

Programsko inženjerstvo

Ak. god. 2022./2023.

Company Database

Dokumentacija, Rev. <1 ili 2>

Grupa: *EkipaZaOcevid*

Voditelj: *Petar Hajduk*

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

Nastavnik: *Manuela Lukić*

Sadržaj

1. Dnevnik promjena dokumentacije

Kontinuirano osvježavanje

| Rev. | Opis promjene/dodatka | Autori | Datum |
|--------|--|--------------|-------------|
| 0.1 | Napravljen predložak. | * | 22.08.2013. |
| 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. |
| | Izradio database sekciju | Nikola Capan | 12.11.2022. |
| | Izradio dijagrame razreda | Nikola Capan | 17.11.2022 |
| 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. |

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

| Rev. | Opis promjene/dodatka | Autori | Datum |
|-------|--|--------|-------------|
| 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. |
| 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

dio 1. revizije

Na osnovi projektnog zadatka detaljno opisati korisničke zahtjeve. Što jasnije opisati cilj projektnog zadatka, razraditi problematiku zadatka, dodati nove aspekte problema i potencijalnih rješenja. Očekuje se minimalno 3, a poželjno 4-5 stranica opisa. Teme koje treba dodatno razraditi u ovom poglavlju su:

- *potencijalna korist ovog projekta*
- *postojeća slična rješenja (istražiti i ukratko opisati razlike u odnosu na zadani zadatak). Dodajte slike koja predočavaju slična rješenja.*
- *skup korisnika koji bi mogao biti zainteresiran za ostvareno rješenje.*
- *moгуćnost prilagodbe rješenja*
- *opseg projektnog zadatka*
- *moгуće nadogradnje projektnog zadatka*

Za pomoć pogledati reference navedene u poglavlju „Popis literature“, a po potrebi konzultirati sadržaj na internetu koji nudi dobre smjernice u tom pogledu.

2.1 Primjeri u 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 | |

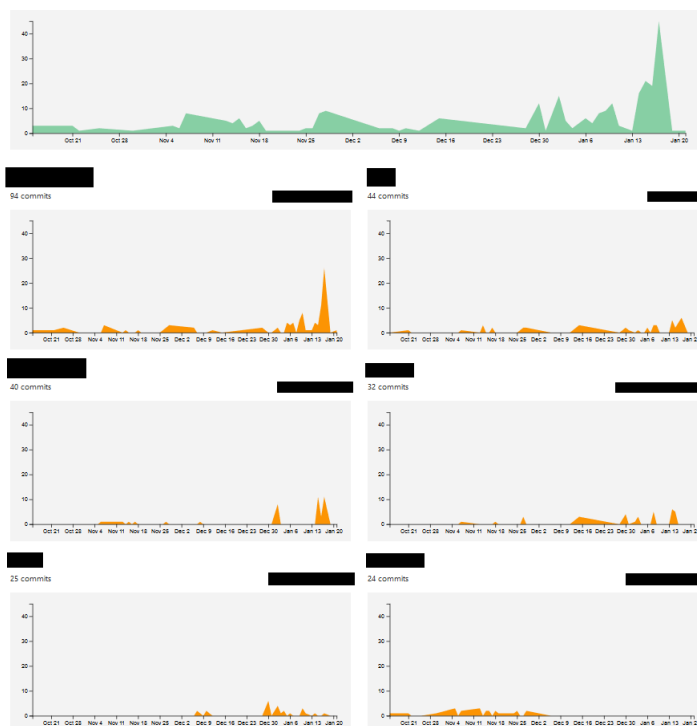
Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

