

Programsko inženjerstvo

Ak. god. 2023./2024.

Digitalni poster

Dokumentacija, Rev. <1 ili 2>

Grupa: *Posterheimer*

Voditelj: *Dominik Papeš*

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

Nastavnik: *Miljenko Krhen*

Sadržaj

1 Dnevnik promjena dokumentacije	2
2 Opis projektnog zadatka	3
3 Specifikacija programske potpore	6
3.1 Funkcionalni zahtjevi	6
3.1.1 Obrasci uporabe	8
3.1.2 Sekvencijski dijagrami	10
3.2 Ostali zahtjevi	11
4 Arhitektura i dizajn sustava	12
4.1 Baza podataka	12
4.1.1 Opis tablica	12
4.1.2 Dijagram baze podataka	13
4.2 Dijagram razreda	14
4.3 Dijagram stanja	15
4.4 Dijagram aktivnosti	16
4.5 Dijagram komponenti	17
5 Implementacija i korisničko sučelje	18
5.1 Korištene tehnologije i alati	18
5.2 Ispitivanje programskog rješenja	19
5.2.1 Ispitivanje komponenti	19
5.2.2 Ispitivanje sustava	19
5.3 Dijagram razmještaja	20
5.4 Upute za puštanje u pogon	21
6 Zaključak i budući rad	22
Popis literature	23
Indeks slika i dijagrama	24

Dodatak: Prikaz aktivnosti grupe

25

1. Dnevnik promjena dokumentacije

Kontinuirano osvježavanje

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napravljen predložak.	V. Javornik	22.10.2023.
0.2	Dodan osnovni opis projektnog zadatka	V. Javornik	23.10.2023.
0.2.1	Manje promjene opisa projektnog zadatka	V. Javornik	24.10.2023.
0.2.2	Ažuriran dnevnik sastajanja	V. Javornik	24.10.2023.
0.3	Dodani aktori	D. Papeš	24.10.2023.
0.3.1	Dopunjeni aktori	D. Papeš	25.10.2023.

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*

Cilj ovog projekta je razviti programsku podršku za stvaranje web aplikacije „Digitalni poster“ koja će olakšati pregled radova sudionicima stručne konferencije prikazom svakog rada i/ili izlaganja odgovarajućim posterom.

Prilikom pokretanja sustava prikazuje se zahtjev za jedinstvenom lozinkom koja je posjetiteljima dodijeljena prilikom dolaska na konferenciju.

Neregistriranom korisniku odabirom konferencije otvaraju se opće informacije o konferenciji: [organizator, pokrovitelji, digitalni poster]. Uz to, navedena su sva izlaganja [mogu biti razvrstana po kategorijama ili vremenu predavanja]: naziv rada, autor, opis. Neregistriranom korisniku je omogućeno prijavljivanje u sustav s postojećim računom (potrebno je upisati korisničko ime i lozinku) ili kreiranjem novog računa. Za kreiranje novog računa potrebni su sljedeći podaci:

- korisničko ime
- lozinka
- ime
- prezime
- email adresa

- broj mobitela, mjesto i sl. ako će biti potrebno za funkcionalnost

Registracijom u sustav korisniku se dodjeljuju prava posjetitelja konferencije, a naknadno mu se mogu dodijeliti prava autora, organizatora ili administratora [*biti će potrebno nadodati ostale dogovorene vrste korisnika*]. Registrirani korisnik može pregledati, mijenjati osobne podatke i izbrisati svoj korisnički račun.

Posjetitelj prijavom i dolaskom na stručnu konferenciju dobiva jedinstvenu lozinku za pristup sustavu. Unošenjem te lozinke u aplikaciju otvara se [*početna stranica konferencije na kojoj se trenutno nalazi, može biti pregled postera, video sadržaj ili izabrane fotografije*]. Nakon prijave dostupni su im promotivni materijali pokrovitelja konferencije. Posjetitelji mogu pratiti trenutna događanja u glavnoj konferencijskoj dvorani pomoću direktnog videa. Imaju mogućnost glasovanja za svaki pojedini poster koji predstavlja svakog pojedinog predavača na konferenciji. Svaki posjetitelj može glasovati samo jednom tijekom određene konferencije. Glasovanje je moguće samo tijekom određenog razdoblja koje je određeno danima i vremenom održavanja konferencije. Nakon završenog postupka glasovanja, obavljaju se rezultati koji su dostupni svim registriranim korisnicima.

Izabrane fotografije dostupne su registriranim korisnicima tijekom konferencija. Fotografije korisnici mogu spremati lokalno na svoj uređaj. Dostupan im je dio s informacijama o mjestu održavanja konferencije koji između ostalog sadrži podatke o trenutnim vremenskim uvjetima i vremenskoj prognozi za navedenu lokaciju.

Uz posjetitelja postoje još x vrste korisnika, a to su

- autor rada
- organizator
- administrator
- *moгуći korisnici*
 - pokrovitelj (dostupna im je analitika vezana uz njihov sadržaj)
 - hotelijer (spomenut tijekom prve laboratorijske vježbe)

Autor koji sudjeluje na stručnom skupu elektroničkom poštom dostavljaju sve potrebne materijale sistemskom administratoru. Autor prima obavijest o rangui svojeg rada prema glasovima posjetitelja. Porukom elektroničke pošte ih se poziva na dodjelu nagrade za prva tri nagrađena rada. Sve sudionike se elektroničkom poštom obavještava o mjestu i vremenu dodjele nagrade.

