Razvojni tim

Aktori i njihovi funkcionalni zahtjevi:

1. Zaposlenik (inicijator/sudionik) može:
 - (a) Prijaviti se u sustav
 - (b) Učitati fotografiju u sustav te izvršiti konverziju te slike u dokument
 - (c) Učitati više fotografija od jednom u sustav te izvršiti njihovu konverziju u dokumente od jednom
 - (d) Potvrditi ili odbiti točnost konverzije dokumenata u sustav
 - (e) Poslati dokument revizoru na pregled
 - (f) Pregledavati povijest svih dokumenata koje je zaposlenik unio u sustav
2. Revizor (inicijator/sudionik) može:

- (a) Provjeriti valjanost dokumenta kojeg mu je poslao zaposlenik
- (b) Učitati fotografiju u sustav te izvršiti konverziju te slike u dokument
- (c) Učitati više fotografija od jednom u sustav te izvršiti njihovu konverziju u dokumente od jednom
- (d) Potvrditi ili odbiti točnost konverzije dokumenata učitanih u sustav
- (e) *Aplikacija sama određuje kategoriju dokumenta no revizor to može ispraviti*
- (f) Proslijediti dokument računovođi odgovornom za vrstu dokumenta kojoj dokument pripada
- (g) Pregledati povijest svih dokumenata koje je revizor provjerio ili unio u sustav

3. Računovođa (inicijator/sudionik) može:

- (a) Računovođa može učitati jednu ili više slika u sustav te izvršiti njihovu konverziju u dokumente
- (b) *Automatsko prevođenje koje treba nekako pojasniti*
- (c) Poslati dokumente direktoru na potpisivanje
- (d) Arhivirati dokumente
- (e) Pregledati povijest svih dokumenata one vrste za koju je računovođa zadužen

4. Direktor (inicijator/sudionik) može:

- (a) Potpisati dokument kojeg mu je poslao računovođa na potpisivanje
- (b) Učitati jednu ili više fotografija u sustav te izvršiti njihovu konverziju u dokumente
- (c) Potvrditi ili odbiti točnost konverzije dokumenata unesenih u sustav
- (d) *Automatizacija detekcije kategorije*
- (e) Proslijediti dokument računovođi odgovornom za vrstu dokumenta kojoj dokument pripada

3.1.1 Obrasci uporabe

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	Primjer slike s potpisom	10
2.2	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: 9. studenoga 2023.
- Prisustvovali: I.Prezime, I.Prezime
- Teme sastanka:
 - opis prve teme
 - opis druge teme

2. sastanak

- Datum: u ovom formatu: 9. 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.