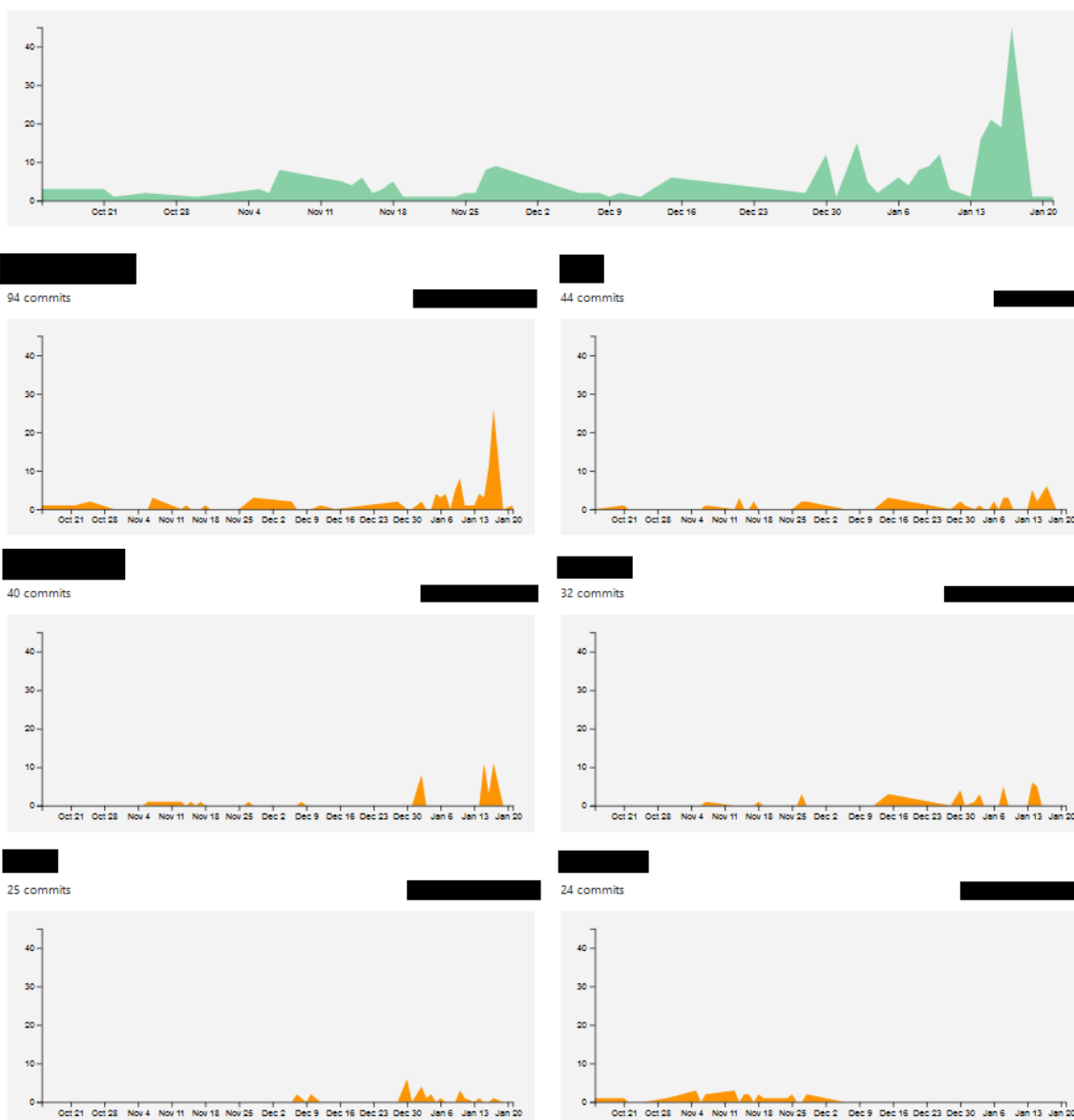
| naslov unutar tablice | | |
|--------------------------|---------|--|
| email | VARCHAR | |
| ime | VARCHAR | |
| primjer | VARCHAR | |

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 ?? 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.

Arhitektura i dizajn sustava

Arhitekturu možemo podijeliti na 3 podsustava:

- Web poslužitelj
- Web aplikacija
- Baza podataka

Web preglednik (*Internetski preglednik, Web browser, Internet browser*) je program koji korisniku omogućuje pregled web-stranica i multimedijalnih sadržaja vezanih uz njih. Svaki internetski preglednik je interpreter (program koji u realnom vremenu izvršava izvorni kod napisan u nekom programskom jeziku, umjesto da ga, prije izvršavanja cijelog prevede u strojni jezik, što inače radi jezični prevoditelj). Dakle, stranica je pisana u kodu koji preglednik interpretira u čitljiv sadržaj. Korisnici putem web preglednika šalju zahtjeve Web poslužitelju.