Organizator konferencije ima širi spektar mogućnosti za rad u aplikaciji. Njemu je omogućeno ažuriranje podataka o vremenima izlaganja i pozvanih izlagača.

Administrator sustava ima najveće ovlasti. On ima pristup bazi s popisom registriranih korisnika i njihovim podacima te ih može brisati. Obavlja prijavu autora, radova i postera. Ima mogućnost ažuriranja tih podataka. Određuje vrijeme u kojem je glasanje moguće. Dostupni su mu svi podaci te može definirati sve potrebne uvjete za ispravan rad sustava.

Sustav treba podržavati rad više korisnika u stvarnom vremenu.

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. Organizator konferencije (naručitelj)
2. Pokrovitelji konferencije
3. Autori radova
4. Glasači
5. Administrator
6. Razvojni tim

Aktori i njihovi funkcionalni zahtjevi:

1. Neregistrirani/Neprijavljeni korisnik (inicijator) može:
 - (a) upisati lozinku za konferenciju
 - (b) registrirati se to jest stvoriti novi korisnički račun za koji su mu potrebni:
korisničko ime, lozinka i adresa e-pošte
 - (c) prijaviti se
 - (d) pregledavati postere, pri čemu ne može glasati
2. Registrirani/Prijavljeni korisnik (inicijator) može:

- (a) pregledavati i glasati za postere pri čemu svaki posjetitelj može glasati najviše za jedan poster
- (b) pregledavati promotivne materijale pokrovitelja
- (c) gledati video-prijenos
- (d) pregledavati i preuzimati fotografije
- (e) vidjeti informacije o mjestu održavanja koje uključuju vremenske uvjete i vremensku prognozu
- (f) pristupiti rezultatima jednom kad postanu dostupni

3. Administrator (inicijator) može:

- (a) prijaviti i ukloniti autore, radove i postere
- (b) pristupiti svim podacima
- (c) definirati sve uvjete za rad sustava

4. Baza podataka (sudionik):

- (a) pohranjuje podatke o korisnicima
- (b) pohranjuje podatke o digitalnim posterima, broju njihovih glasova, njihovim autorima
- (c) pohranjuje fotografije uslikane za vrijeme trajanja konferencije

5. Streaming servis (sudionik):

- (a) pruža uslugu video prijenosa konferencije

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.

UC1 -Pristup konferenciji

- **Glavni sudionik:** Korisnik
- **Cilj:** Pristupiti konferenciji
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik već nije prijavljen
- **Opis osnovnog tijeka:**
 - (a) Korisnik pokreće aplikaciju
 - (b) Pojavljuje se forma za upis lozinke za pristup konferenciji (ista je za sve korisnike za tu konferenciju)
 - (c) Korisnik upiše lozinku i klikne "Potvrdi" ili Enter
 - (d) Nestaje forma za upis lozinke i pojavljuje se ekran s prikazom svih postera
- **Opis mogućih odstupanja:**
 - 3.a Korisnik upiše krivu lozinku
 - (a) Pristup pregledu konferencije je odbijen
 - (b) Pojavljuje se pop-up "Pogrešna lozinka"

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
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.

- (a) Programsko inženjerstvo, FER ZEMRIS, <http://www.fer.hr/predmet/proinz>
- (b) I. Sommerville, "Software engineering", 8th ed, Addison Wesley, 2007.
- (c) T.C.Lethbridge, R.Langanieri, "Object-Oriented Software Engineering", 2nd ed. McGraw-Hill, 2005.
- (d) I. Marsic, "Software engineering book", Department of Electrical and Computer Engineering, Rutgers University, <http://www.ece.rutgers.edu/~marsic/books/SE>
- (e) The Unified Modeling Language, <https://www.uml-diagrams.org/>
- (f) Astah Community, <http://astah.net/editions/uml-new>

Indeks slika i dijagrama

Dodatak: Prikaz aktivnosti grupe

Dnevnik sastajanja

Kontinuirano osvježavanje

U ovom dijelu potrebno je redovito osvježavati dnevnik sastajanja prema predlošku.

(a) sastanak

- Datum: 18. listopada 2023.
- Prisustvovali: D.Papeš, F.Androić, A.Batić, V.Javornik, M.Perhat, D.Tomšić
- Teme sastanka:
 - upoznavanje tima
 - dogovor načina komunikacije

(b) sastanak

- Datum: 23. listopada 2023.
- Prisustvovali: D.Papeš, F.Androić, A.Batić, V.Javornik, M.Perhat, D.Tomšić
- Teme sastanka:
 - podjela članova u dvije podgrupe, prva će napraviti funkcionalne zahtjeve i obrasce uporabe, druga sekvencijske dijagrame i ostale zahtjeve
 - razmjena pitanja koje će grupa postaviti na laboratorijskim vježbama

Tablica aktivnosti

Kontinuirano osvježavanje

Napomena: Doprinosi u aktivnostima treba navesti u satima po članovima grupe po aktivnosti.

	Dominik Papeš	Fran Androić	Ante Batić	Valeria Javornik	Mario Perhat	Dario Tomšić
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						

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

	Dominik Papeš	Fran Androić	Ante Batić	Valeria Javornik	Mario Perhat	Dario Tomšić
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.