Web poslužitelj je temelj svake Web aplikacije jer on služi za komunikaciju klijenta s aplikacijom. Takva vrsta komunikacije odvija se uz pomoć HTTP (*Hyper Text Transfer Protocol*) protokola koji predstavlja glavnu i najčešću metodu prijenosa informacija na Webu. Osnovna namjena ovog protokola je omogućavanje objavljivanja i prezentacije HTML dokumenata, tj. Web stranica. Web poslužitelj nam dakle služi za prosljeđivanje zahtjeva Web aplikaciji te je zadužen za njezino pokretanje.

Od ostalih protokola koji služe za komunikaciju i prijenos informacija i podataka između klijenta i poslužitelja treba napomenuti: IP (*Internet Protocol*) koji služi za prijenos podataka u blokovima (paketi, datagrami), komunikacijski protokol ARP (*Address Resolution Protocol*) te ICMP (*Internet Control Message Protocol*) koji je ugrađen u svaki IP modul kako bi omogućio računalima slanje kontrolnih poruka o greškama. ICMP prijavljuje greške, ali nije zadužen za njihovo ispravljanje.

Klijent koristi Web aplikaciju za obradu traženih zahtjeva. Aplikacija vrši obradu zahtjeva te, u slučaju pozitivnog ishoda obrade, pristupa bazi podataka. Nakon toga preko Web poslužitelja vraća korisniku odgovor koji mu je razumljiv i vidljiv u Web pregledniku.

Programski i korisnički jezici koje smo koristili su: HTML, CSS i JavaScript za Frontend te Java za Backend. Radni okviri koje smo odabrali su React (Frontend)

i Spring Boot (Backend). Od razvojnih okruženja koristili smo IntelliJ u okviru izrade Backenda te Visual Studio Code za izradu Frontenda.

U okviru Backenda korišten je REST (*Representational State Transfer*) API (*Application programming Interface*). REST je softverska arhitektura koja nameće uvjete o tome kako API treba raditi, a API definira pravila koja morate slijediti za komunikaciju s drugim softverskim sustavima.

REST API je sučelje koje dva računalna sustava koriste za sigurnu razmjenu informacija putem interneta. Većina Web aplikacija mora komunicirati s drugim internim aplikacijama i aplikacijama trećih strana kako bi izvršile razne zadatke, a REST API podržava ovu razmjenu informacija jer slijedi sigurne, pouzdane i učinkovite softverske komunikacijske standarde.

3.3 Baza podataka

dio 1. revizije

Podaci u našoj aplikaciji će se spremati u relacijsku bazu podataka koristeći Postgres kao DBMS. Baza će se sastojati od sljedećih entiteta:

- Korisnik
- Tvrtka
- Projekt
- ProjectFRTeamMembers
- Mjesto
- Zaposlenik
- Suradnja

3.3.1 Opis tablica

| Korisnik | | |
|-------------------|---------|--|
| IDKorisnik | INT | ID korisnika |
| Ime | VARCHAR | Ime korisnika |
| Prezime | VARCHAR | Prezime korisnika |
| Nadimak | VARCHAR | Nadimak korisnika |
| LoginEmail | VARCHAR | Email adresa pomoću kojeg se user logira |
| NotificationEmail | VARCHAR | Email adresa na koju korisnik prima obavijesti |
| MaxRazinaOvlasti | VARCHAR | Maksimalna razina ovlasti na projektima |

| Mjesto | | |
|-------------------|---------|----------------------|
| PostanskiBrojGrad | VARCHAR | Poštanski broj grada |
| NazivGrad | VARCHAR | Naziv grada |

| Tvrtka | | |
|-------------------------|---------|---|
| IdTvrtka | INT | Id tvrtke |
| Naziv | VARCHAR | Naziv tvrtke |
| Podrucje | VARCHAR | Područje kojim se tvrtka bavi |
| ABCKategorija | CHAR | Kategorija tvrtke (A B ili C) |
| MjesecPlaniranjaBudzeta | INT | Mjesec u kojem tvrtka planira budžet |
| Drzava | VARCHAR | Država u kojoj tvrtka posluje |
| PBr | VARCHAR | Poštanski broj mjesta tvrtke |
| Adresa | VARCHAR | Adresa tvrtke |
| WebStranica | VARCHAR | Adresa web stranice tvrtke |
| Kontaktirati | BOOLEAN | Treba li ubuduće tu tvrtku treba kontaktirati |

| Zaposlenik | | |
|--------------|---------|---|
| IDZaposlenik | INT | Id zaposlenika |
| IDTvrtke | INT | Id tvrtke za koju zaposlenik radi |
| Ime | VARCHAR | Ime zaposlenika |
| Prezime | VARCHAR | Prezime zaposlenika |
| Email | VARCHAR | Email adresa zaposlenika |
| BrojTelefona | VARCHAR | Broj telefona zaposlenika |
| Uloga | VARCHAR | Pozicija zaposlenika u tvrtci (npr. PR) |

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

| Zaposlenik | | |
|-------------------|---------|--|
| Opis | VARCHAR | Opis zaposlenika (npr. glavni kontakt) |

| Projekt | | |
|---------------------|-----------|---|
| IDProjekt | INT | Id projekta |
| Naziv | VARCHAR | Naziv projekta |
| KategorijaProjekta | INT | Kategorija u koju spada projekt |
| TipProjekta | INT | Eksterni ili interni projekt |
| DatumPočetka | TIMESTAMP | Datum početka projekta |
| DatumZavršetka | TIMESTAMP | Datum završetka projekta |
| FRResponsibleUserId | INT | Id korisnika odgovornog za odnose s tvrtkama projekta |
| FRCilj | INT | Količina novca koja se želi prikupiti za projekt |
| FirstPing | TIMESTAMP | Datum prvog "ping"-a |
| SecondPing | TIMESTAMP | Datum drugog "ping"-a |

| ProjectFRTeamMembers | | |
|-----------------------------|-----|--|
| IDKorisnik | INT | Id korisnika koji je FR team member projekta |
| IDProjekt | INT | Id projekta |

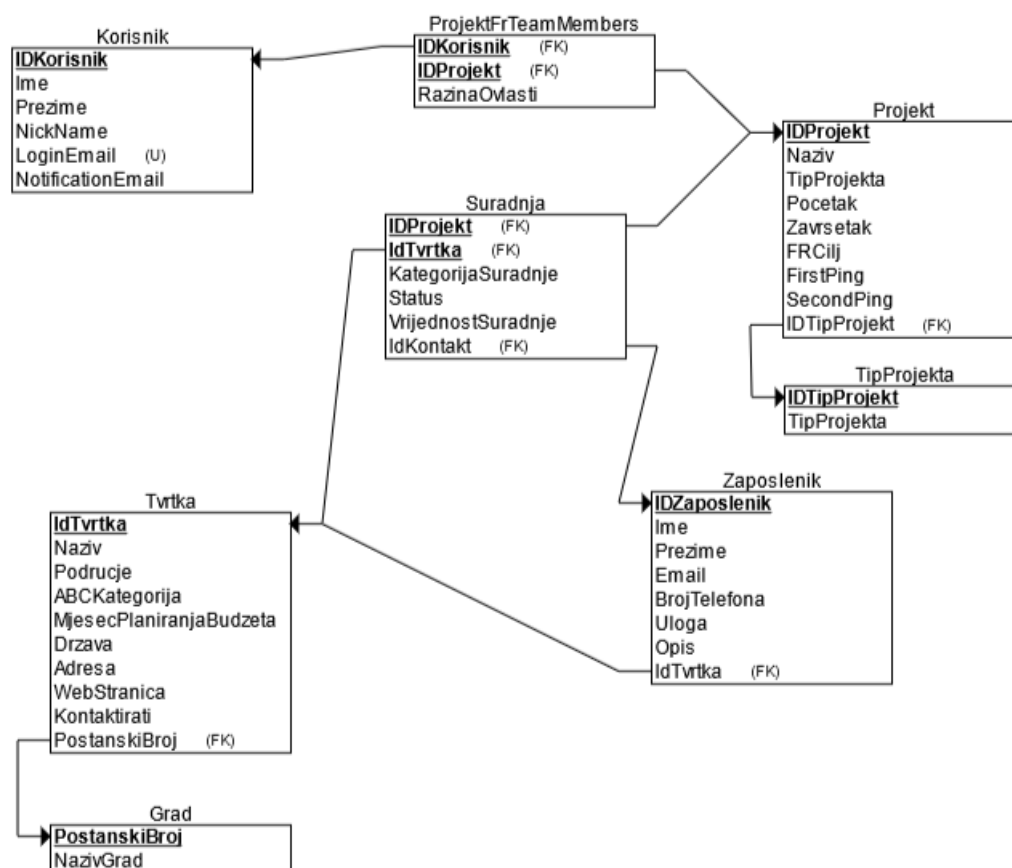
| Suradnja | | |
|-----------------|-----|-------------|
| IDProjekt | INT | Id projekta |

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

| Suradnja | | |
|--------------------|---------|--|
| IdTvrtke | INT | Id tvrtke |
| IdOdgovoran | INT | Id korisnika odgovornog za suradnju s tvrtkom |
| IdKontakt | INT | Id kontakt osobe u tvrtci |
| KategorijaSuradnje | VARCHAR | Kategorija suradnje (Financijska, materijalna ili akademska) |
| Status | VARCHAR | Kontaktirano, ping, dopis, sastanak, uspješno ili neuspješno |
| Komentar | VARCHAR | Komentar na suradnju |
| VrijednostSuradnje | FLOAT | Novčana vrijednost suradnje |

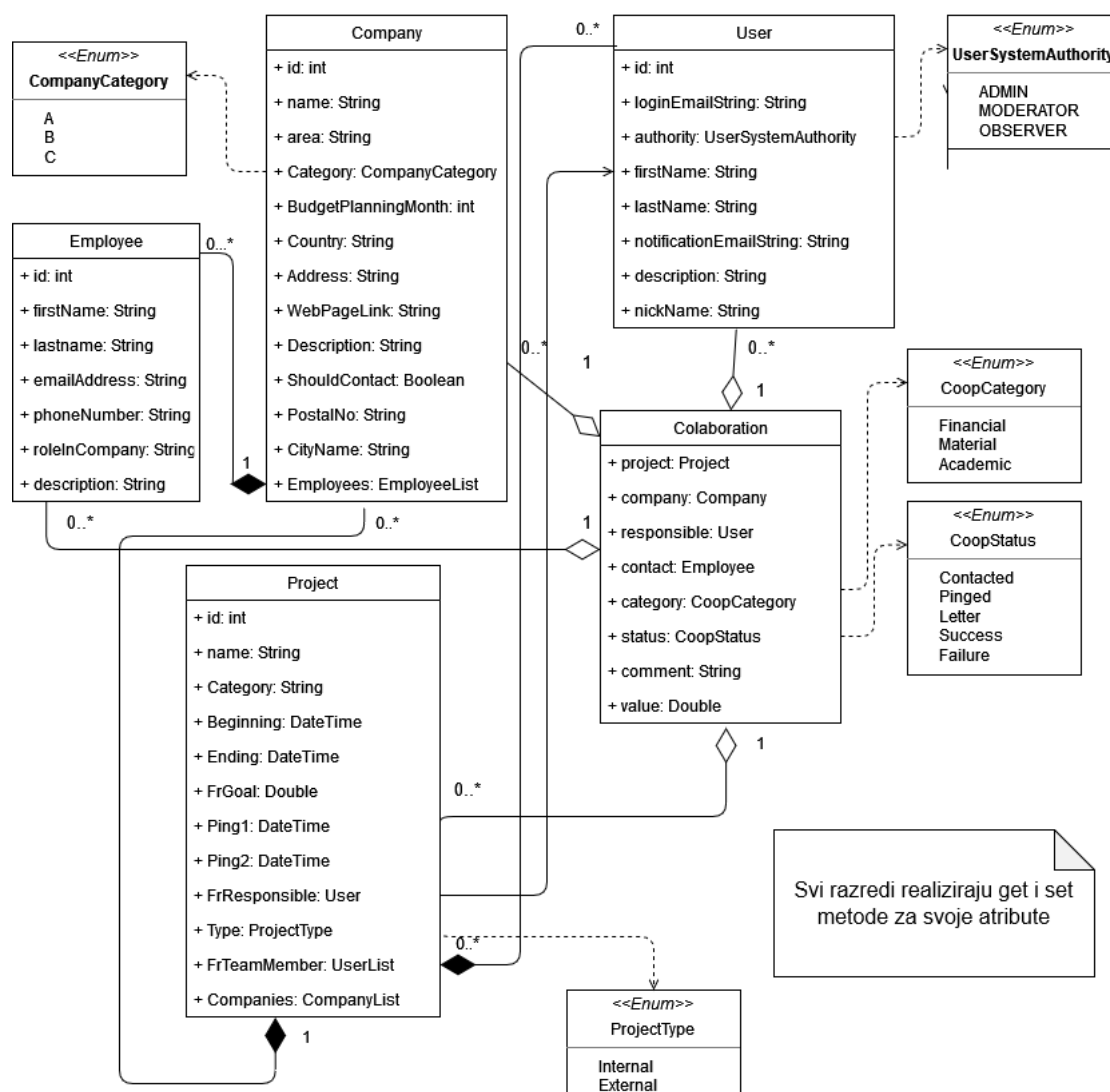
3.3.2 Dijagram baze podataka



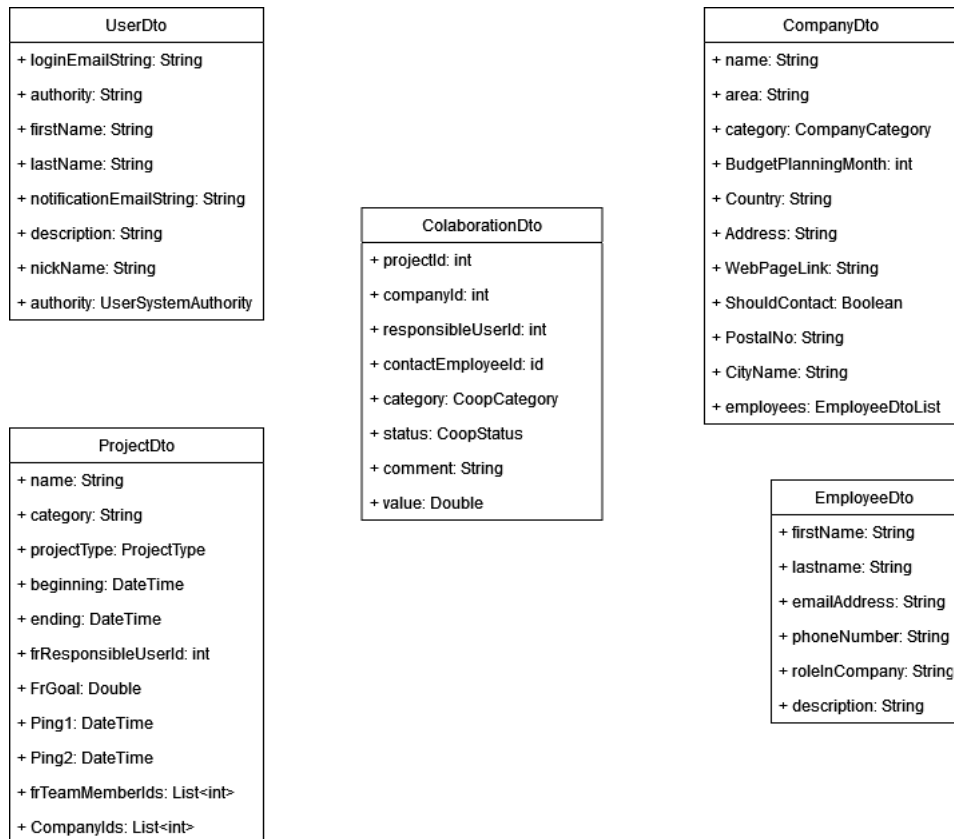
Slika 3.1: Database Diagram

3.4 Dijagram razreda

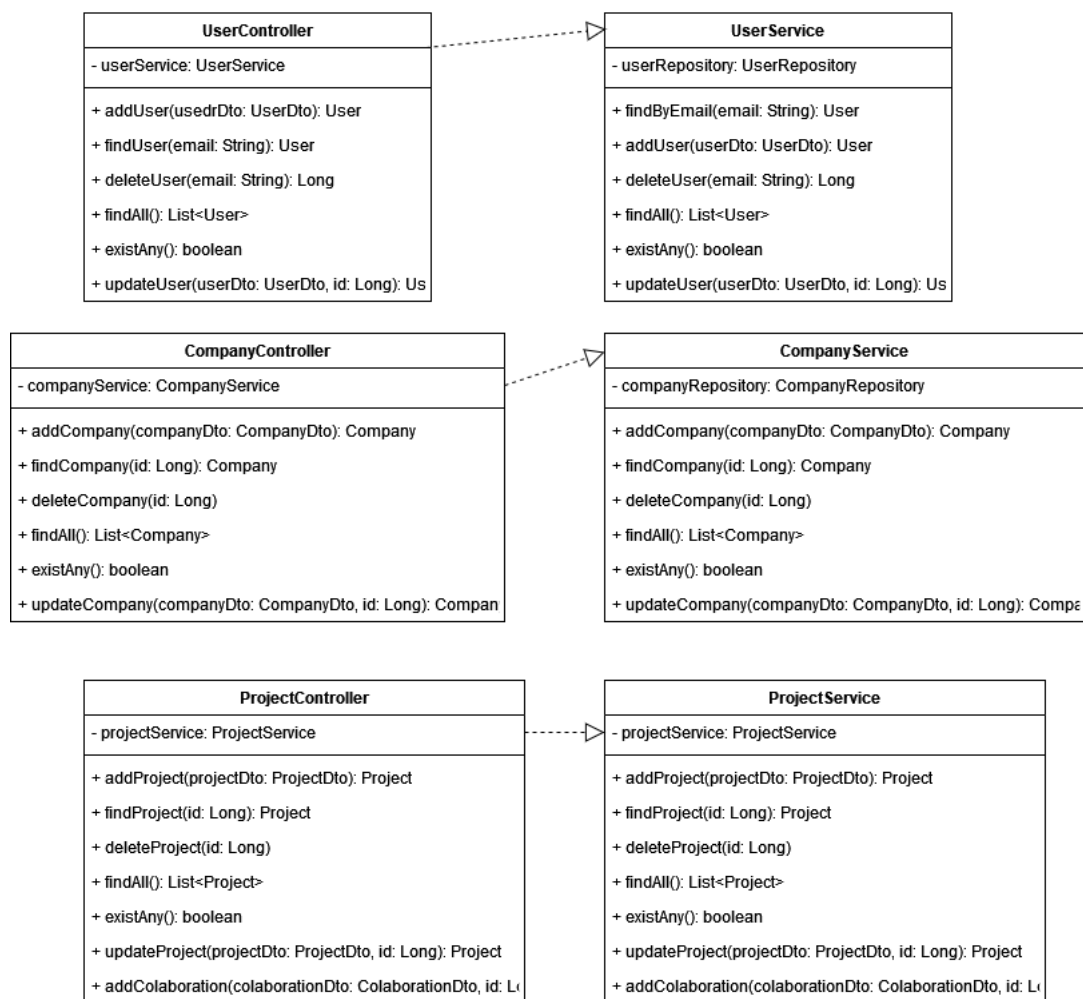
Na sljedećim slikama prikazani su UML dijagrami razreda koji se koriste u aplikaciji. Podijeljeni su u 3 dijela. Prvi dijagram prikazuje klase kojima smo modelirali entitete aplikacije, drugi prikazuje DTO-ove (*Data transfer object*), a treći prikazuje Controllere. Može se uočiti da postoje veze između klasa iz različitih dijagrama (npr. UserDto sadrži UserSystemAuthority), ali su one izostavljene radi preglednosti.



Slika 3.2: Models UML



Slika 3.3: DtoUml



Slika 3.4: Controllers and services UML

3.5 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.

3.6 Dijagram aktivnosti

dio 2. revizije

Potrebno je priložiti dijagram aktivnosti s pripadajućim opisom. Dijagram aktivnosti treba prikazivati značajan dio sustava.

3.7 Dijagram komponenti

dio 2. revizije

Potrebno je priložiti dijagram komponenti s pripadajućim opisom. Dijagram komponenti treba prikazivati strukturu cijele aplikacije.

4. Implementacija i korisničko sučelje

4.1 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.*

4.2 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.

4.2.1 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).*

4.2.2 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/>

4.3 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.

4.4 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.

5. 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

Dodatak: Prikaz aktivnosti grupe

Dnevnik sastajanja

1. sastanak

- Datum: 17. listopada 2022.
- Prisustvovali: P. Hajduk., M. Čenigić, N. Capan, J. Jakovac
- Teme sastanka:
 - Predstavljanje nove teme za prijedlog
 - Rasprava o novoj temi za prijedlog
 - Rasprava o mogućoj podjeli rada
 - Teambuilding

2. sastanak

- Datum: 14. studenoga 2022.
- Prisustvovali: P. Hajduk., M. Čenigić, N. Capan, J. Jakovac, M. Balog, I. Baričević, L. Čunović
- Teme sastanka:
 - Rasprava o implementacijskim detaljima
 - Rasprava o daljnjim koracima

Tablica aktivnosti

| | Petar Hajduk | Jakov Jakovac | Matej Balog | Ivor Baričević | Marko Čengić | Nikola Capan | Lovro Čunović |
|----------------------------------|--------------|---------------|-------------|----------------|--------------|--------------|---------------|
| Upravljanje projektom | | 14 | | 0 | | 0 | |
| Opis projektnog zadatka | | 10 | | 0 | | 0 | |
| Funkcionalni zahtjevi | | 1 | | 0 | | 0 | |
| Opis pojedinih obrazaca | | 3 | | 0 | | 0 | |
| Dijagram obrazaca | | 0 | | 0 | | 0 | |
| Sekvencijski dijagrami | | 0 | | 0 | | 0 | |
| Opis ostalih zahtjeva | | 0 | | 0 | | 0 | |
| Arhitektura i dizajn sustava | | 1 | | 0 | | 0 | |
| Baza podataka | | 1 | | 0 | | 7 | |
| Dijagram razreda | | 0 | | 0 | | 8 | |
| Dijagram stanja | | 0 | | 0 | | 0 | |
| Dijagram aktivnosti | | 0 | | 0 | | 0 | |
| Dijagram komponenti | | 0 | | 0 | | 0 | |
| Korištene tehnologije i alati | | 1 | | 0 | | 0 | |
| Ispitivanje programskog rješenja | | 2 | | 0 | | 0 | |
| Dijagram razmještaja | | 0 | | 0 | | 0 | |
| Upute za puštanje u pogon | | 1 | | 0 | | 0 | |
| Dnevnik sastajanja | | 1 | | 0 | | 0 | |
| Zaključak i budući rad | | 0 | | 0 | | 0 | |
| Popis literature | | 0 | | 0 | | 0 | |

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

| | Petar Hajduk | Jakov Jakovac | Matej Balog | Ivor Baričević | Marko Čengić | Nikola Capan | Lovro Čunović |
|-------------------------------------|--------------|---------------|-------------|----------------|--------------|--------------|---------------|
| Dizajn aplikacije | | 13 | | 1 | | 0 | |
| Rad na responzivnosti aplikacije | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Implementacija Login/Logout | 0 | 7 | 0 | 0 | 0 | 0 | 0 |
| Implementacija Razina ovlasti | | 0 | | 0 | | 0 | |
| Izrada headera | | 2 | | 0 | | 0 | |
| Izrada naslovnice | | 1 | | 0 | | 0 | |
| Izrada users stranice | | 1 | | 3 | | | |
| Izrada user forme | | 4 | | 2 | | 0 | |
| Izrada delete forme | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| Izrada projects stranice | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Izrada project forme | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Izrada companies stranice | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Izrada company forme | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Izrada collaborations komponente | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Izrada collaboration forme | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Izrada spring backenda | | 1 | | 2 | 0 | 0 | 0 |
| Deploy aplikacije | 0 | 6 | 0 | 0 | 0 | 0 | 0 |